DISS. ETH NO. 20191

STRUCTURAL ANALYSIS OF INFERENCE PROBLEMS ARISING IN SYSTEMS BIOLOGY

A dissertation submitted to

ETH Zurich

for the degree of

Doctor of Sciences

presented by

MARKUS UHR
Dipl. Rech. Wiss. ETH

born 20. April 1981

citizen of Menzingen (ZG)

accepted on the recommendation of

Prof. Dr. Jörg Stelling

Prof. Dr. Julio R. Banga

Prof. Dr. Uwe Sauer

2012

Copyright © 2012 Markus Uhr.
All rights reserved.

First edition; revision 2.

Printed by `lulu.com`.
Electronic copy available at ETH e-collection (`e-collection.library.ethz.ch`).

Summary

THIS THESIS is concerned with several problems in the area of computational systems biology. Many of the problems addressed in this thesis have a common aspect: finding results independent of parameter values. The thesis is structured in three parts. The first part is mainly concerned with structural models of cell metabolism and its (transcriptional) regulation. In the second part, the focus lies on chemical reaction network theory. The final part contains a statistical and computational analysis of a large biological experiment.

The contributions of this thesis are the following. In Chapter 2, we propose an equivalent to sensitivity analysis for stoichiometric network models. We develop the theory based on linear algebra, give an efficient implementation using matrix downdating techniques and show the applicability of the method using two biological examples.

Chapter 3 is devoted to a reverse-engineering method for inferring interaction structures of transcription factors and metabolic genes. To this end, we formulate an optimal control problem which is computationally hard to solve and propose a solution method. To demonstrate the feasibility of the approach, we test the method on a network of *B. subtilis* and compare the solution to already known interactions.

In the context of chemical reaction network theory, we first give a computationally more efficient way of solving the deficiency one problem by reformulating it as a mixed-integer linear feasibility problem (Chapter 5). Furthermore, in Chapter 6, we analyze a special class of reaction networks that we call degenerate. To this end, we introduce the concept of *generic deficiency* which has potential applications for the analysis of large-scale biochemical networks arising in biology.

Chapters 7-9 give a comprehensive analysis of a large nutrient shift experiment that was done within the BASYSBIO project. In particular, we give a regression analysis of the physiological rates of *B. subtilis* under transient conditions; we develop and identify a model able to reproduce the measured data; and we use this model to do a simulation study for giving computational support of the hypothesis that the observed behavior is advantageous for *B. subtilis* under evolutionary competition in its natural environmental conditions.

Zusammenfassung

Diese Doktorarbeit beschäftigt sich mit mehreren Problemen in der Systembiologie. Viele der Themen, die behandelt werden, haben einen gemeinsamen Aspekt: es geht darum, Aussagen über Systeme zu machen ohne deren genauen Parameterwerte zu kennen. Die Thesis ist in drei Teile gegliedert. Der erste Teil beschäftigt sich mit strukturellen Modellen des Zellmetabolismus und seiner (transkriptionellen) Regulation. Im zweiten Teil liegt der Fokus auf der Theorie chemischer Reaktionsnetzwerke. Der letzte Teil besteht aus einer statistischen und computerunterstützten Analyse eines grossen biologischen Experiments.

Die Beiträge dieser Thesis sind die folgenden. Im zweiten Kapitel schlagen wir eine Analyse analog der Sensitivitätsanalyse vor, die auf stöchiometrische Netzwerkmodelle angewendet werden kann. Wir entwickeln die Theorie basierend auf linearer Algebra, geben eine effiziente Methode zu deren Berechnung und wir demonstrieren die Anwendbarkeit der Methode anhand zweier biologischer Beispiele.

Das dritte Kapitel widmet sich einer Methode zur Rekonstruktion von Interaktionen zwischen Transkriptionsfaktoren und Genen metabolischer Reaktionen. Dazu formulieren wir ein optimales Regulationsproblem, das schwierig zu lösen ist und schlagen eine Methode zu dessen Lösung vor. Um die Praktikabilität unseres Ansatzes zu zeigen, benutzen wie ein Netzwerk von *B. subtilis* und vergleichen unsere gefundenen Lösungen mit schon bekannten Interaktionen.

Im Kontext der Theorie chemischer Reaktionsnetzwerke präsentieren wir als erstes einen Weg, das Defizienz-eins-Problem effizienter zu lösen indem wir es als gemischt-ganzzahliges Erfüllbarkeitsproblem neu formulieren. Ausserdem analysieren wir eine spezielle Klasse von Netzwerken, die wir “degeneriert” bezeichnen. Dazu führen wir das Konzept der *generischen Defizienz* ein.

Kapitel 7-9 geben eine umfassende Analyse eines Experiments zur Untersuchung des Verhaltens von *B. subtilis*, wenn die Nahrungsquelle gewechselt wird. Dazu führen wir eine Regressionsanalyse zur Ermittlung der physiologischen Raten durch; stellen ein Modell auf, das die gemessenen Werte reproduzieren kann und identifizieren dessen Parameter; und wir benutzen dieses Modell für eine Simulationsstudie um zu Untersuchen, ob das Modell unter evolutionären Wettbewerbsbedingungen das beobachtete Verhalten bestätigen kann.

To my parents

Preface

Systems biology is an interdisciplinary research field in the intersection of mathematics, computer science, and biology. To the novice, the breath of the field may seem overwhelming. Nobody is an expert in all three areas and it takes some time to familiarize oneself with the necessary knowledge to get a grip on systems biology. To me, this was not different. Having had a classical education in computational science and control engineering, the language of biologists was difficult to understand at first, even cryptic. This fact shines through in this thesis as the focus will lie in the "silicon" part of modeling and computational methods rather than the "wet" part of executing actual biological experiments. But what do systems biologists actually do, then?

Researchers in this field aim to analyze and understand complex biological organisms at the system level. This means that instead of the traditional approach of dissecting a system and analyzing its parts in detail, systems biology tries to understand the entirety of a system. In the context of biology this is a formidable goal, since biological systems have a complex, multiscale structure and involve noise and other sources of stochasticity. That this undertaking is not futile show various success stories in systems and synthetic biology. There are two main directions how insight is gained in systems biology, the experimentalist and the theoretical. Of course, none of these approaches alone could succeed on its own; a collaborative and iterative working style between experiment and theoretical analysis is required.

I was in the lucky position to experience this first hand. My PhD work was integrated in the BaSysBio European research project [1] which focused on the model organism *B. subtilis*. The project itself was a fully integrated systems biology project that involved both data generation by conducting biological experiments and data analysis – including modeling and development of novel theoretical methods. This required interdisciplinary collaboration between research groups with expertise in a wide range of different experimental and computational topics. Therefore, besides all the scientific experience I gained during this time (the results of which are the topics of this thesis), a considerable part also falls on "soft skills" experience about how scientists with different backgrounds can communicate successfully (and less so, for that matter).

I hope you, the reader, may have as much pleasure reading this thesis as I had working and writing on it during my time at ETH Zurich.

Acknowledgements

Successfully completing a PhD and writing a thesis would have never been possible without the involvement of many people giving me constant support and encouragement. First and foremost, I would like to thank my thesis supervisor Jörg Stelling for giving me the opportunity, support and freedom for working on this project. Like any excellent advisor, he knew exactly when to let me pursue my own research ideas and when to set me back on track. Also, I would like to thank my co-advisors Julio Banga and Uwe Sauer for the good collaborations we had within the BaSysBio project and for agreeing to be in my examination committee.

Many thanks go to all co-workers and collaborators that helped me in one way or another during my PhD. In particular, I would like to thank my (former and present) office mates Marco Terzer, Mikael Sunnaker, Sotiris Dimopoulos for fruitful discussions about science and otherwise. I'm also indebted to our crnt crowd Hans-Michael Kaltenbach, Markus Dürr and Irene Otero without whom I wouldn't have gotten the insights I have now. A big thanks goes to all other ZH people in our group for making the day-to-day work such an enjoyable experience and all the rest of the CSB group members for having fun group meetings and retreats. Furthermore, I would like to thank Jörg Büscher and Roelco Kleijn from Uwe Sauer's group for providing me with valuable data as well as biological insight and advice when I didn't see the wood for the trees any more.

Additional thanks go to all the people that helped me improving on the quality of presentation of this thesis. All the proofreading and advise on structuring is greatly appreciated. Of course, I take full responsibility of all remaining deficiencies.

A big thank you goes to all the colleagues and friends that supported me during this time alongside my scientific work. Particular thanks go to Jonas Nart and my colleagues from TV Wila who know how to ground me when I'm on the verge of getting lost in the heights of the ivory tower.

Last, and definitely not least, I would like to thank my parents for their continuous support and believing in me through all this time.

Zürich, December 2011

Markus Uhr

Contents

Summary iii

Zusammenfassung iv

Preface vi

Contents viii

1 Mathematical Modeling of Biological Systems 1
- 1.1 Biochemical Reaction Networks 2
- 1.2 Structural Models of Cell Metabolism 4
- 1.3 Interaction Models for Transcriptional Regulation 8

I Structural Modeling and Inference 11

2 Structural Sensitivity 13
- 2.1 Metabolic Network Fundamentals 14
- 2.2 Definition of Structural Sensitivity 15
- 2.3 Feasibility Test 17
- 2.4 Efficient Computation 18
- 2.5 Application: Analysis of *E. coli* Metabolism 22
- 2.6 Application: Analysis of Transcription Transients 26
- 2.7 Conclusion 30

3 Inference of Transcriptional Control Design of Metabolic Networks 35
- 3.1 Formulation of Inference Problem 36
- 3.2 Optimization Procedure 39
- 3.3 Model Evaluation and Predictions 40
- 3.4 Application Example 44
- 3.5 Summary and Conclusions 47

II Chemical Reaction Network Theory 49

4 Introducing Chemical Reaction Network Theory 51
- 4.1 Modeling of Reaction Networks . . . 52
- 4.2 Derivation of Mass-action Equations . . . 53
- 4.3 More CRNT Concepts and Notation . . . 56

5 The Deficiency One Problem 59
- 5.1 Regular Networks . . . 59
- 5.2 Conditions for Multistationarity . . . 61
- 5.3 MILP Reformulation . . . 69
- 5.4 Numerical Computations . . . 73
- 5.5 Discussion & Perspectives . . . 74

6 Analysis of Degenerate Reaction Networks 77
- 6.1 Generic Network Rank and Deficiency . . . 77
- 6.2 Degenerate EFM Subnetworks . . . 81
- 6.3 Conclusions and Perspectives . . . 85

III The Big Experiment 87

7 Data Analysis of the Big Experiment 89
- 7.1 Outline of the Experiment . . . 90
- 7.2 Measurement Data Analysis . . . 90
- 7.3 Biomass Regression . . . 95
- 7.4 External Metabolite Regression . . . 99
- 7.5 Conclusions & Perspectives . . . 103

8 Single Strain Model 105
- 8.1 Building the Model . . . 105
- 8.2 Identifiability Analysis . . . 112
- 8.3 Parameter Identification . . . 114

9 Evolutionary Analysis 121
- 9.1 Double Strain Model . . . 122
- 9.2 Testing Strain Fitness . . . 123
- 9.3 Strain Dominance . . . 123
- 9.4 Conclusions . . . 125

Appendices 127

A Linear Regression 129

B Leave-one-out Formulas in Linear Regression 133

C Pearson's Correlation Coefficient 137

D Maximum Weight Matching 139

Bibliography 143

Curriculum Vitae 149

1

Mathematical Modeling of Biological Systems

At the heart of every computational task in systems biology is a mathematical model representing the system under consideration. A crucial aspect of system biology is therefore to choose the right level of abstraction for modeling a given system. This first chapter is intended to briefly review and familiarize the reader with modeling approaches relevant for understanding this thesis.

Biological systems form complex interaction networks on several levels of abstraction. From a practitioner's point of view, each level of detail must be traded against the breadth of coverage to contain experimental complexity and achieve computational tractability (see Fig. 1.1). One can identify roughly three levels at which chemical and biological processes take place that are relevant for this thesis. Each level leads to its specific kind of model that is able to represent the relevant processes while parametrizing or simplifying effects from other system scales. In the following, we will introduce the corresponding three modeling approaches.

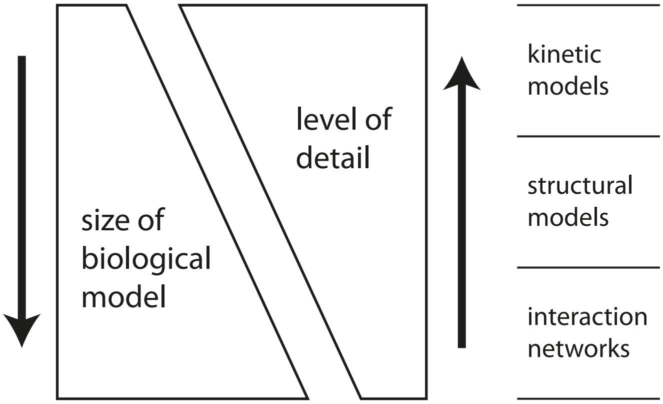


Figure 1.1: Illustration of trade-off between level of detail and size of systems when modeling biological systems. On the right, the three modeling approaches are listed in relation to the modeling trade-offs they make.

1.1 Biochemical Reaction Networks

At the lowest level (meaning the level with the least abstraction), every process that happens in a cell is a *chemical reaction*. In principle, if all possible interactions and all their parameters were known, one could model a cell as one huge chemical factory in which ten-thousands of reactions take place. This is of course an illusionary undertaking; in practice such modeling attempts are limited to specific parts of a cell (such as a metabolic or signaling pathway).

The states of such a biochemical system are the concentrations of its chemical species. That is, we assume that the system consists of a finite set of chemical compounds with concentrations x_i, $i = 1, \dots, m$. Then, a chemical reaction is the process of interconverting molecules from one (or several) species into other species, thereby changing the concentrations x_i. What we are interested in is how the concentrations evolve over time.

The mathematical tool to represent the effect of chemical reactions on species concentrations are *ordinary differential equations* (ODEs). If we collect the concentrations x_i in the vector $x \in \mathbb{R}^m$, we have

$$\frac{dx}{dt} = f(x). \tag{1.1}$$

The system (1.1) can be classified according the form of the function $f(\cdot)$ which is called the *kinetic* of the system.

Mass Action Kinetics

The most basic of all possible kinetics is the so-called mass action kinetic. The underlying assumption here is that a reaction between two chemical species can only happen when two of their molecules collide. Therefore, the rate at which a reaction occurs is proportional to the collision probability of two molecules of the involved species. This probability, in turn, is proportional to the product of the species concentrations.

Example 1.1. Consider a system with three species A_1, A_2, A_3 and the reaction

$$A_1 + A_2 \rightarrow A_3.$$

If we denote the concentrations of A_i with x_i, we get

$$\begin{aligned} \dot{x}_1 &= -kx_1x_2 \\ \dot{x}_2 &= -kx_1x_2 \\ \dot{x}_3 &= kx_1x_2, \end{aligned}$$

where the proportionality constant k is a parameter of the system called *rate constant*. □

The general form of mass action kinetics for a system of m species and n reactions is

$$\dot{x}_l = \sum_{j \in \mathcal{P}(l)} k_j \prod_{i=1}^{m} x_i^{y_{ij}} - \sum_{j \in \mathcal{S}(l)} k_j \prod_{i=1}^{m} x_i^{y_{ij}}, \qquad l = 1, \dots, m. \tag{1.2}$$

Above, we have used the expression $\mathcal{P}(l)$ to denote the (index-)set of all reactions having species l as a product and $\mathcal{S}(l)$ for all reactions having l as a substrate. The exponents y_{ij} are the *molecularity coefficients* of species i in reaction j. The coefficients y_{ij} are non-zero only if species i is a substrate of reaction j.

Part II of this thesis will focus exclusively on biochemical reaction networks that have mass-action kinetics. There, we will discuss a more detailed way of modeling such networks and establish facts about their qualitative behavior irrespective of specific values of the rate constants.

Enzyme Kinetics

An important characteristic of biochemical networks is that basically all reactions are catalyzed by enzymes. This means that the species in biochemical systems can be categorized into small molecules (so-called *metabolites*) and enzymes (which are macro molecules consisting of amino acid chains).

Because of this difference in the participating species, reaction rates can often differ by several orders of magnitude, which means that the system evolves at different time scales. Consider the following prototypical enzyme-catalyzed reaction system (assuming mass action kinetics):

$$S + E \underset{k_{-1}}{\overset{k_1}{\rightleftarrows}} S{-}E \overset{k_2}{\rightarrow} P + E.$$

The first forward and reverse arrows denote the reversible binding of the substrate (S) to the enzyme (E) and the second arrow denotes the irreversible conversion of the substrate into the product (P). Note that the enzyme is not “wasted”, i. e. after each reaction step it can be reused to convert another substrate compound.

The usual assumptions are that (1) the binding parameters k_1, k_{-1} are much larger than the rate constant k_2 (implying that association and dissociation of substrate and enzyme is much faster than conversion of substrate S to product P) and that (2) the total concentration of S is much higher than the total concentration of E. Because of the first assumption, we can assume a quasi steady-state of the two enzyme-binding reactions, i. e.

$$k_1 x_\mathrm{S} x_\mathrm{E} = k_{-1} x_\mathrm{SE}. \tag{1.3}$$

If we denote the total concentrations of S and E as x_{S_t} and x_{E_t} we have

$$x_{\mathrm{S}_t} = x_\mathrm{S} + x_\mathrm{SE} \approx x_\mathrm{S} \tag{1.4a}$$

$$x_{\mathrm{E}_t} = x_\mathrm{E} + x_\mathrm{SE}. \tag{1.4b}$$

The first approximation (1.4a) follows from the assumption that the total concentration $x_{\mathrm{S}_t} \gg x_{\mathrm{E}_t}$, that is, only a small fraction of S can be bound to E.

Using Eqs. (1.4b) and (1.3) together with $K_m \triangleq k_{-1}/k_1$ we get

$$x_\mathrm{SE} = x_{\mathrm{E}_t} \frac{1}{K_m/x_\mathrm{S} + 1} = x_{\mathrm{E}_t} \frac{x_\mathrm{S}}{K_m + x_\mathrm{S}}. \tag{1.5}$$

Plugging this into the ODE for x_{P} yields

$$\begin{aligned} \dot{x}_{\mathrm{P}} &= k_2 x_{\mathrm{SE}} && (1.6a) \\ &= k_2 x_{\mathrm{E}_t} \frac{x_{\mathrm{S}}}{K_m + x_{\mathrm{S}}} && (1.6b) \\ &\approx k_2 x_{\mathrm{E}_t} \frac{x_{\mathrm{S}_t}}{K_m + x_{\mathrm{S}_t}} && (1.6c) \end{aligned}$$

The resulting rate law (1.6c) is called Michaelis-Menten kinetic. It is fundamentally different from the mass action kinetics above. For example, it has a saturation property: if the concentration of S tends to infinity the rate (1.6c) will saturate to the value $k_2 x_{\mathrm{E}_t}$. This system behavior is not obvious from the mass action formulation.

The derivation of this enzymatic rate law reduced a system of three differential equations to only one differential equation. If a biochemical system consists of many enzyme catalyzed reactions, this can simplify the network complexity considerably. Similar laws can be derived for example for reversible reactions and reactions with more than one substrate. Also, regulatory effects such as allosteric inhibition and activation can be modeled within this framework [48].

Typical models where this approach is used are single pathways of metabolic reaction networks. It is difficult, however, to apply such models at a large scale since for each reaction one needs to know the specific reaction mechanism and enzyme kinetic as well as specific parameter values.

1.2 Structural Models of Cell Metabolism

At an intermediary level, one uses a second kind of so-called *structural models*. In this context, *structural* means that these models are free of any kinetic information of a biochemical reaction network. They only contain the topological information of biochemical networks. Justification for this tremendous simplification is again a time-scale separation argument. The structure of metabolic networks allows to separate the effects of metabolite concentrations from the effects of enzyme concentrations. The evolution of metabolite concentrations is governed by enzyme kinetic rate laws, which are much faster than processes affecting enzyme concentrations such as transcription/translation and protein degradation. Thus, with respect to the "enzyme time-scale", it is reasonable to assume a quasi steady state of the cell's metabolites.

Stoichiometry of Metabolic Networks

To capture the network topology of a metabolic system we rewrite the fundamental system equation (1.1) as follows

$$\dot{x} = f(x) = Nv(x). \tag{1.7}$$

The *stoichiometric matrix* $N \in \mathbb{R}^{m \times n}$ is a systems invariant that describes how the chemical reactions interconvert species into each other. The rates at which the reactions occur are captured by the vector $v(x) \in \mathbb{R}^n$. The key point here is that the stoichiometric matrix N depends only on the topology whereas the metabolite concentrations x only affect the reaction rates v.

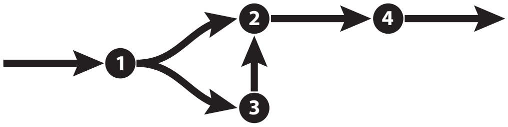


Figure 1.2: Illustration of the network in Example 1.2. The dots symbolize the metabolites (numbers are metabolite indices) and the arrows denote the reactions.

Example 1.2. Consider the example network from Fig. 1.2. The stoichiometric matrix of this network is

$$N = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 1 \\ 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix}.$$

The columns of the matrix N give the stoichiometry of the network reactions. Taking the inner product of the ith row of N with a rate vector $v(x)$ gives the infinitesimal concentration change of the ith metabolite. □

Quasi steady state of the metabolic part of a biochemical system can easily be modeled based on formulation (1.7). We simply require that the metabolite concentrations x to not change over time, i. e.

$$Nv = 0. \tag{1.8}$$

In structural models based on Eq. (1.8), the objects of interest have shifted from the species concentrations x to the reaction rates v. Indeed, there is not much to say about the species any more, since the time derivatives have been explicitly set to zero. The consequences of these simplifying assumptions are vast. To analyze structural models, one can not rely on techniques from classical control theory any more and has to design new analysis methods. The promise of this modeling technique is that analysis methods for structural networks are more likely to scale to large (e. g. genome-scale) models than more traditional system theoretic approaches. Often, for large-scale cellular networks targeted by systems biology, the stoichiometric matrix N is the only available information.

The objects studied on this level of modeling are *solution spaces*, i. e. the feasible values of the fluxes v. The hope is that one can deduce system properties based on the shape of these spaces. To this end, a useful strategy is to further constrain the solution space with additional knowledge. For example, one usually augments the constraints (1.8) with reversibility constraints to reflect whether a reaction is irreversible or not. For this reason, structural modeling is also known as *constraints based modeling*.

Elementary Flux Modes

One possibility of describing the feasible flux space of constraint based models are elementary flux modes (EFMs). Assume we are given the stoichiometric matrix $N \in \mathbb{R}^{m \times n}$ of a metabolic network

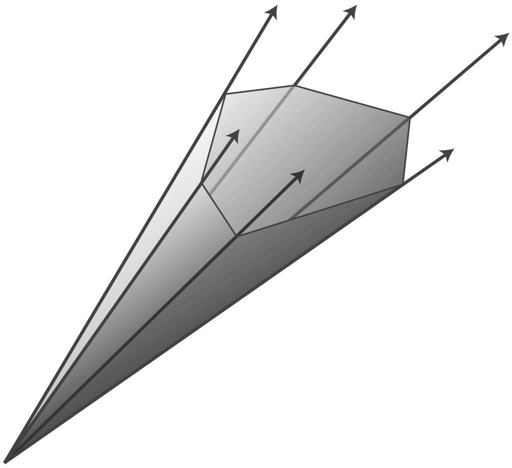


Figure 1.3: Illustration of a polyhedral cone in $\mathbb{R}^3$. The black arrows are the minimal generators (EFMs) of the cone. (Picture adapted from [58].)

and that all reactions are irreversible (if necessary, one introduces an auxiliary reaction in the reverse direction for each reversible reaction). Then, the set of feasible fluxes v can immediately be written as

$$\mathcal{P} = \{v \in \mathbb{R}^n \mid Nv = 0, v \geq 0\}. \tag{1.9}$$

The set $\mathcal{P}$ of all feasible fluxes is a pointed polyhedral cone, and from the above description one can see that it is finitely constrained. By the Minkowski-Weyl theorem [64], polyhedral cones are also finitely generated (see Fig. 1.3). That is, the set $\mathcal{P}$ can equivalently be written as

$$\mathcal{P} = \{v \in \mathbb{R}^n \mid v = E\alpha, \alpha \geq 0\} \tag{1.10}$$

for some matrix $E \in \mathbb{R}^{n \times q}$. It is possible to compute the minimal generating matrix E whose columns are called *elementary flux modes*.

One can show that a minimal generator e of $\mathcal{P}$, i. e. a column of the matrix E, has *minimal support* in the following sense. Let the support of a vector e be

$$\operatorname{supp}(e) = \{i \mid e_i \neq 0\}.$$

Then, a ray e of the cone $\mathcal{P}$ is a minimal generator if and only if for any vector $e' \in \mathcal{P}$

$$\operatorname{supp}(e') \subseteq \operatorname{supp}(e) \implies \operatorname{supp}(e') = \operatorname{supp}(e).$$

Thus, the supports of EFM vectors denote the minimal sets of reactions necessary for a metabolic network to operate in steady state.

Both the finitely constrained and finitely generated representations of $\mathcal{P}$ are dual to each other and there exist algorithms to go from one description to the other. However, given the stoichiometric matrix N, computing the EFM-matrix $E \in \mathbb{R}^{n \times q}$ is nontrivial but possible for many networks with practical relevance [58]. The main difficulty of computing the matrix E is that the number of columns q can grow exponentially with the number of reactions n in the network.

The description of feasible fluxes by EFMs is mostly used to make high-level analyzes of networks. For example, one can study statistics such as the number of reactions per EFM, or the minimal functional units needed to produce all of a cell's biomass components. They prove furthermore useful in generating random feasible fluxes, since it is easy to sample multipliers $\alpha \in \mathbb{R}^q$ according to a prescribed probability distribution, which can be multiplied by E (or some subset of it) to get a feasible random flux distribution v. Metabolic pathway analysis [44], which is based on the decomposition of large metabolic networks into smaller functional units induced by EFMs, is often employed for answering questions from metabolic engineering.

We will also encounter EFMs later in § 6.2 where they are used to analyze multistationarity of EFM-induced subnetworks of a given network. This is related to a method originally described in [12].

Optimality Assumptions

Finding answers to problems such as predicting flux phenotypes or flux adjustments to disturbances is not possible by studying the shape of the flux space alone. A useful tool for arriving at meaningful flux predictions is to make additional optimality assumptions. Justification for these assumptions comes from the theory of evolution, i. e. the selection of individuals in a population according to a fitness criterion. This hypothesis could be experimentally verified in Ibarra et al. [36] who were able to show that the model organism *E. coli* evolves towards an optimal glycerol phenotype when put under appropriate selective pressure.

The first and probably most widely used approach in constraint-based structural models is *flux balance analysis* (FBA) [62] for making predictions on the flux distributions in cell metabolism. This method is based on the assumption that an organism is using its growth substrates in an optimal manner, i. e. to produce as much biomass as possible. The basic skeleton for FBA can thus be written as a standard linear program

$$\begin{aligned} \max_{v} \quad & c^{\mathsf{T}} v \\ \text{s. t.} \quad & Nv = 0 \\ & v \geq 0. \end{aligned} \tag{1.11}$$

As in the definition of EFMs, we assume irreversibility of the reactions (without loss of generality). The direction of optimization c is usually aligned with the biomass reactions. Note that the system (1.11) is in itself not a useful optimization problem since one can readily observe that is is unbounded (assuming the equality constraints are consistent). Thus, useful formulations rely on additional physiological constraints, such as different growth conditions and flux limitations. In this scenario, one replaces the non-negativity constraints $v \geq 0$ with generalized box constraints

$$l_v \leq v \leq u_v.$$

Alternatively, it is possible to generalize the objective function to maximize biomass production *per substrate consumption*, i. e. biomass yield. This case leads to a so-called linear-fractional program with an objective function

$$\max_{v} \quad \frac{c^{\mathsf{T}} v}{d^{\mathsf{T}} v}.$$

The objective vectors c and d define the problem. These problems can still be solved using linear programming [10]. Depending on whether the denominator of the objective $d^{\mathsf{T}} v$ can be negative or not, two or one linear programs must be solved.

Several other biological problems also fit within the constraints-based modeling framework and hence can be solved by linear programming approaches similar to FBA. Among these applications are the prediction of mutant phenotypes, outcomes of evolution, and the analysis of structural couplings in large networks [42].

A more involved generalization of flux balance analysis is to study multiple (non-)linear objective functions simultaneously [47]. The motivation of this approach comes from the fact that the "true" objective function of evolutionary processes is not known, may depend on environmental conditions or can change over time. Potential alternative objectives proposed in the literature are the maximization of ATP yield or the minimization of a cell's redox potential. The result of studying multiple optimality criteria simultaneously is not an optimal solution but a so-called optimal Pareto front that describes the set of non-dominated solutions for all possible weightings of the different, conflicting objective functions.

Another potential application of optimality assumptions lies in predicting metabolic flux adjustments. There, one is usually in the comfortable position to know the (wild-type) flux distribution and the problem is to make predictions of flux adjustments when the organism is subjected to disturbances, for instance by knocking out a gene or changing growth conditions. This is of particular interest in studying genetically engineered organisms. Such organisms were usually not exposed to natural evolutionary pressure and the assumptions underlying FBA may not apply. Instead of directly optimizing the flux distribution, it turns out that minimizing the *flux difference* between the wild-type and a mutant is a good approach in predicting new fluxes. Two methods are published, MOMA (minimization of metabolic adjustments) [49] using the Euclidian norm of flux adjustments as optimization criterion and ROOM (regulatory on/off minimization) [51] using a discrete measure by minimizing the number of nonzero adjustments.

In the next chapter, we will expand on ideas of optimal adjustments by generalizing the basic ideas of MOMA to a sensitivity analysis method for structural networks.

1.3 Interaction Models for Transcriptional Regulation

At the highest level of abstraction (for the scope of this thesis) are general *interaction models* that specify in a qualitative way which parts of a system interact with each other. Such models are used for the description of large (cell-wide) networks including protein-protein interactions (such as in signaling cascades) and gene-protein interactions, also known as genetic regulatory networks.

Experimental data for networks at this level comes mostly from high-throughput methods, for instance high-throughput proteomics (such as yeast two-hybrid or methods based on mass spectrometry) or tiling arrays for gene expression. Such methods deliver experimental evidence that allows for reconstructing network interactions at a global qualitative level of abstraction.

Such models are mostly used for answering biological questions in a qualitative way: whether a target protein is or is not functional; whether a chemical reaction can or cannot take place; whether the expression of a gene is switched on or turned off.

In the following, we will briefly review two modeling formalisms: Boolean networks and graph-based descriptions of interaction networks.

Boolean Models of Metabolic Enzymes

The motivation to use Boolean functions for describing interactions of proteins comes from saturation properties of chemical reactions. Often, protein interactions can be described by Hill kinetics with a sigmoid shape. In a first approximation, the properties of sigmoid functions can be understood as high/low states indicating whether genes are on/off (and thus whether a gene product is present/absent). Such binary system states, depending in a combinatorial fashion on other states, lead to a description using Boolean formulas.

Functional enzymes that catalyze metabolic reactions are often composed of several subunits. Only when all subunits are assembled to form a complex is the enzyme functional. Should one unit be absent because the encoding gene is not transcribed, an enzyme cannot become functional and the chemical reaction will not occur. The requirement that all subunits of a protein complex be present can be encoded by a logical "AND" operator.

For certain metabolic reactions, there exist isoenzymes, i. e. enzymes composed of different subunits catalyzing the same metabolic reaction. In this situation, a chemical reaction occurs whenever at least one of the isoenzymes is functional. This fact can be encoded using a logical "OR" operator. It is sometimes the case that a subunit is part of several enzymes which leads to more complicated Boolean expressions (although every Boolean formula can be brought into disjunctive normal form, of course).

In coarse-grained metabolic models, some metabolic steps are lumped, e. g. in linear pathways without branches. In these cases, the overall reaction from the substrate to the final product takes place only when all intermediate steps a functioning, thus leading to an AND connection between all Boolean formulas of the individual steps.

To summarize, the relation between genetic expression of enzyme subunits and the occurrence of a biochemical reaction in a cell can be described as a Boolean formula. In fact, most metabolic models are augmented with such information, e. g. the genome-scale models of *E. coli* [25] and *B. subtilis* [41].

Interaction Networks of Transcriptional Regulation

Some of a cell's proteins have an effect on the transcription rates of genes. Proteins whose sole purpose is modulating gene expression are called *transcription factors*. Interactions between transcription factors and their target genes leads to a network model where proteins induce forward or feedback control on gene expression.

A convenient tool to capture the topology of interactions between transcription factors (TFs) and genes are *graph-based models*. The edges of the graphs are directed, pointing from the effector (TF) to the affected gene. Furthermore, the edges are labeled to indicate whether a TF has an activating (+) or inhibiting effect (-).

Such graph-based models have very little predictive power. Thus, they are either used in top-down analyzes to make general statements about a network (such as whether it is scale-free or measuring the average connectedness of its nodes). Or they serve as a first model in reverse engineering problems, where the task at hand is to infer the interactions based on empirical data (such as genome sequences).

In this vein, interaction graphs are useful for studying transcriptional control effects on cell metabolism (e. g. switching/regulation of biochemical reactions). The regulated genes of interest are those encoding the enzymes for catalyzing metabolic reactions. This setting leads to a simple case of regulation with only feed-forward signals. In fact, the resulting graph is bipartite with one part being the TFs and the other part consisting of metabolic genes.

Knowledge about such interactions is often collected and stored in publicly accessible databases. For *B. subtilis*, the standard reference of transcriptional interaction networks is DBTBS [52].

In Chapter 3, we will propose a reverse engineering method that is capable to infer potential interaction structures between transcription factors and metabolic genes with high probability. The novelty of this method is that it is based solely on the network structure of cell metabolism.

Part I

Structural Modeling and Inference

2

Structural Sensitivity

Formal sensitivity analysis is a well-established tool for analyzing the influence of perturbations and of control structures on systems behavior. For the area of systems biology, the former type of analysis is mainly concerned with characterizing the overall robustness of biological circuits. The latter can be employed to investigate design principles underlying complex networks in biology [55]. These approaches in general require dynamic, for instance, ODE-based system models. However, establishing realistic and predictive mathematical models for biological circuits is currently a major bottleneck for the field. For most biological systems of interest, our knowledge of control structures and associated kinetic parameters is insufficient for mechanistic modeling. Correspondingly, identification of nonlinear biochemical systems is a challenge [26].

One possible alternative to analyzing the detailed system dynamics is to focus on the set of *possible* behaviors that is consistent with structural constraints, in particular, reaction stoichiometries and reversibilities. This type of 'constraints-based' approaches has been particularly successful for analyzing metabolic networks. For one reason, reaction stoichiometries are well-characterized for metabolic networks such that genome-scale models could be established for several organisms [6]. Also, metabolism operates on faster time-scales than cellular processes such as genetic control systems, which allows for focusing the analysis on the steady-state behavior [42]. Flux-balance analysis (FBA) makes use of this time-scale separation to compute flux distributions in metabolic networks that are optimal in terms of a given objective function by solving a linear optimization problem [62]. Among its many applications, FBA has been used to predict mutant phenotypes, outcomes of evolution, and to analyze structural couplings in large networks [42]. Metabolic pathway analysis, which is based on decomposition of large networks into smaller functional units such as elementary flux modes (EFMs), can be employed for the same purposes. In contrast to FBA, it characterizes the complete solution space of a stoichiometric network model in a generative way instead of a constraints-based way (cf. § 1.2); as a drawback, the determination of elementary pathways is a computationally hard problem. Although some algorithmic progress has been made towards application to genome-scale networks [58], the required computational resources and the sizes of the resulting datasets pose still a challenging problem.

While methods for the analysis of metabolic networks – the controlled system – exist, it is largely unclear how the available biological knowledge could be employed to understand the corresponding control structures. In particular, the genetic control at slow time-scales that establishes different network operation modes is of interest. For instance, by comparison of computational predictions derived from metabolic network structures alone with experimental data on gene expression, the existence of correlations between metabolic fluxes and genetic control has been demonstrated [56, 5]. An apparent path to integrated network representations is the design of hybrid models that represent genetic control in an abstracted form such as Boolean logic models. Such models allow for more accurate predictions [3], but they are of limited use in reverse-engineering of control circuits.

Reverse-engineering of the associated controllers appears feasible because the effective dimension of the control problem may be much smaller than suggested by the complexity of metabolic networks [4]. Several approaches to identify closed-loop reactions to perturbations, or to pinpoint control mechanisms from the structure of the controlled network have been proposed. They rely on assumptions on optimal rejection of perturbations in the sense of a minimal deviation from a given operating point, using continuous ('minimization of metabolic adjustment', MOMA; [49]) or discrete ('regulatory on/off minimization', ROOM; [51]) distance metrics to quantify the deviation between original and perturbed state. Alternatively, 'structural kinetic modeling' makes use of randomly parametrized Jacobians for a given, generic systems model for the same purpose [57].

Despite these advances, an equivalent to formal sensitivity analysis for dynamic systems is lacking. Here, we propose an approach to sensitivity analysis that uses the network structure only and is based on an analytical solution of a least-squares optimization problem (§ 2.2).

In § 2.5, we apply our method to predict control points in the metabolic network of the bacterium *E. coli* as a model system and validate the predictions with published experimental data on metabolic gene expression for a wide range of experimental conditions and perturbations.

2.1 Metabolic Network Fundamentals

Nutrient molecules are taken up by the cell and converted by chemical reactions to intermediary metabolites and finally to components of biomass to enable life. One possibility to model the biochemical processes in a bacterium is to look at the stoichiometry of the chemical reactions occurring in that organism. The interconversion of the different molecules can be modeled as a reaction network. The nodes of this network represent the metabolites that are part of cell metabolism and the edges represent the chemical reactions.

Mathematically, the network structure can be represented as a matrix N, called the *stoichiometric matrix*. The metabolites in a reaction network form the rows of N and the reactions build the columns of the matrix. In a *column* of the stoichiometric matrix, a negative entry denotes a substrate of the reaction and a positive entry denotes a reaction product. Zero entries indicate that a metabolite is not affected by a reaction.

The stoichiometric matrix is a systems invariant that relates the rates v of the reactions (*fluxes*) to the concentrations x of the metabolites. If a metabolic reaction network has m metabolites and n reactions (usually, $m < n$), then $N \in \mathbb{R}^{m \times n}$ and

$$\frac{dx}{dt} = N \cdot v. \tag{2.1}$$

Equation (2.1) is an ordinary differential equation and states that the changes of metabolite concentrations are a linear function of the reaction rates v. Note that the fluxes v generally depend on the concentrations x.

The assumption that the system is in quasi-steady state can be represented as

$$N \cdot v = 0, \tag{2.2}$$

meaning that the flux distribution in the metabolic network lies in the nullspace of N such that metabolite concentrations x do not change. This is the reason why the quantities of interest in metabolic modeling are the fluxes v and not the metabolite concentrations x.

2.2 Definition of Structural Sensitivity

The concept of structural sensitivity is based on a least-squares assumption of metabolic adjustments in biological systems [49]. That is, the structural sensitivity of a metabolic reaction is defined as the "size" of the adjustment required to bring a disturbed flux distribution in steady-state again.

Consider a metabolic network characterized by its stoichiometric matrix $N \in \mathbb{R}^{m\times n}$. A flux $v \in \mathbb{R}^n$ is in steady-state if $Nv = 0$. Assuming an additive disturbance $\delta \in \mathbb{R}^p$ of this flux in a set of p reactions, we split the flux vector as $v = \begin{bmatrix} v_1 & v_2 \end{bmatrix}$, where $v_1 \in \mathbb{R}^{n-p}$ are the undisturbed and $v_2 \in \mathbb{R}^p$ the disturbed reactions. The stoichiometric matrix is split analogously in $N = \begin{bmatrix} N_1 & N_2 \end{bmatrix}$. We require adjustments d such that the disturbed flux is in steady-state again

$$\begin{bmatrix} N_1 & N_2 \end{bmatrix} \begin{bmatrix} v_1 + d \\ v_2 + \delta \end{bmatrix} = 0. \tag{2.3}$$

It is possible that there exist infinitely many possible adjustments d to bring the overall flux in Eq. (2.3) in steady state. We look for optimal adjustments d^* in the sense that their weighted norm $\| W_1 d^* \|_2$ is minimal, where $W_1 = \operatorname{diag}(w_1) \in \mathbb{R}^{n-p\times n-p}$ is a diagonal matrix with positive elements on its diagonal. For symmetry reasons we also consider weighted disturbances $W_2\delta$ with $W_2 = \operatorname{diag}(w_2) \in \mathbb{R}^{p\times p}$ which is also a positive definite diagonal matrix.

In order to find the minimal adjustments d^* of the independent reactions, the following minimization problem must be solved

$$\begin{aligned} \min \quad & \tfrac{1}{2} \| W_1 d \|_2^2 && \text{(2.4a)} \\ \text{s.t.} \quad & N_1 d = -N_2 \delta. && \text{(2.4b)} \end{aligned}$$

For practical applications, we are usually not interested in the additive adjustments d directly but rather in the vector $y \triangleq W_1 d$. Also, it is more interesting to see how y behaves as a function of $\gamma \triangleq W_2 \delta$. Substituting y and γ in (2.4) yields

$$\begin{aligned} \min \quad & \tfrac{1}{2} y^\mathsf{T} y && \text{(2.5a)} \\ \text{s.t.} \quad & N_1 W_1^{-1} y = -N_2 W_2^{-1} \gamma. && \text{(2.5b)} \end{aligned}$$

We can write down an analytical solution for the optimizer of problem (2.5) using the singular value decomposition (SVD) of the matrix $N_1 W^{-1}$

$$N_1 W^{-1} \triangleq U \Sigma V^\mathsf{T} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^\mathsf{T} \\ V_2^\mathsf{T} \end{bmatrix}. \tag{2.6}$$

The splitting of the SVD is according to the rank $r_1 \triangleq \operatorname{rank}(N_1) = \operatorname{rank}(N_1 W^{-1})$ of the constraint matrix, i. e. $\Sigma_{11} \in \mathbb{R}^{r_1 \times r_1}$. Then, the solution is

$$y^* = -V_1 \Sigma_{11}^{-1} U_1^\mathsf{T} N_2 W_2^{-1} \gamma. \tag{2.7}$$

The important point of Eq. (2.7) is that the adjustments y^* are a linear function of the disturbances γ. Thus, we define the matrix of this linear function as the sensitivity to disturbances in the affected reactions.

Definition 2.1 (Structural Sensitivity). *Consider a stoichiometric network defined by its stoichiometric matrix* $N = \begin{bmatrix} N_1 & N_2 \end{bmatrix}$ *and a positive definite diagonal weighting matrix* $W = \operatorname{diag}(W_1, W_2)$. *If we further have the SVD of the matrix* $N_1 W_1^{-1}$ *as in Eq. (2.6), then we define the* structural sensitivity *of the reactions associated with block* N_1 *as a function of disturbances in reactions associated with block* N_2 *as*

$$S \triangleq -V_1 \Sigma_{11}^{-1} U_1^\mathsf{T} N_2 W_2^{-1}. \tag{2.8}$$

The sensitivity S captures the possible adjustments for all directions γ and is therefore an $n - p \times p$ matrix. For an explicitly given disturbance γ we can compute a sensitivity vector $s = S\gamma \in \mathbb{R}^{n-p}$ which is often easier to interpret.

The sensitivity matrix S depends on the choice of the weighting matrices W_1, W_2. There are two canonical cases for defining W_1, W_2 which are described below.

Absolute Sensitivities

If we choose the identity for both matrices, i. e. matrix $W_1 = I$ and $W_2 = I$, we get the well-known 2-norm minimization problem. In this case, we call the sensitivities *absolute sensitivities*, because the objective depends on the absolute values $y_i = d_i$ of the adjustment vector.

Relative Sensitivities

The second important choice for W_1 and W_2 can be made if a reference flux distribution $v^{\text{ref}} = \begin{bmatrix} v_1^{\text{ref}} & v_2^{\text{ref}} \end{bmatrix}$ is known (for example experimentally measured), where the splitting of the vector is according to the (un-)disturbed fluxes. Setting $W_1 = \operatorname{diag}(v_1^{\text{ref}})^{-1}$ and $W_2 = \operatorname{diag}(v_2^{\text{ref}})^{-1}$ leads to minimizing the relative adjustments with respect to the given reference flux. That is, the objective function depends on the relative adjustments $y_i = d_i / v_i^{\text{ref}}$. Therefore, the resulting sensitivities are called *relative sensitivities*.

Scaled Absolute Sensitivities

Historically, the relative sensitivities were defined differently [60]. Instead of minimizing the relative residuals directly, the absolute sensitivity was first computed and then scaled using a given reference flux. This led to various complications in defining the sensitivity properly when a concrete disturbance was given. Thus, we consider this definition to be obsolete.

