
Reduction, Projection, and Simplification of Metabolic Networks

Annika RÖHL

Dissertation zur Erlangung des Grades eines
Doktors der Naturwissenschaften (Dr. rer. nat.)

vorgelegt am Fachbereich Mathematik und Informatik
der Freien Universität Berlin

Berlin, Juni 2018

Betreuer: Prof. Dr. Alexander BOCKMAYR
Fachbereich Mathematik und Informatik
Freie Universität Berlin

Zweitgutachter: Dr. Jürgen ZANGHELLINI
Institut für Biotechnologie
Universität für Bodenkultur Wien

Tag der Disputation: 7. August 2018

Für Jenny

Abstract

Systems biology is an interdisciplinary field of research that combines mathematics, computer science, and engineering in order to analyse biological processes. It has become more and more important in the last two decades, in particular because of successful applications for human health and biotechnology. It aims at simulating biological systems as mathematical models to support time- and cost-intensive research in laboratories. To do so, researchers create formalisms, algorithms, and techniques which can be widely used. One technique to obtain data describing biological entities is genome sequencing. Using modern high-throughput sequencing, increasing knowledge is gained about genomes which can then be used in order to reconstruct metabolic processes and networks of the organisms. Success does not come for free and data gathered with modern techniques is often too large to be analysed by hand. Therefore, methods which extract relevant information from data are in great demand.

In this thesis, we introduce different methods which reduce given data in metabolic networks in a meaningful way. We present a technique which computes minimal metabolic subnetworks which are still able to satisfy predefined functionalities. We also develop a method to compute minimum sets of elementary flux modes which compose the network, where the size is significantly reduced compared to the whole set of elementary flux modes. Furthermore, we provide procedures that reduce the number of variables in a given problem in order to accelerate (already existing) algorithms by using information given by the data. Moreover, we develop a novel procedure to compute minimal cut sets on a projected network. This enables us to compute minimal cut sets of larger cardinality than before and to analyse larger networks. This projection of metabolic networks also gives rise to other applications such as computing minimal metabolic behaviours.

Even though we apply and suit our methods to real metabolic systems, this thesis is focused on the mathematical methods. In order to create and prove the new techniques we make use of (mixed integer) linear optimisation, polyhedral cones, linear algebra, and oriented matroids.

Contents

| | | |
|----------|--|-----------|
| 1 | Introduction | 1 |
| 1.1 | Metabolic networks | 4 |
| 1.2 | Mathematics for metabolic networks | 7 |
| 1.2.1 | Linear programming | 7 |
| 1.2.2 | Polyhedral cones | 8 |
| 1.2.3 | Oriented matroids | 9 |
| 1.3 | Structure of this thesis | 10 |
| 2 | Basics | 11 |
| 2.1 | Notation | 11 |
| 2.2 | Metabolic networks | 12 |
| 2.2.1 | Elementary flux modes | 13 |
| 2.2.2 | Minimal metabolic behaviours | 14 |
| 2.2.3 | Minimal cut sets | 15 |
| 2.2.4 | Bounds | 18 |
| 2.3 | Mathematical basics | 18 |
| 2.3.1 | Polyhedra | 18 |
| 2.3.2 | Linear Programming | 25 |

Contents

| | | |
|----------|--|-----------|
| 2.3.3 | Mixed integer linear programming | 31 |
| 2.3.4 | Numerical instability | 33 |
| 2.3.5 | Oriented Matroids | 36 |
| 2.4 | Handling metabolic networks | 44 |
| 3 | Computing EFMs exploiting FCA | 49 |
| 3.1 | Shortest elementary flux modes | 50 |
| 3.1.1 | Mixed integer linear program to compute the shortest elementary flux modes | 50 |
| 3.1.2 | Computational results | 52 |
| 3.2 | Further work | 57 |
| 4 | Reduction of Networks | 59 |
| 4.1 | Introduction | 59 |
| 4.2 | Background | 59 |
| 4.3 | Methods | 63 |
| 4.3.1 | Basic MILP to compute a minimum subnetwork | 63 |
| 4.3.2 | Conflicting functionalities | 64 |
| 4.3.3 | Computing all minimum subnetworks | 66 |
| 4.3.4 | Reducing the number of binary variables | 67 |
| 4.4 | Results and discussion | 71 |
| 4.4.1 | Comparison with NetworkReducer | 71 |
| 4.4.2 | Comparison with FASTCORE | 72 |
| 4.4.3 | Network reduction for genome-scale metabolic networks | 73 |
| 4.4.4 | Case study: <i>Helicobacter pylori</i> 26695 | 77 |
| 4.5 | Conclusion | 79 |

| | | |
|--------------|---|----------------|
| 5 | Cones, Matroids, Networks | 81 |
| 5.1 | Oriented fundamental circuits and reaction splitting | 82 |
| 5.2 | Computing a basis and oriented fundamental circuits | 88 |
| 5.3 | Different bases | 89 |
| 5.4 | Conclusion | 91 |
| 6 | Finding MEMo | 93 |
| 6.1 | Introduction | 94 |
| 6.1.1 | Intuition | 94 |
| 6.1.2 | Contributions | 96 |
| 6.2 | ERs, EFMs, and MMBs | 98 |
| 6.3 | Extreme rays of pointed augmented flux cones are EFMs | 99 |
| 6.4 | Splitting a minimum number of reversible reactions | 100 |
| 6.5 | MEMo: Minimum set of Elementary Modes | 103 |
| 6.6 | Computational results and discussion | 108 |
| 6.7 | Related work | 113 |
| 6.7.1 | Extreme pathways | 113 |
| 6.7.2 | Minimal metabolic behaviours | 114 |
| 6.7.3 | Minimal generating set | 114 |
| 6.7.4 | Subsets of EFMs | 115 |
| 6.8 | Conclusion and further work | 115 |
| 7 | Projections of flux cones | 117 |
| 7.1 | Introduction | 118 |
| 7.2 | Methods | 119 |
| 7.2.1 | Computing minimal metabolic behaviours via projection | 119 |
| 7.2.2 | Projection and contraction | 124 |
| 7.3 | Results and discussion | 132 |
| 7.4 | Conclusion | 135 |