2.3 Feasibility Test

It is important to see that the computation of the sensitivity matrix S is based on the evaluation of formula (2.8) which always yields a solution. However, the actual optimization problem (2.5) that we are trying to solve might be infeasible.

Because the weighting matrices W_1 and W_2 have both full rank, they do not change the dimension of the subspaces that we will examine below. It is therefore enough to determine feasibility conditions for problem (2.4) or, equivalently, assume w. l. o. g. that W_1 and W_2 are the identity matrices. This will simplify the notation considerably.

If we select p reactions to be disturbed, we can in principle choose any vector $\delta \in \mathbb{R}^p$ as a possible disturbance. To test whether problem (2.4) has a feasible solution for all $\delta \in \mathbb{R}^p$, we have to verify that

$$\operatorname{range}(N_2) \subseteq \operatorname{range}(N_1), \tag{2.9}$$

i. e. whether all disturbances can be "absorbed" by some adjustments d.

Condition (2.9) is equivalent to

$$\operatorname{range}(N_1)^{\perp} \perp \operatorname{range}(N_2) \tag{2.10}$$

that is readily verified by testing the condition

$$U_2^{\mathsf{T}} N_2 = 0, \tag{2.11}$$

where in (2.11), $0 \in \mathbb{R}^{n-p-r_1 \times p}$ is a zero *matrix*.

If condition (2.11) is not met, it is possible to give an orthonormal basis for the subspace of feasible δ. All feasible δ must lie in the null space of $U_2^{\mathsf{T}} N_2$. Assume that the columns of a matrix $Z \in \mathbb{R}^{p \times q}$ are a basis of this null space, where p is the number of disturbed reactions and $q = \dim \operatorname{null}(U_2^{\mathsf{T}} N_2)$. Then, all $\delta = Z\hat{\delta}$ are feasible for any $\hat{\delta} \in \mathbb{R}^q$.

When a concrete disturbance δ is given, we can use a less strict test that only verifies whether a feasible adjustment d exists for the given disturbance. That is, we only require $\delta \in \operatorname{null}(U_2^{\mathsf{T}} N_2)$, or equivalently,

$$U_2^{\mathsf{T}} N_2 \delta = 0, \tag{2.12}$$

with $0 \in \mathbb{R}^{n-p-r_1}$.

2.4 Efficient Computation

Every computation of the structural sensitivity involves a singular value decomposition which is an $O(n^3)$ operation (assuming a square matrix). In many applications, it is necessary to compute the sensitivity for many different disturbances which can become a computationally expensive task. It is thus desirable to speed up the computation of the sensitivity values.

Usually, the number of simultaneously disturbed reactions p is small (often we have $p = 1$). In this case, if we are given n_d disturbances, we need to compute the SVD of n_d very similar matrices: the matrices $N_1^{(i)}$ from Eq. (2.3) derived for the disturbances $i = 1, \dots, n_d$ differ in at most p columns. Instead of computing the SVD for all matrices $N_1^{(i)}$, we could compute the SVD of the complete stoichiometric matrix N and then try to modify this factorization to get factorizations for the $N_1^{(i)}$.

The crucial observation is that we do not need the SVD for computing the structural sensitivities: any complete orthogonal factorization will do [30]. A complete orthogonal factorization of a matrix $A \in \mathbb{R}^{m \times n}$ is given as

$$A \triangleq URV^{\mathsf{T}} = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} R_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^{\mathsf{T}} \\ V_2^{\mathsf{T}} \end{bmatrix} \tag{2.13}$$

As for the SVD, the matrices U and V are orthonormal. The matrix $R_{11} \in \mathbb{R}^{r \times r}$ is upper triangular, with r being the rank of A. For the special case of the SVD, R_{11} would be diagonal.

Downdating a Complete Orthogonal Factorization

It is much easier to remove rows from a factorization (2.13) than it is to remove columns. Therefore, we will focus only on row-downdating.

The core of the downdating method is to remove the *first* row of the matrix. That is, if we are given a matrix

$$A = \begin{bmatrix} a_1^{\mathsf{T}} \\ A_2 \end{bmatrix} = URV^{\mathsf{T}}$$

we want to compute the factorization of $A_2 = U_2 R_2 V_2^{\mathsf{T}}$.

To get the factorization of A_2, we define a sequence of $m - 1$ Givens rotations G_i that annihilate the elements in the first row of the matrix U. Consider the following example with $m = 6$, $n = 5$ and $r = 4$ after the first three rotations:

$$\left(UG_1^{\mathsf{T}} G_2^{\mathsf{T}} G_3^{\mathsf{T}}\right)\left(G_3 G_2 G_1 R\right) = \begin{bmatrix} \times & \boxtimes & \boxtimes & 0 & 0 & 0 \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times & 0 \\ 0 & \times & \times & \times & 0 \\ 0 & 0 & \times & \times & 0 \\ 0 & 0 & \otimes & \times & 0 \\ 0 & 0 & 0 & \otimes & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{2.14}$$

The circled crosses in the R matrix are fills from the second and third rotations. The two highlighted crosses in U determine the next Givens rotation such that the right element gets annihilated. After all $m-1$ rotations are applied, we have

$$\tilde{U}\tilde{R}V^{\top} = \left(UG_1^{\top}\cdots G_{m-1}^{\top}\right)\left(G_{m-1}\cdots G_1 R\right)V^{\top} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \tilde{U}_1 & \tilde{U}_2 \end{bmatrix} \begin{bmatrix} \tilde{r}^{\top} & 0 \\ \tilde{R} & 0 \\ 0 & 0 \end{bmatrix} V^{\top}. \tag{2.15}$$

If the rank of A_2 is equal to the rank of A we are done after these $m-1$ Givens rotations. We can identify

$$U_2 \triangleq \begin{bmatrix} \tilde{U}_1 & \tilde{U}_2 \end{bmatrix} \tag{2.16a}$$

$$R_2 \triangleq \begin{bmatrix} \tilde{R} & 0 \\ 0 & 0 \end{bmatrix} \tag{2.16b}$$

$$V_2^{\top} \triangleq V^{\top}. \tag{2.16c}$$

Note that $\tilde{R}$ is upper triangular. Even though the multiplications with the G_i yield an upper Hessenberg matrix in the upper left block, after removing the top row we have again a triangular $r \times r$ matrix.

A problem arises if $\operatorname{rank}(A_2) = r-1$. In this situation, we need to “shrink” $\tilde{R}$ by one row and column. It is possible to implement this operation efficiently using Givens rotations since the diagonal of $\tilde{R}$ contains a zero. We first shrink the rows by moving the zero on the diagonal to the bottom row. Consider the previous example, where we assume that the second element of the diagonal of $\tilde{R}$ is zero:

$$\begin{bmatrix} \times & \times & \times & \times & 0 \\ 0 & 0 & \boxtimes & \times & 0 \\ 0 & 0 & \boxtimes & \times & 0 \\ 0 & 0 & 0 & \times & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

A Givens rotation that annihilates the lower of the two highlighted elements produces no fill. A series of such rotations can thus be used to zero out all elements on the diagonal after the zero position. This effectively shrinks $\tilde{R}$ by one row.

In the general case, assume that the zero in the diagonal sits at position l ($l \leq r$). We then need $r-l-1$ rotations to achieve the desired result. After applying the rotations, we have

$$A_2 = \left(\tilde{U}G_1^{\top}\cdots G_{r-l-1}^{\top}\right)\left(G_{r-l-1}\cdots G_1 \begin{bmatrix} \tilde{R} & 0 \\ 0 & 0 \end{bmatrix}\right)V^{\top} \tag{2.17a}$$

$$= \bar{U}\begin{bmatrix} \bar{R} & \bar{r} & 0 \\ 0 & 0 & 0 \end{bmatrix} V^{\top} \tag{2.17b}$$

with $\bar{R} \in \mathbb{R}^{r-1 \times r-1}$ upper triangular (it has all zeros in the lower part of the diagonal).

The second step is two shrink the columns, i.e. to zero out the vector $\bar{r}$. This is achieved by $r-1$ Givens rotations applied to the columns of $(\bar{R}, \bar{r})$. In the example, after having applied the first rotation, the situation is

$$\begin{bmatrix} \times & \times & \times & \times & 0 \\ 0 & \boxtimes & \times & \boxtimes & 0 \\ 0 & 0 & \otimes & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The circled element is a fill from the first rotation. The two boxed elements are used to determine the next Givens rotation such that the right element is annihilated.

We then have the final result

$$A_2 = \bar{U}\left(\begin{bmatrix} \bar{R} & \bar{r} & 0 \\ 0 & 0 & 0 \end{bmatrix} G_1^{\mathsf{T}} \cdots G_{r-1}^{\mathsf{T}}\right)\left(G_{r-1} \cdots G_1 V^{\mathsf{T}}\right) \tag{2.18a}$$

$$= \bar{U}\begin{bmatrix} \hat{R} & 0 \\ 0 & 0 \end{bmatrix} \bar{V}^{\mathsf{T}} \tag{2.18b}$$

$$= U_2 R_2 V_2^{\mathsf{T}}. \tag{2.18c}$$

If we want to remove a row other than the first from a matrix A, the procedure above is still useful. We only have to apply a row permutation P to A that moves the desired row to the top.

$$PA = \begin{bmatrix} a_1^{\mathsf{T}} \\ A_2 \end{bmatrix} = (PU)RV^{\mathsf{T}} \tag{2.19}$$

The matrix PU is still an orthonormal matrix and therefore the result will be correct.

Application to Structural Sensitivity

Applying the matrix downdating technique for computing the structural sensitivity is not entirely straightforward since we only know how to remove rows but for the application to structural sensitivity we are actually interested in removing columns. This is why we first need to derive a formula for the structural sensitivity if we are given a factorization of $(NW^{-1})^{\mathsf{T}} = W^{-\mathsf{T}}N^{\mathsf{T}}$.

Assume that we have

$$W^{-\mathsf{T}}N^{\mathsf{T}} = URV^{\mathsf{T}} = \begin{bmatrix} U_1 & U_2 \end{bmatrix}\begin{bmatrix} R_{11} & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} V_1^{\mathsf{T}} \\ V_2^{\mathsf{T}} \end{bmatrix}. \tag{2.20}$$

After removing p rows from the factorization (which is equivalent to removing p rows from N^{T} and the corresponding rows and columns from $W^{-\mathsf{T}}$ since $W^{-\mathsf{T}}$ is a diagonal matrix) we get the factorization

$$W_1^{-\mathsf{T}}N_1^{\mathsf{T}} = \bar{U}\bar{R}\bar{V}^{\mathsf{T}} = \begin{bmatrix} \bar{U}_1 & \bar{U}_2 \end{bmatrix}\begin{bmatrix} \bar{R}_{11} & 0 \\ 0 & 0 \end{bmatrix}\begin{bmatrix} \bar{V}_1^{\mathsf{T}} \\ \bar{V}_2^{\mathsf{T}} \end{bmatrix}. \tag{2.21}$$

Then, analogously to (2.7) and (2.8), we have for the optimal adjustments y^*

$$y^* = -\bar{U}_1 \bar{R}_{11}^{-\mathsf{T}} \bar{V}_1^{\mathsf{T}} N_2 W_2^{\mathsf{T}} y \tag{2.22}$$

and the sensitivities

$$S = -\bar{U}_1 \bar{R}_{11}^{-\mathsf{T}} \bar{V}_1^{\mathsf{T}} N_2 W_2^{-1}. \tag{2.23}$$

Performance Evaluation

Consider the extreme case if we are given a $m \times n$ matrix N and we want to compute the sensitivities for univariate disturbances in all n reactions. To simplify the following calculations, we assume that $m = O(n)$ which is a reasonable assumption for stoichiometric matrices where we often find $m \approx \frac{3}{4}n$. We further assume that the rank of the matrix $r = O(n)$ since often $r \lessapprox m$.

Then, a straightforward implementation would compute n SVDs of all $m \times n-1$ sub-matrices. The FLOP count of this algorithm is $O(n^4)$. The downdating approach, however, requires only $O(n^3)$ FLOPs since one downdating step is of order $O(n^2)$ which is asymptotically much cheaper than a full SVD.

To demonstrate the superiority of the downdating technique, we solved a small benchmark problem using an implementation in Matlab. We computed the sensitivities of random $m \times n$ matrices with $m \approx 0.8 \cdot n$ and coefficients in $[0, 1]$. For each matrix, we considered univariate disturbances in all reactions $1, \ldots, n$.

The running times for both the straightforward implementation and the downdating algorithm are shown in Fig. 2.1. As one can see, the asymptotic advantage of matrix downdating starts to pay off at matrix sizes around $n = 100$. This includes virtually all medium and large scale metabolic networks of biological organisms. For genome-scale networks with $n > 1000$, the downdating technique is the only practically feasible algorithm since computation times for the slower implementation would go into hours.

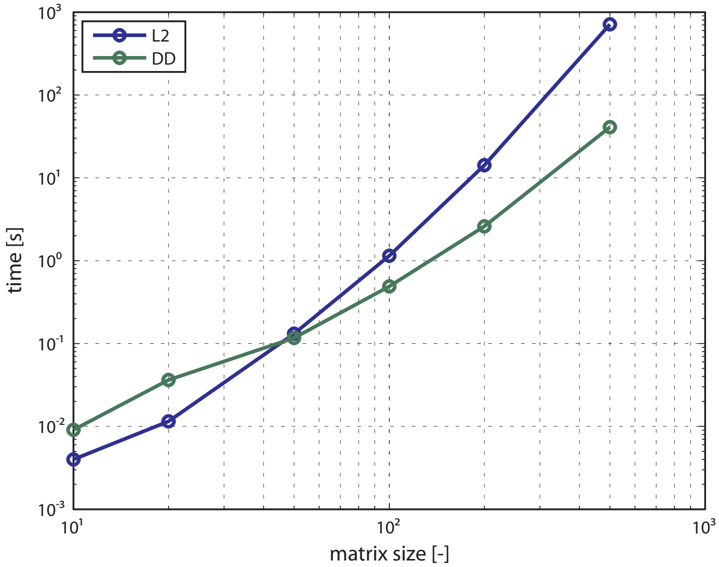


Figure 2.1: Comparison of running times for computing all structural sensitivities based on univariate disturbances. “L2” line denotes running time of straightforward implementation; “DD” line denotes running time of matrix downdating algorithm. Matrix size corresponds to the number of columns n of the $m \times n$ test matrices.

2.5 Application: Analysis of *E. coli* Metabolism

The first application of structural sensitivity is based on a stoichiometric model of *E. coli*. It was published originally in [60] as a proof of concept together with introducing the theory. It starts from the hypothesis that structural network sensitivities are related to variability in gene expression data because if a reaction is very sensitive it requires regulation. Thus, if there are disturbances in the network, sensitive reactions need stronger regulation than insensitive reactions, which should be reflected by high variability of gene expression in the sensitive reactions.

To test this hypothesis, we need a stoichiometric model and a set of gene expression data. For the model, we use a medium scale model of central carbon metabolism of *E. coli* from [56]. Such models capture the basic properties of cell metabolism but have only reduced demands for computational power as compared to full genome-scale models. The model has 97 metabolites and 118 reactions (i.e. the stoichiometric matrix $N \in \mathbb{R}^{97\times 118}$). An interesting property of this network is that the stoichiometric matrix has full row rank, which means that there is no redundancy in the compact stoichiometric model.

For gene expression data, we compiled a set of experimental data from the Many Microbe Microarrays Database (`http://m3d.bu.edu`) [19] that includes 507 experimental conditions among which are different genetic perturbations (knockouts and overexpression), growth conditions (various carbon and nitrogen sources), stresses (antibiotics, heat) and environmental changes (pH, oxygen concentration). Overall, these conditions make up a representative set of random perturbations.

Computation of Network Sensitivities

To be able to analyze the relative sensitivities, we first generate reference fluxes. To this end, we use elementary flux modes (EFMs) of the *E. coli* metabolic network. Elementary flux modes represent independent minimal subsets of reactions that permit steady-state fluxes. More mathematically, they are the minimal generators of the so called “flux cone”

$$\operatorname{null}(N) \cap \mathbb{R}_{\geq 0},$$

which is the set of all feasible fluxes under the steady state assumption and irreversibility constraints (see also the subsection about EFMs in § 1.2). EFMs are unique up to a scaling factor. Building nonnegative linear combinations of EFMs allows for computing all feasible flux distributions, that is, a flux v can be expressed as

$$v = \sum_j \xi_j e_j \tag{2.24}$$

where ξ_j is a non-negative scaling factor for the jth EFM e_j [46].

For the computation of reference fluxes, we consider all EFMs of the *E. coli* model with glucose uptake (and no use of any other substrate) as well as biomass production. This simulates growth of *E. coli* on minimal medium with glucose as its only nutrient source. For the 97×118 model, there are roughly 27 000 EFMs with these properties [56].

EFMs for generating a reference flux are selected according to their biomass yield. This is the ratio of biomass production flux to glucose uptake flux, which is a linear fractional function. Fluxes

Table 2.1: EFM selection for reconstruction of a flux operating point.

α	# EFMs
0.01	21 592
0.1	20 996
0.5	8841
0.9	1425
0.95	708
0.97	400
0.99	90
0.995	35
1	2

with optimal yield $y^{\max}$ are nonnegative linear combinations of the EFMs with highest yield value. The selection of EFMs employs a parameter α such that only EFMs e_j with yield $y_j \geq \alpha \cdot y^{\max}$ are used to generate a reference flux. The values for α and corresponding numbers of EFMs used here are compiled in Table 2.1. To find nonnegative linear combinations of the EFMs, we draw the coefficients randomly from a uniform distribution in the interval $(0, 1)$.

Given the randomized reference fluxes, we are now able to compute the relative sensitivities. We compute the relative sensitivities for a univariate disturbance in every reaction of the network and then take the average over all disturbances. Because the generation of reference fluxes is done by a randomized method, the fluxes might have errors compared to wild-type fluxes. Therefore, we sample 20 different reference fluxes for each disturbance and take the average.

Comparison with Gene Expression Data

For comparing the network sensitivities with gene expression data, we focus on those 42 metabolic genes that are included both in the stoichiometric model and in the data set of experimental data. The experimental data show only small (< 2-fold) variability in average gene expression levels (Fig. 2.2A), but distinctions in the variances. However, as shown in Fig. 2.2B, variability is not related to average expression levels. We find no statistically significant correlation between the coefficient of variation (CV) and the average expression ($r = 0.03$ and $p = 0.85$ for Spearman's rank-order correlation), indicating that prediction of a flux distribution alone will not yield insight into control of the network.

Average structural network sensitivities, in contrast, can show a high correlation with the experimentally determined CV of gene expression, depending on the choice of α (Fig. 2.3). With increasing the threshold α and a corresponding decrease of the share of EFMs used in generating the reference flux distributions, Spearman's rank-ordered correlation coefficient r increases. The two scenarios with $r \approx 0.5$ (Fig. 2.3A) correspond to $\alpha > 0.99$, that is, reference flux distributions with close-to-optimal yield.

To assess the significance of these correlations, we computed the corresponding p-values under the null hypothesis that no correlation exists. As shown in Fig. 2.3B, only the predictions for high

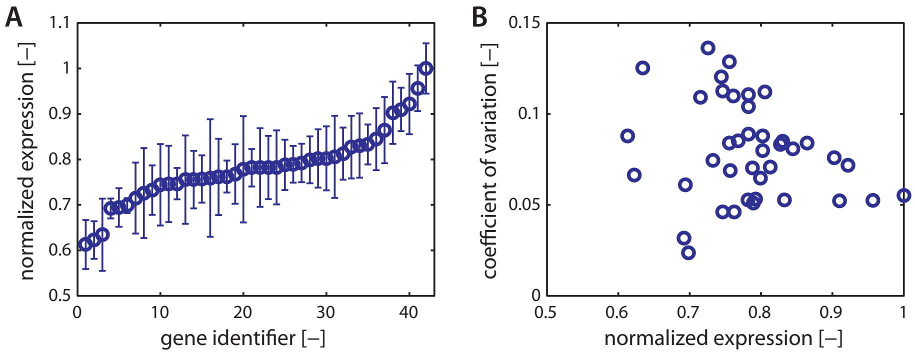


Figure 2.2: Experimentally determined variability in metabolic gene expression for 42 genes in *E. coli* that are included in the stoichiometric model. (A) Average gene expression and standard deviation of expression. (B) Coefficient of variation as a function of the normalized expression level.

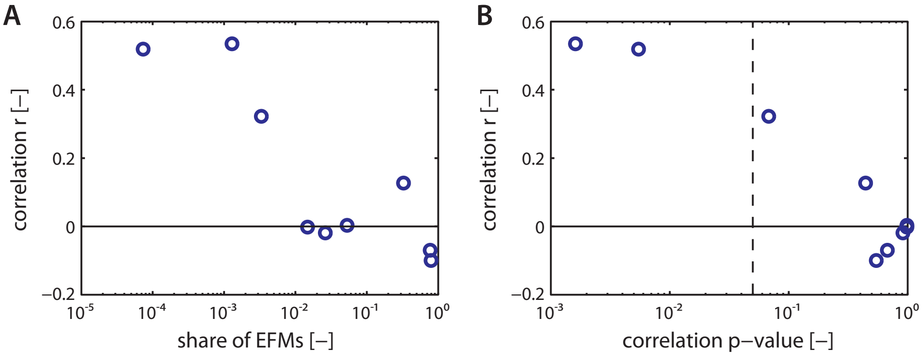


Figure 2.3: Correlation between structural network sensitivities and experimentally observed variation of gene expression for metabolic genes. (A) Spearman's rank correlation coefficient r depending on the share of EFMs with highest biomass yield (number of α-optimal EFMs divided by the total number of EFMs) used for calculating the flux operating point. (B) Statistical significance of the correlations evaluated by the associated p-values; the dashed line indicates a significance level of $p = 0.05$.

α are statistically significant in contrast to, for instance, negative correlations for low α that would contradict our starting hypothesis. This is in accordance with previous findings (e. g. from FBA) that optimal biomass yield is an appropriate objective function to predict fluxes in metabolic networks of microbes [42, 47]. Note also that the correlation coefficients for gene expression variability are higher than those obtained with a simple flux variability score (as defined by the difference between maximal and minimal flux through each reaction) in a previous analysis of a yeast network ($r = 0.17$) [5]; higher p-values compared to that study result from a smaller sample size (number of metabolic genes) considered here.

For practical applications of the approach, a ranking of most sensitive – and, hence, most likely regulated – genes could be useful because this would allow a more targeted, detailed experimental analysis of potential key control points in metabolic networks. Next, we therefore investigated how well structural network sensitivities could identify those genes with most variable expression. More specifically, we rank-ordered the experimental data according to their CV, generated corresponding lists of rank-ordered sensitivities with different lengths, and evaluated the overlap between both.

The null model for statistical evaluation of prediction results follows a hypergeometric distribution. It captures the number of successes in L draws from a finite population of size N without replacement. In our setting, N is the total number of genes and L is the number of most variable genes to be predicted. The expected value for the ratio of correct to total predictions for the null model, φ_0, and the corresponding variance, σ_0^2, are given by

$$\varphi_0 = \frac{L^2}{N} \quad \text{and} \quad \sigma_0^2 = \frac{L^2(N-L)^2}{N^2(N-1)}. \tag{2.25}$$

With this, we can calculate the 95% confidence intervals for the null model as $\varphi_0 \pm 1.96 \cdot \sigma_0$.

Simulation results for different values of α are shown in Fig. 2.4. The share of sensitive genes equals L/N and the prediction accuracy refers to the relative overlap between top-variable genes from the experimental data, and predicted genes with highest structural sensitivity. In agreement with the correlation studies above, we find that the flux operating point has a large influence if the predictions are significantly different from the null model, that is, a random selection of genes. In particular, when only EFMs with highest biomass yield are used for constructing the reference flux distributions, we find an ≈4-fold enrichment of sensitive genes in the predicted lists compared to the null model for the top 10-60% genes (Fig. 2.4D).

Note that these predictions are not perfect; for instance, only in the scenario with $\alpha = 0.01$ (Fig. 2.4A) the genes with highest expression variability are identified correctly. A possible reason for these mismatches between predictions and experimental observations is that the stoichiometric model used here covers only a small core of the metabolic network currently known in *E. coli* whereas the gene expression database contains mostly data collected from stress experiments that affect cell metabolism in peripheral regions. We address the fact that we do not know the resulting disturbance patterns on central carbon metabolism from these experiments by averaging the sensitivities based on univariate perturbations. This lack of detailed knowledge might be another reason for some of the prediction errors.

Figure 2.5 visualizes the correspondence between the reactions' sensitivities with the expression variability of their encoding genes in most of central carbon metabolism. The reactions are divided

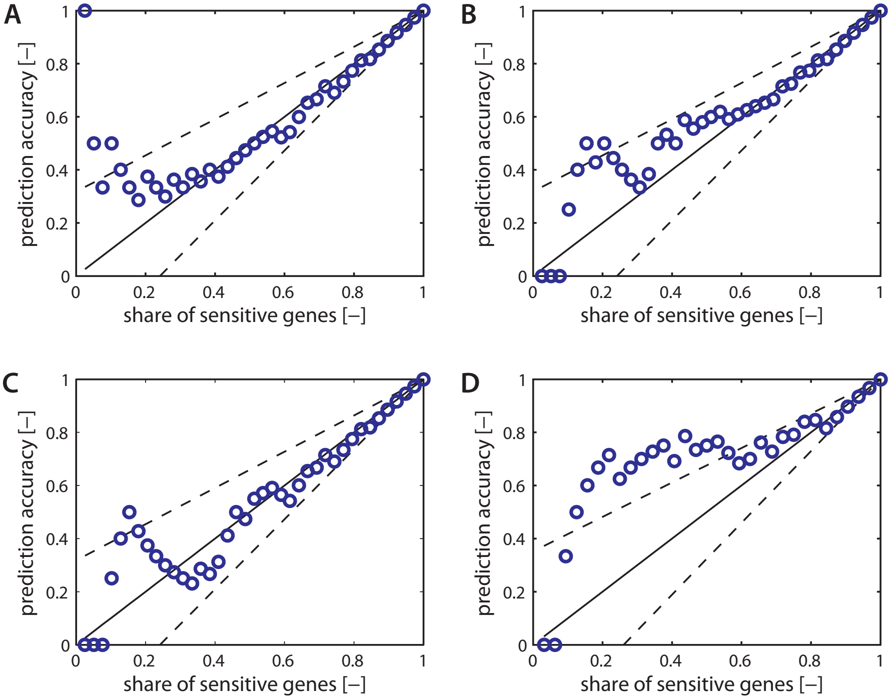


Figure 2.4: Prediction of gene expression variability. Prediction accuracy is determined as the number of genes that show highest experimental variability (CV) *and* are identified as most sensitive by structural sensitivity analysis divided by the number of most variable genes considered for different shares of top-sensitive genes. The panels refer to different scenarios for constructing (random) flux operating points: (A) $\alpha = 0.01$, (B) $\alpha = 0.50$, (C) $\alpha = 0.90$, and (D) $\alpha = 0.995$. Circles refer to structural sensitivity predictions, while the solid line and the dashed lines denote predictions and approximate 95% confidence intervals for the random model, respectively.

into four categories according to their sensitivity and gene variability, showing matches and mismatches of our method. Note that not all reactions could be uniquely assigned to encoding genes. These reactions were not considered in our analysis.

2.6 Application: Analysis of Transcription Transients

The characteristic of the previous application of structural sensitivity analysis was that experimental data was gained under exponential-phase growth conditions. In this last application, we are aiming at analyzing two *dynamic* growth experiments, where a second carbon substrate was added while the bacteria were growing in exponential phase. The overall goal of the experiments was to gain insight into the biochemical processes involved in adapting to a new carbon substrate. In the first experi-

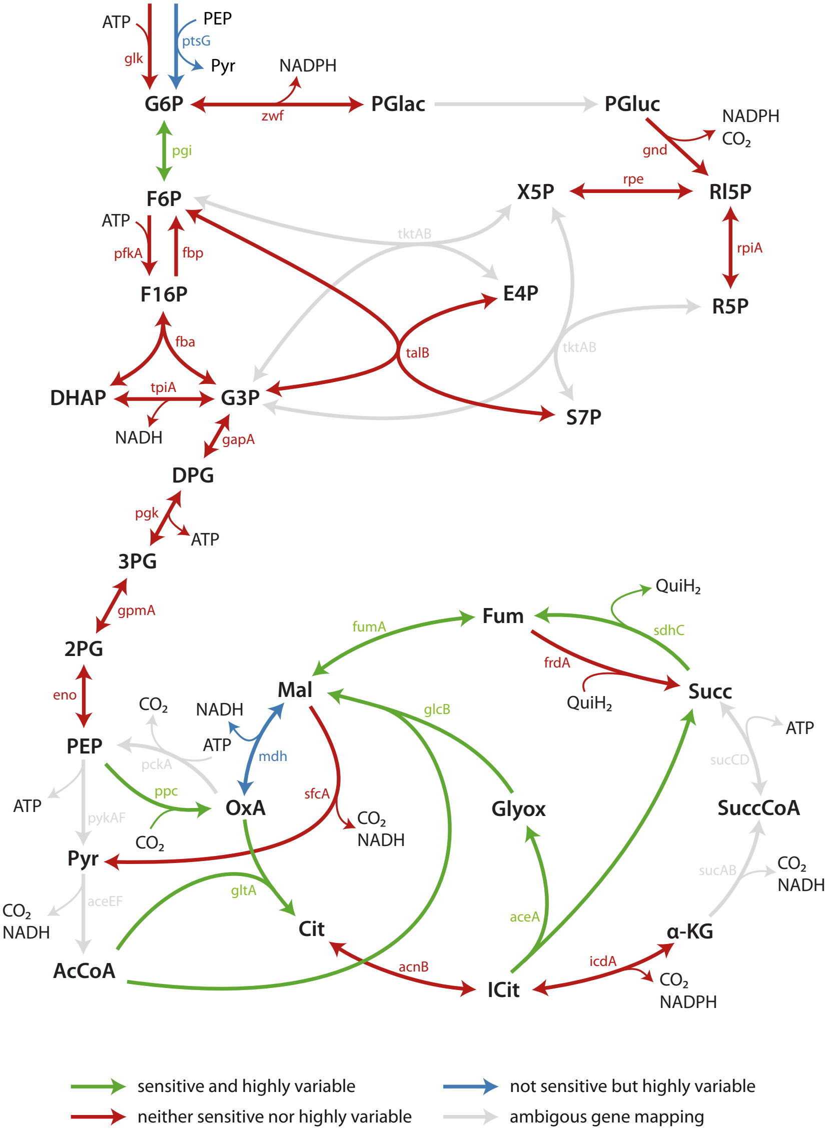


Figure 2.5: Correspondence between structural sensitivity of reactions and gene variability of encoding genes for part of the CCM network of *E. coli* used in this analysis. The reactions are classified as highly sensitive and, correspondingly, gene expression is classified as highly variable if they are among the 14 most sensitive reactions or most variable genes. Three out of 14 reactions are found sensitive but their encoding genes are not highly variable (none shown) and another 3 of 14 are not sensitive but their gene expression is highly variable (2 shown).

ment, *B. subtilis* was disturbed by a malate pulse while it was growing on glucose and in the second experiment, it received a glucose pulse while growing on malate. For a more detailed description and analysis of this experiment see Chapter 7.

The hypothesis drawn from a general analysis of experimental data (exchange rates and internal metabolite pools) was that transcriptional control is only involved in adaptation to glucose, whereas for malate adaptation it only plays a minor role. In the same experiment, intracellular protein concentrations and gene expression profiles were also measured.

Therefore, our aim is to test this hypothesis using structural sensitivity. To this end, recall that the definition of structural sensitivity is based on the assumption that the flux through metabolic reactions adjusts to a new steady-state in response to disturbances. Furthermore, the common approach of modeling the rates v_i of metabolic reactions is using enzyme kinetics of the form

$$v_i = e_i \cdot h(x),$$

where e_i is the concentration of the catalyzing enzyme and $h(x)$ captures the dependence on the metabolite concentrations x. As a first approximation, one can thus assume that the rate changes Δv_i are linear in the enzyme concentration changes Δe_i. Because of the close relation of a reaction's sensitivity and rate change, we are interested in identifying correlations between enzyme concentration changes of metabolic reactions and the structural sensitivities. In that way, it is possible to further support or falsify the above hypothesis that transcriptional control is only involved in one of the nutrient adaptation processes.

Even though we are aiming at analyzing a dynamic experiment, it is valid to use structural sensitivities since we have separation of time scales between metabolic adjustments and transcriptional effects. The latter are the focus of this study and while they might exhibit transient trajectories over the time scale of interest, cell metabolism is always in quasi-steady state.

As already mentioned, for this analysis we are in the comfortable situation of having time course data of intracellular protein concentrations as well as gene expression profiles. We will base our analysis on the protein concentrations, since they are "closer" to the reaction rates and should therefore correlate better with measured flux profiles (given that a flux is mainly transcriptionally regulated, that is). The protein concentrations were measured shortly before and at nine time points after the second substrate was added. The measurements were taken at $t = 0, 5, 10, 15, 25, 45, 60, 90, 120, 150$ minutes after addition of second substrate, where the measurement at $t = 0$ is the reference sample taken during exponential growth. By comparing the other time points with the reference it is possible get estimates of the protein adjustments.

Network Sensitivities

We have used an older version of the medium-scale stoichiometric network of *B. subtilis* with 200 reactions and 186 metabolites. It is missing some uptake reactions that were added at a later point in time. But these are not relevant when *B. subtilis* is growing on glucose and malate as they are neither present in the medium nor can the organism excrete them. Since this network contains a good representation of central carbon metabolism, it is well suited to represent the disturbances of the nutrient shift experiments and thus for this analysis.

Table 2.2: List of reactions that are considered disturbed for computing the network sensitivities. Some reactions were not measured directly but can be assumed to be zero during the course of the experiment. This are the reactions that make up the disturbance vector δ.

reaction	disturbance
Glc transport	measured
Fru transport	assumed zero
Glcn transport	assumed zero
Glu transport	assumed zero
Mal transport	measured
Succ transport	measured
Glyc transport	assumed zero
Pyr transport	measured
Ac transport	measured
Actn transport	assumed zero
growth rate	measured

By design of the experiment, the disturbances affect only the glucose and malate uptake reactions; all other uptake reactions remain undisturbed. We are furthermore in the comfortable situation of knowing the physiological growth rate and exchange rates. This allows not just for indicating the disturbed reactions, but for computing actual disturbance vectors. When splitting the stoichiometric matrix N according to the disturbed and undisturbed reactions, we consider all exchange reactions and the biomass reaction as disturbed and only let internal reactions adjust freely (see Tbl. 2.2). Since we are able to compute an explicit disturbance vector δ, the resulting sensitivities are also vectors (and not matrices), which makes any further analysis easier.

Using this data, we compute absolute structural sensitivities for the given directed disturbances δ for all time points where measurements have been taken in the experiment.

Comparison with Protein Abundances

The protein data set consists of more than 1100 absolute protein quantities in both shift experiments. Because of the technology used, protein detection was not 100% reproducible and not all proteins could be measured in each replicate. We were able to identify over 400 proteins that are related to reactions in the stoichiometric model, e. g. enzymes catalyzing the biochemical reactions.

To compare the sensitivities with the protein data, we have computed the fold-changes of enzyme concentrations for all measured time points, where we took the time point before injection of the second carbon substrate at time $t = 0$ as the baseline concentration. Scatter plots of reaction sensitivities and fold-changes of the corresponding enzymes are shown in Figs. 2.6 and 2.7. Each subplot depicts a measurement sample at a time point after the addition of the second substrate. As a summary of these results, the correlation values (Pearson correlation coefficients) between the sensitivities and the fold-changes are plotted in Figure 2.8. As can be seen, we find significant correlation between the reaction sensitivities and enzyme concentration changes only in the MG shift experiment.

At this point, it is important to recall that this analysis was motivated by the hypothesis that transcriptional control is important to the metabolic adjustment for glucose but not for malate. The findings of the sensitivity analysis are in strong agreement with and thus support this hypothesis.

To verify that these findings are not determined by the enzyme measurements alone, we compared the distributions of enzyme fold-changes (see Figure 2.9 for the fold-change histograms). There is no obvious difference in the distributions between the two shift experiments. Therefore, we can conclude that protein measurements alone could not have revealed these findings.

2.7 Conclusion

We have shown a procedure to identify highly sensitive reactions in metabolic networks. It makes use of stoichiometric information of metabolic networks alone, which is data that is often available early in the investigation of biological organisms. On one hand, this allowed us to neglect the kinetics of the biochemical reactions. On the other hand, this required that we had to make the assumption of least adjustments to disturbances [49]. As a result, the interpretation of the computed sensitivities was possible only in a statistical sense.

Because all computations rely on standard procedures from numerical linear algebra, the method is easy to implement and applicable to larger metabolic networks. Genome-scale models have been established for many model organisms such as *E. coli* [43], *B. subtilis* [41] and could be employed in future work. In particular, calculating structural network sensitivities scales well with network size because the main computational burden lies in the computation of the SVD, which is an $O(n^3)$ process, where n is the number of reactions in a metabolic network. If many sensitivity vectors have to be computed from the same network matrix, improving on this computational complexity is possible. However, for these larger networks, EFMs cannot be computed yet; for specifying the flux operating point, experimental data are needed or sampling methods such as those described in [4] could be employed instead.

We demonstrated the usefulness of the method first in a proof-of-concept application with a model of central carbon metabolism of *E. coli* and experimental data of metabolic gene expression (cf. § 2.5). We were able to show significant correlation between the predicted sensitivities and the variation of gene expression even with such a limited model.

In the final application of § 2.6, we were able to support a hypothesis grounded on the observation of experimental data using computational methods in combination with structural modeling. Because of the good separation between metabolic and transcriptional timescales, it was possible to apply structural sensitivities to the analysis of a dynamic nutrient shift experiment.

Altogether, thus, validation of the structural sensitivity approach with medium-scale stoichiometric models was successful in two cases. It is to say that all presented applications were analysis problems. In combination with the method's scalability, one could further try to make this a useful tool for experimental design to uncover control structures that impinge on metabolic networks.

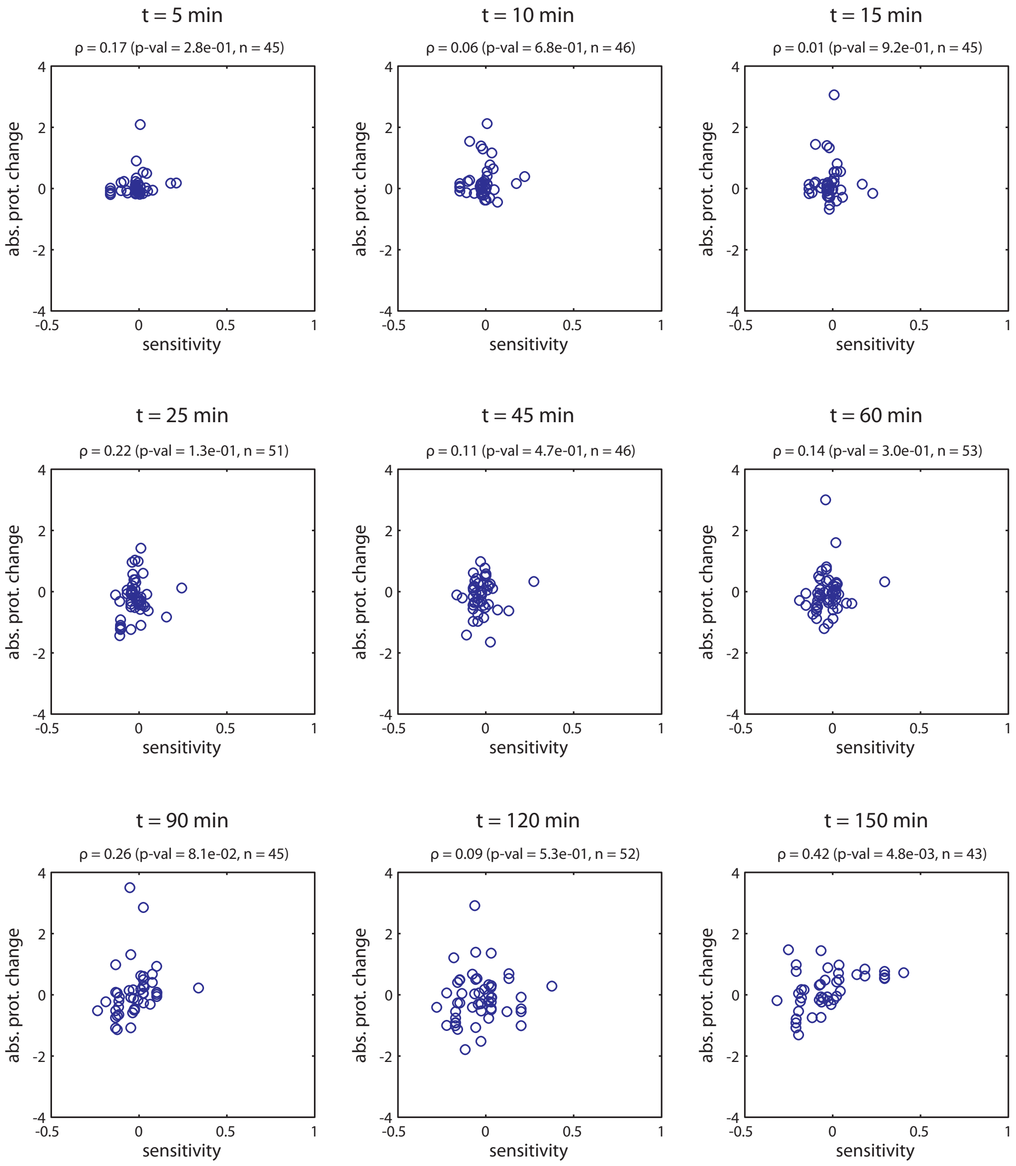


Figure 2.6: Scatter plots of structural sensitivities vs. absolute protein differences at different measurement time points after addition of malate in GM experiment. Numerical correlation values (Pearson correlation coefficient ρ), p-values and number of data points are also given.

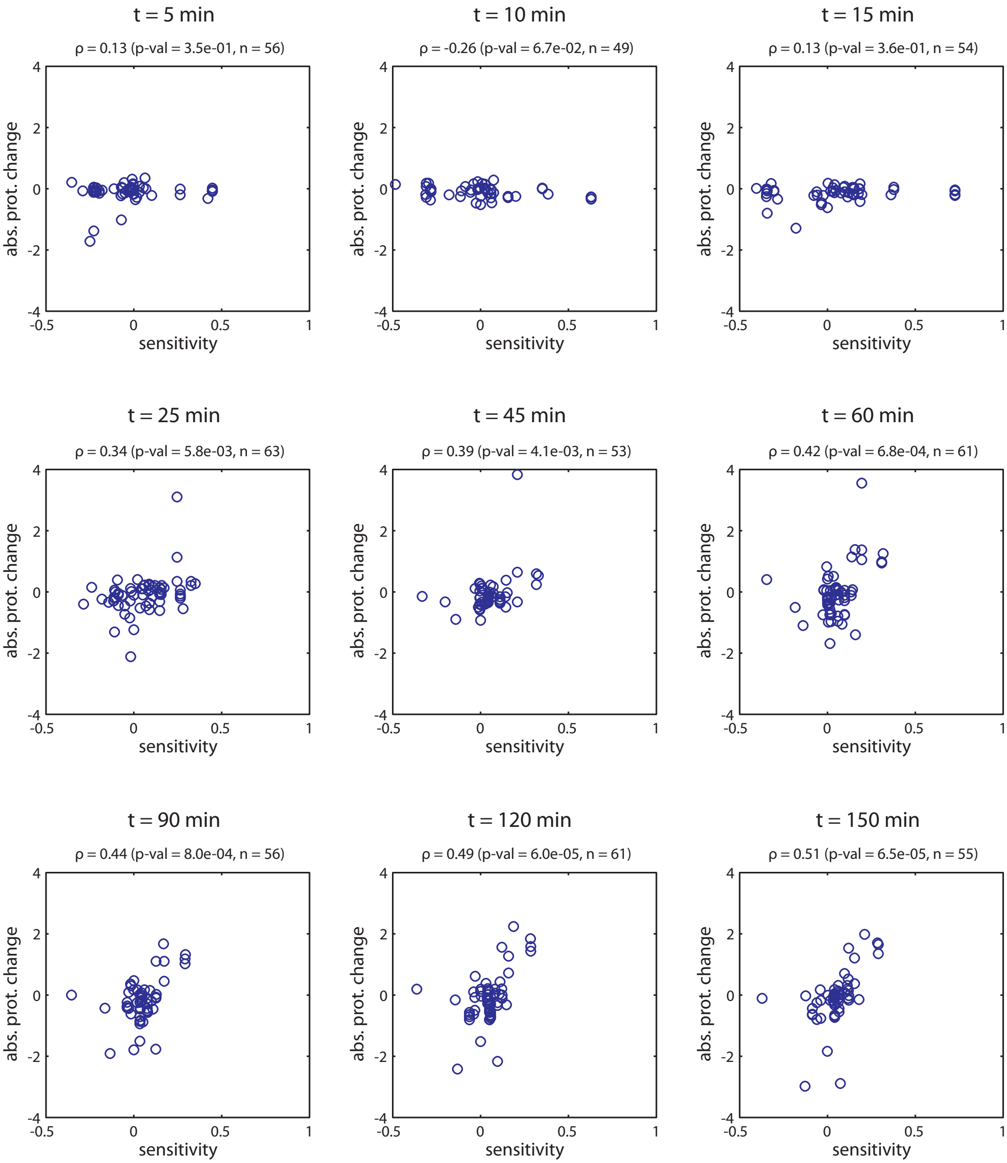


Figure 2.7: Scatter plots of structural sensitivities vs. absolute protein differences at different measurement time points after addition of glucose in MG experiment. Numerical correlation values (Pearson correlation coefficient ρ), p-values and number of data points are also given.