Contents

| | | |
|-----------|--|------------|
| 8 | Computing iMCSs using the dual approach | 137 |
| 8.1 | The dual approach | 138 |
| 8.1.1 | Dual network | 138 |
| 8.2 | Results | 141 |
| 8.2.1 | Projection | 141 |
| 8.2.2 | Computing shortest MCSs | 144 |
| 8.2.3 | Computing MCSs including a knock-out reaction . . . | 148 |
| 8.3 | Conclusion | 151 |
| 9 | Conclusion | 153 |
| | Bibliography | 157 |
| | Notation | 173 |
| | Index | 179 |
| 10 | Appendix | 183 |
| 10.1 | Appendix for RedNet | 183 |
| 10.1.1 | Comparison with NetworkReducer | 183 |
| 10.1.2 | Comparison with FASTCORE | 184 |
| 10.1.3 | New experiments | 185 |
| 10.2 | Appendix for Computing iMCSs using the dual approach . . | 194 |
| 10.2.1 | Time: Computations using Tobalina's tool | 194 |
| 10.2.2 | Cardinality: Computations using Tobalina's tool . . . | 201 |
| | Curriculum vitae | 209 |
| | Declaration | 211 |
| | Acknowledgement | 213 |

Chapter 1

Introduction

Systems biology is an interdisciplinary field of research that uses theoretical approaches from mathematics, computer science, and engineering to analyse complex biological processes such as gene expression, cellular organisation, cellular signalling, or metabolism. Around the year 2000, the research focused on studying the functionalities of single genes and proteins, whereas lately the structure and dynamics of networks got more and more attention [Kitano, 2002]. One reason for this is that the data is available, especially since new techniques are involved such as whole genome-sequencing and high-throughput measurements [Margulies et al., 2005; Bentley et al., 2008]. Another reason is that applications showed promising results in medicine and bio-engineering [Feist and Palsson, 2008; Oberhardt et al., 2009; Shlomi et al., 2011; Bazzani et al., 2012; Erdrich et al., 2014].

A common approach to systematically represent and analyse cellular processes is using networks. Here, the nodes of the network represent biological entities such as enzymes or transcription factors, and edges indicate chemical or molecular interactions between them. There are three major types of biological networks: signalling networks (interactions of receptors of a cell and intracellular actions), gene regulatory networks (interactions of messenger RNA and proteins), and metabolic networks. In this thesis, we consider metabolic networks only, which involve chemical reactions and metabolic pathways of an organism.

The metabolism of organisms tends to be very complex, and it is therefore a major challenge to find a trade-off between abstraction and precision (e.g. having very simple or more complex interactions). Simple interactions lead to networks that can be analysed by efficient algorithms and therefore large networks can be studied. Using more complex interactions often hampers the analysis and only subnetworks can be examined.

1. Introduction

The reconstruction of genome-scale metabolic networks improved remarkably through access to more detailed information by genome sequencing, [Francke et al., 2005; Notebaart et al., 2006; Duarte et al., 2007; Durot et al., 2009; Henry et al., 2010; Thiele and Palsson, 2010; Rupp et al., 2010; Magnúsdóttir et al., 2016] which makes it possible to reconstruct a genome-scale metabolic network including not only some chemical reactions, but all of them. Thus, both the number of available metabolic networks and the size of the networks increased over the past years. This gives the opportunity to study more details of a cell or larger biological systems, as well as to investigate the interactions between different organisms.

To handle this amount of data, mathematical and computational tools were developed [Varma and Palsson, 1994; Segre et al., 2002; Price et al., 2004; Terzer et al., 2009] and applied to study, e.g., the impact of a gene knock-out. Using algorithms enables researchers to perform experiments *in silico* for determining which approaches have the greatest potential for accomplishing a task. This can reduce time and costs of experiments in laboratories. However, depending on the amount of information and complexity of the analysis tool, it is often not feasible to include all given information. Therefore, one has to decide for a level of complexity, i.e., the number of components and the kind of interactions.

There are two fundamentally different approaches to analyse networks: Those that analyse the time-dependent behaviour, the dynamics, and those that analyse the structure, the topology. To do so, different mathematical modelling approaches exist [Walhout et al., 2012]. In the first category, continuous [Tyson et al., 2001; van Heerden et al., 2014; Rügen et al., 2015; Waldherr et al., 2015] or stochastic modelling [Gillespie et al., 2013] methods are used to study the dynamics of networks. However, these approaches are computationally expensive and often do not scale to genome-scale metabolic networks, so one can consider most of the time only small networks or subnetworks. To analyse the topology of a network one can use graph-based methods [Aittokallio and Schwikowski, 2006; Jonnalagadda and Srinivasan, 2014] or constraint-based modelling [Terzer et al., 2009; Bordbar et al., 2014; Klamt et al., 2018].

In this work, we study the topology of large metabolic networks, thus the networks are considered independent from time, i.e., in steady-state. Considering only the topology of the system decreases the complexity by excluding dynamics and therefore allows to analyse much larger networks. But analysing only the topology restricts the accuracy of possible predictions. However, it can be used to detect knock-out targets to suppress unwanted by-product pathways in order to improve the product yield. For example `OptKnock` [Burgard et al., 2003] is a framework for identifying those

knock-out strategies and was successfully applied for the development of *E. coli* strains that produce lactate [Fong et al., 2005]. The SIMUP algorithm [Gawand et al., 2013] was employed to modify *E. coli* to produce biofuel precursors while utilising different sugars in order to reduce the competition with food crops.

Analysing the structure of these networks is challenging by itself due to the size of the networks and focus of many studies [Schuster and Hilgetag, 1994; Varma and Palsson, 1994; Burgard et al., 2004; Klamt and Gilles, 2004; Papin et al., 2004; Larhlimi and Bockmayr, 2009]. We address the issue of complexity by introducing novel approaches for network reduction and analysis.

To reduce metabolic networks we first need to define the desired outcome. Those questions are usually brought up by biological or medical applications. The next step is to find a way on *how* to reduce the network: the *important* information has to be kept while the non-interesting data should be excluded. To do so, mathematical methods have to be developed such that they are applicable to large networks and the result fits the expectations. The contributions of this thesis regarding this topic are the following:

We introduce a method for reducing the search space of an already known method [De Figueiredo et al., 2009] which can be applied to similar methods as well. Thus the running time of the algorithms is decreased and larger networks can be analysed.

Furthermore, we develop a procedure which computes subnetworks maintaining predefined functionalities. The subnetworks can then be used for further analysis. The idea is not new and there exist methods to tackle this problem, but these methods have various disadvantages. Either they compute subnetworks of arbitrary size or only one functionality can be defined. Here we develop a new method where the subnetworks are of minimum size and several functionalities can be defined.

To compute these subnetworks we remove unwanted reactions. This causes a change in the connectivity of the subnetwork compared to the original one. But if we want to study the fragility or robustness of a network we have to keep the connectivity. A solution to this is to not remove reactions but to merge them together by performing a projection. The method we introduce here is fast, applicable to large networks, and the description of the reduced network is smaller than the description of the original network.