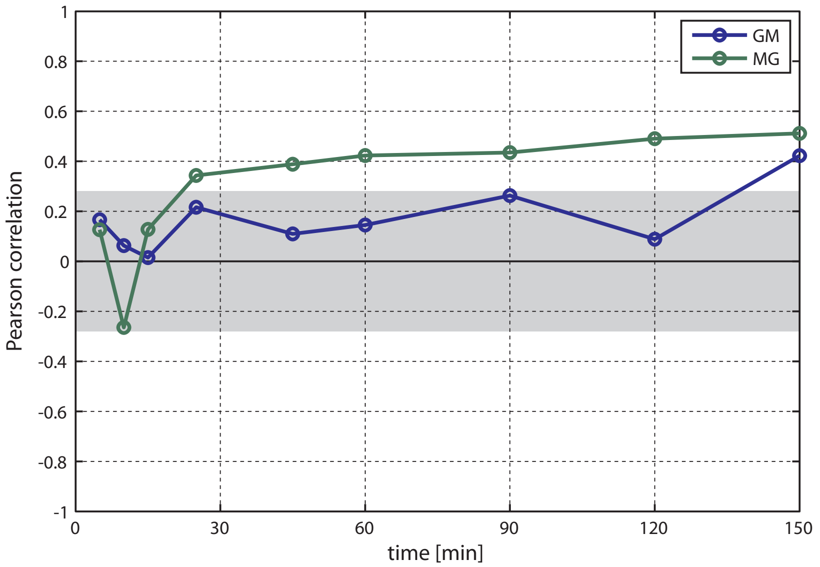


Figure 2.8: Summary plot of correlations for the two nutrient shift experiments. The correlation is between reaction sensitivities and corresponding enzyme concentration changes (log-scale). The concentration changes are taken between time $t = 0$ and the indicated time point. The gray shaded area encloses an approximate interval of correlation with no statistical significance at a 95% confidence level (cf. Appendix C).

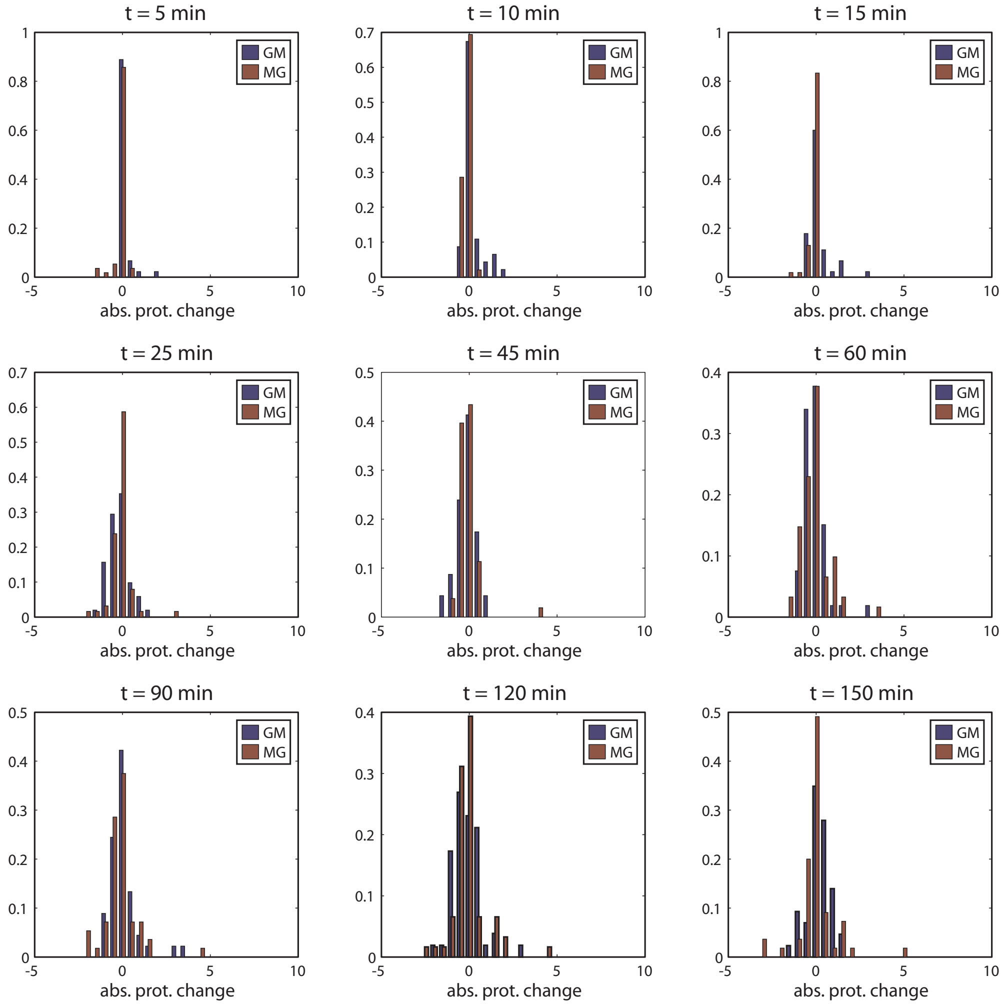


Figure 2.9: Histograms of protein concentration differences at measurement time points. Differences are taken in reference to concentrations at time $t = 0$. Both shift experiments (GM, MG) show a similar spread in the distribution over time.

3

Inference of Transcriptional Control Design of Metabolic Networks

One of the central questions in systems biology is the description and understanding of control in biological systems. The complex architecture of the underlying system and the nonlinear behavior make this task seem difficult. Thus, there exist a wide range of well-established tools for analyzing and describing structures of transcriptional control in biological systems.

However, establishing realistic and predictive mathematical models for biological control circuits is currently a major bottleneck for the field. For most biological systems of interest, our knowledge of control structures and associated kinetic parameters is insufficient for mechanistic modeling. Therefore, identification of nonlinear biochemical systems is a challenge [26]. To cope with current limitations, qualitative and statistical tools are often employed to describe and analyze transcriptional control circuits [37].

Despite this, methods for analyzing metabolic networks – the controlled system – exist. However, it is largely unclear how the available biological knowledge could be employed to understand the corresponding control structures. Methods from bioinformatics that try to infer the structures directly from data (e. g. gene interaction, protein-protein interaction) are often not integrated with the analysis of the controlled system and do not allow conclusions with respect to control aspects. An apparent path to integrated network representation is the design of hybrid models that represent genetic control in an abstracted form such as Boolean logic models. Such models allow for more accurate predictions [3], but are of limited use in reverse engineering control circuits.

Nevertheless, reverse engineering of associated controllers appears feasible because the effective dimension of the control problem may be much smaller than suggested by the complexity of metabolic networks [4]. Several approaches to identify closed-loop reactions to perturbations, or to pinpoint control mechanisms from the structure of the controlled network have been proposed. They rely on assumptions on optimal rejection of perturbations in the sense of a minimal deviation from a given operating point, using continuous [49] or discrete [51] distance metrics to quantify the deviation between original and perturbed state. However, they do not allow to make predictions on the *structure* of the controller.

Here, we propose an approach that tries to infer possible structures of control mechanisms for metabolic networks directly by analyzing different design principle hypotheses. To this end, we formulate a discrete optimization problem with a control objective and compute the optimal realization of such a controller. We demonstrate the feasibility of the approach with a medium-scale model of central carbon metabolism of the bacterium *B. subtilis*.

Parts of this chapter are based on work presented at the 18th IFAC World Congress 2011 [61].

3.1 Formulation of Inference Problem

Metabolic Network Model

Mathematically, the structure of a metabolic network with m metabolites and n reactions can be represented as a matrix $N \in \mathbb{R}^{m \times n}$, called the *stoichiometric matrix*. The metabolites in a reaction network form the rows of N and the reactions build the columns of the matrix. In a *column* of the stoichiometric matrix, a negative entry denotes a substrate of the reaction and a positive entry denotes a reaction product. Zero entries indicate that a metabolite is not affected by a reaction.

The stoichiometric matrix is a systems invariant that relates the rates v of the reactions (*fluxes*) to the concentrations c of the metabolites. If a metabolic network has m metabolites and n reactions (usually, $m < n$), then $N \in \mathbb{R}^{m \times n}$ and

$$\frac{dc}{dt} = Nv. \tag{3.1}$$

Equation (3.1) is an ordinary differential equation and states that the changes of metabolite concentrations are a linear function of the reaction rates.

In the following, we assume that the system is in *quasi steady-state*, that is, the fluxes in the metabolic network are such that metabolite concentrations do not change. Mathematically, this assumption can be represented as

$$Nv = 0. \tag{3.2}$$

The validity of this assumption follows from the fact that we are interested in the effects of transcriptional regulation which evolve on a much slower time scale than the metabolic effects. Hence, the quantities of interest are the reaction rates and not the metabolite concentrations.

Genetic Controller Model

To estimate the structure of genetic controllers for a metabolic network, we use a simplified system to model the effects of transcription factors (TFs) on reaction rates. We assume that a change in transcription factor concentration has a monotone influence (via regulation of enzyme production, considering that fluxes depend linearly on enzyme concentrations) on the reaction rates.

For a metabolic network with n reactions, we use a matrix $Q \in \{-1, 0, 1\}^{n \times n}$ to encode the interaction between transcription factors and reactions. An element q_{ij} of Q is nonzero if reaction i is affected by TF j, otherwise it is zero. The sign of q_{ij} determines the effect of the interaction, which is either activating ($q_{ij} = 1$) or inhibiting ($q_{ij} = -1$). If there is more than one nonzero entry in a *column* of Q, the corresponding TF regulates more than one reaction. It is also possible that a reaction may be affected by more than one transcription factor resulting in multiple nonzeros per *row*. Since the matrix Q captures the whole structure of the interaction of the genetic controller with the metabolic network, it is the decision variable of the inference problem.

Note that the matrix Q may not be quadratic. In fact, biological systems usually have much fewer transcription factors than metabolic reactions. Introducing a quadratic Q has practical reasons: to start the optimization, we need a feasible solution, and the (quadratic) identity matrix is trivially feasible. The estimated control matrix, however, should have many zero columns.

Disturbance Model

To infer the structure of an optimal controller, we need a model of disturbances that such a controller must cope with. For computational tractability, we assume a finite set of disturbances $\Delta = \{\delta^{(1)}, \dots, \delta^{(p)}\}$ with $\delta^{(i)} \in \mathbb{R}^n$. Each disturbance vector $\delta \in \Delta$ has small support, i. e.

$$|\mathrm{supp}(\delta)| \ll n,$$

where the support of a vector is the set of indices of its nonzero components and $|\cdot|$ denotes set cardinality. In practice, disturbances usually have only one or two nonzero components.

A disturbance $\delta \in \Delta$ is assumed to introduce an additive offset to the fluxes of all reactions in its support. Furthermore, we assume that the reactions affected by δ (i. e., the reactions $i \in \mathrm{supp}(\delta)$) remain fixed at $v_i + \delta_i$. Therefore, the controller is only allowed to act on the "free" reactions $\{1, \dots, n\} \setminus \mathrm{supp}(\delta)$. Note that this restriction is necessary (except for some small, pathological example networks). If a controller could counteract also the disturbed reactions, an optimal control structure would have the same structure as the disturbances to neutralize them directly.

Range Space Constraint

The general goal is to find a controller that accounts for all predefined disturbances Δ. Specifically, the steady state condition (2.2) must hold both before and after a disturbance of the network. The assumed operating point of the network before a disturbance is a steady-state flux distribution v. Now, the questions are what are the effects of a disturbance on this flux and how does it look like after the disturbance has happened. Since the disturbed fluxes are fixed to new values defined by the disturbance vector δ, the undisturbed fluxes have to adjust accordingly to achieve a new steady-state. We assume that both the disturbance δ and the "control action" $g \in \mathbb{R}^n$ (via the control structure Q) act additively on the fluxes v, such that the flux distribution after perturbation fulfills

$$N(v + \delta + I_\delta Q g) = 0.$$

Here, I_δ is a diagonal matrix with ones in the rows not affected by δ and zeros otherwise. Hence, with fixed rates of the disturbed reactions, we need to ensure that we can find a compliant vector g for each $\delta \in \Delta$.

Defining $n_\delta \triangleq N\delta$ implies that the existence of a feasible control action g is equivalent to

$$n_\delta = -N I_\delta Q g \qquad \forall \delta \in \Delta. \tag{3.3}$$

It is only possible to find a feasible g if $n_\delta \in \mathrm{range}(N I_\delta Q)$ for all $\delta \in \Delta$, which is why (3.3) is called range space constraint. Note that feasibility of this condition depends crucially on the control structure Q.

Biological Constraints and Objective Function

It is very unlikely that a gene is controlled by a high number of transcription factors. For instance, at most 4 transcription factors control the vast majority of genes in yeast [32]. Therefore, we constrain the number of TF interactions per metabolic reaction to exclude possible, but biologically unrealistic solutions. The maximum number of possible interactions $n_{\max}$ is treated as a parameter in the specification of the optimization problem, modeled as

$$\sum_j |q_{ij}| \le n_{\max} \qquad \forall\, i = 1, \dots, n.$$

To goal of an optimal robust controller is to minimize the control action required to account for the worst case disturbance. Thus, the objective function of the problem can be written as

$$\min_Q \max_\delta \min_g \|g\|_2. \tag{3.4}$$

The choice of the 2-norm for measuring the size of the control action g is analogous to the well-established MOMA method [49], and it has favorable implications for the implementation of the solution algorithm (cf. § 3.2).

The "pure" objective function (3.4) leads to solutions that have many interactions between genes and transcription factors. Comparing such solutions with known interactions gives poor results (data not shown), implying that some sort of regularization is required. We hypothesize that biological cells have to find a compromise between two different (conflicting) objectives:

1. Maximum robustness, that is, minimal flux variations induced by disturbances in the reaction fluxes; and
2. Minimum control effort in terms of the number of interactions required to compensate for the changes.

To account for the second objective, we model the number of interactions as the number of nonzero columns in Q (i. e. the number of active transcription factors) which we denote as $n_c(Q)$.

Since the two objectives are conflicting, we can add the function $n_c(\cdot)$ as a regularization (or penalization) term. It turns out that adding the square of $n_c(Q)$ gives better results. We also penalize the total number of nonzero elements, i. e. the number of gene–transcription factor interactions, which is $n_e(Q)$.

The complete optimization problem is then

$$\begin{aligned} \min_Q \max_\delta \min_g \quad & \|g\|_2 + k_1 \cdot \left(n_c(Q)^2 + k_2 \cdot n_e(Q)^2\right) \\ \text{s. t.} \quad & \textstyle\sum_j |q_{ij}| \le n_{\max} \qquad \forall\, i = 1, \dots, n \\ & n_\delta = -N I_\delta Q g \\ & Q \in \{-1, 0, 1\}^{n \times n} \\ & \delta \in \Delta. \end{aligned} \tag{3.5}$$

The parameters k_1, k_2 determine the trade-off between the two objectives. Good values for these problem-specific tuning parameters have to be found through a "calibration" procedure. Overall, thus, we specify the input by a stoichiometric matrix N of a metabolic network, the set of disturbances Δ, and a choice of the parameters k_i and $n_{\max}$. The output is a robust optimal control structure Q.

3.2 Optimization Procedure

The complete objective function (3.5) of the inference problem consists of three nested optimizations. To solve the optimization problem, we split it into an outer (master) problem consisting of the minimization over Q and an inner problem of evaluating the objective for a fixed Q.

Solving the Master Problem

To solve the master problem, we employ a two-stage procedure. In the first *burn-in* phase, we try to find a good initial feasible solution by running several optimization procedures with $n_{\max} = 1$ and different combinations of the regularization parameters k_1, k_2. For the second *optimization* phase, we employ several solutions from the previous step for initialization and $n_{\max} = 4$. This procedure yields a set of possible solutions. They are evaluated as explained in § 3.3, and the best combination of values for k_1, k_2 is determined.

To solve the actual problem, we used an optimization algorithm inspired by Variable Neighborhood Search (VNS). VNS is a metaheuristic developed in recent years that has been successfully applied to integer optimization problems in different fields [34]. It performs a local search by evaluating the objective function around an incumbent solution, and repeats the procedure visiting different neighborhoods to locate different local optima, among which the global optimum is expected to be found. A key point of the algorithm is the strategy for changing the current neighborhood. VNS usually seeks a new neighborhood by perturbing a set of decision variables using a distance criterion (e.g. using the Hamming distance), followed by a local search in the new neighborhood. Typically, VNS visits proximal neighborhoods (i.e. perturbing a small set of solutions), until no further improvement is achieved. Then, more distant neighborhoods are explored. The alternation between local and global search in VNS may accelerate the convergence to optimal solutions with a reduced risk of getting stuck in local optima.

Here, we implemented advanced strategies to avoid cycles in the search, to increase the efficiency when dealing with large-scale problems, and to modify the search aggressivity to locate high quality solutions (even if not the global optimum) efficiently. Other heuristics, like the "go-beyond" strategy, to exploit promising directions during the local search were adapted from other metaheuristics for continuous optimization [18].

Solving the Inner Problem

The inner problem consists of two optimization tasks, the minimization of the adjustments g and the maximization over the set of disturbances Δ for a fixed control structure Q. Therefore, the penalization term $n_c(Q)^2$ is constant and immaterial for this problem.

We discuss first the minimization problem, i. e. the choice of the optimal control action g for a fixed disturbance δ:

$$\begin{aligned} \min_{g} \quad & \|g\|_2 \\ \text{s. t.} \quad & n_\delta = -NI_\delta Q g. \end{aligned}$$

Note first that g obeys only a set of equality constraints, i. e. finding the minimizer g^* is an equality constrained quadratic optimization problem. Thus, to compute the optimal g^*, we use the singular value decomposition (SVD) of the constraint matrix $NI_\delta Q$:

$$NI_\delta Q = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix}.$$

Then, we have g^* given as

$$g^* = -V_1 \Sigma_{11}^{-1} U_1^T n_\delta.$$

This calculation always yields a value for g^*. To check whether the problem is feasible, we also have to compute the residual

$$r = U_2^T n_\delta.$$

and check if

$$r = 0.$$

In practice, since we are computing with finite precision, we check whether $\|r\|$ (where the norm can be arbitrary) is smaller than a specified tolerance. If the norm is larger, we declare the problem infeasible for the given Q and δ.

The second problem is the maximization over a given set of disturbances Δ. Since we restrict our analysis to a finite set, we solve this problem by enumeration. That is, if we have p disturbances $\delta^{(i)}$, $i = 1, \ldots, p$, we compute the p optimal adjustments $g^*_{(i)}$ for each disturbance and select the largest one

$$\max_i \|g^*_{(i)}\|.$$

This involves p SVDs of the matrices $NI_{\delta^{(i)}} Q$, which becomes computationally expensive for large p.

3.3 Model Evaluation and Predictions

Ultimately, we want to predict possible interactions of transcription factors and metabolic genes. As a control, we first evaluate the quality of our resulting optimal controllers by comparing the predicted controllers with a random model of controllers (the null model). A database of known TF-gene interactions serves as a reference for this comparison. Through this randomized procedure we are able to compute the p-value of the optimal controller under the null hypothesis that it comes from the random model. In other words, we want to find out whether the optimal controller does not predict known interactions better than the random model. The goal, of course, would be to reject the null hypothesis at a high confidence level.

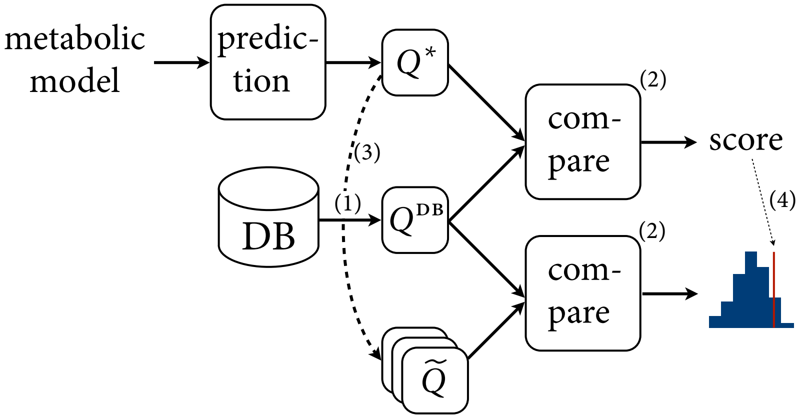


Figure 3.1: Illustration of model comparison procedure. The numbers in parentheses indicate the processing steps – (1) filtering, (2) matching, (3) randomization, (4) computing the p-values – as described in the main text.

Applications for this evaluation procedure are twofold. First, it allows to test whether the modeling assumptions and optimization strategies are able to yield a prediction of practical relevance. Second, we can use this evaluation for “calibrating” the tuning parameters in the model. In particular, it can be used to select the regularization parameters k_1 and k_2 according to the lowest p-values of the solution, as well as the maximum number of TFs per reaction, n_{max}.

Comparison with null model

The “pipeline” of the comparison procedure consists of four steps: (1) filtering, (2) matching, (3) randomization, and (4) computing the p-value. Figure 3.1 gives an overview of how these steps are related to each other. The previous section already explained how to arrive at a prediction Q^* from a stoichiometric matrix N by formulating and solving an optimization problem. This step is represented by the “prediction” box in the figure. The first step (filtering) is to create a control structure Q^{DB} from prior knowledge stored in databases. Then, the matching step follows to compare two control matrices Q, indicated by the “compare” box in the figure. The resulting score is what we will compare to the null model. Thus, the third step is to randomize Q^* and to repeat the comparison for many random structures $\widetilde{Q}$, yielding a null-distribution of scores. The final step is then to compute the p-value of the score from Q^*.

Filtering. First, we have to filter the database for reactions that are present in the metabolic network model as represented by the stoichiometric matrix N. It is possible that some transcription factors have no interactions with the reactions in the model, and therefore those transcription factors have to be removed.

Optionally, one could also consider to filter the reactions further to only include those that appear in the model *and* have an interaction in the DB because, to evaluate the quality of a predicted structure, we can only check whether the already known interactions are reproduced well. Predicted interactions in rows with no known interaction add no information and can be neglected. Filtering this way results in smaller matrices where each row has at least one interaction.

Doing the filtering either way, we obtain a reference interaction matrix Q^{DB} analogous to the predicted (optimal) controller structure Q^*.

Matching. The matching step solves the following problem: in contrast to the structure Q^{DB} extracted from a database, the matrix Q^* has no transcription factor labels associated with it. Therefore, to be able to compare these two matrices, we need to compute these labels. Since transcription factors are associated with matrix columns, we want to find a correspondence between the columns in Q^* and those in Q^{DB} such that the "most similar" columns are matched with each other, in a sense to be made precise. This requires the definition of a score function to measure the "similarity" of two columns of the matrices.

Assume we are given two matrices $Q^{(1)}$, $Q^{(2)}$. Let $q_i^{(a)}$ denote the ith column of matrix $Q^{(a)}$, $a \in \{1,2\}$. Then, we define the column scores $s_{\mathrm{col}}(q_i^{(1)}, q_j^{(2)})$ as the sum of an element-wise comparison of the column vectors, i. e.

$$s_{\mathrm{col}}(q_i^{(1)}, q_j^{(2)}) = \sum_k s_{\mathrm{el}}(q_{ki}^{(1)}, q_{kj}^{(2)}).$$

The elemental scores $s_{\mathrm{el}}(q_{ki}^{(1)}, q_{kj}^{(2)})$ are given as

$$s_{\mathrm{el}}(q_{ki}^{(1)}, q_{kj}^{(2)}) = \begin{cases} 2 & \text{if } q_{ki}^{(1)} = \pm 1 \text{ and } q_{kj}^{(2)} = \pm 1 \\ 1 & \text{if } q_{ki}^{(1)} = \pm 1 \text{ and } q_{kj}^{(2)} = \mp 1 \\ 0 & \text{if } q_{ki}^{(1)} = 0 \text{ and } q_{kj}^{(2)} = 0 \\ -1 & \text{if } q_{ki}^{(1)} = \pm 1 \text{ and } q_{kj}^{(2)} = 0 \\ -2 & \text{if } q_{ki}^{(1)} = 0 \text{ and } q_{kj}^{(2)} = \pm 1. \end{cases}$$

Note that these element scores are not symmetric, i. e.

$$s_{\mathrm{col}}(q_i^{(1)}, q_j^{(2)}) \neq s_{\mathrm{col}}(q_j^{(2)}, q_i^{(1)})$$

in general. The score function is defined such that when we set $Q^{(1)} = Q^*$ and $Q^{(2)} = Q^{\mathrm{DB}}$ we get following properties. True positives are scored 2 or 1 depending on correctness of the sign prediction. False positives are less penalized (−1) than false negatives (−2) because interactions in the database have been inferred from experimental evidence and a prediction missing such an interaction is clearly incorrect. On the other hand, a predicted interaction not present in a database does not prove absence of such an interaction and is therefore not necessarily wrong. In particular, our goal is to find interactions that are not yet discovered.

Assume matrices $Q^{(1)}$ and $Q^{(2)}$ with n_1 and n_2 columns, respectively. To compute the optimal matching between their columns, we model the problem as an assignment problem [8]. To this end, we define the following bipartite graph $G = ((V_1, V_2), E)$ with vertex set V_a being the columns of matrix $Q^{(a)}$, $a \in \{1, 2\}$, and a complete edge set $E = V_1 \times V_2$ that connects every node in V_1 with each node in V_2. The edges are weighted with weights defined by the above column scoring function. Since we have a complete bipartite graph, the edge weights can be arranged as a "score-matrix" $W \in \mathbb{R}^{n_1 \times n_2}$ with elements

$$(W)_{ij} = s_{\text{col}}(q_i^{(1)}, q_j^{(2)}).$$

Then, the best column assignment corresponds to finding a maximum weight matching $M^* \subset E$ in this bipartite graph. We implemented a variant of the Hungarian algorithm [39] that is able to solve the problem for weighted bipartite graphs with negative weights and different numbers of nodes in its two node sets (cf. Appendix D).

Given a maximum weight matching M^*, we construct a similarity measure (score) between two matrices $Q^{(1)}$ and $Q^{(2)}$ as

$$s\left(Q^{(1)}, Q^{(2)}\right) = \sum_i s_{\text{col}}\left(q^{(1)}_{m_i(1)}, q^{(2)}_{m_i(2)}\right).$$

The sum in the above equation goes over the edges in the optimal matching and the column indices $m_i(1)$ and $m_i(2)$ are the node indices from the edges in the matching M^*, i.e.

$$(m_i(1), m_i(2)) \in M^*.$$

Randomization. This step yields the baseline to which we can compare the score of the predicted controller. It is based on randomized variants of Q^* obtained by random row permutations. We consider matrices of the form

$$\widetilde{Q} = PQ^*,$$

where P is a random permutation matrix. This approach preserves the distribution of nonzero elements per column and the signs. Given a random matrix $\widetilde{Q}$, we can repeat the matching of the previous step, which gives a random score. Computing the distribution of these random scores exactly is of course impossible for even a moderate number of rows n in Q^*. We estimate this distribution by computing a large number r of randomly modified control structures $\widetilde{Q}^{(j)}$, $j = 1, \ldots, r$ to approximate the distribution of scores for all $\widetilde{Q}^{(j)}$ (matching for each $\widetilde{Q}^{(j)}$).

Computing the p-value. Since we now know the distribution of scores under a random model, we can compute the p-value of the optimal controller under the assumption that it comes from the random model. To this end, we use the standard, nonparametric rank-estimate for the p-value. Recall that we have constructed r random matrices $\widetilde{Q}^{(i)}$. Define q to be the number of matrices $\widetilde{Q}^{(i)}$ with score equal or higher than the score s^* of the optimal matrix Q^*. Then, the estimated p-value $\hat{p}$ is

$$\hat{p} = \frac{q}{r}.$$

The value $\hat{p}$ indicates the prediction quality in the sense that the smaller it is, the less likely it is that the prediction Q^* is just a "random result" (see Fig 3.3 for an example outcome using the network from § 3.4).

Table 3.1: Systematic evaluation of the influence of the regularization parameters used in the two stages of the optimization procedure (k_1 and k_2). The table shows the p-values of the optimal controllers compared to a random model. The rows (columns) correspond to different values of k_2 (k_1).

k_2/k_1	0.001	0.010	0.100	1.000
0.00	0.668	0.095	0.018	0.554
0.05	0.240	0.005	0.460	0.824
0.10	0.468	0.136	0.018	0.570

Probabilistic predictions

Since the method to compute the optimal controllers is based on a randomized metaheuristic search, there is no guarantee of finding the globally optimal solution after a fixed running time. A good strategy to deal with this problem is thus to start the optimization procedure several times and treat the outcome as a random variable.

Assume we performed n optimizations, resulting in a set of n predicted optimal controllers $Q^{(1)}, \dots, Q^{(n)}$. If we treat each element $q_{ij}^{(k)}$ of a solution $Q^{(k)}$ as an independent Bernoulli variable, we are interested in estimating the interaction probability p_{ij} of a metabolic reaction i with a TF j, given the n realizations from the optimization process.

Note that because of the matching process for identifying the TFs in the solution matrices, not all TFs have the same frequency. Let n_j denote the number of how often TF j was identified. Then, the *minimax* estimator [63] for an interaction between reaction i and TF j is

$$\hat{p}_{ij} = \frac{\sum_k q_{ij}^{(k)} + \sqrt{n_j}/2}{n_j + \sqrt{n_j}}.$$

In our context, there are two reasons for choosing the minimax estimator over the standard maximum-likelihood (ML) estimator. With a relatively low number of observations, minimax estimates are usually more robust than ML estimates. The second reason is based on the following observation: Assume we have two transcription factors, TF1 and TF2 which are matched once and twice in the computed solution matrices, respectively. If we find an interaction of the TFs with a gene (twice for TF1 and once for TF2), which interaction should be ranked higher? Intuitively, one would say that the interaction of TF1 should be ranked higher since we have observed it twice and the interaction of TF2 only once. But the ML-estimator gives $\hat{p} = 1$ for both cases, i. e. they are ranked equally. This is different for the minimax estimator, which ranks TF1 higher than TF2 ($\hat{p} = 0.79$ vs. $\hat{p} = 0.75$), in agreement with intuition.

3.4 Application Example

To establish a proof-of-principle, we used a medium-scale metabolic model of *B. subtilis* with roughly 200 reactions which is based on the genome-scale model of Oh et al. [41]. Since many of the biomass

Table 3.2: Outcomes of 8 independent optimization runs with optimal regularization parameters $k_1 = 0.01$, $k_2 = 0.05$, $n_{\max} = 4$. The column "p-val" denotes the p-value of the comparison with the score of a randomized controller; "#TF" is the number of TFs in the optimal solution; "#matches" is the subset of TFs for reactions with at least one known interaction; "obj" is the objective value.

job	p-val	#TF	#matches	obj.
2585	0.071	15	7	5.7
2587	0.400	20	11	7.0
2611	0.032	20	12	7.4
2612	0.056	15	8	5.6
2613	0.444	18	9	6.1
2615	0.081	17	6	5.9
2642	0.421	12	8	4.5
2650	0.236	18	8	6.3

synthesizing reactions are within linear pathways and could be lumped, we considered only a subset of 62 reactions of central carbon metabolism. This led to a significant speedup of the computations. It also improved the quality of the solutions because the level of detail in the "outer" parts of the model is much lower and gave many false predictions. Note that the alternative of using a genome-scale model could not be pursued because the resulting problem size would have rendered it computationally intractable.

To define the disturbance set, we focused on 7 uptake reactions of possible carbon nutrient sources of *B. subtilis* (glucose, succinate, glycerol, malate, fructose, gluconate and glutamate). We only considered univariate disturbances in these uptake reactions, which corresponds to adding or removing the respective carbon source.

For evaluating the predicted optimal structure, we compared it with already known TF-gene interactions from a database BSBDB (P. Noirot, personal communication) that comprises a superset of the interactions in DBTBS [52]. It contains 1237 interactions between 76 transcription factors and 516 reaction genes. Out of the 76 transcription factors, 24 have interactions with the 62 reactions in our model (with 68 interactions in total, see Fig. 3.2 (right)).

We first "calibrated" the objective function using the evaluation procedure described in § 3.3. We performed a parameter sweep over the regularization parameters k_1 and k_2 of the objective function for both the *burn-in* phase and the *optimization* phase, with one run per (k_1, k_2) pair. The respective p-values resulting from a comparison of the optimal solution with the randomized structures are shown in Table 3.1. This allowed us to identify an optimal reference set of tuning parameters, namely $k_1 = 0.01$ and $k_2 = 0.05$.

To make the actual predictions, we started 8 new optimization runs with the regularization parameter values determined by the "calibration". The results summarized in Table 3.2 clearly indicate that a probabilistic estimation of interactions is needed, given the variability in optimization results.

To examine the potential of predicting true interactions between transcription factors and metabolic reactions, we rank-ordered all interaction pairs according to their minimax score. We labeled

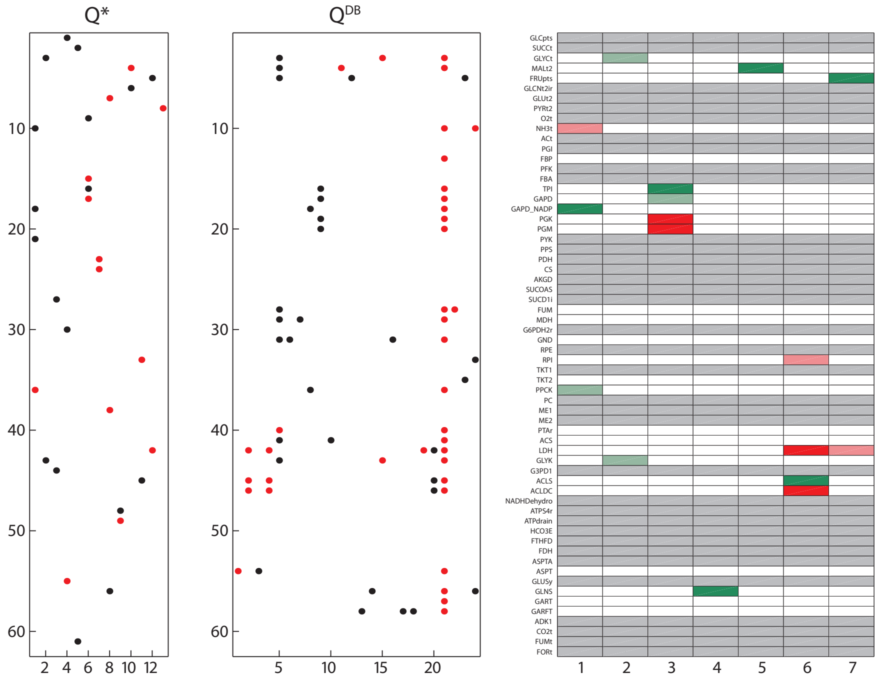


Figure 3.2: *Left:* Example of an optimal predicted interaction structure Q^*. *Middle:* Structure Q^{DB} of known interactions in BSBDB reduced to the 62 reactions in the metabolic network model. In both plots, black dots denote inhibition, red dots activation. *Right:* Optimal matching of Q^* and Q^{DB}. The colors indicate values of elemental scores (dark green: correct prediction including sign; light green: correct prediction, wrong sign; light red: predicted but not in DB; dark red: not predicted but in DB).

each interaction as known or unknown depending on evidence from the database, and analyzed the distribution of labels in the first k elements of this list. The null model of this statistical evaluation follows a hypergeometric distribution with parameters k (number of drawn samples), n (length of list, here: $n = 442$) and p (number of known interactions; here: $p = 27$). The mean and variance of this distribution are

$$\mu_0 = \frac{kp}{n} \quad \text{and} \quad \sigma_0^2 = \frac{kp(n-k)(n-p)}{n^2(n-1)},$$

and we can compute (approximate) 95% confidence intervals for the null model as

$$\mu_0 \pm 1.96 \cdot \sigma_0.$$

Figure 3.4 plots the recovery rate for the *B. subtilis* model and the BSBDB database. In addition, the first 24 interactions are listed in Table 3.3. The predictions are significantly different from the random model, and the top-ranked TF–gene interactions are predicted with high confidence. From

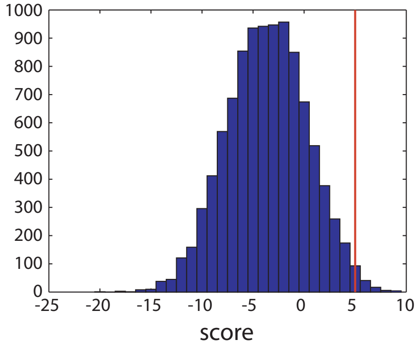


Figure 3.3: Example scores histogram of randomized control structures $\widetilde{Q}^{(i)}$ of 10000 random permutations. The score of the optimal solution Q^* (red line) can be compared to the empirical distribution. Estimating its p-value (here: $\hat{p} = 0.027$) gives an indication of the quality of the solution.

this, we conclude that the optimal-control approach to predict genetic control structures is promising for future applications.

3.5 Summary and Conclusions

We developed a method for predicting cellular control structures which is only based on the well-characterized structures of the controlled (metabolic) network. This type of predictions is not possible with previous methods. For the example application in *B. subtilis*, the predictions resulting from computing optimal control structures reproduce known interactions well. The results not only predict known interactions, but also suggest a number of interactions that had not been reported in the literature yet. Experiments are currently being carried out in order to determine whether these previously unknown interactions exist (see also Tbl. 3.3 for one preliminary result).

In terms of application, we note that there are not many new interactions expected in the central carbon part of cell metabolism. It seems promising to extend the model and the set of disturbances to include, e. g., reactions for modeling cell stress responses. This would be interesting because many transcription factors have interactions in more peripheral parts of cell metabolism. However, our prediction method is computationally demanding. An extension to significantly larger network models would either require recasting the optimization problem into a form that can be solved more efficiently or additional improvements in the implementation, in particular with respect to parallelization of the code.

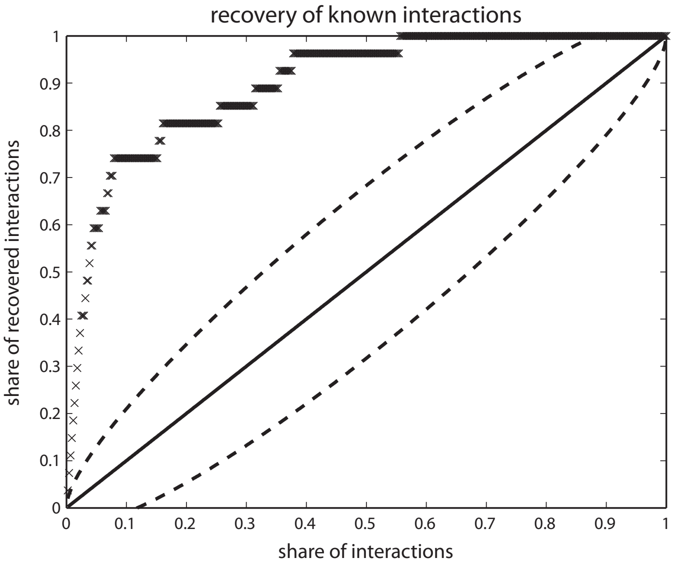


Figure 3.4: Recovery rate of known TF interactions. The x-axis shows the length of the list as a fraction of the total length (k/n). The y-axis shows the percentage of recovered known interactions. The 'x'-line is the actual recovery rate and the solid and dashed lines are the mean and 95% confidence intervals of the null model, respectively.

Table 3.3: List of highest-ranked gene–TF interactions. Gray shaded rows are recovered from BSBDB. Blue row was additionally verified experimentally by P. Noirot et. al (personal communication).

probability	reaction	TF	probability	reaction	TF
0.863	PGM	GlpP	0.750	PFK	Spo0A
0.855	GLYCt	FruR	0.750	FBA	Spo0A
0.845	SUCCt	DctR	0.707	PGI	CcpC
0.833	O2t	CggR	0.667	FRUpts	CcpB
0.833	NH3t	CggR	0.667	TPI	CcpN
0.833	PGK	ResD	0.552	PDH	AnsR
0.833	ACt	SigH	0.500	GLCNt2ir	ASN
0.817	CS	GlnR	0.500	PYRt2	CcpN
0.793	PYK	Rex	0.500	PYRt2	CggR
0.793	PPS	Rex	0.500	GLCNt2ir	CodY
0.793	MALt2	TnrA	0.500	GLUt2	CodY
0.750	PYRt2	GUA	0.500	PYRt2	Rex

Part II

Chemical Reaction Network Theory

4

Introducing Chemical Reaction Network Theory

This chapter intends to familiarize the reader with chemical reaction network theory (crnt) and to introduce its notation and concepts. Originally, crnt was developed as a branch of chemical engineering with the intension to discriminate reaction mechanisms based on qualitative observed properties. The main emphasis of crnt lies on proving the impossibility of a reaction network to have multiple steady states for any set of rate constants. The goal is to find certificates of non-multistationarity (or construct counterexamples) for reaction networks without making any assumptions on the reaction parameters.

Systems biology quickly gained interest in crnt since it faces similar problems. Often, exact mechanisms in metabolic and signaling pathways are unknown and taking measurements in living cells is technically challenging and laborious. It is thus of great interest to the systems biology community to discriminate reaction mechanisms and make qualitative predictions based only on topological properties of reaction networks. Chemical reaction network theory is therefore a tool with potential for applications in systems biology.

There are, however, differences between networks arising in chemical engineering and biology. In particular, the structure of biological networks can become much larger and more complex. This fact can be a stepping stone when applying algorithms from crnt to such networks since they scale poorly with increasing size of the networks. Furthermore, with increasing complexity of reaction networks, preconditions for applying theorems and algorithms are likely not fulfilled any more. Both problem areas require advances in crnt – either algorithmic or theoretical.

Part II of this thesis proposes two extensions to crnt in the hope to bring it closer to the needs of systems biology. In the next chapter we present a reformulation of the so-called *deficiency one algorithm* with tools from discrete optimization. We demonstrate that solving the reformulated versions of the resulting feasibility problems scales much better with increasing network size than naive implementations. In Chapter 6 we present novel ideas for studying networks that are excluded as *irregular* from analysis in crnt. From a historical perspective, it was feasible to exclude this class of networks since they seldom appeared in chemical engineering applications. For biological networks however, it is much more likely to encounter such a network; in particular when dealing with subnetworks resulting from a decomposition procedure proposed by Conradi et al. [12].

In this chapter, we will introduce the modeling concepts and notation used in crnt to describe chemical reaction networks. For an introduction with more thorough exposition of the theory and many more examples see either of Feinberg's review articles [22, 20].

4.1 Modeling of Reaction Networks

The core theory of CRNT focuses on the description of mass-action reaction networks. It models such networks using three sets of chemical species $\mathcal{S}$, of reaction complexes $\mathcal{C}$ and of reactions $\mathcal{R}$. Throughout this report, we assume a fixed ordering of the elements in $\mathcal{S}$, $\mathcal{C}$ and $\mathcal{R}$ and denote by

$$\begin{aligned} m &= |\mathcal{S}| \\ p &= |\mathcal{C}| \\ n &= |\mathcal{R}| \end{aligned}$$

the number of species, complexes, and reactions of a network, respectively. Note that CRNT only deals with sets of finite cardinality.

In chemical reaction systems, one is primarily interested in the time-evolution of species concentrations x_i. We therefore identify the species with their concentrations and with slight abuse of notation we write $\mathcal{S} = \{x_1, \dots, x_m\}$.

The key modeling step of CRNT is to give the substrates and products of chemical reactions their own notion, called *complexes*. A complex $c_j \in \mathcal{C}$, thus, associates a *molecularity coefficient* to each species which could be thought of as a map $c_j : \mathcal{S} \to \mathbb{R}_{\geq 0}$. However, since $\mathcal{S}$ is finite and ordered, we identify the jth complex c_j with its molecularity vector $y_j = (y_{1,j}, \dots, y_{m,j})^\mathsf{T} \in \mathbb{R}^m_{\geq 0}$, where the element $y_{i,j} = c_j(x_i)$ is the molecularity of species i in complex j.