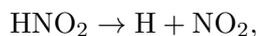
Last but not least we present a technique to reduce the number of generators of a network. The introduced algorithm computes a minimal representation for a network.

1. Introduction

For developing these methods we combine knowledge from different mathematical disciplines, such as linear programming, linear algebra, and oriented matroids. These methods are based on different underlying theories but they share the concept of reducing, in a meaningful way, the information needed to do further research on the underlying metabolic network.

1.1 Metabolic networks

A metabolic network represents a set of chemical reactions within an organism, a cell, or part of a cell. A chemical reaction is a process where substrates are converted to products, e.g.



where HNO_2 is a substrate converted by a reaction into the products H and NO_2 . A substance can function as a product in one metabolic reaction and as a substrate in a different reaction. Substrates and products are also called *metabolites*.

We use the *stoichiometric matrix* $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$ to represent a metabolic network. The columns of S correspond to the set of reactions \mathfrak{R} and the rows to the set of metabolites \mathfrak{M} . The entries of S are called stoichiometric coefficients. They are positive, $S_{j,i} > 0$, if metabolite j is produced by reaction i , and negative, $S_{j,i} < 0$, if metabolite j is consumed by reaction i . Otherwise $S_{j,i} = 0$.

The reactions are divided into two categories: *reversible* and *irreversible*. For example the reaction above can only produce H and NO_2 from HNO_2 . Thus it can only operate in one *direction* and is therefore irreversible. If the reaction were reversible, it would be able to produce HNO_2 from H and NO_2 as well. Furthermore, we distinguish between *exchange* and *internal* reactions. Exchange reactions are transporters of the system. They cross the boundary of the network (e.g. the membrane of a cell) and therefore can transport metabolites in or out of the system. A well-known exchange reaction is the *biomass reaction*. This reaction is an artificial reaction added to the network in order to simulate growth. Biologists construct the biomass reaction such that all necessary components, which the cell needs for growing, are consumed by it. The biomass reaction consumes all components of biomass (e.g. amino-acids, lipids etc.) in the ratios the cell itself is made of. The stoichiometric entries of the biomass reaction are chosen such that they reflect the fractions of those metabolites in one cell, which are typically measured in the laboratory. One can also distinguish between external

1.1 Metabolic networks

and internal metabolites, i.e., metabolites outside of the boundary or inside. Since stoichiometric matrices involve only internal metabolites (if the system is in steady-state) we do not consider external metabolites here.

For a graphical representation of metabolic networks usually directed hypergraphs are used. Here, reactions correspond to edges (or arcs), which are directed to indicate the direction of the reaction and nodes correspond to metabolites. Since a reaction can consume or produce more than one metabolite, some edges are hyperedges connecting more than two nodes. Figure 1.1 gives an illustration of a small network.

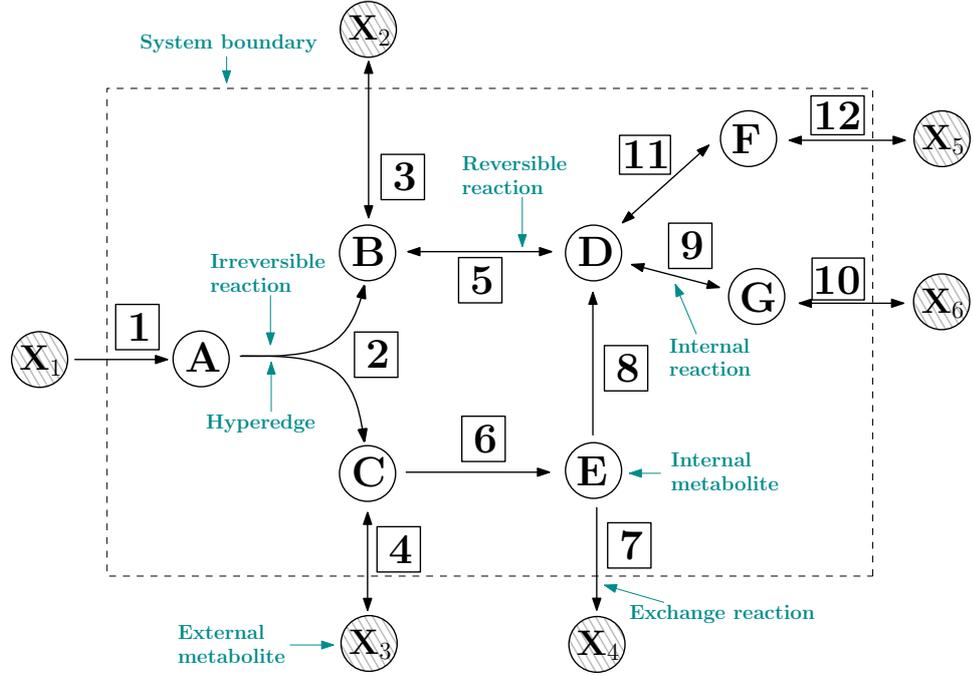
For a reaction $i \in \mathfrak{R}$ to become *active*, i.e., to consume or produce metabolites, an enzyme needs to catalyse reaction i . Whether or not an enzyme is produced depends on the corresponding gene(s) which encode the enzyme. If the gene is not expressed, the enzyme is not produced and therefore cannot catalyse the reaction. The description of a metabolic network can include this information by, e.g., removing the corresponding reaction [Price et al., 2004]. Since we are only interested in the topology of the network, the networks we consider here do not include (time dependent) regulatory information [Covert et al., 2001b], such as switching a gene on and off during different phases. Such a model would require information on metabolite concentrations and even though it is possible to measure such concentrations $C \in \mathbb{R}_{\geq 0}^m$ experimentally, it was until recently very hard to do so [Nielsen and Oliver, 2005; Villas-Bôas and Bruheim, 2007]. Using *metabolomics* [Goldansaz et al., 2017] it became possible to measure metabolite concentrations, so that the concentrations of metabolites are known at a time point t , i.e., $C(t)$ and it is possible to compute the reaction rates $v \in \mathbb{R}^{\mathfrak{R}}$: $v(t) = v(C(t), k)$ which depend on kinetic parameters k , enzyme concentrations and kinetic rate laws, such as mass-action kinetics [Waage and Gulberg, 1864], Michaelis-Menten [Menten and Michaelis, 1913], etc. If this information is available, a kinetic model can be built and we obtain a system of *ordinary differential equations* (ODEs):

$$\frac{dC}{dt} = S \cdot v(C, k). \quad (1.1)$$

Kinetic parameters are hard to obtain [Bailey, 2001; Covert et al., 2001a; Edwards et al., 2002] and the same holds true for kinetic rate laws and enzyme concentrations [Nielsen and Oliver, 2005; Villas-Bôas and Bruheim, 2007]. If the environment does not change it is often assumed that the ODE-system (1.1) will reach a stationary point. This implies that the metabolite concentrations $C \in \mathbb{R}_{\geq 0}^m$ and the reaction rates $v \in \mathbb{R}^{\mathfrak{R}}$ are constant over some time interval. The consequence of considering the system at *steady-state* is enormous since no kinetic parameters are needed and the ODE-system (1.1) becomes a system of homogeneous linear equalities:

$$Sv = 0, \quad (1.2)$$

1. Introduction



$$S = \begin{matrix} & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} & \boxed{9} & \boxed{10} & \boxed{11} & \boxed{12} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \\ G \end{matrix} & \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \end{pmatrix} \end{matrix}$$

The corresponding stoichiometric matrix.

Figure 1.1: Example metabolic network with internal metabolites $\mathfrak{M} = \{A, B, \dots, G\}$, reactions $\mathfrak{R} = \{1, 2, \dots, 11, 12\}$, irreversible reactions $\text{Irr} = \{1, 2, 6, 7, 8\}$, reversible reactions $\text{Rev} = \{3, 4, 5, 9, 10, 11, 12\}$, exchange reactions $\mathfrak{R}_{\text{ex}} = \{1, 3, 4, 7, 10, 12\}$, and external metabolites $\mathfrak{M}_{\text{ex}} = \{X_1, \dots, X_6\}$.

1.2 Mathematics for metabolic networks

which we can solve using linear algebra. To restrict the irreversible reactions to operate in one direction we add inequalities $v_{\text{Irr}} \geq 0$, where v_{Irr} denotes the reaction rates of the irreversible reactions. All vectors v which are in steady-state and fulfil these inequalities will be called feasible *flux vectors*. They are also called flux distributions or pathways.

For the whole thesis, we assume that the network is in steady-state. This implies that we can employ mathematical tools related to systems of linear equations and inequalities.

1.2 Mathematics for metabolic networks

1.2.1 Linear programming

There are several ways to analyse metabolic networks and one of them is using *linear programs* (LPs). LPs are optimisation problems that have a linear objective function and contain linear (in-)equalities as constraints involving real variables. Furthermore, they are solvable in polynomial time, see Chapter 13 in [Schrijver, 1998].

The best known approach for studying metabolic networks using LPs is flux balance analysis (FBA) [Varma and Palsson, 1994]. FBA assumes that cells evolved to grow as fast as possible, i.e., they maximise their growth rate in a given environment. Typically FBA tries to mimic this by maximising the biomass production rate. Without further constraints on uptake rates, such as limitations on nutrients, this would be infinite. Thus, usually the nutrient uptake rates are bounded. However, this results in FBA predicting optimal *growth yield* instead of optimal growth rate, meaning it predicts how efficiently an organism can grow given limited nutrients, but not how fast it can grow. Using additional constraints, e.g. prohibiting some reactions to carry flux, gives insight into the effect of different growth media or gene knock-outs. One can, for example, simulate aerobic and anaerobic growth by imposing or omitting a bound on oxygen uptake.

Solutions of LPs, especially for FBA, do not have to be unique. Computing all solutions is in general not feasible, since there can be infinitely many. One approach is to explore the *optimal solution space*, i.e., the space where all optimal solutions live in, using *flux variability analysis* (FVA) [Mahadevan and Schilling, 2003]. FVA computes for each reaction the maximum and the minimum possible reaction rate assuming maximum yield. For FVA we do not solve one LP only, but two LPs per reaction of the network. If for a reaction the computed flux range (in the whole flux space) does not

1. Introduction

contain 0 the reaction is *essential* for the organism [Klamt and Gilles, 2004]. This implies that the organism cannot grow if the reaction is removed. To compute essential reactions different LP approaches can be used besides FVA [Pharkya and Maranas, 2006; Suthers et al., 2009b]. Another widely used approach where LPs are used is *flux coupling analysis* (FCA) [Burgard et al., 2004; Larhlimi et al., 2012b]. FCA examines how a knock-out of a reaction affects the activity of other reactions to compute qualitative dependencies between reactions. If some or all of the variables of an LP are integers we talk about *mixed integer linear programs* (MILPs). A typical example is to have a boolean variable $a_i \in \{0, 1\}$ for each reaction $i \in \mathfrak{R}$ indicating if reaction i is active ($a_i = 1$) or not ($a_i = 0$). To formulate an MILP is the same routine as for LPs, but solving MILPs is NP-hard while LPs are solvable in polynomial time [Schrijver, 1998].

The first contribution of this thesis is an MILP-based approach to use FCA in order to reduce the size of the solution space of an MILP for metabolic network analysis. This in turn reduces the solving time, making methods applicable to larger networks than before. Furthermore, we develop an MILP which reduces the size of a metabolic network by decreasing the number of reactions while preserving predefined functionalities. We introduce an algorithm which is applicable for user defined functionalities. The reduction enables us to study certain properties of genome-scale metabolic networks in more detail without considering the whole network.

1.2.2 Polyhedral cones

An alternative approach to represent and analyse metabolic networks is using polyhedral cones. If there are no constraints on the reaction rates, besides $Sv = 0$ and $v_{\text{irr}} \geq 0$ (i.e., no upper or lower bounds on the reaction rates), the solution space is unconstrained and called *flux cone*. A flux cone is mathematically a *polyhedral cone* [Terzer, 2009a]. Using techniques from the field of linear algebra and polyhedra, *elementary flux modes* (EFMs) can be computed [Schuster and Hilgetag, 1994; Stelling et al., 2002; Zanghellini et al., 2013]. EFMs are flux vectors that consist of a minimal set of *active* reactions, i.e., reactions with a non-zero flux rate. Minimality means that no proper subset of the active reactions can form a feasible flux vector. EFMs are of great interest, since they are a complete description of the metabolic network and therefore they can be used to represent every possible flux vector.

In this thesis, we develop a method which finds a *minimum set of EFMs* (MEMo) to describe the whole metabolic network and we provide new results and connections between polyhedral cones and metabolic networks. We are

1.2 Mathematics for metabolic networks

able to significantly reduce the number of EFMs needed and therefore we can compute these sets even for genome-scale metabolic networks. Such a MEMo can be used for other applications, especially for applications where the whole set of EFMs is needed but just too large to compute.