In this setting, the ith reaction $r_i \in \mathbb{R}$ is a triple $(k_i, y_{s(i)}, y_{p(i)}) \in \mathbb{R}_{>0} \times \mathcal{C} \times \mathcal{C}$ where k_i is called *rate constant*, $s(i)$ denotes the index of its substrate complex and $p(i)$ the index of its product complex. Since we assume a fixed ordering of the species, complexes and reactions, the two functions $s(\cdot)$ and $p(\cdot)$ mapping a reaction index to the corresponding index of its substrate and product complex are well defined.

Symbolically, a reaction (with index j) is written as

$$\sum_i y_{i,s(j)} x_i \xrightarrow{k_j} \sum_i y_{i,p(j)} x_i .$$

Example 4.1 (Network (4.1) from [21]). As an example to illustrate the above modeling concepts of CRNT, consider the following reaction network.

$$\begin{aligned} x_1 &\underset{k_2}{\overset{k_1}{\rightleftarrows}} 2x_1 \\ x_1 + x_2 &\overset{k_3}{\rightarrow} x_3 \\ x_3 &\underset{k_5}{\overset{k_4}{\rightleftarrows}} x_2 \end{aligned} \tag{4.1}$$

Network (4.1) has species $\mathcal{S} = \{x_1, x_2, x_3\}$, complexes

$$\mathcal{C} = \{y_1, \dots, y_5\} = \left\{ \begin{pmatrix}1\\0\\0\end{pmatrix}, \begin{pmatrix}0\\1\\0\end{pmatrix}, \begin{pmatrix}0\\0\\1\end{pmatrix}, \begin{pmatrix}2\\0\\0\end{pmatrix}, \begin{pmatrix}1\\1\\0\end{pmatrix} \right\}$$

and reactions $\mathcal{R} = \{(k_1, y_1, y_4), (k_2, y_4, y_1), (k_3, y_5, y_3), (k_4, y_3, y_2), (k_5, y_2, y_3)\}$. □

Since all reactions in the framework of CRNT are unidirectional, the next definition gives the "CRNT equivalent" of a reversible reaction.

Definition 4.1 (Reversible Reaction Pair). *Two reactions i, j are called a* reversible reaction pair *if $p(i) = s(j)$ and $s(i) = p(j)$. If a reaction is not part of a reversible reaction pair, it is called* irreversible.

In example network (4.1), the reactions $(1, 2)$ and $(4, 5)$ are reversible reaction pairs and reaction 3 is irreversible.

4.2 Derivation of Mass-action Equations

In CRNT, a reaction network is an autonomous dynamical system whose state variables are the species concentrations $x(t) \in \mathbb{R}^m$. For notational convenience, we will drop the time index and write x instead of $x(t)$. Symbolically, one can thus write

$$\dot{x} = f(x, k), \tag{4.2}$$

where the time derivatives $\dot{x}$ depend on the species concentrations x and the unknown rate constants k. The question, then, is how the right hand side $f : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^m$ of the ODE system is derived from the structure of a reaction network.

In this thesis, we always assume that the chemical reactions have mass action kinetics. Therefore, the network structure together with the rate constants $k \in \mathbb{R}^n_{>0}$ are sufficient to derive the system of differential equations (4.2) for x. To do this, we first write down the expression of reaction rate v_j of the jth reaction $(k_j, y_{s(j)}, y_{p(j)})$ as

$$v_j(x, k_j) = k_j \cdot x^{y_{s(j)}} = k_j \cdot \prod_{i=1}^{m} x_i^{y_{i,s(j)}}. \tag{4.3}$$

For each reaction j with $y_{s(j)} \to y_{p(j)}$, we define a reaction stoichiometry vector $n_j \triangleq (y_{p(j)} - y_{s(j)}) \in \mathbb{R}^n$. By collecting all rates v_j in a vector $v = (v_1, \ldots, v_n)^\top$ and all stoichiometry vectors n_j in the stoichiometric matrix $N = [n_1; \ldots; n_n] \in \mathbb{R}^{m \times n}$, one obtains the following ODE system describing the concentration dynamics:

$$\dot{x} = f(x, k) = Nv(x, k). \tag{4.4}$$

A key observation of CRNT is that the dynamics $f(x, k) = Nv(x, k)$ can be further decomposed according to Figure 4.1 (cf. [28]). On one hand, since the stoichiometric matrix N is constructed as the difference of molecularity vectors y_i, it is possible to factor it into a product

$$N = YM$$

where $Y = [y_1; \ldots; y_p] \in \mathbb{R}^{m \times p}_{\geq 0}$ is the concatenation of the molecularity vectors and $M \in \{-1, 0, 1\}^{p \times n}$ "selects" the substrate and product vectors of the reactions. Note that M has exactly one -1 element

$$\begin{tikzcd}
\mathbb{R}^p \arrow[dd, "Y"'] & & \mathbb{R}^p \arrow[ll, "A(k)"'] \arrow[dl, "U(k)"'] \\
& \mathbb{R}^n \arrow[ul, "M"'] \arrow[dl, "N"'] & \\
\mathbb{R}^m & & \mathbb{R}^m \arrow[ll, "f"'] \arrow[ul, "v"'] \arrow[uu, "\psi"']
\end{tikzcd}$$

Figure 4.1: Commutative diagram of the different decompositions of $\dot{x} = f(x,k)$. (H.-M. Kaltenbach, personal communication.)

and one 1 element per column and all other elements are zero. It can thus be interpreted as the incidence matrix of a (directed) graph (cf. Definition 4.5).

On the other hand, the flux vector $v(x,k)$ can also be further decomposed as

$$v(x,k) = U(k)\psi(x) = \operatorname{diag}(k)U\psi(x).$$

To this end, one first builds the products $\psi(x) \in \mathbb{R}^p$ with

$$\psi_i(x) = x^{y_i} \qquad i = 1,\dots,p.$$

which are sometimes referred to as "complex potentials". Then, it is possible to find a matrix $U \in \{0,1\}^{n\times p}$ that "selects" the substrate complexes of the reactions. Note that the matrix U has exactly one non-zero entry per row. After multiplying with the rate constants k, the decomposition of $v(x,k)$ is complete.

For the scope of this thesis, the most detailed decomposition of the right hand side $f(x,k)$ of the ODE system governing the time evolution of the concentrations x is

$$f(x,k) = YM\operatorname{diag}(k)U\psi(x). \tag{4.5}$$

It is important to note that although $f(x,k)$ is non-linear in x, the only non-linearity occurring in the decomposition (4.5) is the function $\psi(\cdot)$. A large part of the results of CRNT is based on exactly this fact.

Example 4.2 (continued from Example 4.1)**.** Consider again example network (4.1). The complete decomposition of this network's equations is

$$Y = \begin{bmatrix} 1 & 2 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$$M = \begin{bmatrix} -1 & 1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

$$U = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

□

Since the stoichiometric matrix $N = YM$ plays a crucial role in the analysis of chemical reaction networks, the following two definitions give the network properties related to N.

Definition 4.2 (Stoichiometric Subspace; — Compatibility). *The* stoichiometric subspace *of a reaction network is defined as*

$$S \triangleq \text{range}(N).$$

Two concentration vectors x_1 and x_2 are said to be stoichiometrically compatible *if*

$$x_2 - x_1 \in S.$$

Definition 4.3 (Rank of Reaction Network). *The* rank *of a reaction network is defined as*

$$s \triangleq \text{rank}(N).$$

Example 4.3 (continued from Example 4.2). In our example network (4.1), the stoichiometric matrix is

$$N = \begin{bmatrix} 1 & -1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 1 & -1 \\ 0 & 0 & 1 & -1 & 1 \end{bmatrix}.$$

The rank of the network is therefore

$$s = \text{rank}(N) = 2.$$

□

4.3 More CRNT Concepts and Notation

The main goal of chemical reaction network theory is to predict the capability of a reaction network to have multiple positive steady states without making any assumptions on the rate constants k. To be able to do this, we first need a precise notion of what constitutes multistationarity in the sense of CRNT.

The following observation motivates the definition of multistationarity adopted in CRNT (other definitions are used, e.g., when discussing absolute concentration robustness [50]). Consider a trajectory $x(t)$ that is the solution of

$$\dot{x} = f(x, k)$$

with initial condition $x(0) = x_0$. Because of the special form of the right hand side

$$f(x, k) = Nv(x, k),$$

the trajectory $x(t)$ lies in an affine subspace $x_0 + S$ parallel to the stoichiometric subspace S. This fact leads to the following natural definition of multistationarity.

Definition 4.4 (Multistationarity, cf. [20]). *Consider a reaction network and associated ODEs as described in Eq. (4.4). We say that the network admits* multistationarity *if there exist at least two distinct, positive vectors x_1 and x_2 that are stoichiometrically compatible and a positive vector k such that*

$$f(x_1, k) = 0 \tag{4.6a}$$
$$f(x_2, k) = 0 \tag{4.6b}$$
$$x_2 - x_1 \in S. \tag{4.6c}$$

Since CRNT treats the reaction rates k as unknowns, the network topology becomes the crucial object for analyzing multistationarity properties of reaction networks. The following two definitions introduce the elements that characterize the topology of a network in the notation of CRNT.

Definition 4.5 (Network Graph). *The* network graph *of a chemical reaction network $(\mathcal{S}, \mathcal{C}, \mathcal{R})$ is the weighted, directed graph $(\mathcal{C}, \mathcal{R})$ with the complexes as vertices, the reactions as edges (with orientation from substrate to product complexes) and the rate constants as edge weights.*

Sometimes, it is convenient to consider an *undirected* network graph. In this case, we assume that each reversible reaction pair and irreversible reaction induces a single, undirected edge.

Definition 4.6 (Linkage Classes). *The (strongly) connected components of the network graph are called* (strong) linkage classes.

A strong linkage class and its complexes are called terminal *if there exists no reaction having a substrate complex within this class and a product complex outside of this class.*

Because of their outstanding importance in CRNT, the linkage classes and the terminal strong linkage classes are denoted as

$$\mathcal{L}_i, \quad i = 1, \dots, l \qquad \text{and} \qquad \mathcal{T}_i, \quad i = 1, \dots, t,$$

where l denotes the number of linkage classes and t the number terminal strong linkage classes.

Note that the connected components are a feature of the undirected network graph whereas the strongly connected components are defined with respect to the directed graph.

If, for a given network, the linkage classes and the terminal strong linkage classes coincide, the network is called *weakly reversible*.

Example 4.4 (continued from Example 4.3). The network graph of example network (4.1) is

$$x_1 \rightleftarrows 2x_1$$

$$x_1 + x_2 \longrightarrow x_3 \rightleftarrows x_2$$

This graph has $l = 2$ connected components (linkage classes)

$$\begin{aligned}\mathcal{L}_1 &= \{x_1, 2x_2\}\\ \mathcal{L}_2 &= \{x_1 + x_2, x_2, x_3\}.\end{aligned}$$

Furthermore, the network graph has three strongly connected components of which the following $t = 2$ are terminal

$$\begin{aligned}\mathcal{T}_1 &= \{x_1, 2x_2\}\\ \mathcal{T}_2 &= \{x_2, x_3\}.\end{aligned}$$

□

The central element for classifying reaction networks in CRNT is the *deficiency*. In this thesis, we adopt the following definition of this concept.

Definition 4.7 (Deficiency Space; Deficiency). *The* deficiency space *of a reaction network is the linear subspace given by*

$$G \mathrel{\hat{=}} \text{null}(Y) \cap \text{range}(M).$$

The deficiency δ *is the dimension of this subspace:*

$$\delta \mathrel{\hat{=}} \dim(G).$$

The deficiency space of a network is a subspace of $\mathbb{R}^p$ that appears after the multiplication of the fluxes $v(x, k)$ with M and before the multiplication with Y. In Fig. 4.1, this corresponds to the space in the upper left corner. The elements in this space are sometimes referred to as "complex potential changes".

Note that the definition of the deficiency depends only on the completely known network matrices Y and M. It is therefore possible to compute a basis of the deficiency space G and, hence, the network's deficiency without knowing the rate constants. For calculating only the deficiency of a network, it is not necessary to compute a full basis of the deficiency subspace. The following Lemma shows how.

Lemma 4.1 ([33], Proposition 5.1). *The deficiency δ satisfies the relation*

$$\delta = p - l - s, \tag{4.7}$$

where p is the number of complexes, l the number of linkage classes and s is the rank of the reaction network.

We do not give a proof of this basic lemma. But we remark that $p - l = \operatorname{rank}(M)$, which can be seen, for example, by recalling the properties of incidence matrices of directed graphs. Since we also have $s = \operatorname{rank}(YM)$ it is quite easy to see that the difference $p - l - s$ can never become negative (which it should not as δ is the dimension of a linear subspace).

Interestingly, in Feinberg's original publication [20], Eq. (4.7) is taken as the definition of the deficiency of a reaction network. Clearly, the motivation behind this is coming from considerations that led us to Definition 4.7. Other authors (e.g. Gunavardena [33]) define the deficiency as

$$\dim \left(Y \cap A(k) \right).$$

This, however, makes the deficiency a function of the rate constants k. We will come back to this particular subspace in Chapter 6.

For our purposes, Definition 4.7 is most suitable in that it allows to study multistationarity in reaction networks by applying tools from linear algebra more directly and without imposing additional assumptions on the network.

Example 4.5 (continued from Example 4.4). The deficiency space of example network (4.1) is

$$G = \operatorname{span} \begin{bmatrix} 1 \\ -1 \\ 1 \\ 0 \\ -1 \end{bmatrix}.$$

Therefore, the deficiency of the network is $\delta = 1$. This is consistent with Lemma 4.1 as

$$p - l - s = 5 - 2 - 2 = 1.$$

□

5

The Deficiency One Problem

THE DEFICIENCY ONE PROBLEM is a clever formulation of the question of multistationarity for a certain class of reaction networks with deficiency one called *regular* networks. For these networks, it can be regarded as a feasibility problem that asks for the existence of two concentration vectors $x_1, x_2 \in \mathbb{R}^m$ and a parameter vector $k \in \mathbb{R}^n$ that satisfy the conditions of multistationarity in Definition 4.4. In particular, we are looking for a feasible triplet $(x_1, x_2, k) \in \mathbb{R}^m \times \mathbb{R}^m \times \mathbb{R}^n$ such that

$$\begin{aligned} f(x_1, k) &= 0 \\ f(x_2, k) &= 0 \\ x_2 - x_1 &\in S \setminus \{0\} \\ x_1,\ x_2,\ k &> 0. \end{aligned}$$

Note that positivity of concentration and rate vectors are always component-wise. In the following, functions such as $\log(x)$, when applied to a vector x, are understood as component-wise application of the function to the vector elements. The same holds for “vector division” x/y when both x and y are vectors.

Assuming that the function f describes the kinetics of a mass-action network with deficiency one, one can derive a reformulation of the above problem that works around the nonlinearities in f and always finds a set of feasible vectors (x_1, x_2, k) or proves that no such vectors exist. The underlying idea is to derive feasibility conditions for two proxy quantities $z \in \mathbb{R}^m$ and $s \in \mathbb{R}^m$ defined as

$$\begin{aligned} z &\triangleq \log\left(\frac{x_2}{x_1}\right) \\ s &\triangleq x_2 - x_1, \end{aligned}$$

from which it is possible to recover a feasible set of vectors (x_1, x_2, k).

5.1 Regular Networks

The conditions of the deficiency one problem apply only to a certain subset of reaction networks with deficiency one. These networks are called *regular*. For a network to be regular it must fulfill three regularity conditions. This section will state these three conditions and briefly review their implications with respect to the formulation of the deficiency one problem.

(R1) The first condition is called *positive dependency*. This means that there must exist positive reaction rates v in the nullspace of the stoichiometric matrix, i.e.

$$\begin{aligned} Nv &= 0 \\ v &> 0. \end{aligned}$$

It is clear that this condition is necessary for a network to have *any* positive steady state. That is, if there exists no v satisfying this condition it is not possible to find a positive concentration vector x with $f(x,k) = 0$.

Testing for this condition can be done using linear programming. Since the constraints are homogenous, one could, for example, solve the following linear feasibility problem:

$$\begin{aligned} Nv &= 0 \\ v &\geq e, \end{aligned}$$

where $e \in \mathbb{R}^n$ is a vector of all ones.

The reason for having this condition is that for formulating some of the feasibility constraints of the deficiency one problem, one implicitly assumes that one can compute such a v. One could also opt to augment the feasibility constraints with this condition but since this is an easy to check, necessary precondition it was historically formulated as a regularity condition.

(R2) The definition of the second regularity condition involves the reaction graph of a network. Recall that l denotes the number of linkage classes and t the number of terminal strong linkage classes in the network graph. Clearly, each linkage class has *at least* one terminal component. Then, the condition is

$$t = l,$$

meaning that each linkage class must have *exactly* one terminal strong linkage class.

The point of this condition is related to the definition of multistationarity. It holds that

$$\operatorname{range}(A(k)) \subseteq \operatorname{range}(M)$$

with equality exactly when $t = l$ since the dimension of the nullspace of $A(k)$ is equal to the number of terminal strong linkage classes in a network (cf. [24]). Thus, if $t > l$ for a network with deficiency one, exactly one of the following two conditions

$$\operatorname{null}(Y) \cap \operatorname{range}(A(k)) = G \qquad \operatorname{range}(YA(k)) \subset S \tag{5.1a}$$

$$\operatorname{null}(Y) \cap \operatorname{range}(A(k)) = \{0\} \subset G \qquad \operatorname{range}(YA(k)) = S \tag{5.1b}$$

is true, which depends on the values of the rate constants k. This inherent dependency on the position of the rate constants k is not taken into account by the deficiency one problem. Such networks are thus excluded from consideration.

(R3) The last condition is called *cut-pair property*. It is defined on subgraphs of the network graph given by the terminal strong linkage classes $\mathcal{T}_i$, $i = 1, \ldots, t$. A terminal strong linkage class is said to have the cut-pair property if, when one removes any of its reversible reaction pairs or irreversible reactions, it becomes disconnected.

An immediate consequence of condition (R3) is that terminal strong linkage classes cannot have irreversible reactions. The importance of this condition with respect to the deficiency one problem is that the structure of the nullspace of the matrix M simplifies dramatically. Under (R3), the dimension of the nullspace of M is equal to the number of reversible reaction pairs in the terminal strong linkage classes and the nullspace basis vectors of M have support in the reactions of these pairs. In other words, any *complex balancing* flux v (i. e. a flux with $Mv = 0$) is *detailed balancing* in its terminal reactions (i. e. $v_i = v_j$ if (i, j) are a reversible reaction pair).

The reason for including this condition is that, although the constraints of the deficiency one problem are sufficient for all deficiency one networks, only for networks with the *cut-pair property* they are also necessary.

5.2 Conditions for Multistationarity

As already alluded in the introduction of this chapter, the goal is to find equivalent conditions for multistationarity of (x_1, x_2, k) in two proxy variables

$$\begin{aligned} z &= \log\left(x_2/x_1\right) \\ s &= x_2 - x_1. \end{aligned}$$

The reason for doing this is of course to get conditions that are computationally better tractable than the original system of nonlinear equations. In fact, it will turn out that the new conditions in (z, s) involve only linear (in-)equalities. However, the problem itself is not a linear programming problem since the *sense* of the inequalities is not fixed.

The intention of this section is to give a new derivation of the conditions for the deficiency one problem that allow a transformation into the framework of mixed integer linear programming. There are in total three kinds of conditions which will be derived in the following subsections.

Sign Compatibility Condition

The first set of feasibility conditions for the vectors z and s results from the problem of reconstructing two feasible concentration vectors x_1, x_2 from a given vector z. The difficulty lies in the condition that x_1 and x_2 must be stoichiometrically compatible. Hence, there must exist a vector $s = x_2 - x_1 \in S \setminus \{0\}$. Since $z = \log\left(x_2/x_1\right)$, we find that

$$s = x_2 - x_1 = x_1 \circ \left(\exp(z) - e\right), \tag{5.2}$$

where e is a conforming vector of all ones and "$\circ$" denotes the Hadamard product (i. e. element-wise multiplication). Since we want x_1 to be positive, we can choose $s \in S \setminus \{0\}$ only with the same sign structure as the expression $\exp(z) - e$. Since we have

$$\operatorname{sgn}(\exp(z) - e) = \operatorname{sgn}(z),$$

the condition can be propagated to z, i.e. we require

$$\operatorname{sgn}(s_i) = \operatorname{sgn}(z_i) \qquad \forall i = 1, \ldots, m. \tag{5.3}$$

Constraints (5.3) are related to step 8 of the deficiency one algorithm as presented in [21].

Complex Inequalities

The second set of constraints applies at the level of “complex potentials”. It is clear that if x is a stationary point then $\psi(x)$ must lie in the null space of $YA(k)$ (cf. Fig. 4.1). If a network exhibits multistationarity, we must be able to find two vectors $\psi(x_1)$ and $\psi(x_2)$ for which this is true. To see when this is possible one has to study the nullspace structure of $YA(k)$.

For regular deficiency one networks, the dimension of $\operatorname{null}(YA(k))$ is equal to $l+1$ ($= t+1$). The nullspace of the kinetic matrix $A(k)$ is known to be t-dimensional and decomposes into t orthogonal one-dimensional subspaces with support in the complexes of the terminal strong linkage classes [24, 27, 33]. That is, one can find positive vectors $b^{(1)}, \ldots, b^{(t)}$ with

$$\operatorname{null} A(k) = \operatorname{span}\left\{b^{(1)}, \ldots, b^{(t)}\right\} = \bigoplus_{i=1}^{t} \operatorname{span} b^{(i)}.$$

Thus, there exists exactly one additional vector $c \in \operatorname{null}(YA(k))$ with $A(k)c \neq 0$. It is also known that if one can find a steady-state concentration x_1 with $A(k)\psi(x_1) = 0$, the network cannot be multistationary in the sense of CRNT [33]. This fact allows to conclude that if x_1 is one of multiple stationary points, one can choose $c = \psi(x_1)$ as the additional basis vector of $\operatorname{null}(YA(k))$.

Since $\psi(x_2)$ must also lie in $\operatorname{null}(YA(k))$, it can be written as a linear combination of the basis vectors chosen above. A necessary condition for x_2 to be a stationary vector is therefore

$$\psi(x_2) = \alpha c + \lambda_1 b^{(1)} + \ldots + \lambda_t b^{(t)}. \tag{5.4}$$

To derive the final complex inequalities, we have to find another expression for the “complex potentials” $\psi(x_2)$ in terms of the concentrations x_1 and z. For each complex $i = 1, \ldots, p$ it holds that

$$\begin{aligned} \psi_i(x_2) &= x_2^{y_i} \\ &= \left(\frac{x_2}{x_1}\right)^{y_i} x_1^{y_i} \\ &= \exp\left(\log\left(\left(\frac{x_2}{x_1}\right)^{y_i}\right)\right) x_1^{y_i} \\ &= \exp\left(y_i^{\mathsf{T}} \log\left(\frac{x_2}{x_1}\right)\right) x_1^{y_i}. \end{aligned}$$

Using the definition $z = \log(x_2/x_1)$, we get

$$\psi_i(x_2) = \exp(y_i^{\mathsf{T}} z)\psi_i(x_1) \qquad i = 1, \ldots, p. \tag{5.5}$$

The conditions (5.4) and (5.5) can be combined into a system of linear inequalities for the vector z. The sense of the inequalities is determined by the signs of the coefficients $\lambda_1, \dots, \lambda_t$ (which is a degree of freedom in the feasibility problem). Looking at the complexes $i = 1, \dots, p$ individually, we have

$$\psi_i(x_2) = \exp(y_i^T z)c_i = \alpha c_i + \lambda_{\tau(i)} b_i^{(\tau(i))} \qquad \text{if } \lambda_{\tau(i)} \neq 0 \tag{5.6a}$$

$$\psi_i(x_2) = \exp(y_i^T z)c_i = \alpha c_i \qquad \text{if } \lambda_{\tau(i)} = 0 \text{ or } \tau(i) = \varnothing \tag{5.6b}$$

Here, we have introduced the notation $\tau(i)$ which associates a complex i with the index of its terminal strong linkage class (we write $\tau(i) = \varnothing$ if complex i is nonterminal, i.e. if there is no association). Note that each complex is affected by at most one of the basis vectors $b^{(i)}$ since they have disjunct support defined by the terminal strong linkage classes (as noted above).

Dividing each equality i by c_i and forgetting about the first equality yields

$$\exp(y_i^\mathsf{T} z) = \alpha + \lambda_{\tau(i)} \widetilde{b}_i^{(\tau(i))} \qquad \text{if } \lambda_{\tau(i)} \neq 0 \tag{5.7a}$$

$$\exp(y_i^\mathsf{T} z) = \alpha \qquad \text{if } \lambda_{\tau(i)} = 0 \text{ or } \tau(i) = \varnothing \tag{5.7b}$$

where we have made the substitution $\widetilde{b}_i^{(\tau(i))} = b_i^{(\tau(i))}/c_i$. Equation (5.7b) implies equality between the expression $\exp(y_i^\mathsf{T} z)$ and some arbitrary reference value α with $\alpha > 0$ (cf. Proposition A.1 in [23] for why α must always be larger than zero). Equation (5.7a) implies an *in*equality between the expressions $\exp(y_i^\mathsf{T} z)$ and α where the sense of the inequality depends on the sign choice of $\lambda_{\tau(i)}$.

Since $\exp(\cdot)$ is a strictly monotone function, these considerations directly apply to its argument, i.e. the expressions $y_i^\mathsf{T} z$. More precisely, we have the following system of linear equalities and inequalities (with $a = \log(\alpha)$):

$$y_i^\mathsf{T} z > a \qquad \text{if } \lambda_{\tau(i)} > 0 \tag{5.8a}$$

$$y_i^\mathsf{T} z = a \qquad \text{if } \lambda_{\tau(i)} = 0 \text{ or } \tau(i) = \varnothing \tag{5.8b}$$

$$y_i^\mathsf{T} z < a \qquad \text{if } \lambda_{\tau(i)} < 0 \tag{5.8c}$$

The inequality system (5.8) is equivalent to the conditions of steps 4 and 5 of the deficiency one algorithm as presented in [21].

Connection to Upper-Middle-Lower Partitions

The original formulation [21] of the algorithm eliminates the variable a from the system of inequalities (5.8) and yields what is called an *upper-middle-lower* partition of the complexes. To derive it, we partition the complexes into three sets U, M, L as follows:

$$i \in \begin{cases} U & \text{if } \lambda_{\tau(i)} > 0 \\ M & \text{if } \lambda_{\tau(i)} = 0 \\ L & \text{if } \lambda_{\tau(i)} < 0. \end{cases}$$

Then, we can derive from system (5.8) the following pairwise (in-)equalities between the expressions $y_i^\top z$ in the different sets U, M and L:

$$\begin{aligned} y_i^\top z &> y_j^\top z && \text{if } i \in U,\ j \in M \\ y_i^\top z &= y_j^\top z && \text{if } i \in M,\ j \in M \\ y_i^\top z &< y_j^\top z && \text{if } i \in L,\ j \in M \end{aligned}$$

Although this is mathematically appealing, it is an inefficient transformation from a computational point of view. In particular, it is cumbersome to model set membership and pairwise inequalities in a mixed integer programming framework. It is much easier to carry over the value a in Eq. (5.8) as an additional degree of freedom.

Flux Inequalities

The last set of constraints is necessary to ensure the existence of positive fluxes when reconstructing two steady-state concentrations x_1, x_2 from z and s. To derive these constraints, it is necessary to study the nullspace structure of the matrix M. Assume we are given a fixed $g \in G$, a basis vector of the deficiency subspace. The set of feasible fluxes v with $Mv = g$ depends on the structure of the nullspace of M, i.e. it can be parametrized as

$$v = \hat{v} + w,$$

where $\hat{v}$ is an arbitrary "test flux" with $M\hat{v} = g$ and $w \in \operatorname{null}(M)$. Note that since w lies in the nullspace of M, it is a complex-balancing flux. The crucial observation is that for reaction networks with the cut-pair property, i.e. those satisfying regularity condition (R3), the flux w is not just complex-balancing but in fact detailed-balancing in its terminal reactions. In other words, for reactions in terminal strong linkage classes the flux through irreversible reactions and the net flux through reversible reaction pairs is uniquely determined by the choice of $g \in G$. The only degree of freedom for v is the "futile cycling" in its reversible reaction pairs. This means that if reactions (i, j) are a reversible reaction pair, the components w_i, w_j of a vector $w \in \operatorname{null}(M)$ must be equal, i.e. $w_i = w_j$.

This observation leads to an additional set of inequalities for the vector z. The derivation is based on the fact that the fluxes $v(x_2, k)$ can be written in terms of the concentrations x_1 and z in analogy to Eq. (5.4) for the complex potentials. The trick for doing this is to see that the product $U \operatorname{diag}(\exp(Y^\top z))$ "commutes" in the sense that

$$U \operatorname{diag}(\exp(Y^\top z)) = \operatorname{diag}(\exp(UY^\top z))U.$$

We thus get for $v(x_2, k)$:

$$\begin{aligned} v(x_2, k) &= \operatorname{diag}(k) U \psi(x_2) && (5.9a) \\ &= \operatorname{diag}(k) U \operatorname{diag}(\exp(Y^\top z)) \psi(x_1) && (5.9b) \\ &= \operatorname{diag}(\exp(UY^\top z)) \operatorname{diag}(k) U \psi(x_1) && (5.9c) \\ &= \operatorname{diag}(\exp(UY^\top z)) v(x_1, k). && (5.9d) \end{aligned}$$

Assume that $Mv(x_1, k) = g \in G$. Then, the following holds for $v(x_2, k)$:

$$\begin{aligned} Mv(x_2, k) &= M \operatorname{diag}(\exp(UY^\top z))v(x_1, k) \\ &= \alpha g \\ &= M\alpha v(x_1, k). \end{aligned}$$

The second equality above follows from (5.4). By equating the first and last line, we get

$$\operatorname{diag}(\exp(UY^\top z))v(x_1, k) - \alpha v(x_1, k) \in \operatorname{null}(M). \tag{5.10}$$

This is the critical condition based on which one can formulate the flux inequalities.

First, note that for non-terminal reactions (i. e. reactions whose substrate complex is non-terminal), condition (5.10) is fulfilled trivially if (z, α) satisfy the complex inequalities. For a non-terminal reaction i it holds that

$$\exp(y_{s(i)}^\top z)v_i(x_1, k) - \alpha v_i(x_1, k) = \left(\exp(y_{s(j)}^\top z) - \alpha\right) v_i(x_1, k) = 0.$$

Therefore, we only need to consider reactions in terminal strong linkage classes. From the discussion above we know that for the expression (5.10) to lie in the nullspace of M it must be detailed-balancing in its terminal reactions. Assume that $\hat{v}_i$, $\hat{v}_j$ are the forward and backward fluxes of a reversible reaction pair of an arbitrary flux distribution $\hat{v}$ with $M\hat{v} = g \in G$. Since the net flux through reactions (i, j) is fixed by the choice of g, we can parametrize all feasible fluxes with a parameter β as $\hat{v}_i + \beta$ and $\hat{v}_j + \beta$. Since the expression in Eq. (5.10) lies in the nullspace of M, its ith and jth components must be equal, i. e. we get

$$\exp(y_{s(i)}^\top z)(\hat{v}_i + \beta) - \alpha(\hat{v}_i + \beta) = \exp(y_{s(j)}^\top z)(\hat{v}_j + \beta) - \alpha(\hat{v}_j + \beta)$$

or equivalently

$$\left(\alpha - \exp(y_{s(i)}^\top z)\right)(\hat{v}_i + \beta) = \left(\alpha - \exp(y_{s(j)}^\top z)\right)(\hat{v}_j + \beta). \tag{5.11}$$

We need to study when conditions (5.11) have a solution with $\hat{v}_i + \beta$ and $\hat{v}_j + \beta$ positive. To do this, note first that since (i, j) are a terminal reversible reaction pair, they belong to the same terminal strong linkage class, say τ. For (z, α) satisfying the complex inequalities we know that the sign of the factors $\alpha - \exp(y_{s(i)}^\top z)$ and $\alpha - \exp(y_{s(j)}^\top z)$ must be the same and depends on the choice of the sign of λ_τ (the multiplier of the basis vector $b^{(\tau)} \in \operatorname{null}(A(k))$ from Eq. (5.4)). Therefore, finding positive factors $\hat{v}_i + \beta$ and $\hat{v}_j + \beta$ for having equality in Eq. (5.11) is possible if and only if the following set of conditions holds:

$$\begin{aligned} \alpha - \exp(y_{s(i)}^\top z) &= \alpha - \exp(y_{s(j)}^\top z) && \text{if } \hat{v}_i = \hat{v}_j \text{ or } \lambda_\tau = 0 \\ \alpha - \exp(y_{s(i)}^\top z) &> \alpha - \exp(y_{s(j)}^\top z) && \text{if } \hat{v}_i > \hat{v}_j,\ \lambda_\tau > 0 \text{ or } \hat{v}_i < \hat{v}_j,\ \lambda_\tau < 0 \\ \alpha - \exp(y_{s(i)}^\top z) &< \alpha - \exp(y_{s(j)}^\top z) && \text{if } \hat{v}_i > \hat{v}_j,\ \lambda_\tau < 0 \text{ or } \hat{v}_i < \hat{v}_j,\ \lambda_\tau > 0 \end{aligned}$$

After eliminating α and using monotonicity of the function $\exp(\cdot)$, this simplifies to

$$
\begin{aligned}
& y_{s(i)}^{\mathsf{T}} z = y_{s(j)}^{\mathsf{T}} z && \text{if } \hat{v}_i = \hat{v}_j \text{ or } \lambda_\tau = 0 && (5.12a)\\
& y_{s(i)}^{\mathsf{T}} z < y_{s(j)}^{\mathsf{T}} z && \text{if } \hat{v}_i > \hat{v}_j,\ \lambda_\tau > 0 \text{ or } \hat{v}_i < \hat{v}_j,\ \lambda_\tau < 0 && (5.12b)\\
& y_{s(i)}^{\mathsf{T}} z > y_{s(j)}^{\mathsf{T}} z && \text{if } \hat{v}_i > \hat{v}_j,\ \lambda_\tau < 0 \text{ or } \hat{v}_i < \hat{v}_j,\ \lambda_\tau > 0 && (5.12c)
\end{aligned}
$$

Assume that we have ordered all reversible reaction pairs (i, j) such that $\hat{v}_i > \hat{v}_j$ for some $\hat{v} > 0$ with $M\hat{v} \in G$. If the reaction network contains an irreversible reaction, this ordering is fixed, i.e. there is no flux $\hat{v}' > 0$ with $M\hat{v}' \in G$ and $\hat{v}'_i < \hat{v}'_j$. If the network contains only reversible reactions, however, we can find such a flux $\hat{v}'$. This is true because we can find β's for each reaction pair such that $-\hat{v}_{i,j} + \beta > 0$. Hence, the case where $\hat{v}_i < \hat{v}_j$ in (5.12) has to be considered only for reversible networks.

Note that this "switch" between the two regimes $\hat{v}_i > \hat{v}_j$ and $\hat{v}_i < \hat{v}_j$ happens for all reaction pairs simultaneously, not for each pair individually. This is because the relation between $\hat{v}_i$ and $\hat{v}_j$ is determined by the vector $g = M\hat{v}$. If we choose a g' pointing in the opposite direction of g, we will find flux vectors $\hat{v}'$ with all relations inverted.

The inequalities of system (5.12) are called the "preliminary inequalities" in step 2 in the deficiency one algorithm as presented in [21].

Example 5.1 (Network (4.1)). As an example to illustrate the construction of inequalities for a deficiency one network, we take again the reaction network (4.1) from the previous chapter. For convenience of the reader, we reproduce the network here:

$$
\begin{aligned}
x_1 &\underset{k_2}{\overset{k_1}{\rightleftarrows}} 2x_1\\
x_1 + x_2 &\xrightarrow{k_3} x_3\\
x_3 &\underset{k_5}{\overset{k_4}{\rightleftarrows}} x_2
\end{aligned}
$$

We start with the complex inequalities. Recall that the network has two terminal strong linkage classes (cf. Example 4.4). Therefore, we have to choose a "reference value" a and the sign of two multipliers λ_1, λ_2. For example, we can choose

$$
\begin{aligned}
a &= 2.5\\
\operatorname{sgn}(\lambda_1) &= 1\\
\operatorname{sgn}(\lambda_2) &= -1.
\end{aligned}
$$

This leads to the following set of (in-)equalities for z:

$$\begin{aligned} y_1^{\mathsf{T}} z &> a & \qquad & & z_1 &> a \\ y_2^{\mathsf{T}} z &> a & & & 2z_1 &> a \\ y_3^{\mathsf{T}} z &= a & & \text{or in components} & z_1 + z_2 &= a \\ y_4^{\mathsf{T}} z &< a & & & z_3 &< a \\ y_5^{\mathsf{T}} z &< a & & & z_2 &< a \end{aligned}$$

For the flux inequalities, we need to fix a “test flux” $\hat{v}$ such that $M\hat{v} = g \in G$. For example, we can choose

$$\hat{v} = (3, 1, 2, 3, 1)^{\mathsf{T}}$$

leading to

$$M\hat{v} = (-2, 2, -2, 0, 2)^{\mathsf{T}}$$

which lies clearly in the deficiency subspace. For the reversible reaction pair (1, 2) we have $\hat{v}_1 > \hat{v}_2$. These reactions are associated with multiplier $\lambda_1 > 0$. Thus, condition (5.12b) applies, i. e.

$$y_1^{\mathsf{T}} z < y_2^{\mathsf{T}} z.$$

The second reversible reaction pair is (4, 5) where we have $\hat{v}_4 > \hat{v}_5$. But since the multiplier $\lambda_2 < 0$, in this case condition (5.12c) applies, i. e.

$$y_4^{\mathsf{T}} z > y_5^{\mathsf{T}} z.$$

All in all, we get the following (in-)equalities for the vector z:

$$y_2^{\mathsf{T}} z > y_1^{\mathsf{T}} z > y_3^{\mathsf{T}} z = a > y_4^{\mathsf{T}} z > y_5^{\mathsf{T}} z$$

that are satisfied, for example, by

$$z = (3.75, -1.25, 1)^{\mathsf{T}}.$$

The last step consists of checking the sign compatibility. To this end, we have to find an $s \in S$ that is sign compatible with z. In the present case this is possible. For example, choosing

$$s = (5, -1, 1)^{\mathsf{T}}$$

has the same sign structure as z and lies in range(N). Therefore, we can conclude that network (4.1) has the capability of admitting multiple steady states for certain rate constants k. Indeed, we can reconstruct a pair (x_1, x_2) as

$$\begin{aligned} x_1 &= s/(\exp(z) - 1) \approx (0.1204, 1.4016, 0.5820)^{\mathsf{T}} \\ x_2 &= x_1 + s \approx (5.1204, 0.4016, 1.5820)^{\mathsf{T}}. \end{aligned}$$

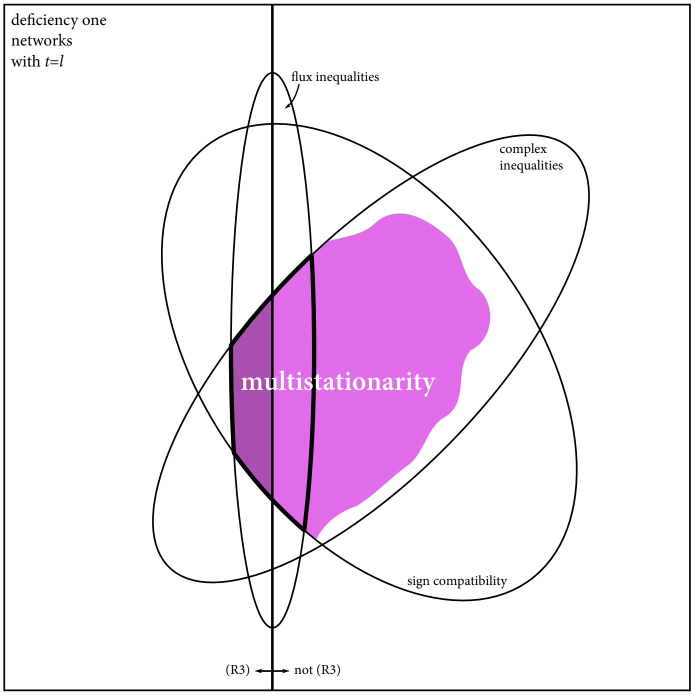


Figure 5.1: Illustration of how the three types of constraints relate to multistationarity in deficiency one networks. The pink area is the set of all multistationary networks. The sign compatibility conditions and the complex inequalities are both necessary conditions. The flux inequalities make the system sufficient (bold-bordered area). For all networks for which regularity condition (R3) holds, the conditions are also necessary (dark pink area).

Note that it could happen that $s_i = z_i = 0$ for some components of the vectors z and s. Then, the quotient $s_i/(\exp(z_i) - 1)$ is not defined. But in this case, one can choose an arbitrary positive value for $x_{i,1}$ (see also Eq. (5.2)). Then, one can find rate constants

$$k \approx (0.6190, 0.0868, 0.4342, 0.6160, 0.2035)^{\mathsf{T}}$$

that yield

$$f(x_1, k) = f(x_2, k) = 0.$$

□

Necessity and Sufficiency of the Conditions

The complete constraint system resulting from the reformulation of the deficiency one problem consists of three types of conditions: the sign compatibility conditions (5.3), complex inequalities (5.8) and flux inequalities (5.12). It is instructive to discuss necessity and sufficiency of these conditions with respect to the regularity of a deficiency one network. See Figure 5.1 for an illustration of the following arguments.

Clearly, feasibility of all three conditions together is sufficient for establishing multistationarity in a deficiency one network. The sign compatibility condition as well as the complex inequalities are both necessary. The sign compatibility condition ensures that the concentration vectors x_1, x_2 reconstructed from z and s are both positive. The complex inequalities are required for finding a common set of rate constants k for both steady-state concentrations x_1 and x_2.

The most interesting set of constraints are the flux inequalities. They basically state that the expression

$$\operatorname{diag}(\exp(UY^\top z)) \cdot v(x_1, k) - \alpha \cdot v(x_1, k)$$

must be a detailed-balancing flux. Together with the other two types of conditions, this is for general deficiency one networks only a sufficient condition for having multistationarity. Only when the network satisfies regularity condition (R3), the conditions are also necessary.

5.3 MILP Reformulation

In the previous section, we have derived feasibility conditions for vectors z and s that are mostly linear but the sense of the linear (in-)equalities is an additional degree of freedom. Furthermore, they rely on some sign conditions between the elements of z and s. Here, we show that it is possible to formulate these conditions as a mixed integer linear program. As such, the feasibility problems become amenable to all the efficient solution strategies developed for these kinds of problems.

Modeling Preliminaries

The most critical problem in modeling the constraints from the deficiency one problem using the tools from integer and linear programming are the strict inequalities. Numerical algorithms are only able to solve linear programs with equality and non-strict inequality constraints. It is thus not possible to have strict inequalities in a MILP formulation directly. One has to aim at deriving an *equivalent* problem that fits the framework of integer programming.

The relevant constraints (5.3), (5.8) and (5.12) allow to find such an equivalent formulation since they are homogenous. That is, if (z, s) are a feasible solution, scaled versions $(\beta \cdot z, \beta \cdot s)$ are also feasible for all $\beta > 0$. For a variable $x \in \mathbb{R}$, this allows to rewrite sign-conditions of the form

$$\left\{ \begin{array}{l} x < 0 \\ x = 0 \\ x > 0 \end{array} \right\} \quad \text{as} \quad \left\{ \begin{array}{l} x \leq -\varepsilon \\ x = \ \ 0 \\ x \geq \ \ \varepsilon \end{array} \right\}$$

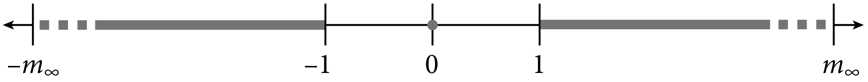


Figure 5.2: Disjoint domain used for modeling the sign conditions.

for some $\varepsilon > 0$ in the sense that if the first set of constraints is feasible the second is so as well. Note that this only holds because one can rescale the variables of homogenous constraints. For practical proposes, one often chooses $\varepsilon = 1$ (which we will do in the remainder of this section).

In essence, what this modeling approach does is to transform the original domain $\mathbb{R}$ of the variable x with sign constraints

$$\left\{ \begin{array}{c} x < 0 \\ x = 0 \\ x > 0 \end{array} \right\}$$

into a disconnected domain

$$\{x \leq -1\} \cup \{x = 0\} \cup \{x \geq 1\}.$$

Unfortunately, it is not possible to model such a disconnected domain that is unbounded in the $\pm\infty$ directions by using binary variables and linear constraints. The best one can do is an approximation of the form

$$\{-m_\infty \leq x \leq -1\} \cup \{x = 0\} \cup \{m_\infty \geq x \geq 1\}$$

which is depicted in Fig. 5.2. The value m_∞ is known as “big M” and represents an arbitrary large but finite value. The restriction of x to one of the three feasible intervals is achieved by introducing two binary auxiliary variables u_1, $u_2 \in \{0,1\}$. Then, we can write

$$x - m_\infty u_1 + u_2 \leq 0 \tag{5.13a}$$

$$x - u_1 + m_\infty u_2 \geq 0 \tag{5.13b}$$

$$u_1 + u_2 \leq 1 \tag{5.13c}$$

which yields the desired result. The possible assignments for the variables u_1, u_2 and the resulting effects for x are listed in Table 5.1.

The problems with such a “big M” approximation are twofold. The first is of conceptual nature. On one hand it is always possible to find an m_∞ that is large enough for a given problem instance such that the approximation and the original problem are equivalent. On the other hand, however, for any fixed m_∞ it is usually possible to find a problem instance where the approximation is infeasible but the original problem is not. The second problem is more of practical nature. A large value for m_∞ yields a high condition number of the constraint matrix, which might result in numerical difficulties when solving the problem.

In practice, these issues with the “big M” approximation are not a problem in the context of the deficiency one problem. An intuitive explanation for this is that we will use such “big M” inequalities

Table 5.1: Truth table of the auxiliary binary variables u_1, u_2 used to model the sign of the continuous variable x as well as the resulting lower (lb) and upper (ub) bounds of x. Note that the assignment $u_1 = u_2 = 1$ is not a feasible assignment.

x	u_1	u_2	lb	ub
< 0	0	1	$-m_\infty$	-1
$= 0$	0	0	0	0
> 0	1	0	1	m_∞

to enforce conditions on the variable z which is the logarithm of the concentration vectors. If m_∞ is chosen too small, it would mean that there exists no feasible solution x_1, x_2 with

$$\frac{x_1}{x_2} < e^{m_\infty}.$$

From this point of view, one could expect the algorithm to work for fairly small values of m_∞ when applied to realistic, non-pathological networks. For example, all test networks from Table 5.3 succeed with a value of m_∞ as small as 10. An interesting open problem would be to give a lower bound on m_∞ depending on the network data Y, M such that the feasibility problem is guaranteed to give the correct answer.

Stoichiometric & Sign Compatibility

The condition that s has to lie in the stoichiometric subspace S is encoded using a matrix $W \in \mathbb{R}^{(m-s)\times m}$, whose rows span $S^\perp$, the orthogonal complement of the stoichiometric subspace. Then, we have

$$Ws = 0. \tag{5.14}$$

The more complicated condition is the sign compatibility (5.3), which can be captured using two vectors of binary variables $u_1, u_2 \in \{0,1\}^m$. We use them to enforce the sign condition as in Eqns. (5.13) simultaneously on z and s. We thus have

$$z - m_\infty u_1 + u_2 \leq 0 \tag{5.15a}$$
$$z - u_1 + m_\infty u_2 \geq 0 \tag{5.15b}$$
$$s - m_\infty u_1 + u_2 \leq 0 \tag{5.15c}$$
$$s - u_1 + m_\infty u_2 \geq 0 \tag{5.15d}$$
$$u_1 + u_2 \leq e \tag{5.15e}$$
$$e^\mathsf{T}(u_1 + u_2) \geq 1 \tag{5.15f}$$

The last constraint (5.15f) is needed to enforce a non-zero solution. Otherwise, one could assign $z = s = 0$ which would be feasible irrespective of multistationarity of the reaction network.

Complex Inequalities

For modeling the possible sign choices of the variables $\lambda_1, \dots, \lambda_t$ in condition (5.8), we introduce two more vectors with t binary variables v_1 and v_2 and use again the technique described in Eqns. (5.13) to couple the sign of λ_i with the values of $v_{i,1}$ and $v_{i,2}$. For associating the terminal strong linkage classes with their complexes, we define a matrix $T \in \mathbb{R}^{p \times t}$ with

$$(T)_{ij} = \begin{cases} 0 & \text{if } \tau(i) = \varnothing \text{ or } \tau(i) \neq j \\ 1 & \text{if } \tau(i) = j \end{cases}$$

This allows the constraints (5.8) to be written as

$$Y^{\mathsf{T}} z - ae - m_\infty T v_1 + T v_2 \leq 0 \tag{5.16a}$$

$$Y^{\mathsf{T}} z - ae - T v_1 + m_\infty T v_2 \geq 0 \tag{5.16b}$$

$$v_1 + v_2 \leq e \tag{5.16c}$$

$$e^{\mathsf{T}} (v_1 + v_2) \geq 1 \tag{5.16d}$$

$$v_1, v_2 \in \{0, 1\}^t \tag{5.16e}$$

Note that constraint (5.16d) is needed to make sure that not all λ_i, $i = 1, \dots, t$, are zero at the same time. Otherwise, $z = 0$, $a = 0$ would be a feasible solution and therefore feasibility of the inequality system (5.16) would say nothing about multistationarity in the underlying reaction network.

Flux Inequalities

The flux inequalities are the most complicated constraints to be modeled in a MILP framework. This is because if the network is reversible, the sense of the inequality signs depends on two preconditions. One is the sign of multipliers λ_i and the other is the direction of the deficiency vector $g \in G$.

For modeling the flux inequalities (5.12) we assume that we are given a feasible flux vector $\hat{v} \in \mathbb{R}^n$ with $\hat{v} > 0$ and $M\hat{v} \in G$. Such a vector could be computed, for example, by solving an auxiliary linear program (see also the discussion of regularity condition (R1)). We further assume that all reversible reaction pairs (i, j) in terminal strong linkage classes are ordered such that $\hat{v}_i \geq \hat{v}_j$ (for the given flux $\hat{v}$).

If the relation of the fluxes $\hat{v}_i$, $\hat{v}_j$ would be fixed (as is the case for irreversible networks), the constraints (5.12) could be written as

$$\begin{aligned} (y_{s(i)} - y_{s(j)})^{\mathsf{T}} z &= 0 && \text{if } \hat{v}_i = \hat{v}_j \\ (y_{s(i)} - y_{s(j)})^{\mathsf{T}} z - m_\infty v_{\tau,1} + v_{\tau,2} &\leq 0 && \text{if } \hat{v}_i > \hat{v}_j \\ (y_{s(i)} - y_{s(j)})^{\mathsf{T}} z - v_{\tau,1} + m_\infty v_{\tau,2} &\geq 0 && \text{if } \hat{v}_i > \hat{v}_j \end{aligned}$$

where we have used τ to denote the index of the terminal strong linkage class of the reaction pair (i, j). For reversible networks this does not work, however. In this case, $w_i \leq w_j$ is also possible, which could be modeled by interchanging the roles of the binary vectors v_1 and v_2 in the above

Table 5.2: Truth table of the binary variables v_1, v_2, v_s and the resulting lower and upper bounds of p_1. The columns "(a)" and "(b)" denote the bounds resulting from constraints (5.18a) and (5.18b). The columns "max" and "min" are the effective lower and upper bounds, respectively.

			p_1 lb			p_1 ub		
v_1	v_2	v_s	(a)	(b)	max	(a)	(b)	min
1	0	0	1	−1	1	1	1	1
0	1	0	0	0	0	0	2	0
0	0	0	0	−1	0	0	1	0
0	0	1	−1	0	0	1	0	0
1	0	1	0	0	0	2	0	0
0	1	1	−1	1	1	1	1	1

equations. To represent both cases in the same problem instance, we need a new binary variable v_s that determines in which regime we are (i.e. whether $\hat{v}_i \geq \hat{v}_j$ or $\hat{v}_i \leq \hat{v}_j$). We further need auxiliary variables $p_1, p_2 \in \mathbb{R}^t$ that take on the roles of v_1, v_2 if $v_s = 0$ or v_2, v_1 if $v_s = 1$. That is, we write the constraints as

$$\begin{aligned}
(y_{s(i)} - y_{s(j)})^\mathsf{T} z &= 0 && \text{if } \hat{v}_i = \hat{v}_j && (5.17\text{a}) \\
(y_{s(i)} - y_{s(j)})^\mathsf{T} z - m_\infty p_{\tau,1} + p_{\tau,2} &\leq 0 && \text{if } \hat{v}_i \neq \hat{v}_j && (5.17\text{b}) \\
(y_{s(i)} - y_{s(j)})^\mathsf{T} z - p_{\tau,1} + m_\infty p_{\tau,2} &\geq 0 && \text{if } \hat{v}_i \neq \hat{v}_j && (5.17\text{c})
\end{aligned}$$

The variables p_1 and p_2 obey the following conditions

$$\begin{aligned}
v_1 - v_s \leq p_1 &\leq v_1 + v_s & v_2 - v_s \leq p_2 &\leq v_2 + v_s && (5.18\text{a}) \\
v_2 - (1 - v_s) \leq p_1 &\leq v_2 + (1 - v_s) & v_1 - (1 - v_s) \leq p_2 &\leq v_1 + (1 - v_s) && (5.18\text{b})
\end{aligned}$$

That these constraints indeed yield the desired behavior of the variables p_1, p_2 is shown in Table 5.2 (only for p_1). The important point to note is that p_1 and p_2 are fixed to integer values for all feasible values of v_1, v_2, v_s. It is therefore not necessary to declare them as binary variables.

Another important point to note is that if the network is irreversible, the "switching variable" v_s must be fixed to zero.

5.4 Numerical Computations

To demonstrate the versatility of the approach of reformulating the deficiency one problem as a mixed integer linear program, we solve the example networks from the ERNEST toolbox [54]. The toolbox contains in total 33 example networks collected from various references of which 18 networks have deficiency one. We solve the deficiency one problems from these 18 networks both with the toolbox and a state-of-the-art MILP solver using our reformulation and compare the solution times.

For solving the integer programs, we have used the software package Gurobi [31] which is free for academic purposes. The test computer used for the comparison has an AMD Athlon X2 processor with 2.8 GHz and 4 GB of RAM running a 64 bit Linux operating system. The results are summarized in Table 5.3.

The table reveals that the algorithm implemented in the ERNEST toolbox has most difficulties in proving infeasibility of the constraints since it uses a brute-force method to enumerate and test all combinations of sign conditions. Even for moderately sized reaction networks, solution times can exceed 20 minutes. On the other hand, the MILP solver is able to solve all problems quickly and without any problems. In fact, the solution times indicated in Table 5.3 are mostly dominated by program startup overhead. The actual time spent for solving the problem is always below 0.05 seconds. This is confirmed by the low number of simplex iterations the solver needs to find a feasible solution of prove infeasibility. Surprisingly, more than half of the problems can be solved by the solver's heuristics and *no* simplex iterations are necessary. This clearly demonstrates the power of the reformulation technique which results in problems that are "easy" for state-of-the-art MILP solvers.

5.5 Discussion & Perspectives

As the examples solved in the previous section demonstrate, the reformulation of the feasibility conditions as a mixed-integer linear program gives clear computational advantages compared to brute-force enumeration of all combinations. However, the tested networks are all relatively small examples and it would be interesting to run our reformulation on larger networks. The problem with larger networks is that they often have larger deficiency and thus the techniques from this chapter do not apply.

Currently, two methods show potential for scaling the analysis of deficiency one networks to larger networks. The first consists of restricting the analysis to subnetworks that are constructed in a particular way involving elementary flux modes. We address certain aspects of this idea related to the deficiency one problem in the following chapter (cf. § 6.2). The second method applies to a certain class of networks with higher deficiency that is decomposable in the coordinates of the reactions into deficiency one parts [17].

Another open problem is the choice of the constant m_∞ as was already mentioned in the introduction of the MILP reformulation. Although it seems not to be a problem for practical networks, choosing m_∞ *a posteriori* based on the network matrices Y and M would still be a final safeguard for the correctness of our MILP approach.

Table 5.3: Example networks from the ERNEST toolbox and solution times using the Gurobi 4.5.1 MILP solver and the ERNEST toolbox (running in Matlab 2009b). The "size" columns denote the number of species (m), complexes (p) and reactions (n) of the reaction network. The column "m/s" indicates whether the network can have multistationarity or not. The "time" column of the MILP solver includes the actual time spent in the solver (in parenthesis) since the overall time is largely dominated by program startup overhead.

		size				MILP		ERNEST	
	network	m	p	n	m/s	time [s]	iter	time [s]	ref
1	single phosphorylation	6	6	6	n	0.50 (0.01)	96	4.44	[2]
2	EGF pathway (RKIP network)	11	11	11	n	0.53 (0.04)	354	1252.81	[2]
3	(De-)phosphorylation mechanism 2	9	9	11	n	0.50 (0.02)	121	154.00	[13]
4	(De-)phosphorylation mechanism 3	8	8	10	n	0.50 (0.02)	125	46.04	[13]
5	Elementary enzyme catalysis (1)	4	6	6	n	0.49 (0.00)	0	1.01	[14]
6	Elementary enzyme catalysis (2)	6	9	10	n	0.49 (0.00)	0	10.23	[14]
7	Elementary enzyme catalysis (3)	6	9	10	n	0.49 (0.00)	0	10.39	[14]
8	Elementary enzyme catalysis (5)	6	9	10	n	0.48 (0.00)	0	10.82	[14]
9	Edelstein biochemical reactor	3	5	6	y	0.49 (0.00)	0	0.21	[20]
10	Example network (1.3)	2	5	6	n	0.50 (0.02)	138	0.37	[21]
11	Example network (4.1)	3	5	5	y	0.50 (0.00)	23	0.17	[21]
12	Example network (4.24)	2	4	2	y	0.49 (0.00)	0	0.14	[21]
13	Example network (4.3)	2	3	3	n	0.48 (0.00)	0	0.00 [a]	[22]
14	Example network (2.6)	7	9	13	?	0.49 (0.00)	0	0.00 [b]	[23]
15	Example network (3.1)	6	10	10	y	0.50 (0.00)	0	11.39	[23]
16	Example network 6 (Tbl. 1)	2	5	6	n	0.51 (0.02)	159	0.19	[45]
17	Example network 7 (Tbl. 1)	2	5	6	y	0.49 (0.00)	0	0.09	[45]
18	Example network 8 (Tbl. 1)	2	5	6	n	0.51 (0.02)	201	0.19	[45]

a. DF1 algorithm was not run since DF1 theorem applies.

b. DF1 algorithm was not run since (R3) is violated.

6

Analysis of Degenerate Reaction Networks

Chemical reaction networks arising in biochemical applications often fail to satisfy the regularity conditions stated in Section 5.1. Interestingly, it is often the second condition that is violated. Already in the early publications [21, 23] it was argued that such networks have "strange" properties. This chapter is an attempt to put these arguments on a more solid mathematical foundation and propose some ideas on how to proceed if such a network emerges in an application. The following definition gives these networks a proper name.

Definition 6.1. *A reaction network violating regularity condition (R2) is called* degenerate. *That is, a* degenerate *network has at least one linkage class with more than one terminal strong linkage class.*

An important motivation for doing this analysis comes from the EFM decomposition of reaction networks [12]. This is a method of decomposing networks with high deficiency into subnetworks with deficiency zero or one. In this process, it often happens that the resulting subnetworks are degenerate. Some of the material in this chapter that concerns degeneracy of EFM subnetworks is discussed in [59].

6.1 Generic Network Rank and Deficiency

The main idea for the discussion of degenerate reaction networks is to use the notion of a *generic* property. The following definition (adapted from [53]) makes precise what this means.

Definition 6.2 (Generic). *A property P is said to hold* generically *on a set X if the set of points V on which the property does not hold has measure zero.*

Furthermore, the points $X \setminus V$ on which the property holds are called generic points *(with respect to property P).*

An example is in order to illustrate this concept.

Example 6.1 ([53]). Consider the function $f(x_1, x_2) = x_1 x_2$ and the set $X = \mathbb{R}^2$. The property $f(x_1, x_2) \neq 0$ does hold generically on X since the set $V = \{x_1 = 0 \vee x_2 = 0\}$ for which $f(x_1, x_2) = 0$ has measure zero. $\square$

The concept of a generic property can be applied to generalize the definitions of network rank and deficiency. Recall the discussion of regularity condition (R2), where it was already indicated that the definition of the deficiency δ is of limited use for degenerate networks because they have

$$\operatorname{range}(Y \cdot A(k)) \subseteq \operatorname{range}(N),$$

whereas for non-degenerate networks these subspaces always coincide. For degenerate networks, then, the object of interest becomes the "kinetic subspace"

$$\operatorname{range}(Y \cdot A(k)) \subseteq S$$

and – in analogy to the deficiency space G – the subspace $\operatorname{null}(Y) \cap \operatorname{range}(A(k))$. In fact, one could argue that these are always the objects of interest, but for the special case of non-degenerate networks, these two subspaces are identical to the stoichiometric subspace and the deficiency subspace. For degenerate networks, however, they depend on the parameter vector k such that they are not an invariant of the network topology any more.

The properties we are interested in are the dimensions of these subspaces. Since their orientation depends on the rate constants k, their dimension is also a function of k. Recall that the rank of $A(k)$ equals $p - t$ where p is the number of complexes in the network and t is the dimension of its null space (equal to the number of terminal strong linkage classes [33]) and that the rank of the network is $\operatorname{rank}(N) = s$. A simple counting argument shows that

$$\begin{aligned} \dim[\operatorname{range} YA(k)] &\in \{\max(s-(t-l),0),\dots,\min(s,p-t)\} = \{s_{\min},\dots,s_{\max}\} \\ \dim[\operatorname{null}(Y)\cap\operatorname{range}(A(k))] &\in \{\max(\delta-(t-l),0),\dots,\min(\delta,p-t)\} = \{\delta_{\min},\dots,\delta_{\max}\}. \end{aligned}$$

For defining the generic network rank, consider the following construction (based on [7]). Let $\mathcal{M}_j$ denote the set of all $j \times j$ minors of the matrix $YA(k)$, for $j = s_{\min},\dots,s_{\max}$. Then,

$$p_j(k) = \sum_{M\in\mathcal{M}_j} (\det M)^2$$

is a polynomial in the rate constants k. Using the polynomials $p_j(k)$, we can define the generic network rank and generic deficiency as follows.

Definition 6.3 (Generic Network Rank and Deficiency). *The* generic network rank *is defined as*

$$s_G = \max\{j : \exists k \in \mathbb{R}^n_{>0} \text{ such that } p_j(k) \neq 0\}.$$

The generic deficiency *is defined as*

$$\delta_G = p - t - s_G.$$

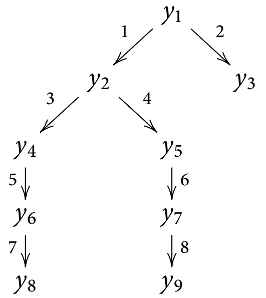
Note that the set of k for which $\operatorname{rank}(YA(k)) < s_G$ satisfies the algebraic condition $p_{s_G}(k) = 0$. Thus, the generic network rank and deficiency are indeed generic properties for the rate constants $k \in \mathbb{R}^n_{>0}$.

The point of this definition is that it is essentially an invariant for all networks, whether degenerate or not. As one would expect from a generalization, it reduces to the traditional rank s and deficiency δ for "well-behaved" (i. e. non-degenerate) networks.

The price one has to pay for this generalization is the restriction to "only" almost all $k \in \mathbb{R}^n_{>0}$. This is not as problematic as it might first seem for one can safely argue that all physical realizations of reaction networks have rate constants in general position. (At least, it is impossible to disprove such a claim by trying to identify the rate constants using statistical methods in the presence of measurement noise.)

It is possible to construct degenerate reaction networks whose generic ranks attain any value between the lower bound $s_{\min}$ and upper bound $s_{\max}$. We demonstrate this in the following example.

Example 6.2. We consider chemical reaction networks with the following network graph



(6.1)

and $m = 6$ different species. From the graph, we see that it has $p = 9$ complexes, $n = 8$ reactions, $l = 1$ linkage class and $t = 3$ terminal strong linkage classes.

In the following, we will construct three networks, each having rank $s = 6$ and deficiency $\delta = 2$, with different generic network rank s_{G} and deficiency δ_{G}.

1. By choosing the complex molecularity matrix Y as

$$Y^{(1)} = \begin{bmatrix} & 1 & & & 1 & & & & 1 \\ & 1 & & 1 & & & & 1 & \\ 1 & & 2 & & & & & & \\ & & & & 1 & & 1 & & \\ & & & 1 & & 1 & & & \\ 1 & & 1 & & & 1 & & 1 & \end{bmatrix}$$

   the network has $s_{\text{G}} = 6$ and $\delta_{\text{G}} = 0$.

2. By choosing the complex molecularity matrix Y as

$$Y^{(2)} = \begin{bmatrix} & 1 & & & 1 & & & & 1 \\ 1 & & & 1 & 1 & & & 2 & \\ & 1 & 1 & & & & & & \\ 1 & & 2 & & & & & & \\ & & & & & 1 & & & \\ & & & & & & 1 & & \end{bmatrix}$$

the network has $s_{\text{G}} = 5$ and $\delta_{\text{G}} = 1$.

3. By choosing the complex molecularity matrix Y as

$$Y^{(3)} = \begin{bmatrix} & & & & 1 & & 1 & & 1 \\ 2 & & & & & 1 & 1 & & \\ & & & & 1 & & & 1 & \\ & & & 1 & & & & & 1 \\ & 1 & & & & & & & \\ & & 1 & & & & & & \end{bmatrix}$$

the network has $s_{\text{G}} = 4$ and $\delta_{\text{G}} = 2$. □

Note that in the above example, the third network (with $s_{\text{G}} = 4$ and $\delta_{\text{G}} = 2$) does not satisfy regularity condition (R1). It has no flux vector $v \in \mathbb{R}^n_{>0}$ with $Nv = 0$. This is in fact generally true as we show in the following

Lemma 6.1. *For degenerate networks satisfying regularity condition (R1), it holds that*

$$\delta_{\text{G}} < \delta.$$

Proof. First, note that degenerate networks satisfying (R1) must have $\delta > 0$. For if $\delta = 0$, we have $v \in \text{null}(M)$. This, however, implies a weakly reversible network contradicting the assumption that the network is degenerate.

Then, the proof goes by contradiction. The condition $\delta_{\text{G}} = \delta$ implies that, for all k,

$$\operatorname{rank} YA(k) = s_{\text{G}} = \max(s - (t - l), 0) = s - (t - l).$$

On the other hand, regularity condition (R1) implies the existence of some $\tilde{v} \in \mathbb{R}^{n-1}_{<0}$ such that

$$n_j = N^{(-j)} \cdot \tilde{v} \tag{6.2}$$

for all $j = 1, \dots, n$, where n_j denotes the stoichiometry vector of reaction j and $N^{(-j)}$ is the stoichiometric matrix with the jth column deleted.

Let $[n]$ denote the set $\{1, \dots, n\}$ and $\binom{[n]}{s}$ the set of subsets of $[n]$ with cardinality s. Furthermore, let $N_{\rho,\sigma}$ denote the submatrix of N with rows $\rho \subseteq [m]$ and columns $\sigma \subseteq [n]$.

Now, because Eq. (6.2) holds for *every* j, we can always choose a reaction j at a branching point for which we can find a set of reactions $\sigma \in \binom{[n]-j}{s}$ with

$$\operatorname{rank} N_{[m],\sigma} = s$$
$$\operatorname{rank} U_{\sigma,[p]} > s - (t - l).$$

The important point here is to see that when reaction j is a branching reaction, the reactions in σ have at least $s - (t - l - 1)$ different substrate complexes and thus the rank of $U_{\sigma,[p]}$ is larger than $s - (t - l)$. This, however, implies

$$\operatorname{rank} N_{[m],\sigma} \cdot K_{\sigma,\sigma} \cdot U_{\sigma,[p]} > s - (t - l)$$

(where $K = \text{diag}(k)$), which contradicts the assumption that $s_{\text{G}} = s - (t - l)$. □

Degenerate Networks with Deficiency One

In view of the deficiency one algorithm, an important class of degenerate reaction networks are those with deficiency one. If $\delta = 1$, Lemma 6.1 specializes as

Corollary 6.2. *For a reaction network satisfying regularity condition (R1) with* $\delta = 1$ *and* $t > l$ *the property* $\delta_G = 0$ *holds generically for* $k \in \mathbb{R}^n_{>0}$.

Intuitively, one would expect networks with $\delta_G = 0$ to have the same stationarity properties as regular deficiency zero networks. The next lemma establishes that this is, at least for almost all rate constants k, true.

Lemma 6.3. *Degenerate networks with* $\delta_G = 0$ *have no positive steady-state for almost all parameter values* k.

Proof. For reaction networks with $\delta_G = 0$ and rate constants k in general position, the only possible steady-states are complex-balancing steady-states, i. e. steady-states x with

$$A(k) \cdot \psi(x) = 0.$$

A necessary condition for the existence of a positive complex-balancing steady-state is that the network is weakly-reversible. This, in turn, implies that $t = l$ which contradicts the assumption that the network is degenerate. ☐

Thus, degenerate networks with deficiency one are more similar to deficiency zero networks with respect to their stationarity properties. Such networks are never good candidates for finding multistationarity. In fact, for all practical purposes, they will never exhibit multiple positive steady-states in the same stoichiometric compatibility class.

6.2 Degenerate EFM Subnetworks

As many reaction networks arising in applications have $\delta > 1$, applying the deficiency one algorithm to these networks is not possible. Because of this, a decomposition of networks with deficiency higher one was suggested in Conradi et al. [12] that yields subnetworks that always have deficiency $\delta \leq 1$.

Elementary Flux Mode Decomposition

The decomposition is based on the elementary flux modes (EFMs) of a reaction network. EFMs are flux vectors $v \in \mathbb{R}^n$ in the nullspace of N with minimal support (cf. the subsection on EFMs in § 1.2). Recall that the support of a vector is the set of indices of its nonzero elements

$$\operatorname{supp}(v) = \{ i \mid v_i \neq 0 \},$$

and denote the number of nonzero elements of a vector v as

$$\operatorname{nz}(v) = |\operatorname{supp}(v)|.$$

Then, we define the restriction of a matrix $A = [a_1, \dots, a_n] \in \mathbb{R}^{m \times n}$ to the support of a vector $v \in \mathbb{R}^n$ as

$$A|_v = [a_{i_1}, \dots, a_{i_{\mathrm{nz}(v)}}] \in \mathbb{R}^{m \times \mathrm{nz}(v)} \qquad \text{with } i_k \in \mathrm{supp}(v).$$

Then, EFMs can be defined in the following, equivalent way.

Definition 6.4 (EFM; Stoichiometric Generator)**.** *The minimal generators of the pointed polyhedral cone* $\mathrm{null}(N) \cap \mathbb{R}^n_{\geq 0}$ *are called* elementary flux modes (EFMs)*. In particular, an EFM is a vector* $v \in \mathbb{R}^n_{\geq 0}$ *that satisfies*

$$\begin{aligned} Nv &= 0 \\ \mathrm{rank}(N|_v) &= \mathrm{nz}(v) - 1. \end{aligned}$$

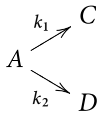
A stoichiometric generator *is an EFM* v *that additionally satisfies*

$$Mv \neq 0. \tag{6.3}$$

From this definition of elementary flux modes, it becomes immediately clear that EFM subnetworks have deficiency one if they are induced by stoichiometric generators and deficiency zero otherwise.

Subnetwork Regularization

Subnetworks defined by stoichiometric generators are guaranteed to satisfy regularity conditions (R1) and (R3) but not necessarily condition (R2) [12]. In this section, we will explain how to analyze degenerate subnetworks (i. e. those failing condition (R2)). Because the nullspace of the stoichiometric matrix $N|_v$ is exactly one dimensional, the ratio between any two steady-state fluxes is known. This allows to rewrite two reactions with a common substrate complex as one reaction. The following degenerate network fragment illustrates the procedure:



(6.4a)

can be rewritten as

$$A \xrightarrow{k_1 + k_2} \tfrac{k_1}{k_1+k_2} C + \tfrac{k_2}{k_1+k_2} D. \tag{6.4b}$$

The following lemma establishes the equivalence of the dynamics of the two systems.

Lemma 6.4 (Regularization Lemma)**.** *The networks* (6.4a) *and* (6.4b) *have the same dynamics.*

Proof. The first network (6.4a) has the following dynamics:

$$\begin{aligned} f_{(6.4\mathrm{a})}(x, k) &= \begin{bmatrix} y_1 & y_2 & y_3 \end{bmatrix} \cdot \begin{bmatrix} -k_1 - k_2 & 0 & 0 \\ k_1 & 0 & 0 \\ k_2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x^{y_1} \\ x^{y_2} \\ x^{y_3} \end{bmatrix} \\ &= y_1(-k_1 - k_2)x^{y_1} + y_2 k_1 x^{y_1} + y_3 k_2 x^{y_1}. \end{aligned}$$

The second network (6.4b) has the following dynamics:

$$\begin{aligned} f_{(6.4\mathrm{b})}(x,k) &= \begin{bmatrix} y_1 & \frac{k_1}{k_1+k_2}y_2 + \frac{k_2}{k_1+k_2}y_3 \end{bmatrix} \cdot \begin{bmatrix} -k_1-k_2 & 0 \\ k_1+k_2 & 0 \end{bmatrix} \cdot \begin{bmatrix} x^{y_1} \\ x^{(\cdots)} \end{bmatrix} \\ &= y_1(-k_1-k_2)x^{y_1} + \left(\frac{k_1}{k_1+k_2}y_2 + \frac{k_2}{k_1+k_2}y_3\right)(k_1+k_2)x^{y_1}. \end{aligned}$$

□

Since the function $f(x,k)$ is not modified when rewriting network (6.4a) into (6.4b), this step can be repeated at every branching point in an EFM subnetwork that is a source of degeneracy. If there are more than two reactions emanating from a substrate, it can also be applied recursively. In the end, an EFM network becomes regular, i. e. will also satisfy regularity condition (R2). Note that it is indeed enough to focus on motives (6.4a) for making EFM subnetworks regular deficiency one networks, since their network graphs do not contain any cycles. The only exception to this are reversible reaction pairs, which are themselves EFM subnetworks and thus never degenerate.

This rewriting procedure is in general applicable to any reaction network. The problem is that it makes the matrix Y and the function $\psi(x)$ depend on the rate constants k. For EFM-induced subnetworks this problem can be overcome, however. Consider again network (6.4a). We can immediately write down a necessary condition on the rate constants k_1, k_2 for the network to have a steady state, namely

$$\frac{k_1}{k_2} = \frac{v_1}{v_2} = \text{const.} \tag{6.5}$$

where v_1 and v_2 are the coefficients of an arbitrary nullspace flux vector (whose direction is unique for EFM networks). Therefore, it is possible to replace the coefficients $k_1/(k_1+k_2)$ and $k_2/(k_1+k_2)$ in (6.4b) by $v_1/(v_1+v_2)$ and $v_2/(v_1+v_2)$ which are known quantities.

Thus, the network (6.4a) has a steady state if and only if the network

$$A \xrightarrow{k^*} \frac{v_1}{v_1+v_2}C + \frac{v_2}{v_1+v_2}D \tag{6.6}$$

has one, where the new rate constant $k^* = k_1 + k_2$ captures the remaining degree of freedom.

Note that this is only guaranteed to work in EFM networks since the nullspace of $N|_v$ has such a simple structure. General networks can have more complex sources of degeneracy where this simple procedure can not be applied. For example, take the following network

$$C \longleftarrow A \rightleftarrows B \longrightarrow D. \tag{6.7}$$

Since the complexes C and D are connected by a non-singleton strongly connected component, the procedure fails.

Multistationarity

In the course of regularizing a degenerate EFM subnetwork, even though the network dynamics $f(x,k)$ are not modified, the same is not true for the stoichiometric matrix N. Since, by assumption,

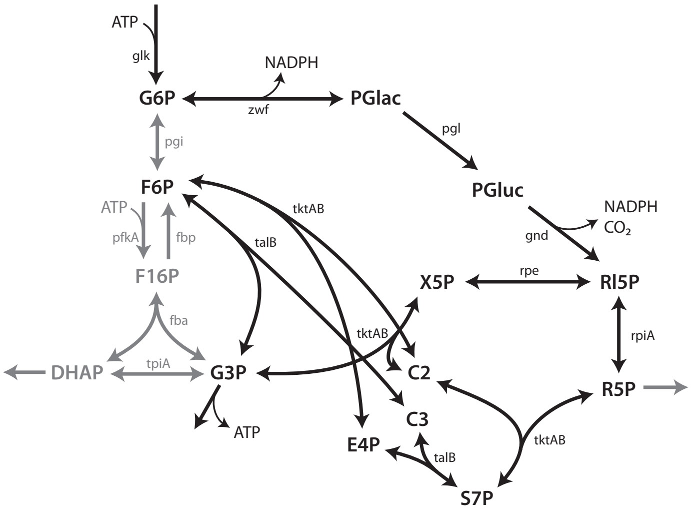


Figure 6.1: Simplified illustration of a mass-action model for the upper part of glycolysis and the pentose phosphate pathway. Black reactions form an EFM subnetwork that can be analyzed with the deficiency one algorithm. Gray reactions are inactive in the EFM subnetwork.

the original network is degenerate it can only exhibit multistationarity if the rate constants k satisfy the conditions implied by the direction of the EFM nullspace vector.

Under these conditions, we have the following relation between kinetic and stoichiometric subspaces with respect to multistationarity.

Lemma 6.5. *The kinetic subspace of the original network is equal to the stoichiometric subspace of the regularized network if the rate constants satisfy the conditions implied by the EFM nullspace vector.*

Proof. Since both networks have the same dynamics $f(x,k)$, the kinetic subspaces of the networks coincide. In the regularized network, it further holds that the stoichiometric and kinetic subspaces are the same. ☐

Application Example

We use the upper part of glycolysis (including the pentose phosphate pathway) as an application example for EFM-based analysis of multistationarity. The corresponding network shown in Fig. 6.1 is an important part in bacterial metabolism. It is the central pathway for growing on monosaccharides (e. g. glucose) and for producing energy for the cell. It is also important for the synthesis of precursors for amino acids and nucleotides. Notably, even for such well-characterized pathways the dynamic features are not fully understood.

Our network model consists of 4 uptake reactions, 7 outflows, and 13 enzyme-catalyzed reactions. When converted to mass-action kinetics, this results in a model with 53 species, 79 complexes and 78 reactions. The network deficiency is 8, which hampers CRNT-based analysis without prior network decomposition. In analyzing the model, we found 123 EFMs and 88 of these EFMs were stoichiometric generators. The subnetworks that correspond to these generators have between 19 and 34 reactions (see Fig. 6.1 for an example) and they all are degenerate. However, the only degenerate linkage class is that containing the uptake reactions with the zero complex as their substrate.

We regularized the subnetworks to enable analysis with the deficiency one algorithm. Interestingly, none of the subnetworks corresponding to an EFM can admit multiple steady states. However, as an EFM-based analysis only provides sufficient conditions for multistationarity of the entire network, multistationarity of the overall network is still possible [12]. Moreover, multistationarity in the overall network seems very likely, given its complex structure, which contains several feedback loops (e. g. via the energy carrier ATP).

6.3 Conclusions and Perspectives

The application of some of the main methods and theorems of CRNT (in particular the deficiency one algorithm) is currently restricted to regular reaction networks, which limits their applicability to real-world problems. In this chapter we proposed a generalized notion of network deficiency – *generic deficiency* – that broadens the applicability of existing results to a broader class of reaction networks including degenerate ones.

Furthermore, we developed and applied a regularization method for EFM subnetworks to circumvent the limitations of the deficiency one algorithm with respect to those networks. Again, this broadens the class of networks for which the preconditions in [12] are fulfilled.

Future generalizations of more elements of CRNT, for instance of the deficiency one theorem, are interesting and preliminary results indicate that such extensions of the theory are feasible. Finally, our case-study of a medium-sized network model indicates that this is a promising approach, for instance, to understand the remarkable robustness of biological systems [55].

Part III

The Big Experiment

7

Data Analysis of the Big Experiment

In this final part of the thesis, we describe the analysis of a series of experiments where *B. subtilis* cells were subjected to nutrient source shifts (referred to colloquially as the *Big Experiment* for brevity) conducted under the umbrella of the BaSysBio project [1]. The experiment was a consortium-wide undertaking with the aim of gaining a comprehensive understanding of system adaptation during the transient phase at all levels of the cellular network. Specifically, the focus of the experiment was to identify the key regulatory effects and their interplay among the metabolic and transcriptional levels. To achieve this ambitious goal, measurement data was acquired at several *omics* levels including transcripts, proteins and metabolites.

More specifically, the experiment was designed to expose the cells to shifts between a glycolytic (glucose) and a gluconeogenic (malate) carbon substrate. These substrates were chosen because it is known that *B. subtilis* can consume both substrates simultaneously. For most other substrate combinations, one can commonly observe two sequential growth phases where one substrate is inhibiting uptake of the other until it is exhausted (diauxic shift).

The present study is a contribution to the data analysis of this *Big Experiment* and as such, its aims are more modest. Of particular interest in our analysis are the physiological rates, i.e. the growth rate and metabolite exchange rates of *B. subtilis* during the different phases of the experiment. By means of a general regression analysis, we will first focus on the statistical estimation of the transient trajectories of the physiological quantities during the second phase of the experiment. Based on the conclusions from the regression analysis – and supported by other experimental evidence not presented here – one can hypothesize about the transcriptional adaptation strategies of *B. subtilis* with respect to the nutrient shifts. The description of all this work will be the contents of the present chapter.

After that, we will present a “maximally reduced” parametric model that is able to reproduce the observed data and mimic the hypothesized adaptation strategies. The main challenges are to derive model equations that are plausible, particularly with respect to mass/energy conservation, and to identify the model parameters based on the measurement data (Chapter 8). Through insights gained from the analysis of this model, it is then possible to study the adaptation strategies in an evolutionary context (Chapter 9).

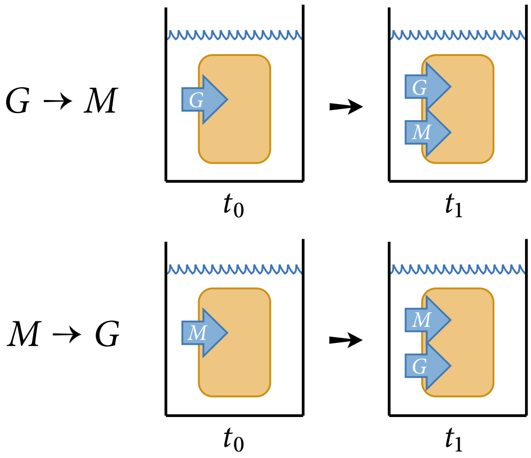


Figure 7.1: Outline of the Big Experiment.

7.1 Outline of the Experiment

The data underlying the Big Experiment was acquired in six nutrient shift experiments. The outline of these experiments is as follows. Starting at time t_0, the cells were initially grown in a bioreactor on a single carbon substrate (either glucose, G, or malate, M). During this first phase of the experiment, they were in exponential growth, i. e. their cell metabolism was in steady-state. When they reached a certain concentration, say at time t_1, a certain amount of the second substrate was added to the bioreactor. Figure 7.1 shows an outline of this experimental setup. The experiment was done in both directions, referred to as the $G \rightarrow M$ shift (for first growing on glucose, then adding malate) and the $M \rightarrow G$ shift (for first growing on malate, then adding glucose). Each shift experiment was repeated three times yielding three biological replicates, referred to as BR 1-3.

For each biological replicate, times series data was collected by measuring the optical density (proportional to cell concentration in the reactor) and metabolite concentrations in the reactor broth at 5-6 time points in phase 1 (before addition of the second substrate) and at 9-10 time points in phase 2 (after addition).

7.2 Measurement Data Analysis

The goal of this section is to survey the physiological data acquired during the Big Experiment to get a first idea of what processes are involved in the nutrient shift adaptations. That is, we want to derive the physiological rates (i. e. growth and exchange rates) by formulating and solving a regression problem based on the noisy measurements of the experiment. Since we have no general idea yet of the system's behavior, we try to make as modest assumptions as possible about the underlying biological processes and, thus, the regression model.

Because of the experimental design we know that in phase 1, the bacteria are growing in (quasi) steady-state conditions. That is, we assume constant growth and metabolite exchange rates. In phase 2, however, the system is in a transient phase where we have no *a priori* knowledge about its behavior. The safest way to proceed is thus to use a nonparametric regression technique during phase 2.

In the next section, we derive a regression framework that allows for doing parametric (during phase 1) and nonparametric (phase 2) regression simultaneously. In the two following sections, we will explain how this is applied to estimate the biomass trajectories (§ 7.3) and substrate concentrations (§ 7.4). This way, the notation can be kept simpler and the exposition of the basic ideas should become clearer.

General Regression Setting

The goal is to estimate a function $\bar{y}(t)$ of time representing the evolution of a physiological quantity in the bioreactor, e. g. the biomass concentration. As hypothesis space, we choose p basis functions $b_i(t)$, $i = 1, \ldots, p$ such that a linear combination

$$\sum_i \alpha_i b_i(t) \approx \bar{y}(t)$$

is able to approximate the underlying function $\bar{y}(t)$ closely over the time interval $[t_0, t_2]$. By "closely", we mean that if we knew the function $\bar{y}(t)$ we could choose the parameters α_i such that the approximation error

$$\left|\bar{y}(t) - \sum_i \alpha_i b_i(t)\right| \qquad t \in [t_0, t_2]$$

would be negligible for practical purposes. Because of the special protocol of the Big Experiment, the basis functions $b_i(t)$ should have the following characteristics: for $t_0 < t < t_1$ (phase 1) we have steady-state growth and thus know the parametric form of the trajectory $\bar{y}(t)$; for $t_1 < t < t_2$ (phase 2) we face unknown system behavior and thus have to use a rich hypothesis space (i. e. nonparametric regression). Furthermore, the transition from phase one to phase two has implications on the regularity of the function $\bar{y}(t)$ (i. e. its continuity and differentiability properties) at switching time t_1.

The regression problem is based on noisy measurements y_i of the trajectory at known time points t_i, $i = 1, \ldots, n$ with (unknown) standard deviation σ:

$$y_i = \bar{y}(t_i) + \sigma \cdot \varepsilon_i$$

where ε_i are i.i.d. $\mathcal{N}(0, 1)$ random variables.

Using the noisy measurements y_i, we can estimate the parameters α_i by minimizing the residual sum of squares. Because of the rich hypothesis space over the time interval (t_1, t_2), we need to include a regularization of function complexity over this interval. A standard technique is to penalize the second derivative of the trajectory $\hat{y}(t)$ which leads to a linear function in the limit of infinite penalization. The regression problem thus becomes

$$\min_\alpha \quad \frac{1}{2}\sum_i (y_i - \hat{y}(t_i))^2 + \frac{\lambda}{2}\int_{t_1}^{t_2} (\hat{y}''(\tau))^2 d\tau$$

where we have introduced a penalization parameter $\lambda > 0$. In the limit $\lambda \to \infty$ the problem reduces to a standard linear regression problem.

It is convenient to rewrite the regression problem in matrix form which is also useful for a computationally efficient implementation. To this end, we collect the measurements in a vector $y = (y_1, \ldots, y_n)^\top$ and the parameters in $\alpha = (\alpha_1, \ldots, \alpha_p)^\top$. Define the basis matrix $B \in \mathbb{R}^{n \times p}$ with elements

$$(B)_{ij} = b_j(t_i)$$

and the penalty matrix $R \in \mathbb{R}^{p \times p}$ with elements

$$(R)_{ij} = \int_{t_1}^{t_2} b_i''(\tau) b_j''(\tau) d\tau.$$

Using this notation, we can rewrite the regression problem as the following least-squares problem

$$\min_{\alpha} \quad \frac{1}{2} \|y - B\alpha\|^2 + \frac{\lambda}{2} \left(\alpha^\top R \alpha\right).$$

It has a well-known solution

$$\hat{\alpha} = \left(B^\top B + \lambda R\right)^{-1} B^\top y.$$

To reconstruct the function estimates $\hat{y}(t)$ at arbitrary time points t we define the vector

$$b(t) = \left(b_1(t), \ldots, b_p(t)\right)^\top$$

which allows for writing $\hat{y}(t)$ as

$$\hat{y}(t) = b^\top(t) \left(B^\top B + \lambda R\right)^{-1} B^\top y.$$

Of particular interest are the estimated values at the measurement time points t_i. When collected in a vector $\hat{y} = (\hat{y}(t_1), \ldots, \hat{y}(t_n))^\top \in \mathbb{R}^n$, we get the regression values as

$$\hat{y} = B \left(B^\top B + \lambda R\right)^{-1} B^\top y,$$

which shows that they depend linearly on the measured sample points y. The matrix

$$H = B \left(B^\top B + \lambda R\right)^{-1} B^\top$$

projects the measured data points y into the p-dimensional linear subspace spanned by the columns of the basis matrix B. The matrix H is called *hat matrix*, because applying it to the measured data y yields the regression estimates which are commonly denoted with a hat symbol ($\hat{y}$). In the "parametric" case (i. e. when $\lambda = 0$), this projection is orthogonal which implies that the sum of the eigenvalues of H equals p. This fact can be used to extend to the general case $\lambda > 0$, leading to the concept of *effective degrees of freedom* p_{eff}, defined as the sum of the eigenvalues of the hat matrix, i. e.

$$p_{\mathrm{eff}} = \operatorname{tr} H.$$

For $\lambda = 0$, i. e. the "parametric" case, this yields $p_{\text{eff}} = p$. The p_{eff} are used for estimating the unknown variance $\hat{\sigma}^2$. Define the residuals r as

$$r = y - \hat{y} = (I - H)y.$$

Then, the variance estimate becomes (cf. Appendix A)

$$\hat{\sigma}^2 = \frac{1}{n - p_{\text{eff}}} r^{\mathsf{T}} r.$$

Derivatives

The purpose of solving the above regression problem in the context of the Big Experiment is to get the physiological rates of *B. subtilis* during the experiment, i. e. the growth rate and the specific uptake rates (where "uptake" also includes substrate excretion). These rates are the time derivatives of the regression trajectories $\hat{y}(t)$ (see the following §§ 7.3-7.4 for a discussion of the exact details). Because of the chosen ansatz for $\hat{y}(t)$, the time derivative $d/dt\, \hat{y}(t)$ is

$$\frac{d}{dt}\hat{y}(t) = \sum_i \hat{\alpha}_i \cdot b_i'(t),$$

where $b_i'(t)$ is the time derivative of the ith basis function.