Closely related to the EFMs are the *minimal cut sets* (MCSs) which are minimal sets of reactions as well [Klamt and Gilles, 2004; Klamt, 2006; Pharkya and Maranas, 2006]. However, MCSs do not represent behaviours of the network but can be used to prevent them. Whenever a set of reactions of an MCS is removed from a metabolic network, certain behaviours are not possible anymore. MCSs were employed to develop bacterial strains for the biotechnological production of a variety of chemicals, from bio-fuel precursors to pharmaceuticals or to better understand a given organism [Melzer et al., 2009; Trinh et al., 2011].

1.2.3 Oriented matroids

The general strength of mathematics in analysing biological systems lies in the abstraction. If a network or a question regarding the network can be formulated mathematically, we can use all tools which fit to the underlying theoretical concept. But also within mathematics there exist structures to abstract certain properties. One of these structures are oriented matroids. Since they are an abstraction they can be used to represent polyhedral cones or metabolic networks [Reimers, 2014], but also linear spaces, oriented hypergraphs, etc. Here, we explore the connections between polyhedral cones, metabolic networks, and oriented matroids to gain new insights on metabolic networks.

Using oriented matroids we gain new results about polyhedral cones of metabolic networks and develop the method to compute a MEMo, as mentioned above. In addition, oriented matroids give rise to a procedure to compute a new type of knock-out sets. With the newly created underlying theory these knock-out sets can be computed efficiently. Thus, searching for knock-out sets can be done in genome-scale metabolic networks, which was not possible before or only with a huge effort.

1.3 Structure of this thesis

First, we present in Chapter 2 notation and background information needed for this work. This includes linear programming and mixed integer linear programming as well as polyhedra and polyhedral cones, and finally oriented matroids. Therefore, Chapter 2 forms the mathematical basis for this thesis.

In Chapter 3 a method to compute EFMs will be discussed. We reduce the solution space and decrease the running time of a known MILP by using coupling information of the reactions.

In Chapter 4 we use mixed integer linear programming to find minimum subnetworks which fulfil given properties. A method is developed that allows to define different functionalities which should be maintained by the subnetwork and where the resulting subnetwork consists of a minimum number of reactions. We use FCA to decrease the running time of the method.

In Chapter 5 we focus on oriented matroids. We first clarify the connection between an oriented matroid, a polyhedral cone, and a metabolic network, where all reactions are reversible. We show new links between the three objects. With these links we develop a procedure for finding a minimum generating set of EFMs for a given metabolic network, where all reactions are reversible.

In Chapter 6, we introduce an algorithm which, for a given metabolic network, finds a minimum set of EFMs fully representing the underlying network. To do so, we use the knowledge from Chapter 5 and develop new theory about polyhedral cones and the structure of their related metabolic networks.

Inspired by the Chapters 5 and 6 we introduce a new concept of cut sets and a new method to compute them in Chapter 7. Based on projections of polyhedral cones and the connection between projecting variables of polyhedral cones and reactions of metabolic networks we additionally employ oriented matroids to project certain reactions such that the projected network can be used to compute cut sets. Using projections leads again to networks containing fewer reactions and therefore a lot of known methods can be applied to genome-scale metabolic networks to compute cut sets where it was not possible before. We discuss these methods and the results in Chapter 8.

In Chapter 9 we summarise the results of this thesis and give an outlook for possible next steps and projects using this thesis as a basis or setup.

Chapter 2

Basics

In this chapter we introduce some basic notation, definitions and theorems necessary for different parts of this work. Knowledge needed for a specific topic only will be introduced in the section where it is used. We start with some notation, then we introduce metabolic networks and some well-known insights formally. Finally, we present some mathematical concepts regarding polyhedral cones, linear programming, and oriented matroids.

2.1 Notation

For a vector $x \in \mathbb{R}^n$ we denote by x_i the i -th element of x . For a set H of indices, x_H is the subvector of x corresponding to these indices. We use superscripts to refer to different vectors. For example, $\{x^1, \dots, x^t\}$ denotes a set of t vectors and we use a set of indices I here as well: $x^I = \{x^i \mid i \in I\}$. If all elements of a vector should be greater or equal to zero we write $x \geq 0$ instead of $x_i \geq 0$ for all i . With $\hat{e}_i \in \mathbb{R}^n$ we denote the i -th unit vector and the identity matrix is denoted by $E \in \mathbb{R}^{n \times n}$. We denote by $A_{i,*}$ the i -th row and by $A_{*,j}$ the j -th column of the matrix A . With $A_{H,*}$ resp. $A_{*,H}$, we refer to a set of rows, resp. to a set of columns of A . We denote the rank of a matrix $A \in \mathbb{R}^{m \times n}$ with $\rho(A)$, where $\rho(A)$ is the maximum number of linearly independent columns in A . The horizontal resp. vertical concatenation of matrices A, B is denoted by $(A|B)$ resp. $\begin{pmatrix} A \\ B \end{pmatrix}$. $\mathbf{0}_{m,n}$ are zero matrices of the size $m \times n$. Finally, \mathbb{B} are the binary numbers, thus $\mathbb{B} = \{0, 1\}$.

2. Basics

2.2 Metabolic networks

As already mentioned the focus of this thesis lies on metabolic networks. To fully describe a metabolic network, the set of internal metabolites \mathfrak{M} , reactions \mathfrak{R} , irreversible reactions $\text{Irr} \subseteq \mathfrak{R}$, and the stoichiometric matrix $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$ are needed:

Definition 2.1 (Metabolic network). *A tuple $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ with $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$ denotes a metabolic network. Its internal metabolites are denoted by \mathfrak{M} . The reactions are denoted by the set \mathfrak{R} , and $\text{Irr} \subseteq \mathfrak{R}$ are the irreversible reactions. S is the stoichiometric matrix.*

Note that the set of reversible reactions, denoted by Rev , is the complement of the irreversible reactions $\text{Rev} = \mathfrak{R} \setminus \text{Irr}$.

We noted before that the entries of the stoichiometric matrix S , the *stoichiometric coefficients* are positive, $S_{j,i} > 0$, if reaction $i \in \mathfrak{R}$ produces metabolite $j \in \mathfrak{M}$, negative, $S_{j,i} < 0$, if metabolite $j \in \mathfrak{M}$ is consumed by reaction $i \in \mathfrak{R}$, and 0 otherwise. We assume that the stoichiometric coefficients are rational numbers for computational reasons. For the sake of completeness we assume as well that we only deal with finite models, thus the number of metabolites and reactions describing the network \mathcal{N} is finite.

Since we do not consider a dynamical system, our network will always be in steady-state, i.e., we only consider vectors $v \in \mathbb{R}^{\mathfrak{R}}$ such that $Sv = 0$. Additionally, the entries corresponding to the irreversible reactions are all non-negative, i.e., $v_{\text{Irr}} \geq 0$. v is called a *feasible flux vector* and can be interpreted as a steady-state flux distribution of the metabolic network \mathcal{N} . We call v_i the *reaction rate* or *flux rate* of $i \in \mathfrak{R}$ or the *flux through* reaction i . If $v_i \neq 0$ we call i an *active* reaction. The set of all feasible flux vectors is called the *flux cone*:

Definition 2.2 (Flux cone and flux vector of a metabolic network). *Let $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$, with $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$, be a metabolic network. $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ is called the *flux cone* of \mathcal{N} . The vectors $v \in \Gamma_{\mathcal{N}}$ are called *flux vectors*.*

The constraints $v_{\text{Irr}} \geq 0$ restrict the irreversible reactions to carry flux in one direction only. We consider flux vectors $v \in \Gamma_{\mathcal{N}}$ as reversible or irreversible as well. If all non-zero entries of v correspond to reversible reactions only then we call v a *reversible flux vector*. Note that v being reversible implies $-v$ being a feasible flux vector too, thus $-v \in \Gamma_{\mathcal{N}}$. If any of the non-zero elements of v corresponds to an irreversible reaction then v is an *irreversible flux vector*, and therefore $-v$ is not feasible.

2.2.1 Elementary flux modes

One important set of flux vectors are the *elementary flux modes* (EFMs). To define the EFMs we need the notion of *support* of a flux vector $v \in \Gamma_{\mathcal{N}}$:

Definition 2.3 (Support). *The set $\text{supp}(v) := \{i \in \mathfrak{R} \mid v_i \neq 0\}$ is the support of a vector $v \in \mathbb{R}^n$.*

In metabolic networks the support of a flux vector $v \in \Gamma_{\mathcal{N}}$ corresponds to the set of *active reactions* in v . We use the notion of support for a set of vectors too. For example $\text{supp}(\Gamma_{\mathcal{N}}) = \{\text{supp}(v) \mid v \in \Gamma_{\mathcal{N}}\}$.

Definition 2.4 (Min). *Let \mathcal{V} be a family of subsets of a set U . With the operator Min we define the set of (inclusion) minimal subsets: $\text{Min}(\mathcal{V}) := \{v \in \mathcal{V} \mid \nexists w \in \mathcal{V} \setminus \{v\} \text{ s.t. } w \subset v\}$.*

We can now define the elementary flux modes of a metabolic network [Schuster and Hilgetag, 1994].

Definition 2.5 (Elementary flux mode (EFM)). *For a given metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$, an elementary flux mode (EFM) is a feasible flux vector $v \in \Gamma_{\mathcal{N}} \setminus \{0\}$ which has minimal support with respect to set inclusion, i.e., the set of EFMs is $\text{Min}(\text{supp}(\Gamma_{\mathcal{N}}))$.*

We denote by $\mathcal{E}_{\mathcal{N}}$ the finite set of EFMs in the metabolic network \mathcal{N} . An EFM $e \in \mathcal{E}_{\mathcal{N}}$ is called *reversible* if $e, -e \in \Gamma_{\mathcal{N}}$, otherwise e is called *irreversible*. The set of reversible resp. irreversible EFMs by $\mathcal{E}_{\mathcal{N}}^{\text{Rev}}$ resp. $\mathcal{E}_{\mathcal{N}}^{\text{Irr}}$.

EFMs define minimal sets of reactions that can operate together in steady-state. Minimality means that if any of the reactions is deleted, then the whole flux vector cannot operate in steady-state anymore. We refer to EFMs not only by flux vectors v but also using the indices of the active reactions of the flux vector v .

Example 2.6 (Elementary flux modes). *For the network of Figure 1.1 there exist 18 EFMs with the following sets of active reactions: $\{1, 2, 3, 4\}$, $\{1, 2, 3, 6, 7\}$, $\{1, 2, 3, 5, 6, 8\}$, $\{1, 2, 3, 6, 8, 9, 10\}$, $\{1, 2, 3, 6, 8, 11, 12\}$, $\{1, 2, 4, 5, 9, 10\}$, $\{1, 2, 4, 5, 11, 12\}$, $\{1, 2, 5, 6, 7, 9, 10\}$, $\{1, 2, 5, 6, 7, 11, 12\}$, $\{1, 2, 5, 6, 8, 9, 10\}$, $\{1, 2, 5, 6, 8, 11, 12\}$, $\{3, 5, 9, 10\}$, $\{3, 5, 11, 12\}$, $\{3, 4, 5, 6, 8\}$, $\{4, 6, 7\}$, $\{4, 6, 8, 9, 10\}$, $\{4, 6, 8, 11, 12\}$, $\{9, 10, 11, 12\}$. The sets $\{3, 5, 9, 10\}$, $\{3, 5, 11, 12\}$, $\{9, 10, 11, 12\}$ are reversible EFMs, which can carry flux in both directions. We computed the EFMs using the *efmtool* [Terzer, 2017a] which is explained in Subsection 2.3.1.4.*

2. Basics

EFMs are a popular approach to analyse metabolic networks because every steady-state behaviour of the network can be represented with help of the EFMs [Schuster and Hilgetag, 1994; Schuster et al., 2000]. Formally speaking, the set $\mathcal{E}_{\mathcal{N}}$ is a *generating set* or *conic basis* of the flux cone $\Gamma_{\mathcal{N}}$. This means that every flux vector $v \in \Gamma_{\mathcal{N}}$ can be represented as a *conical combination* $v = \sum_{e \in \mathcal{E}_{\mathcal{N}}} \lambda_e e$, for some $\lambda_e \geq 0$. Note that for a reversible EFM, we use two flux vectors, each representing one direction.

EFMs can be applied to study different properties of the underlying metabolic network [Stelling et al., 2002; Zanghellini et al., 2013]. For instance, they can be used to study the robustness of the network [Behre et al., 2008] or to search a decomposition into EFMs for given flux vectors [Schwartz and Kanehisa, 2005, 2006; Chan and Ji, 2011]. EFMs can also be used to detect important pathways [Carlson and Srienc, 2004a,b], for example pathways which produce certain metabolites. Metabolic engineering can be done using EFMs [Trinh et al., 2006, 2008; Melzer et al., 2009; Trinh and Srienc, 2009; Unrean et al., 2010; Trinh et al., 2011; Ruckerbauer et al., 2015]. EFMs do not only play an important role in the research of metabolic networks but also in this thesis. We improve an already existing method to compute EFMs in Chapter 3. We reduce the solution space and therefore speed up the running time of the algorithm for computing EFMs. In Chapter 6 we introduce a new method to compute EFMs. The method from Chapter 6 finds a minimum set of EFMs which describes the whole metabolic network. The minimum number of EFMs needed to describe the network is significantly smaller than the whole set of EFMs. Thus such a set can be computed for genome-scale metabolic networks whereas computing the whole set of EFMs is not possible.