Choosing the Penalization Parameter

The next question we will address is how to choose the penalization parameter λ. Usually, one defines a "prediction risk" and then tries to choose a λ such as to minimize it. A typical choice is the *squared error loss*

$$\min_{\lambda} \quad \mathbb{E}\Big[\sum_i \big(\bar{y}(t_i) - \hat{y}(t_i)\big)^2\Big].$$

Of course, it is not possible to minimize this quantity directly without knowing the true trajectory $\bar{y}(t)$. What one can do, however, is to choose an information criterion that approximates this problem. A famous choice is leave-one-out cross validation for selecting the penalization parameter λ: choose λ such that

$$\text{CV} = \frac{1}{n}\sum_i \big(y_i - \hat{y}^{-i}(t_i)\big)^2$$

is minimized, where $\hat{y}^{-i}(t)$ denotes a fit by leaving out data point i. Instead of recomputing the fits $\hat{y}^{-i}$ for all $i = 1, \dots, n$, there is a formula for $\hat{y}^{-i}$ involving only values computed by the original fit $\hat{y}$, i. e.

$$\text{CV} = \frac{1}{n}\sum_i \left(\frac{y_i - \hat{y}(t_i)}{1 - h_{ii}}\right)^2,$$

where h_{ii} is ith diagonal element of the hat matrix H (see Appendix B for a derivation of this formula). We use a slight modification called generalized cross validation (GCV) [15]. The motivation to use generalized cross validation is that it is better behaved when the variance is unknown. It is also believed to be less susceptible to overfitting. The definition of the GCV score is

$$\mathrm{GCV} = \frac{1}{n} \sum_i \left(\frac{y_i - \hat{y}(t_i)}{1 - p_{\mathrm{eff}}/n} \right)^2 .$$

The difference to leave-one-out CV is that it uses the average p_{eff}/n for each i instead of the specific diagonal elements h_{ii}. Note that in the sum over all $i = 1, \dots, n$ they are equal, i.e.

$$\sum_i h_{ii} = \sum_i \frac{p_{\mathrm{eff}}}{n} = p_{\mathrm{eff}}.$$

Outlier Detection

The data of the Big Experiment is noisy and some measurements seem to have very large deviations that could be interpreted as outliers. To detect and assess the "extremity" of such measurements, we use Studentized residuals. That is, we analyze the ith residual when data point i was not used in the fitting procedure. As for leave-one-out cross validation, there is a formula for computing the residual r_i^{-i} of data point i from a fit with omitted measurement i without recomputing the regression:

$$r_i^{-i} = \frac{r_i}{1 - h_{ii}}.$$

The Studentized residual τ_i is defined as

$$\tau_i = \frac{r_i}{\hat{\sigma}^{-i} \sqrt{1 - h_{ii}}},$$

where $\hat{\sigma}^{-i}$ is the estimated standard deviation when point i is omitted. It can be computed efficiently from the following relation

$$(n - p_{\mathrm{eff}} - 1) \left(\hat{\sigma}^{-i} \right)^2 = (n - p_{\mathrm{eff}}) \hat{\sigma}^2 - \frac{r_i^2}{1 - h_{ii}}.$$

In parametric regression, i.e. when $p_{\mathrm{eff}} = p$ holds, the residuals τ_i follow a Student's t-distribution with $n - p - 1$ degrees of freedom. Extreme events (i.e. outliers) could thus be detected using the critical value of a t-distribution. When using penalized estimates with $\lambda > 0$, this is not true but still useful as an approximation. A detailed derivation of these formulas can be found in Appendix B.

Confidence Intervals

Define in analogy to the hat matrix H the "hat function"

$$h(t) = B(B^{\mathsf{T}} B + \lambda R)^{-1} b(t)$$

such that we can write the function value estimate at time t as

$$\hat{y}(t) = h^{\mathsf{T}}(t) \cdot y.$$

We would like to specify a confidence interval of the form

$$P\left[|\bar{y}(t) - \hat{y}(t)| \leq z_\delta \hat{\sigma} \sqrt{h^\top(t) h(t)}\right] = 1 - \delta,$$

where δ is the desired confidence level and z_δ is a critical value to be determined. It turns out that in the linear, non-penalized case z_δ is the critical value of a Student's t-distribution with $n - p$ degrees of freedom (cf. Appendix A), i. e.

$$z_\delta = t(n - p, 1 - \delta/2).$$

We use the above confidence intervals also as an approximation in the penalized case. However, this neglects any bias in the estimates. The reason for doing this is that bias estimation (e. g. using bootstrapping) proved to be very unreliable in numerical experiments with synthetic data, therefore we refrained from including it in confidence interval estimates. In effect, confidence intervals are correct over phase 1 of Big Experiment and approximate over phase 2.

7.3 Biomass Regression

We now turn to applying the regression to the biomass measurements of the Big Experiment. Let us denote the biomass in the reactor at time t as $m(t)$. The relation between the biomass and the growth rate is given by the following ordinary differential equation

$$\dot{m}(t) = \mu(t) m(t)$$

where $\mu(t)$ is the growth rate. The solution of this ODE is

$$m(t) = m_0 \cdot \exp\left(\int_{t_0}^{t} \mu(\tau) d\tau\right)$$

where $m_0 = m(t_0)$ is the initial condition. Since we have heteroscedasticity in the biomass measurements, we continue to work with the logarithm

$$\log m(t) \triangleq \ell(t).$$

This makes the biomass measurements and residuals look more equally distributed (see Figs. 7.2 and 7.3). We thus have:

$$\ell(t) = \ell_0 + \int_{t_0}^{t} \mu(\tau) d\tau.$$

When the actual experiments were conducted, the switching time t_1 was extrapolated for each replicate based on previously taken OD measurements. This makes the initial condition ℓ_0 replicate dependent. Thus, biomass trajectories must be estimated per replicate $e = 1, \ldots, r$, (where r is the number of replicates). We make the following ansatz for the log-biomass trajectory $\ell^e(t)$ of replicate e as

$$\ell^e(t) = \ell_0^e + \sum_i \alpha_i b_i(t) \qquad \forall e = 1, \ldots, r.$$

The basis functions $b_i(t)$ and parameters α_i are the same for all replicates, whereas the replicate spe-

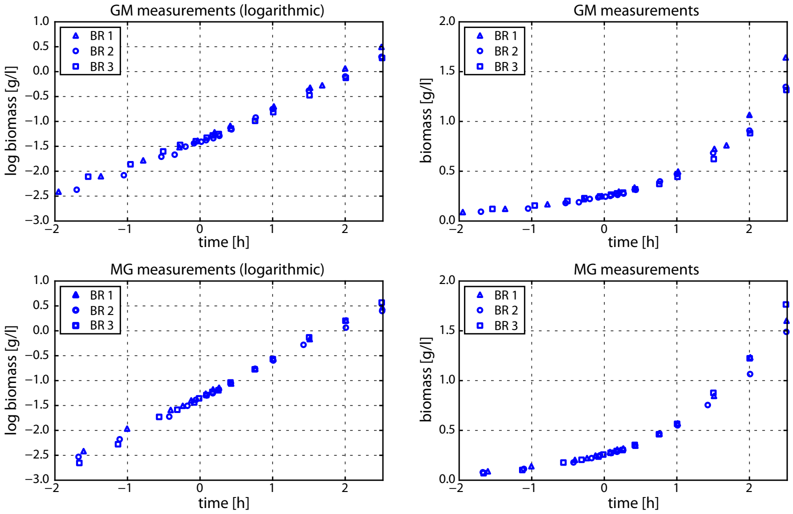


Figure 7.2: Biomass measurements of Big Experiment (three replicates BR 1-3). Left column shows the measurements on log-scale, right column shows measurements in absolute scale.

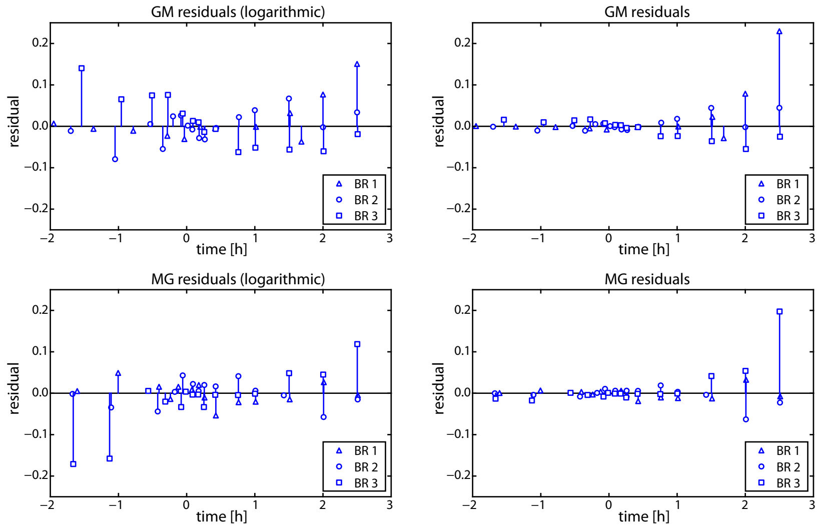


Figure 7.3: Residuals of biomass regression (three replicates BR 1-3). Left column shows residuals on log-scale, right column shows residuals in absolute scale. In the right column, one can see some indication of heteroscedasticity. The residuals are computed based on a regression of the log-measurements.

cific initial conditions ℓ_0^e capture the differences between the replicates. The actual basis functions used for fitting $\ell^e(t)$ are depicted in Fig. 7.4. During phase one, from $t_0 = -2$ to $t_1 = 0$, the basis functions are linear. In phase two, from $t_1 = 0$ to $t_2 = 2.5$, we use cubic splines with natural boundary conditions. The spline knots are non-uniformly spaced according to the planned measurement time points of the Big Experiment (note that the actual measurements were not taken exactly at these points). Close to the switching time t_1, the knot spacing it denser than towards the end of the experiment. At the ith knot position, the ith basis function equals 1 and all other functions are 0. The only exception is the very first knot at $t = -2$, where all basis functions are zero. This is an arbitrary normalization of the basis functions needed because of the replicate-specific initial conditions. Failing to normalize the basis functions would result in an additional degree of freedom rendering the regression model unidentifiable. In total, we have $p = 10$ basis functions together with $r = 3$ initial conditions, resulting in a total of 13 degrees of freedom.

This replicate-specificity makes the formulation of the regression problem slightly more complicated. Concatenate the measurements $\ell_1, \dots, \ell_r$ into one large measurement vector

$$y \triangleq \begin{bmatrix} \ell^1 \\ \vdots \\ \ell^r \end{bmatrix} \in \mathbb{R}^{(r \cdot n)}.$$

The dimension $r \cdot n$ of this vector stipulates that in all replicates the same number of n measurements was taken. This is an idealization to keep the notation simple. In reality, we would need to introduce replicate-specific numbers n_i, $i = 1, \dots, r$ and the dimensions of y would be $\sum_i n_i$ and even the measurement time points t_i would need be represented per-replicate. It is straight-forward to incorporate this in an actual implementation.

Next, we also need to assemble the "basis matrix" as

$$A \triangleq \begin{bmatrix} B & \mathbb{1} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ B & 0 & \cdots & \mathbb{1} \end{bmatrix} \in \mathbb{R}^{(r \cdot n) \times (p+r)}$$

which implies an augmented "parameter vector" with added initial conditions

$$x = (\alpha_1, \dots, \alpha_p, \ell_0^1, \dots, \ell_0^r)^\top \in \mathbb{R}^{p+r}.$$

Finally, we need to adjust the dimension of the regularization matrix with zeros as

$$\Omega \triangleq \begin{bmatrix} R & 0 \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{(p+r) \times (p+r)}.$$

The simultaneous regression estimate $\hat{x}$ of the trajectory parameters and initial conditions is

$$\hat{x} = (A^\top A + \lambda \Omega)^{-1} A^\top y.$$

Applying this procedure to the biomass measurements y of the Big Experiment yields the result plotted in Fig. 7.5.

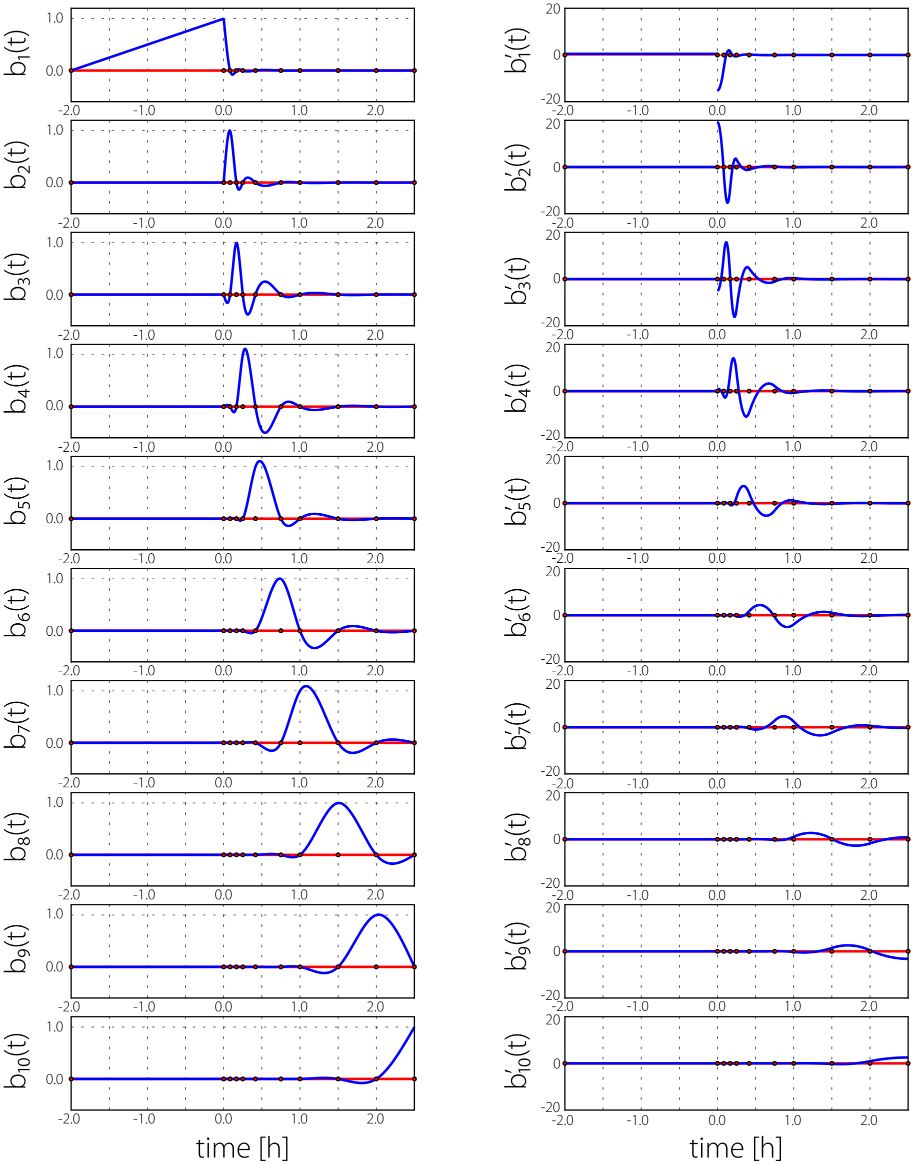


Figure 7.4: The basis functions for fitting the biomass trajectories in the reactor. Left column shows the values of the basis functions, the right column the derivatives. Phase one extends from $t_0 = -2$ to $t_1 = 0$. In this phase, the basis is a linear function. In phase two, from $t_1 = 0$ to $t_2 = 2.5$, we use cubic splines. The red bullets indicate the location of the spline knots which are non-uniformly placed at the planned measurement time points of the Big Experiment. Note that the ith basis function is 1 at knot i and 0 at all other knots. The regularity condition at switching time t_1 is that the function be continuous.

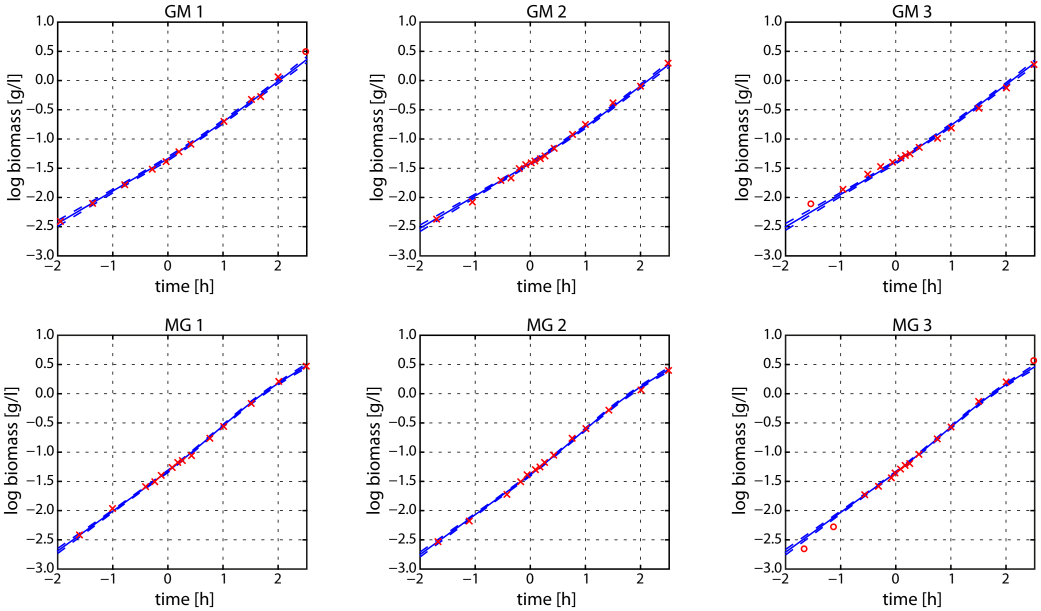


Figure 7.5: Regression of biomass. Red crosses are measured data points used for fitting, red circles are measurements classified as outliers. Blue solid line is regressed biomass trajectory, blue dashed lines denote (point wise) confidence intervals.

As already hinted earlier, the actual biological quantity of interest is the *growth rate* $\mu(t)$. We have

$$\mu(t) = \dot{\ell}(t) = \sum_i \alpha_i b_i'(t),$$

where $b_i'(t)$ is the time derivative of basis function i. As assumed, the growth rate is the same for all replicates since the replicate-specific initial conditions ℓ_0^e drop out. The actual estimate $\hat{\mu}(t)$ is plotted in Fig. 7.6.

7.4 External Metabolite Regression

Estimation of external metabolite concentrations works basically analogous to biomass estimation. There is, however, one main complicating factor: the estimate depends intrinsically on the amount of biomass in reactor. To accommodate for this we plug in the biomass estimate $\hat{m}(t)$ of the previous section. Taking into account the uncertainty associated with $\hat{m}(t)$ for the metabolite regression would lead to an error-in-variables model that is beyond the scope of this study. The proposed approach is feasible because the uncertainty in $\hat{m}(t)$ is significantly smaller than the variation in the concentration measurements.

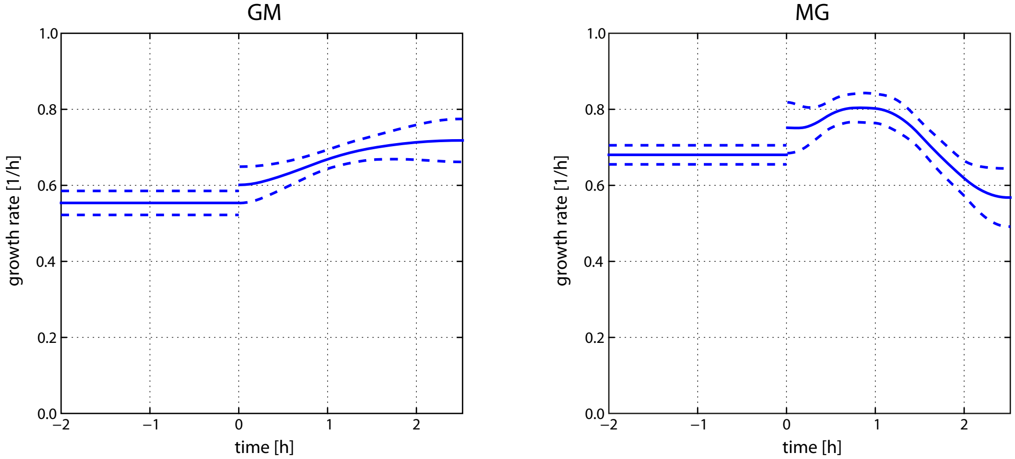


Figure 7.6: Regression of growth rate, derived from biomass fits. Note that there is only one (consensus) growth rate for all three replicates of the Big Experiment.

Let $c(t)$ denote the (true) concentration of an external metabolite in the reactor at time t. Then,

$$\dot{c}(t) = -v(t) \cdot m(t) + u(t),$$

where $v(t)$ is the specific uptake rate and $u(t)$ is the external input (needed since during the Big Experiment there is input applied to the system for some substrates). The minus sign is chosen such that substrate uptake leads to a positive rate $v(t)$. The solution of $c(t)$ can be directly written down

$$c(t) = c_0 + \int_{t_0}^{t} -v(\tau)m(\tau) + u(\tau)d\tau.$$

Due to the experimental setup and the dependence on $m(t)$, not only the concentration $c(t)$ but also the initial condition c_0 is replicate dependent. Because of this, and the experiment-specific input function $u(t)$, the strategy of specifying a common basis for the concentration trajectories of all replicates does not work, i.e. it is not possible to make a replicate-agnostic ansatz for $c(t)$. However, the uptake rate $v(t)$ is again assumed to be the same for all replicates. Thus, in analogy to the growth rate $\mu(t)$, we can make an ansatz for $v(t)$ with the same regularity properties, i.e.

$$v(t) = \sum_i \alpha_i b_i'(t).$$

For the concentrations $c(t)$ we thus get

$$c(t) = \int_{t_0}^{t} \Big(\sum_i \alpha_i b_i'(\tau)\Big) m(\tau) + u(\tau)d\tau = \sum_i \alpha_i \int_{t_0}^{t} b_i'(\tau)m(\tau)d\tau + \int_{t_0}^{t} u(\tau)d\tau.$$

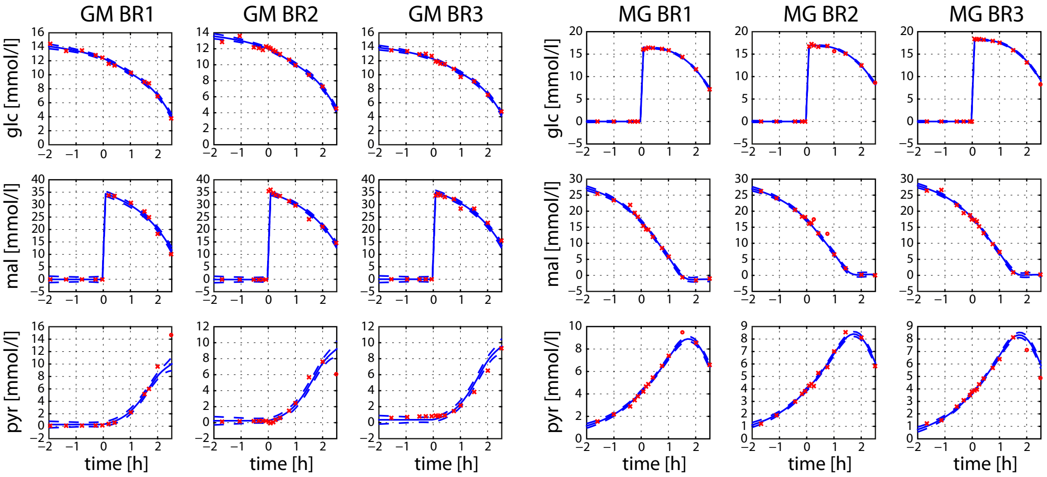


Figure 7.7: Regression of external metabolite concentrations. Red crosses are measured data points used for fitting, red circles are measurements classified as outliers. Blue solid line is regressed concentration trajectory, blue dashed lines denote (point wise) confidence intervals.

One can see that the regression is still linear in α, but the elements (i, j) of the basis matrix

$$\int_{t_0}^{t_i} b_j'(\tau)m(\tau)d\tau$$

are now replicate-specific and must be computed numerically. The input function $u(t)$ is

$$u(t) = \begin{cases} w_0 \cdot \delta(t - t_1) & \text{for the switching substrate} \\ 0 & \text{otherwise,} \end{cases}$$

where $\delta(\cdot)$ is the Dirac function and w_0 is a weight indicating the amount of substrate added at time point t_1. Similar to the unknown initial condition c_0, the pulse weight w_0 must also be assumed unknown, thereby adding another degree of freedom in the regression problem.

Using this approach, the basis functions $b_i(t)$ from biomass regression can also be used for metabolite regression. Indeed, we have the same requirements for the basis functions in both cases. The uptake rates are also assumed to be constant during phase 1 (exponential growth on one substrate) and the metabolite concentration trajectory is continuous at switching time t_1, except for the substrate that is added. But this jump is due to the user input function $u(t)$, and not the "autonomous" part related to substrate uptake that is captured by the basis functions $b_i(t)$. Furthermore, there is no reason why we should choose different knot positions of the spline functions in phase 2.

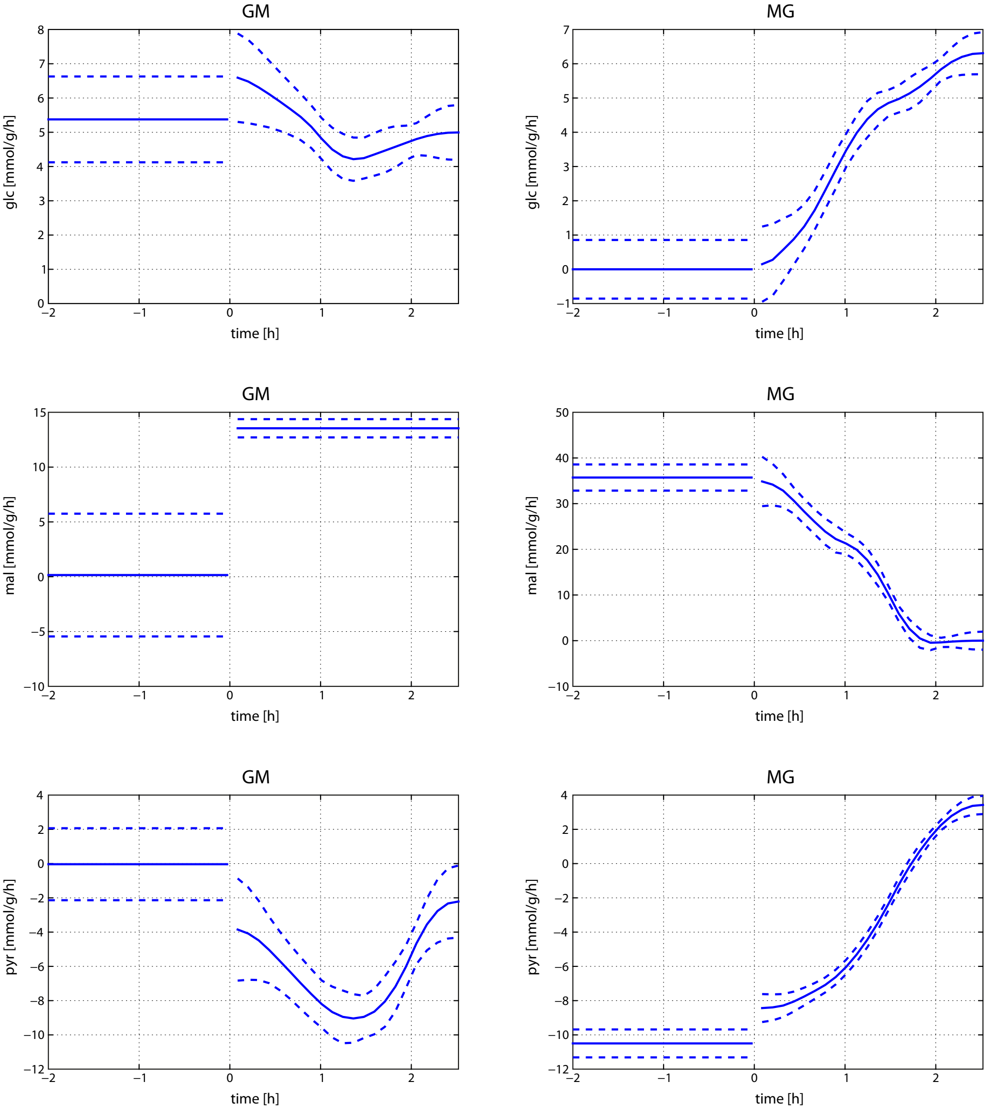


Figure 7.8: Regression of uptake rates, derived from concentration fits. Note that there is only one (consensus) uptake rate for all three replicates of the Big Experiment.

7.5 Conclusions & Perspectives

Using penalized splines for solving the regression problems proved flexible enough to account for some of the characteristics of the Big Experiment. In particular, the combination of parametric and non-parametric estimation within the same framework is possible by appropriate choice of basis functions and limiting the penalization to a subset of the time interval. Furthermore, the dependencies of the exchange rate on the biomass content required numerical computation of the regression coefficients.

Because of this non-standard use of penalized spline regression, a few open problems remain. For one, the given confidence intervals are only approximate and it is open how one could combine parametric estimates of uncertainty with more advanced methods from non-parametric regression (such as those presented in [63]). A second open problem is how to account for the uncertainty in the biomass estimates used for exchange rate regression. At the moment, this information is neglected and current state of errors-in-variables theory is limited for advanced regression settings such as the Big Experiment.

By looking at the regression results, the most striking observation is the different behavior of the cell with respect to the added substrate. If malate is added to cells growing on glucose, malate uptake rate jumps immediately up to its final level (Fig. 7.8, left column, middle row). In the reverse case, however, when glucose is added to cells growing on malate, the uptake rate is only gradually increasing, reaching its final level after about 1.5 hours (Fig 7.8, right column, top row).

This is strong support for the hypothesis that *B. subtilis* employs different control strategies for growth on these substrates. The machinery for malate, it seems, is maintained throughout, but its glucose capabilities can be switched on and off based on external conditions (i. e. glucose availability).

Another interesting observation comes from the relation of malate and pyruvate. When *B. subtilis* is growing on malate, a lot of the taken up substrate is excreted as pyruvate. In the MG shift experiment one can see that the malate concentration goes to zero around time $t = 1.5$h (Fig. 7.7, right part, middle rows) and hence the uptake rate also drops to zero (Fig. 7.8, right column, middle row). At that time, the accumulated external pyruvate starts to be used as substrate. The exchange rate of pyruvate crosses zero at about the same time (Fig. 7.8, right column, lower row), indicating that uptake of pyruvate is setting in.

8

Single Strain Model

THE MODEL we seek to build and analyze in this chapter has to reproduce the growth curves and the physiological substrate uptake rates of the Big Experiment as shown in Fig. 8.1. In particular, it should account for the findings of the previous regression analysis saying that it has different strategies for adapting to glucose and malate. For glucose, it should have slowly increasing uptake whereas malate uptake should be high from the start.

Furthermore, *B. subtilis* exhibits an effect of mutual uptake inhibition when it grows on both substrates simultaneously. To understand this effect, consider the data in Tbl. 8.1. One can see that the substrate uptake rates are smaller when both substrates are present than when only one substrate is available. Note that this data was acquired from shake-flask experiments [38] and the measured values are not compatible with the Big Experiment data. It is only shown to demonstrate the effect. The cause for this is not known precisely, but it is commonly believed that metabolic jamming between glycolytic and gluconeogenic pathways is the reason. Clearly, the model should also take this effect into account.

8.1 Building the Model

We now turn to the modeling ideas behind the parametric model. To explain the nutrient adaptation between two different carbon/energy sources, the simplest conceivable model must have at least three states: one for the (generic) biomass in the reactor (x^{BM}) and two for the nutrient subsystems (e^{G}, e^{M})

Table 8.1: Growth rate and specific substrate uptake rates of *B. subtilis* in exponential phase. When growing on both substrates simultaneously, all rates a inhibited compared to their steady-state values on a single substrate. Data taken from [38].

condition	growth rate	Glc uptake	Mal uptake
G	0.43	7.3	0
M	0.75	0	26.4
G & M	0.55	5.3	14.3

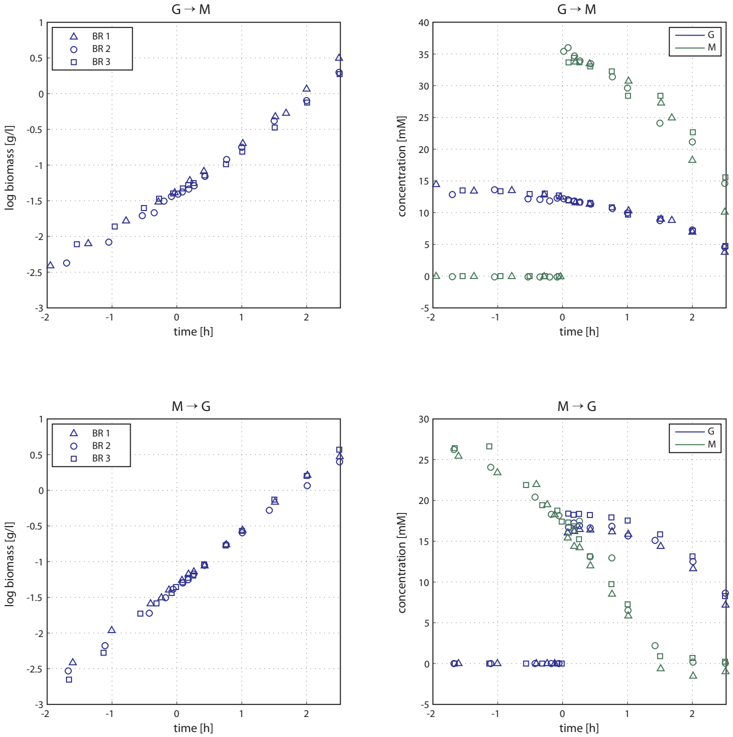


Figure 8.1: Measurement data of the "Big Experiment" relevant for the single strain model. Top row shows the replicates of the $G \rightarrow M$ shift, bottom row shows the $M \rightarrow G$ replicates. Left column are the biomass measurements in log-scale. Right column are the external substrate concentrations.

that enable the cell to grow on the respective substrates. The basic idea behind these subsystems is that if a state e^S (where $S \in \{G, M\}$) is "high", the cell has the ability to take up and grow on substrate S (if it is available, that is). If this state is "low", however, the cell is missing this capability (even if the substrate is present). A schematic picture of such a model is shown in Fig. 8.2. The units of the state variables e^G and e^M are "absolute concentrations" in the reactor, not specific concentrations per biomass.

As already mentioned in the previous paragraph, the model depends on the concentrations of the external substrates. This introduces another two states x_G and x_M needed to describe the whole system in the reactor. The complete system has therefore five states

$$(e^G, e^M, x^{BM}, x_G, x_M).$$

In the remainder of this section, we will give a detailed description and analysis of the mathematical equations that govern the dynamics of these five states.

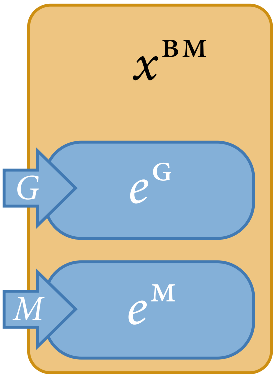


Figure 8.2: Schematic of the cell model. The "generic" biomass in the reactor is given by the state x^{BM} and the substrate-specific subsystem concentrations are denoted e^{G} and e^{M}. The subsystems facilitate growth on the respective substrates G and M (indicated by the labeled arrows).

Subsystem Cost

The two substrates glucose (G) and malate (M) serve as raw material for assembling biomass components as well as an energy source for catalyzing the chemical reactions. Thus, in our model the cell has to divide the energy/mass of the substrates on the generic biomass x^{BM}, and the two subsystems e^{G}, e^{M}.

Assume that the cell divides the substrate according to the weights w_{BM}, w_{G} and w_{M} on the biomass production, and the two subsystems, respectively. Then, the amount of substrate that is spent on the production of the three systems is

$$\begin{aligned}\mathrm{BM}: &\quad \frac{w_{\mathrm{BM}}}{w_{\mathrm{BM}} + w_{\mathrm{G}} + w_{\mathrm{M}}}\\ \mathrm{G}: &\quad \frac{w_{\mathrm{G}}}{w_{\mathrm{BM}} + w_{\mathrm{G}} + w_{\mathrm{M}}}\\ \mathrm{M}: &\quad \frac{w_{\mathrm{W}}}{w_{\mathrm{BM}} + w_{\mathrm{G}} + w_{\mathrm{M}}}.\end{aligned}$$

We assume that the wild-type *B. subtilis* cell spends a constant amount of energy for biomass production, i. e. $w_{\mathrm{BM}} = \mathrm{const.}$ However, the amount of energy it spends on the expression of its subsystems is not constant over time but depends on external conditions such as substrate availability. To account for this in the model, the cost factors w_{G}, w_{M} become functions of the external substrate concentrations. We propose an ansatz like

$$\begin{aligned}w_{\mathrm{G}}(x_{\mathrm{G}}) &= w_{\mathrm{G}}^{\max}\lambda_{\mathrm{G}}(x_{\mathrm{G}})\\ w_{\mathrm{M}}(x_{\mathrm{M}}) &= w_{\mathrm{M}}^{\max}\lambda_{\mathrm{M}}(x_{\mathrm{M}}),\end{aligned}$$

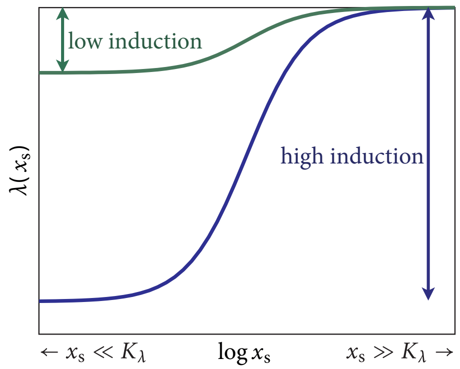


Figure 8.3: Shape of $\lambda(\cdot)$ function.

where we introduce a "modulation function"

$$\lambda_s : \mathbb{R}_{\geq 0} \to [0,1]$$

that determines how much of the maximal possible resources is spent on the subsystem S, depending on the external substrate concentration x_s. As ansatz for this function, we choose

$$\lambda_s(x_s) = a_\lambda^s + (1 - a_\lambda^s)\frac{x_s}{K_\lambda + x_s}.$$

If the external substrate concentration x_s is low (i. e. $x_s \ll K_\lambda$) the expression of the subsystem is close to the *baseline expression* a_λ^s. If the concentration is high (i. e. $x_s \gg K_\lambda$), the value of the modulation function is close to 1, which means that the cell fully expresses the subsystem. Cross-over between these two extremes happens around K_λ (see Fig. 8.3).

With the parameter a_λ^s it is possible to describe different control strategies of the cell with respect to subsystem expression. If $a_\lambda^s \ll 1$, we say that the cell *regulates* the subsystem, since the value of λ_s clearly depends on the concentration x_s. On the other hand, if $a_\lambda^s \approx 1$, the value of λ_s depends only slightly on the concentration x_s and we say that the cell expresses the subsystem *constitutively*. Note that the parameter a_λ^s must lie in the interval $[0,1]$ so that the function λ_s takes values in $[0,1]$ as required.

With this ansatz for the subsystem weight we get

$$\begin{aligned}
\mathrm{BM}: &\quad \frac{w_{\mathrm{BM}}}{w_{\mathrm{BM}} + w_{\mathrm{G}}^{\max}\lambda_{\mathrm{G}}(x_{\mathrm{G}}) + w_{\mathrm{M}}^{\max}\lambda_{\mathrm{M}}(x_{\mathrm{M}})} \\
\mathrm{G}: &\quad \frac{w_{\mathrm{G}}^{\max}\lambda_{\mathrm{G}}(x_{\mathrm{G}})}{w_{\mathrm{BM}} + w_{\mathrm{G}}^{\max}\lambda_{\mathrm{G}}(x_{\mathrm{G}}) + w_{\mathrm{M}}^{\max}\lambda_{\mathrm{M}}(x_{\mathrm{M}})} \\
\mathrm{M}: &\quad \frac{w_{\mathrm{W}}^{\max}\lambda_{\mathrm{M}}(x_{\mathrm{M}})}{w_{\mathrm{BM}} + w_{\mathrm{G}}^{\max}\lambda_{\mathrm{G}}(x_{\mathrm{G}}) + w_{\mathrm{M}}^{\max}\lambda_{\mathrm{M}}(x_{\mathrm{M}})}.
\end{aligned}$$

The scaling of the weights w_{BM}, $w_{\mathrm{G}}^{\max}$, $w_{\mathrm{M}}^{\max}$ is arbitrary, i.e. choosing weights $c \cdot w_{\mathrm{BM}}$, $c \cdot w_{\mathrm{G}}^{\max}$, $c \cdot w_{\mathrm{M}}^{\max}$ yields the same division of resources to subsystems. This is a degree of freedom that we can use to our advantage. A particularly useful scaling is to choose $c = 1/w_{\mathrm{BM}}$ which effectively measures all quantities in units of "generic biomass". For the subsystem weights we get

$$\begin{aligned} w_{\mathrm{G}}^{\max}/w_{\mathrm{BM}} &\triangleq \rho_{\mathrm{G}}^{\max} \\ w_{\mathrm{M}}^{\max}/w_{\mathrm{BM}} &\triangleq \rho_{\mathrm{M}}^{\max}. \end{aligned}$$

We call the factors $\rho_{\mathrm{G}}^{\max}$, $\rho_{\mathrm{M}}^{\max}$ the *(maximum) costs* of the two subsystems, since they measure how much energy is spent on the subsystems in relation to the biomass of the cell.

For simplicity, we assume throughout the remainder of this report that the maximum costs for the subsystems are equal, i.e.

$$\rho_{\mathrm{G}}^{\max} = \rho_{\mathrm{M}}^{\max} \triangleq \rho.$$

For the expressions that determine how the resources are spent on the three systems, we finally get

$$\begin{aligned} \mathrm{BM}: &\quad \frac{1}{1 + \rho\lambda_{\mathrm{G}}(x_{\mathrm{G}}) + \rho\lambda_{\mathrm{M}}(x_{\mathrm{M}})} \\ \mathrm{G}: &\quad \frac{\rho\lambda_{\mathrm{G}}(x_{\mathrm{G}})}{1 + \rho\lambda_{\mathrm{G}}(x_{\mathrm{G}}) + \rho\lambda_{\mathrm{M}}(x_{\mathrm{M}})} \\ \mathrm{M}: &\quad \frac{\rho\lambda_{\mathrm{M}}(x_{\mathrm{M}})}{1 + \rho\lambda_{\mathrm{G}}(x_{\mathrm{G}}) + \rho\lambda_{\mathrm{M}}(x_{\mathrm{M}})}. \end{aligned}$$

Specific Uptake Rates

To derive an expression for the specific uptake rates v^{S} of the cell, we have to make certain assumptions. For the lack of more detailed knowledge, we make a simple Michaelis-Menten ansatz assuming that substrate uptake is similar to an enzyme-catalyzed reaction

$$v^{\mathrm{S}} = k_{\mathrm{S}} \bar{e}^{\mathrm{S}} \frac{x_{\mathrm{S}}}{K_v + x_{\mathrm{S}}},$$

where k_{S} is a rate constant, $\bar{e}^{\mathrm{S}} \triangleq e^{\mathrm{S}}/x^{\mathrm{BM}}$ is the *specific* concentration of the subsystem S (in analogy to an enzyme) and the last term describes the binding of the substrate molecules to the transport proteins.

Since we also want to include the effect of mutual inhibition when both substrates are present, we include an inhibition factor

$$\gamma_{\mathrm{S}} : \mathbb{R}_{\geq 0} \to [0, 1]$$

that represents the strength of the uptake rate "damping" and depends on the alternate substrate concentration $\bar{e}^{\mathrm{T}}$ (we use the superscript T to denote the alternate of substrate S, e.g. if $S = G$ then $T = M$ and vice versa). The formulas for the substrate uptake reaction rates thus become

$$v^{\mathrm{S}} = k_{\mathrm{S}} \bar{e}^{\mathrm{S}} \gamma(\bar{e}^{\mathrm{T}}) \frac{x_{\mathrm{S}}}{K_v + x_{\mathrm{S}}}.$$

The mechanism that causes this inhibition is not known. For this analysis, we assumed an allosteric effect on the protein levels. This means that the concentration of the internal subsystems $\bar{e}^{\mathrm{S}}$ affects the functionality of the other enzymes $\bar{e}^{\mathrm{T}}$, lowering the *flux* v^{T}. Note that this mechanism has no direct effect on the expression of the *subsystem* $\bar{e}^{\mathrm{T}}$.

Another open problem is the choice for an ansatz of γ_{S}. It is not possible to derive the shape of γ_{S} from first principles. There are, however, two properties that any such function should have. First, if the uptake flux of the other substrate, v^{T}, is zero, $\gamma_{\mathrm{S}}(e^{\mathrm{T}})$ should be 1, i.e. there is no inhibition of the flux v^{S}. Second, the function should be monotonically decreasing. A simple possible choice satisfying these properties is a linear function

$$\gamma_{\mathrm{S}}(e^{\mathrm{T}}) = \left(1 - a_{\gamma}^{\mathrm{S}}\right)\left(1 - \frac{\bar{e}^{\mathrm{T}}}{\bar{e}_{\max}}\right) + a_{\gamma}^{\mathrm{S}}$$

where $a_{\gamma}^{\mathrm{S}} \in [0,1]$ is a parameter that determines the maximum possible inhibition. The quantity $\bar{e}_{\max}$ is a property of the system independent of the choice of $\gamma_{\mathrm{S}}(\cdot)$ and also of the subsystem S (cf. subsection on subsystem concentrations in § 8.1 for how it is calculated).

Subsystem Yields

The previous sections discussed the distribution of the resources available to the cell on the three biomass components BM, G, and M. Here, we describe the efficiency of conversion (yields), i.e. how much biomass the cell can get in return of the invested resources.

We denote the biomass yields for glucose and malate as η_{G} and η_{M}, respectively. In general, the yields of the three biomass components could be different which would result in three different values for η_{G} and η_{M} each. To simplify model analysis (and to facilitate parameter identification later on), we make two assumptions. First, we assume that the yields for the G and M subsystems are equal (analogous to the assumption that the cost parameter ρ is the same). Second, we assume that the yield ratios between glucose and malate are equal, i.e.

$$\frac{\eta_{\mathrm{G}}}{\eta_{\mathrm{M}}} = \frac{\eta_{\mathrm{G}}^{\mathrm{E}}}{\eta_{\mathrm{M}}^{\mathrm{E}}}$$

where the superscript E denotes the yields for the enzyme subsystems. Equivalently, we can also write

$$\frac{\eta_{\mathrm{G}}^{\mathrm{E}}}{\eta_{\mathrm{G}}} = \frac{\eta_{\mathrm{M}}^{\mathrm{E}}}{\eta_{\mathrm{M}}} \triangleq c_{\mathrm{E}}.$$

At this point, we thus introduce three parameters η_{G}, η_{M}, and c_{E} to describe all yields of the system.

System Equations

For completeness, we state the full system of ODEs for the five states of the single strain model.

$$\dot{x}^{\mathrm{BM}} = \eta_{\mathrm{G}} \frac{1}{1+\rho\lambda(x_{\mathrm{G}})+\rho\lambda(x_{\mathrm{M}})} \nu^{\mathrm{G}} x^{\mathrm{BM}} + \eta_{\mathrm{M}} \frac{1}{1+\rho\lambda(x_{\mathrm{G}})+\rho\lambda(x_{\mathrm{M}})} \nu^{\mathrm{M}} x^{\mathrm{BM}} - d x^{\mathrm{BM}} \tag{8.1a}$$

$$\dot{e}^{\mathrm{G}} = c_{\mathrm{E}} \eta_{\mathrm{G}} \frac{\rho\lambda(x_{\mathrm{G}})}{1+\rho\lambda(x_{\mathrm{G}})+\rho\lambda(x_{\mathrm{M}})} \nu^{\mathrm{G}} x^{\mathrm{BM}} + c_{\mathrm{E}} \eta_{\mathrm{M}} \frac{\rho\lambda(x_{\mathrm{G}})}{1+\rho\lambda(x_{\mathrm{G}})+\rho\lambda(x_{\mathrm{M}})} \nu^{\mathrm{M}} x^{\mathrm{BM}} - d e^{\mathrm{G}} \tag{8.1b}$$

$$\dot{e}^{\mathrm{M}} = c_{\mathrm{E}} \eta_{\mathrm{G}} \frac{\rho\lambda(x_{\mathrm{M}})}{1+\rho\lambda(x_{\mathrm{G}})+\rho\lambda(x_{\mathrm{M}})} \nu^{\mathrm{G}} x^{\mathrm{BM}} + c_{\mathrm{E}} \eta_{\mathrm{M}} \frac{\rho\lambda(x_{\mathrm{M}})}{1+\rho\lambda(x_{\mathrm{G}})+\rho\lambda(x_{\mathrm{M}})} \nu^{\mathrm{M}} x^{\mathrm{BM}} - d e^{\mathrm{M}} \tag{8.1c}$$

$$\dot{x}_{\mathrm{G}} = -\nu^{\mathrm{G}} x^{\mathrm{BM}} \tag{8.1d}$$

$$\dot{x}_{\mathrm{M}} = -\nu^{\mathrm{M}} x^{\mathrm{BM}} \tag{8.1e}$$

The first three equations (8.1a)-(8.1c) describe the growth of biomass and the subsystems. The first terms give the growth for glucose and the second terms give the growth for malate. Note that we have added a generic degradation term d to these biomass components. This is mainly for the later use of the model to study evolutionary aspects of adaptation strategies to nutrient shifts.

The last two equations (8.1d)-(8.1e) denote the dynamics of the external substrate concentrations.

Specific Subsystem Concentration

The biologically relevant quantity is not the subsystem concentration e^{S} directly, but the *specific* subsystem concentration $\bar{e}^{\mathrm{S}} = e^{\mathrm{S}}/x^{\mathrm{BM}}$. For example, the specific uptake rates ν^{S} are proportional to $\bar{e}^{\mathrm{S}}$. The differential equation for $\bar{e}^{\mathrm{S}}$ is

$$\dot{\bar{e}}^{\mathrm{S}} = \frac{\dot{e}^{\mathrm{S}} x^{\mathrm{BM}} - e^{\mathrm{S}} \dot{x}^{\mathrm{BM}}}{(x^{\mathrm{BM}})^2}.$$

To understand the asymptotic behavior of $\bar{e}^{\mathrm{S}}$, it is instructive to look at this quantity under specific physiological conditions. Consider, for example, exponential growth on only one substrate S. This condition means basically that $x_{\mathrm{S}} \gg K_\nu, K_\lambda$ and $x_{\mathrm{T}} = 0$. We can thus make the following simplifications for the λ-functions

$$\lambda_{\mathrm{S}}(x_{\mathrm{S}}) \approx 1 \qquad \lambda_{\mathrm{S}}(x_{\mathrm{T}}) = a_\lambda^{\mathrm{S}}$$

and for the specific uptake rates

$$\nu^{\mathrm{S}} \approx k_{\mathrm{S}} \bar{e}^{\mathrm{S}} \qquad \nu^{\mathrm{T}} = 0.$$

Then, we get for $\bar{e}^{\mathrm{S}}$

$$\dot{\bar{e}}^{\mathrm{S}} = c_{\mathrm{E}} \eta_{\mathrm{S}} \frac{\rho}{1+\rho} k_{\mathrm{S}} \left(\bar{e}^{\mathrm{S}}\right)^2 - \eta_{\mathrm{S}} \frac{1}{1+\rho} k_{\mathrm{S}} \bar{e}^{\mathrm{S}}$$

which has an analytic solution

$$\bar{e}^{\mathrm{S}}(t) = \frac{c_{\mathrm{E}}\rho}{1 + C \exp\left(-c_{\mathrm{E}} \eta_{\mathrm{S}} \frac{\rho}{1+\rho} k_{\mathrm{S}} t\right)}$$

where the constant C depends on the initial conditions. From the above equation, it can be seen that the maximal specific concentration $\bar{e}_{\max}$ of the two subsystems is subsystem independent and given by

$$\bar{e}_{\max} = c_{\mathrm{E}}\rho.$$

Using this scaling, we can rewrite the equations of the specific uptake rates v^{S} as

$$v^{\mathrm{S}} = v^{\mathrm{S}}_{\max} \frac{\bar{e}^{\mathrm{S}}}{c_{\mathrm{E}}\rho} \gamma(\bar{e}^{\mathrm{T}}) \frac{x_{\mathrm{S}}}{K_v + x_{\mathrm{S}}}$$

where we have replaced the original rate constant k_{S} with the expression $v^{\mathrm{S}}_{\max}/c_{\mathrm{E}}\rho$. The parameter $v^{\mathrm{S}}_{\max}$ has an immediate interpretation as the maximal possible uptake rate of substrate S that is nearly attained under the conditions above (exponential growth on single substrate).

8.2 Identifiability Analysis

Now that we have described the mathematical structure of the model, the next step is to identify its parameters. We will first give a theoretical identifiability analysis of the model based on results of observability of nonlinear systems [35]. After that, we will explain how to solve the actual parameter estimation problem.

Observability

For discussing observability of nonlinear systems, consider the following general autonomous model (i. e. a model with no inputs):

$$\Sigma: \qquad \begin{aligned} \dot{x} &= f(x) \\ y &= h(x) \end{aligned}$$

where x is the state variable and y are the output variables. The functions $f : \mathbb{R}^n \to \mathbb{R}^n$ and $h : \mathbb{R}^n \to \mathbb{R}^m$ are assumed to be smooth.

Two initial conditions x_0' and x_0'' are called *indistinguishable* over a time interval $[t_0, t_1]$ if their system output is equal, i. e. if

$$y(x_0', t) \equiv y(x_0'', t) \quad \forall t \in [t_0, t_1].$$

The set of initial conditions indistinguishable from x_0 is denoted $I(x_0)$.

Using this concept of indistinguishability, we are able to state the first two of the four commonly used definitions of observability [35].

1. *Observable.* A system is *observable at the point* x_0 if $I(x_0) = \{x_0\}$. It is *observable* if $I(x) = \{x\}\ \forall x \in \mathbb{R}^n$.

2. *Weakly observable.* A system is *weakly observable at* x_0 if there exists a neighborhood U around x_0 with $I(x_0) \cap U = \{x_0\}$. It is *weakly observable* if $I(x) \cap U = \{x\}$ for every $x \in \mathbb{R}^n$.

Note that the "weak" version of observability can only discriminate a point x_0 from its neighbors, but not from points far away. Both concepts are global in the sense that it might be necessary to go far away form x_0 or to wait a long time until discrimination is possible.

It is often convenient to have a stronger, "localized" notion of observability. To state this, we first need a stricter definition of indistinguishability. Two initial conditions x_0' and x_0'' in an open set $U \subset \mathbb{R}^n$ are *U-indistinguishable* over the time interval $[t_0, t_1]$ if $\forall t \in [t_0, t_1]$

$$\begin{aligned} x(x_0', t) &\in U \\ x(x_0'', t) &\in U \\ y(x_0', t) &\equiv y(x_0'', t). \end{aligned}$$

The set of initial conditions U-indistinguishable from x_0 is denoted as $I_U(x_0)$.

We are now in the position to state the "localized" versions of observability.

3. *Local observability*. A system is *locally observable at the point* x_0 if for every neighborhood U of x_0 it holds that $I_U(x_0) = \{x_0\}$. It is *locally observable* if it is locally observable at every $x \in \mathbb{R}^n$.
4. *Local weak observability*. A system is *locally weakly observable at* x_0 if there exist a neighborhood U of x_0 such that for every neighborhood $V \subset U$ it holds that $I_V(x_0) = \{x_0\}$. It is *locally weakly observable* if it is so at every $x \in \mathbb{R}^n$.

For the last definition of observability (local weak observability), there exists a well-known, sufficient condition called the *observability rank condition (O.R.C)* [35, Theorem 3.1]. Let the *observability distribution* σ be the smallest algebra over $C^\infty(\mathbb{R})$ containing the output functions h_i ($i = 1, \dots, m$) that is closed under Lie-differentiation along f. Let further denote

$$\Omega_\sigma = \operatorname{span}\{d\lambda | \lambda \in \sigma\}$$

the associated observability co-distribution of the system Σ. Then, Σ satisfies the O.R.C at x_0 if

$$\operatorname{rank} \Omega_\sigma(x_0) = n.$$

The reverse is "almost" true in the following sense [35, Theorem 3.11]. If the system Σ is locally weakly identifiable then the O.R.C holds generically (i.e. for almost all $x \in \mathbb{R}^n$). This is a useful theorem for practically checking whether a given system is locally weakly observable or not.

Observability theory can be used for identifiability as follows. Consider a family of nonlinear, autonomous systems, parameterized by a vector $\theta \in \mathbb{R}^p$:

$$\Sigma^\theta : \qquad \begin{aligned} \dot{x} &= f_\theta(x) \\ y &= h_\theta(x). \end{aligned}$$

By augmenting the state vector x with the parameters θ as constant states in a vector $\xi = (\theta, x) \in \mathbb{R}^{p+n}$, we get an augmented system

$$\Sigma^A : \qquad \begin{aligned} \dot{\xi} &= f_A(\xi) \\ y &= h_A(\xi). \end{aligned}$$

If the O.R.C is satisfied generically for the augmented system Σ^A, the parameters and the initial conditions are distinguishable simultaneously.

Observability of the Single Strain Model

To fit the single strain model into the framework described above, we collect the states into a vector

$$x = (e^{\mathrm{G}}, e^{\mathrm{M}}, x^{\mathrm{BM}}, x_{\mathrm{G}}, x_{\mathrm{M}})^{\mathsf{T}} \in \mathbb{R}^5$$

and analogously for the parameters

$$\theta = (\eta_{\mathrm{G}}, \eta_{\mathrm{M}}, c_{\mathrm{E}}, \rho, d, k_{\mathrm{G}}, k_{\mathrm{M}}, a_\lambda^{\mathrm{G}}, a_\lambda^{\mathrm{M}}, a_\gamma^{\mathrm{G}}, a_\gamma^{\mathrm{M}}, K_\lambda, K_\nu)^{\mathsf{T}} \in \mathbb{R}^{13}.$$

Thus, the augmented single strain model has a state vector $\xi \in \mathbb{R}^{18}$ (i. e. $n = 18$).

Checking observability of the augmented system completely symbolically is not possible. However, it is possible to compute all Lie-derivatives

$$L_f^k h_i(x) \qquad i = 1, \ldots, m$$

up to order $k = 6$ as well as their partial derivatives symbolically (using a computer algebra system) to construct a test matrix of size 21×18. After that, the rank computation must be done numerically for a randomly chosen x_0. It turns out that

$$\operatorname{rank} \Omega_\sigma^A(x_0) = n - 1,$$

i. e. the O.R.C is not satisfied (generically). It is thus instructive to compute (numerically) a nullspace vector of the test matrix and inspect the pattern of the non-zeros. As it turns out, the orthogonal complement $(\Omega_\sigma^A)^\perp$ contains an algebraic relation between c_{E}, e^{G} and e^{M}. This is directly related to the choice of units in which the subsystem states are represented. Since these subsystem states cannot be measured, the choice of units is arbitrary and should not affect the overall system behavior. This is exactly why the system is not locally observable.

To remedy this problem, we have to fix the units of the subsystems. A particularly convenient choice would be if $\bar{e}_{\max}$, the maximum specific concentration of the subsystems, would be one. As was shown in § 8.1 (in the subsection about specific subsystem concentrations), this can be achieved by

$$c_{\mathrm{E}} \equiv \frac{1}{\rho}. \tag{8.2}$$

Using this scaling, the model equations simplify considerably. Note that this is not a constraint in the sense that the model becomes more "rigid". It simply fixes the (arbitrary) units of the subsystems.

8.3 Parameter Identification

This section describes the regression problem that we have built around the single strain model (8.1) for estimating its parameters and initial conditions as well as the methods used to solve it.

The experimental setting for acquiring the data was as follows: there were two nutrient shift experiments (G → M and M → G), each with three biological replicates (BR 1-3), see Fig. 8.1. We will therefore use two indices s (shift index) and r (replicate index) to denote the time series data measured during an experiment.

As in the previous section, we collect the five system states in state vector

$$x = (e^{\mathrm{G}}, e^{\mathrm{M}}, x^{\mathrm{BM}}, x_{\mathrm{G}}, x_{\mathrm{M}})^{\mathsf{T}}.$$

However, the parameter vector also has to include those initial conditions that were measured, which makes it experiment dependent. Assume that the system is initially growing on substrate S and at switching time t_1 substrate T is added. In this case, the uncertain initial conditions are x_0^{BM}, $x_{\mathrm{S,0}}$ and w_{T} (pulse weight of added substrate). The initial value $x_{\mathrm{T,0}}$ of substrate T is zero and the "internal states" e^{S}, e^{T} are at their maximum and minimum value, since the system is initially evolving under single-substrate conditions. Therefore, the parameter vector for describing shift experiment $S \to T$ is denoted

$$\tilde{\theta} = (\eta_{\mathrm{G}}, \eta_{\mathrm{M}}, c_{\mathrm{E}}, \rho, d, k_{\mathrm{G}}, k_{\mathrm{M}}, \dots, x_0^{\mathrm{BM}}, x_{\mathrm{S,0}}, w_{\mathrm{T}})^{\mathsf{T}} \in \mathbb{R}^{16}.$$

We abbreviate the system (8.1) as

$$\dot{x} = f(x, \tilde{\theta}) \tag{8.3a}$$

$$y = h(x), \tag{8.3b}$$

where the system outputs $y = (y_1, y_2, y_3)^{\mathsf{T}}$ correspond to the measured data: $y_1 \equiv x^{\mathrm{BM}}$ (biomass), $y_2 \equiv x_{\mathrm{G}}$ (external glucose concentration) and $y_3 \equiv x_{\mathrm{M}}$ (external malate concentration).

Since the Big Experiment data set consists of two shift types and three replicates, the complete parameter vector θ to be estimated has 31 dimensions (13 system parameters, 6 initial biomass concentrations, 6 initial substrate concentrations and 6 pulse weights).

Objective Function

The objective function of the regression problem is the residual sum-of-squares between the model trajectory and the measurement points. The first problem is that the measured quantities have different physical units (biomass concentration in [g/l], metabolite concentrations in [mmol/l]). As we want to combine all measurements in one objective function, we need to get comparable values. To achieve this, we scale the residuals by their standard deviation. Assuming normally distributed errors, this leads to dimensionless, standard normal random numbers. Since we have three system outputs, we assume that the errors come from three measurement processes with different variance (σ_{BM}^2, σ_{G}^2, σ_{M}^2). For the biomass measurements, as discussed in Chapter 7 (and shown in Figs. 7.2 and 7.3), we can find heteroscedasticity in the residuals. We thus use the logarithm of the biomass measurements and the scaling σ_{BM}^2 is applied to the log-values.

We define the residuals including the scalings as

$$r_{s,r,i}^{\mathrm{BM}} \triangleq \frac{\log y_{s,r,i}^{\mathrm{BM}} - \log y_{s,r}^{\mathrm{BM}}(t_i, \theta)}{\sigma_{\mathrm{BM}}}$$

$$r_{s,r,i}^{\mathrm{G}} \triangleq \frac{y_{s,r,i}^{\mathrm{G}} - y_{s,r}^{\mathrm{G}}(t_i, \theta)}{\sigma_{\mathrm{G}}}$$

$$r_{s,r,i}^{\mathrm{M}} \triangleq \frac{y_{s,r,i}^{\mathrm{M}} - y_{s,r}^{\mathrm{M}}(t_i, \theta)}{\sigma_{\mathrm{M}}}$$

such that the objective function becomes

$$\min_{\theta} f_{\text{obj}}(\theta) \triangleq \frac{1}{2} \sum_{s} \sum_{r} \sum_{i} \left(r^{\text{BM}}_{s,r,i}\right)^2 + \left(r^{\text{G}}_{s,r,i}\right)^2 + \left(r^{\text{M}}_{s,r,i}\right)^2 . \tag{8.4}$$

Gradient Computation

To solve the optimization problem efficiently, it is necessary to compute the gradient $\nabla f_{\text{obj}}(\theta)$ of the objective function. In our case, this leads to computing the derivatives of state trajectories defined by the ODE system (8.3). This can be achieved numerically by employing specialized ODE solvers that can compute ODE sensitivities, such as CVODE [11].

The problem for computing the gradient of the objective function is that it is defined for the optimization vector θ, but the ODE system (8.3) is defined in terms of a different parameter vector $\tilde{\theta}$. We assume that there exists a differentiable function $a : \theta \mapsto \tilde{\theta}$ that maps θ to $\tilde{\theta}$. Since $\tilde{\theta}$ also contains some initial conditions, this function is parametrized by the shift index s and the replicate index r, i. e. we have

$$\tilde{\theta} = a_{s,r}(\theta).$$

To get the gradient of $f_{\text{obj}}(\theta)$, we clearly need the ODE sensitivities $\partial x/\partial\theta$. Since the ODEs are defined in terms of their “system parameters” $\tilde{\theta}$, a solver returns the solution to

$$\begin{aligned} \dot{x} &= f(x, \tilde{\theta}) \\ x(0) &= x_0(\tilde{\theta}) \end{aligned}$$

with the associated sensitivities

$$\begin{aligned} \frac{d}{dt}\tilde{S} &= \frac{\partial f}{\partial x}\tilde{S} + \frac{\partial f}{\partial \tilde{\theta}} \\ \tilde{S}(0) &= \frac{\partial x_0}{\partial \tilde{\theta}}. \end{aligned}$$

To come back to the desired sensitivities, we can apply the chain rule for which we need the Jacobian of the parameter map $\partial a/\partial\theta$. The solution $x(t)$ in terms of the parameter vector θ is

$$x(t, a(\theta)) = x_0(a(\theta)) + \int_0^t f(x(s, a(\theta)), a(\theta))\, ds.$$

The sensitivities with respect to the parameters θ are thus

$$\frac{\partial x(t, a(\theta))}{\partial \theta} = \frac{\partial x_0(a(\theta))}{\partial \theta} + \int_0^t \frac{\partial f}{\partial x} \frac{\partial x(s, a(\theta))}{\partial \theta} + \frac{\partial x}{\partial \theta}\, ds.$$

Applying the chain rule yields

$$\frac{\partial x(t, \tilde{\theta})}{\partial \tilde{\theta}} \frac{\partial a}{\partial \theta} = \frac{\partial x_0(\tilde{\theta})}{\partial \theta} \frac{\partial a}{\partial \theta} + \int_0^t \frac{\partial f}{\partial x} \frac{\partial x(s, \tilde{\theta})}{\partial \tilde{\theta}} \frac{\partial a}{\partial \theta} + \frac{\partial x}{\partial \tilde{\theta}} \frac{\partial a}{\partial \theta}\, ds.$$

We thus see that the sensitivities S can be computed from $\tilde{S}$ as

$$S = \frac{\partial x}{\partial \theta} = \tilde{S}\frac{\partial a}{\partial \theta}.$$

We finally arrive at the following expression of the gradient

$$\nabla f_{\text{obj}}(\theta) = -\sum_s \sum_r \sum_i \frac{r_{\text{BM}}}{\sigma_{\text{BM}}} \frac{1}{y_{s,r}^{\text{BM}}} \left[\frac{\partial y_{s,r}^{\text{BM}}}{\partial \theta}\right]^{\mathsf{T}} + \frac{r_{\text{G}}}{\sigma_{\text{G}}}\left[\frac{\partial y_{s,r}^{\text{G}}}{\partial \theta}\right]^{\mathsf{T}} + \frac{r_{\text{M}}}{\sigma_{\text{M}}}\left[\frac{\partial y_{s,r}^{\text{M}}}{\partial \theta}\right]^{\mathsf{T}} \tag{8.5}$$

where the sensitivities can be computed as

$$\begin{aligned}
\frac{\partial y_{s,r}^{\text{BM}}}{\partial \theta} &= \frac{\partial y_{s,r}^{\text{BM}}}{\partial x}\frac{\partial x_{s,r}}{\partial \tilde{\theta}}\frac{\partial a_{s,r}}{\partial \theta}\\
\frac{\partial y_{s,r}^{\text{G}}}{\partial \theta} &= \frac{\partial y_{s,r}^{\text{G}}}{\partial x}\frac{\partial x_{s,r}}{\partial \tilde{\theta}}\frac{\partial a_{s,r}}{\partial \theta}\\
\frac{\partial y_{s,r}^{\text{M}}}{\partial \theta} &= \frac{\partial y_{s,r}^{\text{M}}}{\partial x}\frac{\partial x_{s,r}}{\partial \tilde{\theta}}\frac{\partial a_{s,r}}{\partial \theta}.
\end{aligned}$$

We transpose the sensitivities in (8.5) since we interpret the expressions $\partial y_{s,r}^{\text{BM}}/\partial x$, $\partial y_{s,r}^{\text{G}}/\partial x$, and $\partial y_{s,r}^{\text{M}}/\partial x$ as row vectors, selecting effectively the respective rows of the sensitivity matrix $\tilde{S}$ that is returned by the ODE solver. Transposing makes the gradient a column vector, which is the common convention.

Estimating Parameter Uncertainty

Once problem (8.4) is solved, we have to assess the “quality” of its minimizer $\hat{\theta}$. This is usually done by specifying confidence intervals in which the true parameter value is to be found with high probability.

It is generally only possible to give exact confidence intervals for linear problems. Since we are dealing with a highly nonlinear problem, we have to accept approximate intervals. There are several approaches available for deriving approximations of confidence intervals. To simplify exposition, collect all residuals $r_{s,r,i}^{\text{BM}}$, $r_{s,r,i}^{\text{G}}$, and $r_{s,r,i}^{\text{M}}$ in one large vector $r \in \mathbb{R}^n$ and rewrite objective (8.4) as

$$f_{\text{obj}}(\theta) = \frac{1}{2}r^{\mathsf{T}}r.$$

We assume further that the scalings σ_{BM}, σ_{G}, and σ_{M} are chosen such that the variance estimate at the optimal $\hat{\theta}$ is equal to one, i. e.

$$\frac{r^{\mathsf{T}}(\hat{\theta})r(\hat{\theta})}{n-p} = 1. \tag{8.6}$$

The problem here is that the residual scalings $\hat{\sigma}_i$ cannot be chosen independently from $\hat{\theta}$ to make the above equation hold. This means that we would need to solve the problem for $\hat{\sigma}_i$ and $\hat{\theta}$ simultaneously. A simpler alternative is to fix some $\hat{\sigma}_i$, then solve for $\hat{\theta}$. Once we have a value for $\hat{\theta}$, we can adjust the $\hat{\sigma}_i$ such that Eq. (8.6) holds. Then, we can iterate by computing a new $\hat{\theta}$. In practice, this works very well, and we find that 2-3 iterations suffice to satisfy Eq. (8.6) "well enough" (i.e. such that the $\hat{\sigma}_i$ before and after reoptimization differ by less than 2%).

When Eq. (8.6) holds, the two most widely used approximations for the parameter covariance can be written as

$$\hat{V}_1 \triangleq \left(\left(\nabla r(\hat{\theta}) \right)^{\mathsf{T}} \left(\nabla r(\hat{\theta}) \right) \right)^{-1}$$

$$\hat{V}_2 \triangleq \frac{1}{2} \left(\nabla^2 f_{\mathrm{obj}}(\hat{\theta}) \right)^{-1} .$$

The former estimate $\hat{V}_1$ is based on applying linear least-squares directly to a first-order approximation of the objective function around $\hat{\theta}$. The latter approximation $\hat{V}_2$ can be derived from maximum-likelihood considerations [16]. Both estimates are identical if the underlying model is linear.

Since we already know how to compute the gradients of the residual, using the first approximation $\hat{V}_1$ is more convenient. For $\hat{V}_2$, we would need to derive second order information, which is much more involved algorithmically as well as numerically. Also, in [16], the authors claim that

> "among the variants of the linearization method, the one using $\hat{V}_1$ is the best choice because it is the cheapest, and for each dataset tested it gives results that are never considerably worse and are sometimes considerably better than the other variants."

Using the covariance estimate $\hat{V}_1$, it is possible to specify approximate confidence intervals for the parameters θ_i of the form

$$P\left[\left| \theta_i - \hat{\theta}_i \right| \le z_\delta \sqrt{\left(\hat{V}_1 \right)_{ii}} \right] = 1 - \delta$$

where δ is the desired confidence level. The value z_δ is the critical value $t(n-p, 1-\delta/2)$ from a Student's t-distribution with $n-p$ degrees of freedom (see Appendix A for a more detailed derivation in the linear case).

Issues & Discussion

Solving the regression problem numerically turned out to be difficult. We experimented with different non-linear solvers (`fminunc` and `fmincon` from the Matlab optimization toolbox [40] and KNITRO [9]). The ODE sensitivities were always computed using CVODE.

An obvious problem is the unidentifiability of the system (cf. § 8.2). This implies that the Hessian $\nabla^2 f_{\mathrm{obj}}(\theta)$ is singular at a local optimum making it more difficult for solvers to converge. Using the scaling (8.2) for the parameter c_{E}, the system becomes locally identifiable and the objective function of the resulting regression problem should thus have unique (local) minima.

However, even with this improvement, the system parameters were difficult to identify practically. Therefore, we assigned to parts of the parameters fixed values shown in Tbl. 8.2. This reduces the dimension of the optimization parameter to 22 (4 system parameters, 18 initial conditions).

Table 8.2: Fixed system parameters.

parameter	value	parameter	value
$\eta_{\text{G}}/\eta_{\text{M}}$	5	a_γ^{G}	0.1
c_{E}	$1/\rho$	a_γ^{M}	0.1
d	0.1	K_λ	0.1
a_λ^{G}	0.1	K_ν	0.1
a_λ^{M}	0.8		

Table 8.3: Estimated system parameters. Note that the ratio $\eta_{\text{G}}/\eta_{\text{M}}$ was held fixed during the optimization. The confidence interval is with respect to a confidence level of 5%.

parameter	value	confidence interval
η_{G}	0.1285	± 0.0203
η_{M}	0.0257	± 0.0041
ρ	0.0611	± 0.0977
k_{G}	6.3103	± 0.2102
k_{M}	38.9689	± 1.2336

We give the following justification for fixing these system parameters to the proposed values. Fixing c_{E} goes without loss of generality, as shown in § 8.2. We approximated the ratio of $\eta_{\text{G}}/\eta_{\text{M}}$ as well as the values of a_λ^{G}, a_λ^{M}, a_γ^{G} and a_γ^{G} from uptake measurements of independent steady-state batch culture experiments [38] (see also Tbl. 8.1). The dissociation constants K_λ and K_ν were fixed to values small enough such that high external glucose and malate concentrations drive the kinetic expressions into saturation. The exact value of these parameters has basically no influence on the regression estimates of the system trajectory. From the Big Experiment data, the ratio between the maximum uptake speeds k_{G}, k_{M} and the degradation rate d is practically not identifiable. We therefore fixed d to an arbitrary value of 0.1.

In Fig. 8.4, we show the system trajectories of a simulation using the parameter values of Tbls. 8.2 and 8.3 for reproducing the measurement data of the Big Experiment. The model is able to explain the data relatively well, except for the biomass trajectory in the $M \to G$ shift after around 1.5h from addition of the second substrate. There, the simulation predicts a slowdown in growth whereas the measured data does not indicate such an effect. The likely reason for this is the depletion of malate in the reactor. In the experimental data, one can see at the same time beginning uptake of pyruvate (cf. Figs. 7.7 and 7.8), acting as a replacement of malate for keeping the growth rate at a constant level. Since this effect is not modeled in the simulated system, the observed discrepancy occurs.

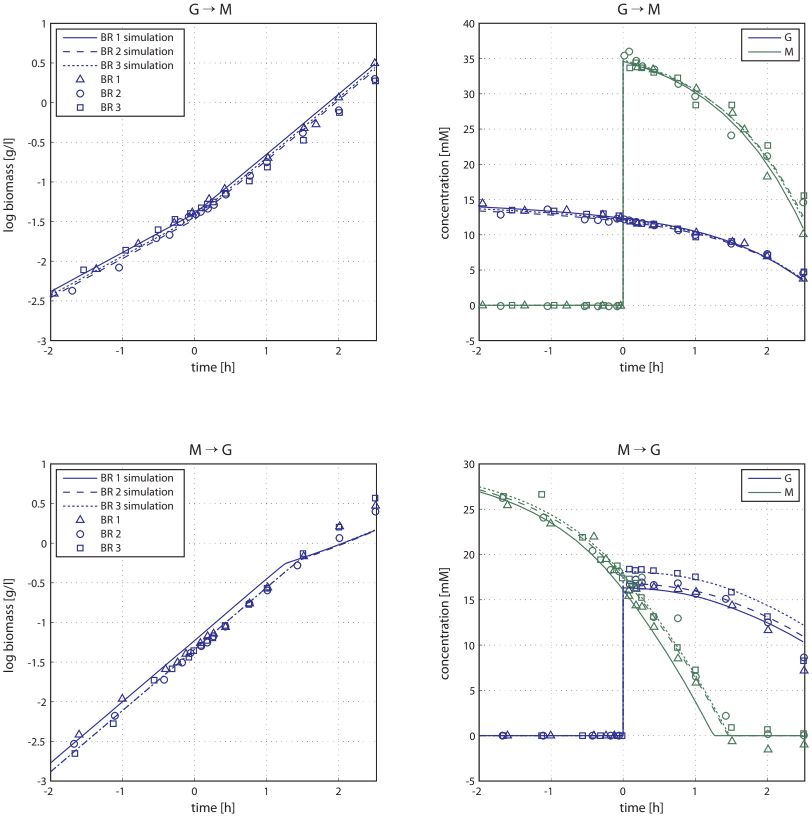


Figure 8.4: Simulated model trajectories for reproducing the measured data.

9

Evolutionary Analysis

THE MAIN CONCLUSIONS from the data analysis of the Big Experiment so far are: *B. subtilis* is prepared for malate substrate where we can observe immediate uptake upon addition, but not for glucose where we see a gradual increase in uptake speed over time. We attribute this difference to two distinct control strategies of its subsystems. A natural question to ask is whether this difference is just a whim of nature or whether there is a deeper biological reason for this.

The goal of this section is to derive model based support for the hypothesis that the observed control strategy is advantageous in an evolutionary context. To test this, we need to define alternative strategies to be compared with the wild-type strategy. As described in the discussion of the single strain model, the subsystem control strategies are directly related to the parameters a_λ^s in the $\lambda(\cdot)$ functions. Since we have two subsystems (glucose and malate) and two control strategies (constitutive expression and regulated expression), a natural way to create alternative strategies is to assign the control strategies to subsystems in different combinations. This leads to four *in silico* strains in total, shown in Tbl. 9.1.

Using these strains, we can test their fitness by letting them compete for limited nutrient resources. To keep the evaluation tractable, we will only look at two strains competing directly. To this end, we need to be able to simulate two strains growing in a bioreactor at the same time, which will be the topic of the following subsection. Then, we need to define a simulation scenario that allows us to evaluate the strain fitness. Finally, we will need to aggregate the results of the pairwise comparisons to an analysis encompassing all four strains.

Table 9.1: Four *in silico* strains of *B. subtilis* with alternate strategies for substrate uptake. Note that strain 2 corresponds to the wild-type strategy.

subsystem	strain 1	strain 2	strain 3	strain 4
Glc	regulated	regulated	constitutive	constitutive
Mal	regulated	constitutive	regulated	constitutive

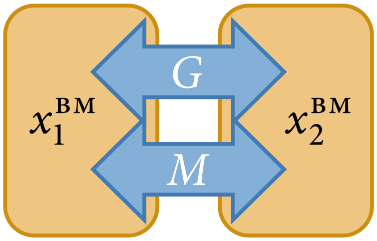


Figure 9.1: Conceptual setting for simulating strain fitness. Two different strains (x_1^{BM}, x_2^{BM}) are competing for the available substrate (G, M).

9.1 Double Strain Model

We want to derive the system equations for two strains growing in a bioreactor simultaneously. Each strain will be characterized by three states; the two subsystems for growth competence and the biomass. To differentiate between the two stains in the reactor, we introduce indices $i, j \in \{1, 2, 3, 4\}$. Therefore, the state vector of the double strain model has eight states

$$\left(e_i^{\mathrm{G}}, e_i^{\mathrm{M}}, x_i^{\mathrm{BM}}, e_j^{\mathrm{G}}, e_j^{\mathrm{M}}, x_j^{\mathrm{BM}}, x_{\mathrm{G}}, x_{\mathrm{M}}\right)^T.$$

We assume that the strains grow independently from each other, i. e. there is no direct interaction between the quantities $\left(e_i^{\mathrm{G}}, e_i^{\mathrm{M}}, x_i^{\mathrm{BM}}\right)$ and $\left(e_j^{\mathrm{G}}, e_j^{\mathrm{M}}, x_j^{\mathrm{BM}}\right)$ for any $i \neq j$. Of course, the available substrate $(x_{\mathrm{G}}, x_{\mathrm{M}})$ is used by both strains.

For completeness, we give the full set of equations for two strains i, j. Note that we have included the scaling $\rho = 1/c_{\mathrm{E}}$ which simplifies the equations a bit.

$$\dot{x}_i^{\mathrm{BM}} = \eta_{\mathrm{G}} \frac{1}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_i^{\mathrm{G}} x_i^{\mathrm{BM}} + \eta_{\mathrm{M}} \frac{1}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_i^{\mathrm{M}} x_i^{\mathrm{BM}} - d x_i^{\mathrm{BM}} \tag{9.1a}$$

$$\dot{e}_i^{\mathrm{G}} = \eta_{\mathrm{G}} \frac{\lambda(x_{\mathrm{G}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_i^{\mathrm{G}} x_i^{\mathrm{BM}} + \eta_{\mathrm{M}} \frac{\lambda(x_{\mathrm{G}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_i^{\mathrm{M}} x_i^{\mathrm{BM}} - d e_i^{\mathrm{G}} \tag{9.1b}$$

$$\dot{e}_i^{\mathrm{M}} = \eta_{\mathrm{G}} \frac{\lambda(x_{\mathrm{M}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_i^{\mathrm{G}} x_i^{\mathrm{BM}} + \eta_{\mathrm{M}} \frac{\lambda(x_{\mathrm{M}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_i^{\mathrm{M}} x_i^{\mathrm{BM}} - d e_i^{\mathrm{M}} \tag{9.1c}$$

$$\dot{x}_j^{\mathrm{BM}} = \eta_{\mathrm{G}} \frac{1}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_j^{\mathrm{G}} x_j^{\mathrm{BM}} + \eta_{\mathrm{M}} \frac{1}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_j^{\mathrm{M}} x_j^{\mathrm{BM}} - d x_j^{\mathrm{BM}} \tag{9.1d}$$

$$\dot{e}_j^{\mathrm{G}} = \eta_{\mathrm{G}} \frac{\lambda(x_{\mathrm{G}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_j^{\mathrm{G}} x_j^{\mathrm{BM}} + \eta_{\mathrm{M}} \frac{\lambda(x_{\mathrm{G}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_j^{\mathrm{M}} x_j^{\mathrm{BM}} - d e_j^{\mathrm{G}} \tag{9.1e}$$

$$\dot{e}_j^{\mathrm{M}} = \eta_{\mathrm{G}} \frac{\lambda(x_{\mathrm{M}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_j^{\mathrm{G}} x_j^{\mathrm{BM}} + \eta_{\mathrm{M}} \frac{\lambda(x_{\mathrm{M}})}{1 + \rho\lambda(x_{\mathrm{G}}) + \rho\lambda(x_{\mathrm{M}})} v_j^{\mathrm{M}} x_j^{\mathrm{BM}} - d e_j^{\mathrm{M}} \tag{9.1f}$$

$$\dot{x}_{\mathrm{G}} = -v_i^{\mathrm{G}} x_i^{\mathrm{BM}} - v_j^{\mathrm{G}} x_j^{\mathrm{BM}} \tag{9.1g}$$

$$\dot{x}_{\mathrm{M}} = -v_i^{\mathrm{M}} x_i^{\mathrm{BM}} - v_j^{\mathrm{M}} x_j^{\mathrm{BM}} \tag{9.1h}$$

9.2 Testing Strain Fitness

For designing an “experimental” setup to test the strains’ fitness, we need to consider that the main difference between strains is their uptake strategy. The goal is to find a setup that exhibits these differences. This is possible by making the strains switch between the two substrates, analogous to the Big Experiment. Therefore, we need to make sure that there are periods during the simulation when substrate is available, and periods with no substrate.

Furthermore, we want to test strain fitness in an evolutionary setting. This means that we need repeated switches between substrates, akin to repeated games in a game theoretic sense. It is then possible to study the behavior of the system in the long run: will there emerge certain patterns such as cooperation, limit cycles, extinction of one (or possibly both!) strains?

A solution to all these considerations is to simulate the system under periodic pulses of substrate where the kind of substrate is chosen randomly (the probability of choosing one or the other substrate is a simulation parameter). In this setting, there are periods of substrate availability (shortly after a substrate pulse) and substrate scarcity (shortly before the next pulse). The cells will have to adjust to the added substrate repeatedly whenever the newly added substrate is different from the previous pulse.

Figure 9.2 displays an example trajectory of the system simulated under these conditions. In this example, the pulse period is three hours and substrate probabilities are equal between glucose and malate. The pulse weights of glucose and malate are chosen inversely proportional to the yields such that the same amount of biomass results from each pulse (modulo the resources spent on the subsystems). One can see that the amount of biomass in the reactor grows to a value where all substrate is used quickly and most of the time, the cells are starved (this is a general observation, not just valid for this example).

9.3 Strain Dominance

Since we are interested in testing the different strains’ fitness, a natural question to ask is: when is a particular strain fitter than all other three strains? We call this situation *strain dominance*.

To answer this question, we conducted a systematic study of all 6 pair-combinations of the 4 strains. For each combination, we simulated their behavior under different conditions. In particular, we were interested in the influence of three parameters: the subsystem cost ρ for maintaining the competence of growing on a substrate (because the control strategies of these subsystems are the main distinguishing features of the strains), the probability for choosing a substrate when adding a pulse, and the pulse period itself. We divided the parameter space in a grid of 11 points for the cost factor ρ (from factor 1/100 to 100 of estimated (“true”) cost ρ_0), 11 points for the substrate probability (from 0% chance of adding malate to 100%) and 3 points for the pulse period (1, 3 and 6 hours). Doing this for all 6 pairs of strains results in a total of 2178 different configurations. Because of the randomized substrate selection, we have to run multiple simulations for each configuration to make conclusions of the “expected” behavior. To keep the computational workload manageable, we repeated each configuration 30 times.