2.2.2 Minimal metabolic behaviours

Computing the whole set of EFMs for genome-scale metabolic networks is often not possible, since the number of EFMs grows exponentially with the size of the network. [Larhlimi and Bockmayr, 2009] introduced a minimum description of the flux cone $\Gamma_{\mathcal{N}}$ based on *minimal metabolic behaviours* (MMBs). Each MMB corresponds to a subset of EFMs with the same set of active irreversible reactions. Larhlimi and Bockmayr defined them as follows:

Definition 2.7 (Minimal metabolic behaviour). *A metabolic behaviour is a non-empty set of irreversible reactions $D \subseteq \text{Irr}$ which carries flux in a flux vector $v \in \Gamma_{\mathcal{N}}$: $D = \{i \in \text{Irr} \mid v_i \neq 0\}$. A metabolic behaviour is minimal if there is no other metabolic behaviour D' contained in D .*

Example 2.8 (Minimal metabolic behaviours). For the network in Figure 1.1 we can find three different MMBs: The first MMB consists of two reactions, namely reactions 1 and 2. $\{1, 2\}$ is the set of active irreversible reactions of the EFM $\{1, 2, 3, 4\}$ or $\{1, 2, 4, 5, 11, 12\}$. The second MMB is $\{6, 7\}$ which is the set of active irreversible reactions of the EFM $\{4, 6, 7\}$. The last MMB is $\{6, 8\}$, which is the set of active irreversible reactions of the EFM $\{4, 6, 8, 9, 10\}$. Other EFMs contain this MMB as well, e.g. $\{4, 6, 8, 11, 12\}$.

It can be shown that in order to obtain a minimal generating set of $\Gamma_{\mathcal{N}}$, for each MMB one of the corresponding EFMs has to be chosen [Larhlmi and Bockmayr, 2009], see also [Jevremović and Boley, 2013]. We make use of MMBs to prove that a method we introduce in Chapter 6 computes a minimum set of EFMs that represent a metabolic network \mathcal{N} . Furthermore, we use MMBs to compute minimal cut sets, which are introduced next.

2.2.3 Minimal cut sets

EFMs can be used for metabolic engineering. One strategy is to search for genes whose knock-outs lead to a desired modification of the metabolic network. One or several genes encode an enzyme which then catalyses a reaction. Therefore, instead of searching directly for the genes, one can search for reactions to be knocked-out, resp. removed. Those intervention strategies are for example used to analyse the structural robustness of metabolic networks [Deutscher et al., 2006; Behre et al., 2008]. Furthermore, they can be employed to block a certain behaviour [Haus et al., 2008] or, the other way around, to find most efficient pathways for the production of certain compounds [Trinh et al., 2006; von Kamp and Klamt, 2014].

Suppose the goal is to prevent a *target reaction* from carrying flux. A set of reactions which fulfils this goal is called a *cut set* [Klamt and Gilles, 2004]:

Definition 2.9 (Cut set). Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$, a target reaction $\text{tar} \in \mathfrak{R}$, and a flux rate $v_{\text{tar}}^* \in \mathbb{R}$. A set of reactions $\xi \subseteq \mathfrak{R}$ is called a cut set (with respect to the defined target reaction tar) if after the removal of these reactions ξ from the network \mathcal{N} it is no longer possible to achieve a flux rate of most v_{tar}^* for tar . Thus there exists no $v \in \Gamma_{\mathcal{N}}$ with $v_{\text{tar}} > v_{\text{tar}}^*$ and $v_{\xi} = 0$.

A cut set which consists of only one reaction is called an *essential reaction*:

2. Basics

Definition 2.10 (Essential reaction). *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$, a target reaction $\text{tar} \in \mathfrak{R}$, and a flux rate $v_{\text{tar}}^* \in \mathbb{R}$. We call a reaction $j \in \mathfrak{R}$ essential if, after removing j , it is no longer possible to achieve a flux rate of at least v_{tar}^* through the target reaction tar .*

Typically, tar is the biomass reaction of \mathcal{N} and v_{tar}^* is 20% of the maximal flux rate of tar [Klamt and Stelling, 2003].

We explain how to find the maximal rate through a given reaction in Section 2.3.2.1. An obvious essential reaction (a cut set of cardinality 1) would be the target reaction itself. But it can be interesting to know other essential reactions as well, especially since the target reaction is most of the time the biomass reaction and it is not feasible to knock out an artificial reaction in the laboratory. In Definition 2.9 we do not only consider a single reaction but also a set of reactions as cut sets. Besides, it is possible to have not only one but several target reactions. The minimality of cut sets is defined, again, w.r.t. set inclusion:

Definition 2.11 (Minimal cut set (MCS)). *A cut set $\zeta \subseteq \mathfrak{R}$ (with respect to a defined target reaction) is a minimal cut set (MCS) if no proper subset of ζ is a cut set (w.r.t. the target reaction), thus if there does not exist a cut set ξ (w.r.t. the target reaction) with $\xi \subset \zeta$.*

Since all feasible flux vectors of a network can be represented using EFMs, one method for finding reactions of cut sets is searching for *hitting sets* within the EFMs [Klamt and Gilles, 2004; Klamt, 2006].

MCSs can be computed from EFMs in the following way [Klamt and Gilles, 2004]: Suppose all EFMs are known and a target reaction is given. Let EFMs_{tar} be the set of those EFMs that involve the target reaction. Given EFMs_{tar} , we search for *hitting sets*, i.e., sets of reactions such that each EFM in EFMs_{tar} contains at least one reaction of the hitting set. Inclusion-minimal hitting sets are the MCSs we are looking for. Note that MCSs found with this technique do not allow any flux through the target reaction, i.e., $v_{\text{tar}} = 0$ for all $v \in \Gamma_{\mathcal{N}}$ with $v_{\xi} = 0$.