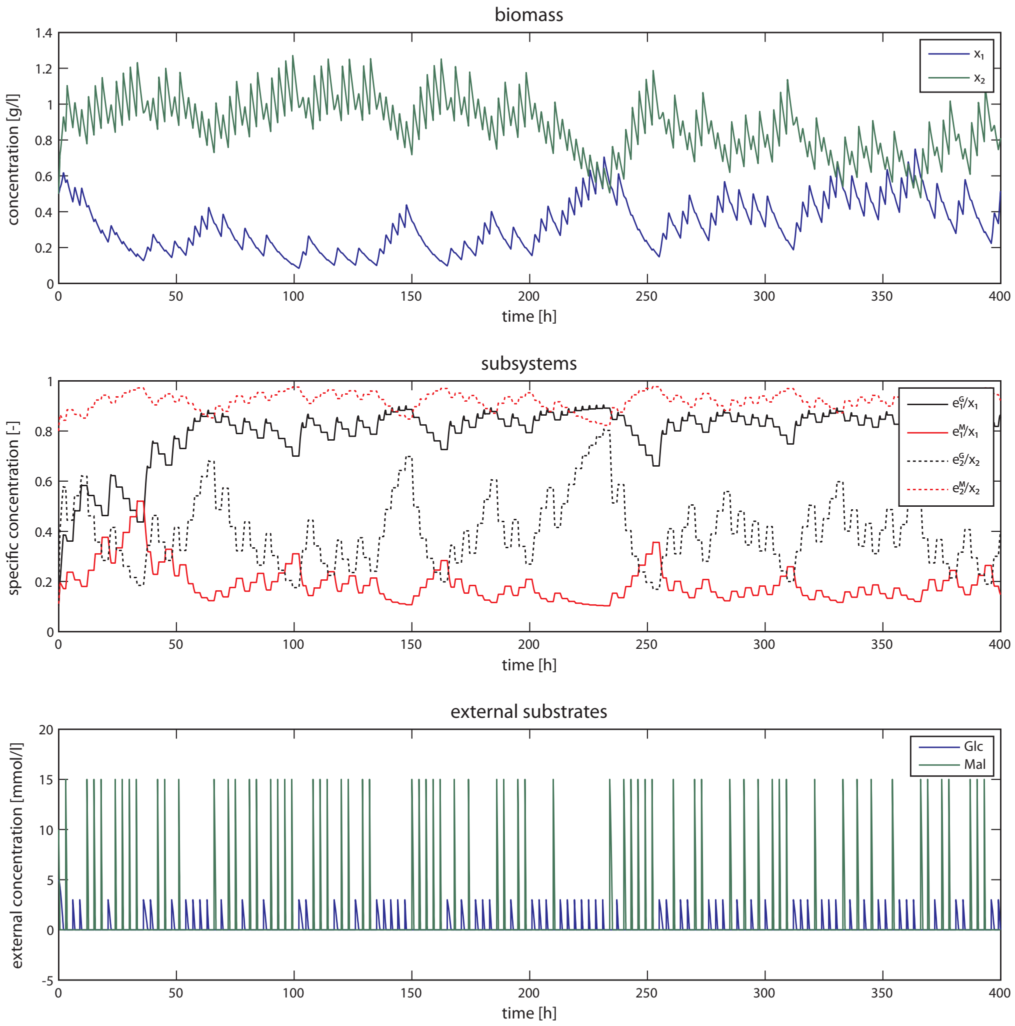


Figure 9.2: Example trajectory of strains 1 and 2 growing. Upper panel shows biomass, middle panel specific subsystem "concentration", lower panel external substrate concentration. Pulse period is 3h, substrate probability is equal between glucose and malate. Total simulated time is 400 hours.

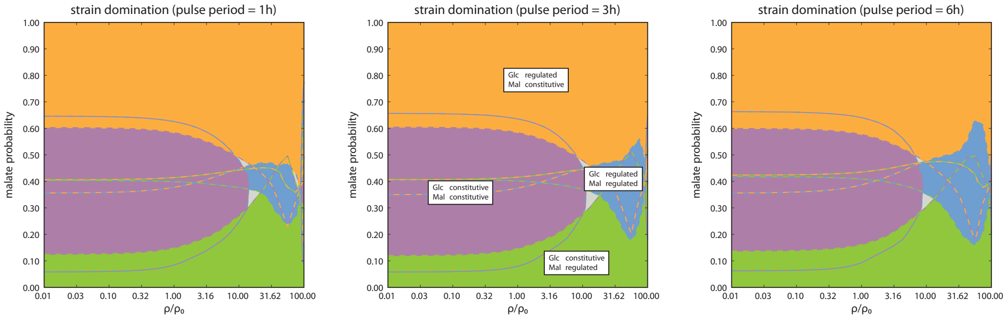


Figure 9.3: Plots showing strain dominance across the examined parameter region. Colored areas denote regions with single strain dominance (small gray areas mean no dominance). Dashed lines are demarcation lines between strain pairs.

The evaluation consisted of averaging the biomass contents of the reactor over the last 200h of a 1000h simulation and then taking the mean over all 30 repetitions. Because of the oscillating behavior of the system caused by the periodic excitations of the substrate pulses, taking just the biomass contents after 1000 hours would increase the variance of the result.

To decide about strain dominance, we looked at the ratio $x_i^{\mathrm{BM}}/x_j^{\mathrm{BM}}$ of the averaged biomass contents. If the ratio is larger than 1, we say that strain i dominates strain j; if the ratio is smaller than 1, we say that strain j dominates strain i.

The results are shown in Fig. 9.3. The colored areas of the plot denote the regions with strain dominance, i.e. where one strain performs consistently better than the other three (the small gray areas denote areas with no dominance). The boundaries of the colored regions are defined by the lines $x_i^{\mathrm{BM}}/x_j^{\mathrm{BM}} = 1$. The color pattern of the dashed lines indicate which strains are involved. Note that these lines are approximate in two ways. First, they are random because they are based on the average of 30 random simulations and second, they are bilinearly interpolated from the grid points where the fitness function was actually evaluated.

A curious phenomenon happens for subsystem costs $\rho/\rho_0 \gtrsim 50$. In this case, the denominator in Eqs. (9.1a)-(9.1f) becomes so large that the resources spent on generic biomass cannot account for the degeneration rate d. This means that $x_i^{\mathrm{BM}}, x_j^{\mathrm{BM}} \to 0$ as $t \to \infty$ implying that both strains get extinct. We call a cost causing this effect *lethal cost*. Because of that, in regions of Fig. 9.3 with $\rho/\rho_0 \gtrsim 50$ the system behavior is qualitatively different than what the colors might imply and must be interpreted with care. In particular, these regions are not biologically meaningful.

9.4 Conclusions

The first and most obvious conclusion of this analysis is probably that the success of the switching strategy depends mostly on the ratio of glucose/malate probability. The other parameters pulse period and subsystem cost are not (period) or much less (cost) influential on strain dominance.

Only a significant increase in subsystem cost can affect strategy 4 (both subsystems constitutively expressed).

A more interesting observation is that in most regions, one strategy dominates the other three. This is a surprising result as it could have been that there are large indeterminate areas with no dominance. This is a strong hint that only one strain is possible in nature, if the ecosystem of the bacterium is not changing. Of course, this is not a mathematically rigorous claim; but then, it is difficult to see how one could arrive at this without a large computational study.

The most reassuring observation is that the wild-type strategy dominates in regions that are similar to *B. subtilis*' natural habitat. This is a clear indication of consistency of evidence gathered in the Big Experiment.

Appendices

A

Linear Regression

In this appendix, we derive many probability theoretic properties of regression problems when the problems are linear and the nuisance variables are i. i. d. normally distributed. In this case, the results are exact. Most of the results presented here are used in the main text.

Assume the following “true” linear process

$$\bar{y}(t) = \sum_k \alpha_k b_k(t)$$

with known basis functions $b_k(t)$ but unknown parameters α_k, $k = 1, \dots, p$ and n noisy measurements

$$y_i = \bar{y}(t_i) + \sigma\varepsilon_i = \sum_k \alpha_k b_k(t_i) + \sigma\varepsilon_i \qquad i = 1, \dots, n.$$

We assume that the noise variables ε_i are i. i. d. $\mathcal{N}(0,1)$ distributed. Using matrix notation, we can collect the measurements in a vector $y = (y_1, \dots, y_n)^\mathsf{T}$, the parameters in $\alpha = (\alpha_1, \dots, \alpha_p)^\mathsf{T}$ and the error terms in $\varepsilon = (\varepsilon_1, \dots, \varepsilon_n)^\mathsf{T} \sim \mathcal{N}(0, I_n)$, which have a multivariate normal distribution with covariance matrix I_n (the n-dimensional identity matrix). Define further the basis matrix $B \in \mathbb{R}^{n \times p}$ with elements

$$(B)_{ij} = b_j(t_i).$$

This lets us rewrite the above equation as

$$y = B\alpha + \sigma\varepsilon.$$

We assume that the system is identifiable, i. e. that rank $B = p$.

The standard least-squares estimate for the parameters α is

$$\hat{\alpha} = (B^\mathsf{T} B)^{-1} B^\mathsf{T} y.$$

The estimate $\hat{\alpha}$ is a random variable that has a normal distribution. In particular, we have

$$\alpha - \hat{\alpha} \sim \mathcal{N}(0, \sigma^2 (B^\mathsf{T} B)^{-1}).$$

The estimated value $\hat{y}(t)$ of the function at arbitrary time t is therefore given as

$$\hat{y}(t) = b^\mathsf{T}(t)\hat{\alpha}.$$

If we define the hat matrix H as

$$H = B(B^\top B)^{-1}B^\top$$

we can write the function estimates $\hat{y} = (\hat{y}(t_1), \dots, \hat{y}(t_n))^\top$ at the measurement time points t_i as

$$\hat{y} = Hy.$$

The hat matrix is an orthogonal projection from $\mathbb{R}^n$ to the p-dimensional subspace spanned by the range of B. Since the parameter vector $\hat{\alpha}$ is a normal random variable, so is the prediction $\hat{y}(t)$. We have

$$\bar{y}(t) - \hat{y}(t) \sim \mathcal{N}(0, \sigma^2 b^\top(t)(B^\top B)^{-1}b(t))$$

At the measurement time points t_i, we get

$$\bar{y}(t_i) - \hat{y}(t_i) \sim \mathcal{N}(0, \sigma^2 h_{ii}),$$

where h_{ii} is the ith diagonal element of the hat matrix H. The residual vector r is defined as the difference between the measurements and the predicted values, i. e.

$$r = y - \hat{y}$$

which can be written using the hat matrix as

$$r = (I - H)y.$$

Since H is a projection matrix, $I - H$ is also a projection into the orthogonal complement of range of B, which is an $n - p$ dimensional subspace. The residual vector r is thus also a random variable with distribution

$$r \sim \mathcal{N}\left(0, \sigma^2(I - H)\right).$$

Note that even though r is an n-dimensional vector, the covariance matrix $(I - H)$ only has rank $n - p$. Since the vectors $\hat{y}$ and r lie in orthogonal subspaces, they are independent random variables. Using the residuals r, we can get an estimate for the variance σ^2 of the process as

$$\hat{\sigma}^2 = \frac{1}{n-p}\sum_i r_i^2 = \frac{1}{n-p} r^\top r.$$

To find the distribution of $\hat{\sigma}^2$, we have to analyze the hat matrix H and the residual vector in more detail. We have for the residuals

$$r = (I - H)y = (I - H)(\bar{y} + \sigma\varepsilon) = \sigma(I - H)\varepsilon.$$

We know that H is an orthogonal projection matrix into a p-dimensional subspace. Therefore, the Eigendecomposition of H has the following form

$$H = U\Lambda U^\top = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U_1^\top \\ U_2^\top \end{bmatrix}$$

where U is orthonormal and the splitting is such that $U_1 \in \mathbb{R}^{n\times p}$ and $U_2 \in \mathbb{R}^{n\times(n-p)}$. We thus get

$$\begin{aligned} r^\top r &= \sigma^2 \varepsilon^\top \begin{bmatrix} U_1 & U_2 \end{bmatrix} \left(\begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} - \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \right) \begin{bmatrix} U_1^\top \\ U_2^\top \end{bmatrix} \varepsilon \\ &= \sigma^2 \varepsilon^\top U_2 U_2^\top \varepsilon. \end{aligned}$$

Since the columns of U_2 are orthonormal, the value $U_2^\top \cdot \varepsilon \in \mathbb{R}^{n-p}$ has a distribution

$$U_2^\top \cdot \varepsilon \sim \mathcal{N}(0, I_{n-p}).$$

This shows that summing up the squared residuals is equivalent to summing $n-p$ i. i. d. standard normal random variables. One can thus directly see that

$$(n-p)\frac{\hat{\sigma}^2}{\sigma^2} \sim \chi^2_{n-p},$$

i. e. it is χ^2-distributed with $n-p$ degrees of freedom.

Using the knowledge about the distributions of the parameters $\hat{\alpha}$ and the residuals r, it is possible to specify confidence intervals. Of interest are single parameter confidence intervals for $\hat{\alpha}_i$ and simultaneous confidence regions of the whole parameter vector $\hat{\alpha}$.

For the parameter $\hat{\alpha}$, we want to specify the region for which

$$P\left[(\alpha - \hat{\alpha})^\top B^\top B(\alpha - \hat{\alpha}) \le z_\delta p \hat{\sigma}^2\right] = 1 - \delta,$$

where α is the true parameter value, p is the dimension of the parameter vector and z_δ is a critical value to be determined. Dividing both sides of the inequality with σ^2 yields

$$P\left[\frac{1}{\sigma^2}(\alpha - \hat{\alpha})^\top B^\top B(\alpha - \hat{\alpha}) \le z_\delta p \frac{\hat{\sigma}^2}{\sigma^2}\right] = 1 - \delta.$$

Making the substitutions

$$\begin{aligned} u &\triangleq (\alpha - \hat{\alpha})^\top \frac{B^\top B}{\sigma^2}(\alpha - \hat{\alpha}) \\ v &\triangleq (n-p)\frac{\hat{\sigma}^2}{\sigma^2} \end{aligned}$$

lets us rewrite the condition as

$$P\left[\frac{u/p}{v/(n-p)} \le z_\delta\right] = 1 - \delta.$$

Since we know that

$$u \sim \chi^2_p \qquad \text{and} \qquad v \sim \chi^2_{n-p}$$

we see that

$$\frac{u/p}{v/(n-p)}$$

is the definition of an $F_{p,n-p}$-distributed random variable. We must therefore choose z_δ as the critical value $F(p, n-p, 1-\delta)$ of an F-distribution with parameters p and $n-p$.

We turn now to the second problem of giving confidence intervals for a single parameter $\hat{\alpha}_i$. However, we give a more general derivation of intervals for arbitrary linear combinations $w^\top \hat{\alpha}$ for a fixed vector $w \in \mathbb{R}^p$ of which a single parameter interval is just a special case (by choosing $w = e_i$, the ith canonical basis vector). It is possible to give a confidence interval of the form

$$P\left[\left|w^\top(\alpha - \hat{\alpha})\right| \le z_\delta \hat{\sigma}\sqrt{w^\top (B^\top B)^{-1} w}\right] = 1 - \delta.$$

Dividing by σ and rearranging yields

$$P\left[\frac{\left|w^\top(\alpha - \hat{\alpha})\right|}{\sigma\sqrt{w^\top (B^\top B)^{-1} w}} \le z_\delta \frac{\hat{\sigma}}{\sigma}\right] = 1 - \delta.$$

Making substitutions

$$\begin{aligned} u &\triangleq \frac{w^\top(\alpha - \hat{\alpha})}{\sigma\sqrt{w^\top (B^\top B)^{-1} w}} && \sim \mathcal{N}(0,1) \\ v &\triangleq (n-p)\frac{\hat{\sigma}^2}{\sigma^2} && \sim \chi^2_{n-p} \end{aligned}$$

leads to

$$P\left[|u| \le z_\delta \sqrt{\frac{v}{n-p}}\right] = 1 - \delta.$$

Using symmetry of the normal distribution, we can remove the absolute value and rewrite this as

$$P\left[u \ge z_\delta \sqrt{\frac{v}{n-p}}\right] = \frac{\delta}{2}.$$

Since

$$\frac{u}{\sqrt{v/(n-p)}}$$

is the definition of a t_{n-p}-distributed random variable, we must choose z_δ as the critical value $t(n-p, 1-\delta/2)$ of a Student's t-distribution with $n-p$ degrees of freedom.

B

Leave-one-out Formulas in Linear Regression

In many applications of regression, e.g. in cross-validation or outlier detection, it is important to find the statistical quantities discussed in the previous appendix *while leaving out point i in the estimation*. Usually, one is interested in all n regression estimates when each sample i has been left out in turn. When implemented in a naive way, the statistical estimations have to be repeated n times using $n-1$ sample points, which is an expensive process.

The central concern of this appendix is to find formulas of the statistical quantities when the ith sample point has been removed, based on an estimate taken over all n points. This can potentially save a lot of computational work when the calculation is to be repeated n times. The general approach for deriving the formulas is to understand the removal of a sample point as a rank one modification of basis matrix B and to apply the Sherman-Morisson formula to the inverses $(B^\top B)^{-1}$.

We begin with the parameter estimates $\hat{\alpha}$. Let denote $\hat{\alpha}^{-i}$ the estimate when point i has been removed from estimation. Then, we have

$$\begin{aligned}\hat{\alpha}^{-i} &= (B^\top B - b_i^\top b_i)^{-1}(B^\top - b_i^\top e_i)y\\ &= \left(B^+ + \frac{B^+ b_i^\top b_i B^+}{1 - b_i^\top B^+ b_i}\right)(B^\top - b_i^\top e_i)y,\end{aligned}$$

where we have written $B^+ = (B^\top B)^{-1}$. The vector $b_i \in \mathbb{R}^p$ is the ith row of the basis matrix B, understood as a column vector, and $e_i \in \mathbb{R}^n$ denotes the ith canonical basis vector. Note that $b_i^\top B^+ b_i = h_{ii}$ is the ith diagonal element of the hat matrix. Factoring out we get

$$\begin{aligned}\hat{\alpha}^{-i} &= B^+ B^\top y - B^+ b_i e_i^\top y + \frac{1}{1-h_{ii}} B^+ b_i b_i^\top B^+ B^\top y - \frac{1}{1-h_{ii}} B^+ b_i b_i^\top B^+ b_i e_i^\top y\\ &= \hat{\alpha} - B^+ b_i \cdot y_i + \frac{1}{1-h_{ii}} B^+ b_i b_i^\top \hat{\alpha} - \frac{1}{1-h_{ii}} B^+ b_i \cdot h_{ii} y_i\\ &= \hat{\alpha} - B^+ b_i \left(y_i - \frac{1}{1-h_{ii}} \hat{y}_i + \frac{h_{ii}}{1-h_{ii}} y_i\right)\\ &= \hat{\alpha} - B^+ b_i \frac{1}{1-h_{ii}} (y_i - \hat{y}_i)\\ &= \hat{\alpha} - B^+ b_i \frac{r_i}{1-h_{ii}}.\end{aligned}$$

Note that all quantities $\hat{\alpha}$, r_i, and h_{ii} in the final expression are based on the estimate with all n points. Therefore, we can compute $\hat{\alpha}^{-i}$ as a simple modification of the original estimate $\hat{\alpha}$.

Using $\hat{\alpha}^{-i}$, we can compute the function estimates $\hat{y}^{-i}(t)$ without point i as

$$\hat{y}^{-i}(t) = b^{\mathsf{T}}(t)\hat{\alpha}^{-i}.$$

Of particular interest is the ith residual r_i when point i was not used in the estimation process. We have

$$\begin{aligned} r_i^{-i} &= y_i - \hat{y}^{-i}(t_i) \\ &= y_i - b_i\left(\hat{\alpha} - B^{+}b_i\frac{r_i}{1-h_{ii}}\right) \\ &= y_i - \hat{y}_i + h_{ii}\frac{r_i}{1-h_{ii}} \\ &= \frac{r_i}{1-h_{ii}}. \end{aligned}$$

The most complicated update is the estimate of the standard deviation $\hat{\sigma}$. To derive a formula for it, we have to analyze the updating of the residual vector r^{-i}:

$$\begin{aligned} r^{-i} &= y - \hat{y}^{-i} = y - B\hat{\alpha}^{-i} \\ &= y - B\left(\hat{\alpha} - B^{+}b_i\frac{r_i}{1-h_{ii}}\right) \\ &= y - \hat{y} + BB^{+}b_i\frac{r_i}{1-h_{ii}} \\ &= r + BB^{+}b_i\frac{r_i}{1-h_{ii}}. \end{aligned}$$

Note that this includes the residual r_i^{-i} of the omitted sample point i. The inner product of this vector is thus

$$\begin{aligned} (r^{-i})^{\mathsf{T}}r^{-i} &= r^{\mathsf{T}}r + \frac{r_i}{1-h_{ii}}\left(r^{\mathsf{T}}BB^{+}b_i + b_i^{\mathsf{T}}B^{+}B^{\mathsf{T}}r\right) + \frac{r_i^2}{(1-h_{ii})^2}b_i^{\mathsf{T}}B^{+}B^{\mathsf{T}}BB^{+}b_i \\ &= r^{\mathsf{T}}r + \frac{h_{ii}}{(1-h_{ii})^2}r_i^2. \end{aligned}$$

The middle term in the upper line is zero since the residual vector r is orthogonal to the range of the basis matrix B. We therefore get for the updated estimate of the standard deviation $\hat{\sigma}^{-i}$

$$\begin{aligned} (n-p-1)\left(\hat{\sigma}^{-i}\right)^2 &= (r^{-i})^{\mathsf{T}}r^{-i} - (r_i^{-i})^2 \\ &= r^{\mathsf{T}}r + \frac{h_{ii}}{(1-h_{ii})^2}r_i^2 - \frac{1}{(1-h_{ii})^2}r_i^2 \\ &= (n-p)\hat{\sigma}^2 - \frac{r_i^2}{1-h_{ii}}. \end{aligned}$$

Application: Derivation of Studentized residuals

One application of the leave-one-out update formulas is outlier detection. A robust way to assess the quality of data point i is to look at the residual when the point has not been used for the regression.

From the covariance matrix $\sigma^2(I-H)$ of the residuals r one sees that

$$\operatorname{Var} r_i = \sigma^2(1-h_{ii}).$$

Therefore, the residual r_i^{-i} has

$$\operatorname{Var} r_i^{-i} = \frac{\sigma^2}{1-h_{ii}}.$$

Furthermore, the distribution of the updated standard deviation is given as

$$(n-p-1)\frac{\left(\hat{\sigma}^{-i}\right)^2}{\sigma^2} \sim \chi^2_{n-p-1}.$$

The ratio

$$\begin{aligned}\tau_i &= \frac{\frac{\sqrt{1-h_{ii}}}{\sigma} r_i^{-i}}{\sqrt{\frac{(\hat{\sigma}^{-i})^2}{\sigma^2}}} \\ &= \frac{\sqrt{1-h_{ii}}\,\frac{r_i}{1-h_{ii}}}{\hat{\sigma}^{-i}} \\ &= \frac{r_i}{\hat{\sigma}^{-i}\sqrt{1-h_{ii}}}\end{aligned}$$

is called *Studentized residual* and has a Student's t-distribution with $n-p$ degrees of freedom.

C

Pearson's Correlation Coefficient

For measuring the dependence of two random quantities, one often uses correlation. In many cases, one is not only interested in the magnitude of the correlation, but also whether the correlation is statistically significant. In this appendix, we give a derivation of the confidence intervals of Pearson's correlation coefficient ρ.

The definition of Pearson's correlation coefficient ρ between two samples x_i and y_i, $i = 1, \ldots, n$ is

$$\rho = \sum_i \frac{x_i - \bar{x}}{\sqrt{\sum_k x_k - \bar{x}}} \cdot \frac{y_i - \bar{y}}{\sqrt{\sum_k y_k - \bar{y}}},$$

where the bar over the symbols x, y denotes the sample mean

$$\bar{x} = \frac{1}{n} \sum_i x_i.$$

Assuming that the samples x_i and y_i are n realizations of two normal random variables X and Y it is possible to derive p-values for the null hypothesis that they are uncorrelated. To save some notation, we collect the sample points in vectors $x, y \in \mathbb{R}^n$ and introduce the residual vectors

$$\begin{aligned} r_X &= x - \bar{x} \\ r_Y &= y - \bar{y}. \end{aligned}$$

By the same reasoning as in Appendix A on linear regression, the residual vectors can be thought of as an orthogonal projection into an $n-1$ dimensional subspace and the residual sum of squares is thus χ^2-distributed with $n-1$ degrees of freedom.

Using the residuals, we can write the squared correlation coefficient more compactly as

$$\rho^2 = \frac{(r_X^\top r_Y)^2}{(r_X^\top r_X)(r_Y^\top r_Y)}.$$

To arrive at an expression for the p-values, we have to study the projection of one residual vector, r_X say, on the second residual vector. To this end, we can rewrite the expression for ρ^2 as

$$\rho^2 = \frac{\left(r_X^\top \frac{r_Y}{\sqrt{r_Y^\top r_Y}} \right)^2}{(r_X^\top r_X)}.$$

The numerator is the square of the projection of r_X on r_Y. Since we assume, under the null hypothesis, that X and Y are independent we can invoke the same arguments as above for the sample mean. That is, the squared projection is χ^2-distributed with one degree of freedom. To deal with the denominator, we have to look at the residual vector of the projection, i. e.

$$r_X - \left(r_X^\top \frac{r_Y}{\sqrt{r_Y^\top r_Y}} \right) \frac{r_Y}{\sqrt{r_Y^\top r_Y}}.$$

The sum of squares of this vector is

$$\begin{aligned}
&\left[r_X - \left(r_X^\top \frac{r_Y}{\sqrt{r_Y^\top r_Y}} \right) \frac{r_Y}{\sqrt{r_Y^\top r_Y}} \right]^\top \left[r_X - \left(r_X^\top \frac{r_Y}{\sqrt{r_Y^\top r_Y}} \right) \frac{r_Y}{\sqrt{r_Y^\top r_Y}} \right] \\
&= r_X^\top r_X - 2\frac{\left(r_X^\top r_Y\right)^2}{r_Y^\top r_Y} + \frac{\left(r_X^\top r_Y\right)^2}{r_Y^\top r_Y} \\
&= r_X^\top r_X \left(1 - \frac{\left(r_X^\top r_Y\right)^2}{(r_X^\top r_X)(r_Y^\top r_Y)} \right) \\
&= r_X^\top r_X (1 - \rho^2).
\end{aligned}$$

Since this is a projection into an $n-2$ dimensional subspace, this sum is χ^2-distributed with $n-2$ degrees of freedom. Thus, the expression

$$(n-2)\frac{\rho^2}{1-\rho^2} \quad \sim F(1, n-2)$$

has an F-distribution with parameters 1 and $n-2$. Taking the square root yields

$$\rho\sqrt{\frac{n-2}{1-\rho^2}} \quad \sim t(n-2),$$

a Student's t-distribution with $n-2$ degrees of freedom.

Note that throughout the derivation, we have only used the normality assumption for X and independence of X and Y. Thus, the requirement that both random variables must be normally distributed can be relaxed.

D

Maximum Weight Matching

This appendix details the implementation of the algorithm to compute maximum weight matchings in bipartite graphs (also known as assignment problems). It follows the description in [29]. For simplicity of exposition, assume that the bipartite graph is balanced (i.e. has the same number of nodes in both its node sets), complete and has positive integer weights. All of these requirements go without loss of generality, or – in other words – all problems can be reduced to this case (see comments at the end of this appendix chapter).

The actual algorithm that we describe is based on the dual formulation of the problem. That is, instead of iteratively increasing the weight of a matching it decreases a so-called *labeling* of the nodes. Consider a complete bipartite graph with node sets X and Y and weights $w : X \times Y \to \mathbb{N}$. A *labeling* is a function

$$\ell : X \cup Y \to \mathbb{R}.$$

A labeling is *feasible* iff

$$\ell(x) + \ell(y) \geq w(x,y) \qquad \forall\, x \in X,\ y \in Y.$$

The difference $\ell(x) + \ell(y) - w(x,y)$ is called the *slack* of the edge (x,y).

To state the algorithm we need one more definition. The *equality graph* of a labeling is the graph $G_\ell = (X, Y, E_\ell)$ with edges

$$E_\ell = \{(x,y) \mid \ell(x) + \ell(y) = w(x,y)\}.$$

Given all these definitions, the following property establishes the connection between a labeling and a matching of a weighted bipartite graph.

Theorem D.1 (Kuhn-Munkres). *If a labeling ℓ is feasible and a matching M on E_ℓ is perfect, then M is also a maximum weight matching of the original graph.*

This theorem suggests the following combinatorial procedure for finding a maximum weight matching.

```
Find feasible labeling $\ell$.
Set matching $M = \emptyset$.
while $M$ is not perfect do
    1. Find augmenting path in $E_\ell$ and increase matching $M$.
    2. If $M$ is maximal in $E_\ell$ improve $\ell \to \ell'$ such that $E_\ell \subset E_{\ell'}$.
end while
```

The crucial point of this overall procedure is to see that whenever M is maximal but not yet perfect, there always exists an improvement in the labeling $\ell \to \ell'$ with the property $E_\ell \subset E_{\ell'}$, i.e. $E_{\ell'}$ is strictly larger than E_ℓ. To establish this fact, we need the *neighbor set* of a node x and a subset $S \subseteq X$:

$$\begin{aligned} N_\ell(x) &= \{y \mid (x,y) \in E_\ell\} \\ N_\ell(S) &= \bigcup_{x \in S} N_\ell(x). \end{aligned}$$

Then, for any $S \subseteq X$ such that $T = N_\ell(S) \neq Y$, we can find

$$\alpha_\ell = \min_{x \in S, y \notin T} \ell(x) + \ell(y) - w(x,y)$$

and set

$$\ell'(v) = \begin{cases} \ell(v) - \alpha_\ell & \text{if } v \in S \\ \ell(v) + \alpha_\ell & \text{if } v \in T \\ \ell(v) & \text{otherwise.} \end{cases}$$

It is easy to see that $\ell'(\cdot)$ is also a feasible labeling. Furthermore, edges in E_ℓ are also in $E_{\ell'}$ and at least one additional edge is in the new equality graph (namely the one that attains the minimum α_ℓ).

The key to an efficient implementation is to find augmenting paths in E_ℓ quickly. To achieve this, the algorithm maintains a tree of alternating paths in E_ℓ rooted at a free node $x \in X$. The tree is represented using two sets $S \subseteq X$ and $T \subseteq Y$. For each element $x \in S$, we store its parent element $\text{pred}(x) \in T$ (except for the root element) and, likewise, for each $y \in T$ we store $\text{pred}(y) \in S$. Our implementation uses breadth-first search to build this tree. Two situations can occur during this exploration phase. First, if we find an augmenting path, we can increase the matching M in E_ℓ, choose a free vertex as a new root node and start building a new augmenting tree (step 1 of algorithm outline above). The second possibility is that we have constructed a spanning tree in the connected component of E_ℓ containing the root node x. Then, to continue the search for an augmenting path, we have to increase the equality graph by improving the labeling $\ell \to \ell'$ using the procedure described above. This yields at least one new edge, which allows to continue the breadth-first search in the new equality graph $E_{\ell'}$ (step 2 of algorithm outline).

To initialize the algorithm, finding an initial feasible labeling is trivial. One can put all the "weight" of the labeling on the nodes $x \in X$, i.e.

$$\ell(v) = \begin{cases} \max_{y \in Y} w(v,y) & \text{if } v \in X \\ 0 & \text{if } v \in Y. \end{cases}$$

This construction assures that each node is adjacent to at least one opposite node in the initial equality graph G_ℓ since for each node $x \in X$ there is at least one edge with zero slack.

For the matching M, one can start with the empty matching $M = \varnothing$, or slightly more efficient, one can add zero-slack edges from the initial labeling if their end nodes $y \in Y$ are not yet matched.

If the bipartite graph is not complete one can add “dummy edges” with weight zero between unconnected nodes. If the algorithm terminates with such a dummy edge in the matched edges, one simply removes that edge from the final matching. If the graph is not balanced one can add “dummy nodes” which have only zero-weight edges. It is also straightforward to implement the algorithm for unbalanced graphs. The stopping criterion must then be modified to terminate the algorithm when all nodes of the smaller node set are matched.

Dealing with negative weights is possible by making them positive before starting the algorithm. To this end, one can add a large enough value w_0 to the weights such that all “shifted” weights are (strictly) positive. When the algorithm terminates, the constant w_0 is subtracted again and all matched edges with a non-positive weight are removed from the final result.

Bibliography

[1] BaSysBio integrated European project, `http://basysbio.eu`.

[2] D. Angeli, P. De Leenheer, and E. Sontag. Graph-theoretic characterizations of monotonicity of chemical networks in reaction coordinates. *Journal of Mathematical Biology*, 2009.

[3] C. Barrett, C. Herring, J. Reed, and B. Ø. Palsson. The global transcriptional regulatory network for metabolism in Escherichia coli exhibits few dominant functional states. *Proceedings of the National Academy of Sciences USA*, 102(52):19103–19108, 2005.

[4] C. L. Barrett, N. D. Price, and B. Ø. Palsson. Network-level analysis of metabolic regulation in the human red blood cell using random sampling and singular value decomposition. *BMC Bioinformatics*, 7:132, 2006.

[5] Y. Bilu, T. Shlomi, N. Barkai, and E. Ruppin. Conservation of expression and sequence of metabolic genes is reflected by activity across metabolic states. *PLoS Computational Biology*, 2(8):e106, 2006.

[6] I. Borodina and J. Nielsen. From genomes to in silico cells via metabolic networks. *Current Opinions in Biotechnology*, 16(3):350–5, 2005.

[7] R. Budney (mathoverflow.net/users/1465). Rank(A) and other algorithms as a polynomial. MathOverflow. `http://mathoverflow.net/questions/12238` (version: 2010-01-18).

[8] R. Burkard, M. Dell'Amico, and S. Martello. *Assignment Problems*. Society for Industrial and Applied Mathematics, Philadephia, PA, 2009.

[9] R. H. Byrd, J. Nocedal, and R. A. Waltz. Knitro: An integrated package for nonlinear optimization. In *Large Scale Nonlinear Optimization*, volume 83 of *Nonconvex Optimization and Its Applications*, pages 35–59. Springer Verlag, 2006.

[10] A. Charnes and W. W. Cooper. Programming with linear fractional functionals. *Naval Research Logistics Quarterly*, 9(3-4):181–186, 1962.

[11] S. D. Cohen and A. C. Hindmarsh. CVODE, a stiff/nonstiff ODE solver in C. *Computers in Physics*, 10(2):138–143, 1996.

[12] C. Conradi, D. Flockerzi, J. Raisch, and J. Stelling. Subnetwork analysis reveals dynamic features of complex (bio)chemical networks. *Proceedings of the National Academy of Sciences USA*, 104(49):19175–19180, 2007.

[13] C. Conradi, J. Saez-Rodriguez, E.-D. Gilles, and J. Raisch. Using chemical reaction network theory to discard a kinetic mechanism hypothesis. *Systems biology*, 152(4):243–248, 2005.

[14] G. Craciun, Y. Tang, and M. Feinberg. Understanding bistability in complex enzyme-driven reaction networks. *Proceedings of the National Academy of Sciences*, 103(23):8697–8702, 2006.

[15] P. Craven and G. Wahba. Smoothing noisy data with spline functions. *Numerische Mathematik*, 31(4):377–403, 1978.

[16] J. R. Donaldson and R. B. Schnabel. Computational experience with confidence regions and confidence intervals for nonlinear least squares. *Technometrics*, 29(1):67–82, 1987.

[17] M. Dürr. personal communication, 2009.

[18] J. A. Egea, R. Martí, and J. R. Banga. An evolutionary method for complex-process optimization. *Computers & Operations Research*, 37(2):315–324, 2010.

[19] J. J. Faith, B. Hayete, J. T. Thaden, I. Mogno, J. Wierzbowski, G. Cottarel, S. Kasif, J. J. Collins, and T. S. Gardner. Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biology*, 5, 2007.

[20] M. Feinberg. Chemical reaction network structure and the stability of complex isothermal reactors – I. The deficiency zero and deficiency one theorems. *Chemical Engineering Science*, 42(10):2229–2268, 1987.

[21] M. Feinberg. Chemical reaction network structure and the stability of complex isothermal reactors – II. Multiple steady states for networks of deficiency one. *Chemical Engineering Science*, 43(1):1–25, 1988.

[22] M. Feinberg. The existence and uniqueness of steady states for a class of chemical reaction networks. *Archive for Rational Mechanics and Analysis*, 132(4):311–370, 1995.

[23] M. Feinberg. Multiple steady states for chemical reaction networks of deficiency one. *Archive for Rational Mechanics and Analysis*, 132(4):371–406, 1995.

[24] M. Feinberg and F. Horn. Chemical mechanism structure and the coincidence of the stoichiometric and kinetic subspaces. *Archive for Rational Mechanics and Analysis*, 66(1), 1977.

[25] 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*, 3:121, 2007.

[26] X.-J. Feng and H. Rabitz. Optimal identification of biochemical reaction networks. *Biophysical Journal*, 86(3):1270–1281, 2004.

[27] K. Gatermann and B. Huber. A family of sparse polynomial systems arising in chemical reaction systems. *Journal of Symbolic Computation*, 33(3):275–305, 2002.

[28] K. Gatermann and M. Wolfrum. Bernstein's second theorem and Viro's method for sparse polynomial systems in chemistry. *Advances in Applied Mathematics*, 34(2):252–294, 2005.

[29] M. J. Golin. Bipartite matching and the Hungarian method. Course Notes, 2007.

[30] G. H. Golub and C. F. V. Loan. *Matrix Computations*. Johns Hopkins University Press, Baltimore, 3rd edition, 1996.

[31] Z. Gu, E. Rothberg, and R. Bixby. *Gurobi Optimizer Reference Manual Version 4.5*. Houston, Texas, 2011.

[32] N. Guelzim, S. Bottani, P. Bourgine, and F. Képès. Topological and causal structure of the yeast transcriptional regulatory network. *Nature Genetics*, 31(1):60–63, 2002.

[33] J. Gunawardena. Chemical reaction network theory for in-silico biologists. Technical report, Bauer Center for Genomics Research, 2003.

[34] P. Hansen, N. Mladenović, and J. A. Moreno Pérez. Variable neighbourhood search: methods and applications. *Annals of Operations Research*, 175(1):367–407, 2010.

[35] R. Hermann and A. J. Krener. Nonlinear controllability and observability. *IEEE Transactions on Automatic Control*, 22(5):728–740, 1977.

[36] R. U. Ibarra, J. S. Edwards, and B. Ø. Palsson. Escherichia coli K-12 undergoes adaptive evolution to achieve in silico predicted optimal growth. *Nature*, 420(6912):186–189, 2002.

[37] G. Karlebach and R. Shamir. Modelling and analysis of gene regulatory networks. *Nature Reviews Molecular Cell Biology*, 9(10):770–780, 2008.

[38] R. Kleijn. personal communication, 2008.

[39] H. W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[40] The MathWorks, Inc., Natick, MA. *Optimization Toolbox User's Guide, Version 4.3*, 2009.

[41] Y.-K. Oh, B. Ø. Palsson, S. M. Park, C. H. Schilling, and R. Mahadevan. Genome-scale reconstruction of metabolic network in Bacillus subtilis based on high-throughput phenotyping and gene essentiality data. *Journal of Biological Chemistry*, 282(39):28791–28799, 2007.

[42] 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):886–97, 2004.

[43] J. L. Reed, T. D. Vo, C. H. Schilling, and B. Ø. Palsson. An expanded genome-scale model of Escherichia coli K-12 (iJR904 GSM/GPR). *Genome Biology*, 4(9), 2003.

[44] C. H. Schilling, S. Schuster, B. Ø. Palsson, and R. Heinrich. Metabolic pathway analysis: Basic concepts and scientific applications in the post-genomic era. *Biotechnology Progress*, 15(3):296–303, 1999.

[45] P. M. Schlosser and M. Feinberg. A theory of multiple steady states in isothermal homogeneous CFSTRs with many reactions. *Chemical Engineering Science*, 49(11):1749–1767, 1994.

[46] S. Schuster, A. von Kamp, and M. Pachkov. Understanding the roadmap of metabolism by pathway analysis. In W. Weckwerth, editor, *Metabolomics: Methods and Protocols*, volume 358 of *Methods in Molecular Biology*, pages 199–226. Springer, 2007.

[47] R. Schütz, L. Küpfer, and U. Sauer. Systematic evaluation of objective functions for predicting intracellular fluxes in Escherichia coli. *Molecular Systems Biology*, 3:119, 2007.

[48] I. H. Segel. *Enzyme kinetics: behavior and analysis of rapid equilibrium and steady-state enzyme systems*. Wiley, New York, 1993.

[49] D. Segrè, D. Vitkup, and G. M. Church. Analysis of optimality in natural and perturbed metabolic networks. *Proceedings of the National Academy of Sciences USA*, 99(23):15112–7, 2002.

[50] G. Shinar and M. Feinberg. Structural sources of robustness in biochemical reaction networks. *Science*, 327(5971):1389–1391, 2010.

[51] 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 USA*, 102(21):7695–700, 2005.

[52] N. Sierro, Y. Makita, M. de Hoon, and K. Nakai. DBTBS: a database of transcriptional regulation in Bacillus subtilis containing upstream intergenic conservation information. *Nucleic Acids Research*, 36(suppl 1):D93–96, 2008.

[53] A. J. Sommese and C. W. Wampler. *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science*. World Scientific, 2005.

[54] N. Soranzo and C. Altafini. ERNEST: a toolbox for chemical reaction network theory. *Bioinformatics*, 25(21):2853–2854, 2009.

[55] J. Stelling, E. D. Gilles, and F. J. Doyle. Robustness properties of circadian clock architectures. *Proceedings of the National Academy of Sciences USA*, 101(36):13210–5, 2004.

[56] J. Stelling, S. Klamt, K. Bettenbrock, S. Schuster, and E. D. Gilles. Metabolic network structure determines key aspects of functionality and regulation. *Nature*, 420(6912):190–3, 2002.

[57] R. Steuer, T. Gross, J. Selbig, and B. Blasius. Structural kinetic modeling of metabolic networks. *Proceedings of the National Academy of Sciences USA*, 103(32):11868–73, 2006.

[58] M. Terzer. *Large Scale Methods to Enumerate Extreme Rays and Elementary Modes*. PhD thesis, ETH Zurich, 2009.

[59] M. Uhr, H.-M. Kaltenbach, C. Conradi, and J. Stelling. Analysis of degenerate chemical reaction networks. In R. Bru and S. Romero-Vivó, editors, *Positive Systems*, volume 389 of *Lecture Notes in Control and Information Sciences*, pages 163–171. Springer Berlin/Heidelberg, 2009.

[60] M. Uhr and J. Stelling. Structural sensitivity analysis of metabolic networks. In *Proceedings of the 17th IFAC World Congress*, 2008.

[61] M. Uhr, A. F. Villaverde, J. A. Egea, J. R. Banga, and J. Stelling. Inference of transcriptional control design of metabolic networks. In *Proceedings of the 18th IFAC World Congress*, 2011.

[62] A. Varma and B. Ø. Palsson. Metabolic flux balancing: Basic concepts, scientific and practical use. *Nature Biotechnology*, 12(10):994–998, 1994.

[63] L. Wasserman. *All of nonparametric statistics*. Springer Texts in Statistics. Springer-Verlag, New York, 2006.

[64] G. M. Ziegler. *Lectures on Polytopes*, volume 152 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1995.

Curriculum Vitae

Personal Details

Name	Markus Uhr
Address	Geissackerstr. 22, 8492 Wila
Email	markus.uhr@bsse.ethz.ch
Date of Birth	20. April 1981
Citizenship	Menzingen (ZG), Switzerland

Education

2007 - 2012	PhD Studies in Computer Science and Systems Biology at ETH Zurich
2003 - 2006	Graduate Studies in Computational Science and Engineering at ETH Zurich
2001 - 2003	Undergraduate Studies in Electrical Engineering at ETH Zurich

Work/Practical Experience

2003 - 2005	Layouting for *blitz* (student magazine for electrical engineering students)
2002	Programmed a data exchange/synchronization tool for SIWID AG
2001 - 2005	Sales Assistant at Coop (20% Job, parallel to studies)

Languages

German	native tongue
English	good spoken and written
French	spoken and written knowledge

Computer Skills

Programming	C/C++, Java, C#, F#, Python, Matlab
OS	Linux, Mac OS X
Applications	LaTeX, Adobe Illustrator, Adobe InDesign

Publications

- M. Uhr, Optimal Operation of a Hydroelectric Power System Subject to Stochastic Inflows and Load. *Diploma Thesis*. 2006.
- I. Guyon, J. Li, T. Mader, P. A. Pletscher, G. Schneider, and M. Uhr, Competitive baseline methods set new standards for the NIPS 2003 feature selection benchmark. *Pattern Recognition Letters*, 28(12):1438–1444, 2007.
- K. Nolde, M. Uhr, and M. Morari. Medium term scheduling of a hydro-thermal system using stochastic model predictive control. *Automatica*, 44(6):1585–1594, 2008.
- K. Nolde, M. Uhr, and M. Morari. Medium term production scheduling of a hydro-thermal system under uncertainty using nested benders decomposition. *Automatisierungstechnik*, 56 (4):189–196, 2008.
- M. Uhr and J. Stelling. Structural sensitivity analysis of metabolic networks. In *Proceedings of the 17th IFAC World Congress*, 2008.
- M. Uhr, H.-M. Kaltenbach, C. Conradi, and J. Stelling. Analysis of degenerate chemical reaction networks. In R. Bru and S. Romero-Vivó, editors, *Positive Systems*, volume 389 of *Lecture Notes in Control and Information Sciences*, pages 163–171. Springer Berlin/Heidelberg, 2009.
- M. Uhr, A. F. Villaverde, J. A. Egea, J. R. Banga, and J. Stelling. Inference of transcriptional control design of metabolic networks. In *Proceedings of the 18th IFAC World Congress*, 2011.
- J. M. Büscher, W. Liebermeister, M. Jules, M. Uhr, J. Muntel et al. Global network reorganization during dynamic adaptations of Bacillus subtilis metabolism. *Science*, submitted.