Example 2.12 (How to compute MCSs using EFMs). *For the network in Figure 1.1 there exist 18 EFMs, see Example 2.6. We choose reaction 7 as the target reaction, see Figure 2.1 for illustration. There are 4 EFMs using reaction 7 with the supports $\{1, 2, 3, 6, 7\}$, $\{1, 2, 5, 6, 7, 9, 10\}$, $\{1, 2, 5, 6, 7, 11, 12\}$, $\{4, 6, 7\}$. In all these EFMs, reaction 6 is active. Therefore, if reaction 6 is removed, there will be no feasible flux vector using reaction 7. Consequently, reaction 6 is an MCS for the target reaction 7. Minimal cut sets of cardinality 2 are $\{1, 4\}$ and $\{2, 4\}$. The set $\{1, 6\}$ is*

2.2 Metabolic networks

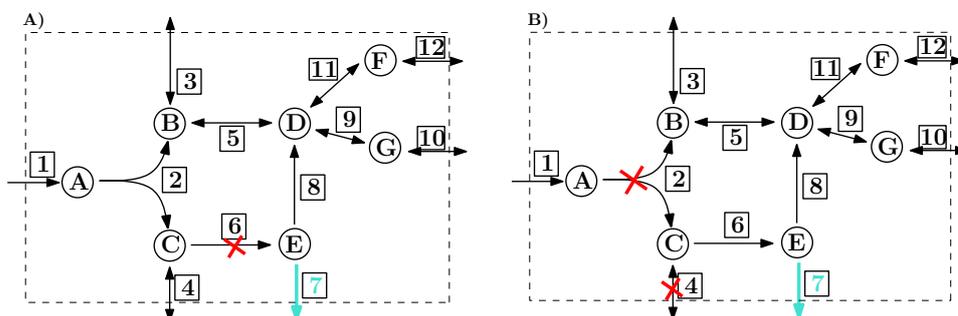


Figure 2.1: Examples of minimal cut sets where reaction 7 is the target reaction. **A)** If reaction 6 is not present, e.g. knocked out, there is no flux through reaction 7. Therefore reaction 6 is an MCS of cardinality 1. **B)** If the reactions 2 and 4 are not present, there is no flux through reaction 7. Therefore reaction 2 and 4 form an MCS of cardinality 2.

not an MCS since it contains the MCS {6}. The remaining MCSs are {3, 4, 5}, {3, 4, 9, 11}, {3, 4, 10, 12}, {3, 4, 10, 11}, {3, 4, 9, 12}.

Computing EFMs for genome-scale metabolic networks is often not possible, since the number of EFMs grows exponentially with the size of the network. Therefore the method described above may not be applicable in practice.

To address this problem, we propose in Chapter 7 to make use of *minimal metabolic behaviours* (MMBs) [Larhlimi and Bockmayr, 2009], see Definition 2.7. The number of MMBs may be several orders of magnitude smaller than the number of EFMs, see Table 7.1. Our idea is to use MMBs in the same way as EFMs in order to compute MCSs that consist of irreversible reactions only. The method can be applied to genome-scale metabolic networks whereas most algorithms fail to compute the whole set of EFMs.

There exist other methods as well to compute MCSs of a metabolic network. Several are based on linear programming [Burgard et al., 2003; Pharkya and Maranas, 2006; Suthers et al., 2009b], while in [Goldstein and Bockmayr, 2015] lattice theory is used. In [Jungreuthmayer et al., 2013] the Berge algorithm [Berge, 1984] is exploited, whereas [Nair et al., 2017] use swarm optimisation. Methods to compute MCSs such that desired functionalities of the network are not affected are presented in [Li et al., 2009; Hädicke and Klamt, 2011]. Some utilise the fact that MCSs are EFMs in the dual network [Klamt and Stelling, 2003; Ballerstein et al., 2012] or combine this fact with LP formulations [von Kamp and Klamt, 2014; Apaolaza et al., 2017]. The complexity of computing minimal cut sets using different approaches is discussed in [Acuña et al., 2009].

2. Basics

2.2.4 Bounds

If we consider the flux cone $\Gamma_{\mathcal{N}}$ of a metabolic network \mathcal{N} we do not have any limitations on the flux rates. Therefore, we simulate an environment where an infinite amount of nutrients is available. To compute the EFMs, this is a feasible approach. But *in silico* experiments are often used to predict the behaviour of an organism for *in vivo* experiments and therefore the nutrient uptake rates have to be bounded. This is done by introducing for each flux rate v_i an upper bound $u_i \in \mathbb{R}$ and lower bound $l_i \in \mathbb{R}$. The flux cone $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ is thus transformed into $P_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, l \leq v \leq u\}$. We assume $l_{\text{Irr}} \geq 0$, thus the direction of the irreversible reactions is already determined by their lower bounds and we do not have to include the constraints $v_{\text{Irr}} \geq 0$ into $P_{\mathcal{N}}$.

2.3 Mathematical basics

In this section we give some mathematical basics needed for this thesis. We go into more details in the corresponding sections. The concepts we consider here are polyhedral cones, linear programs (LPs), mixed integer linear programs (MILPs), and oriented matroids.

We start with two definitions related to dependencies of vectors.

Definition 2.13 (Linearly independent). *A set of vectors $\{x^1, \dots, x^t\}$ is called linearly independent if $\sum_{i=1}^t \alpha_i x^i = 0$ for $\alpha_i \in \mathbb{R}$ implies that $\alpha_i = 0$ for all i .*

Definition 2.14 (Affinely independent). *A set of vectors $\{x^1, \dots, x^t\}$ is called affinely independent if $\sum_{i=1}^t \alpha_i x^i = 0$ and $\sum_{i=1}^t \alpha_i = 0$ for $\alpha_i \in \mathbb{R}$ implies that $\alpha_i = 0$ for all i .*

2.3.1 Polyhedra

To use the mathematical concepts of polyhedra and cones we introduce in this section some basics about polyhedra. We start by some fundamental definitions and theorems related to polyhedral cones. The basis of this section is the book [Schrijver, 1998].

2.3.1.1 Basics and connection to metabolic networks

We begin with the definition of a cone:

Definition 2.15 (Cone). *A subset $\Gamma \subseteq \mathbb{R}^n$ is called a cone if any conic combination of two elements $x, y \in \Gamma$ belongs to Γ again, i.e., $\lambda x + \mu y \in \Gamma$, for any non-negative $\lambda, \mu \in \mathbb{R}_{\geq 0}$.*

The cones we consider here are polyhedral cones:

Definition 2.16 (Polyhedral Cone). *A cone $\Gamma \subseteq \mathbb{R}^n$ is called polyhedral if there exists a matrix $A \in \mathbb{R}^{m \times n}$ such that $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$.*

They are called polyhedral cones because they are cones as well as polyhedra:

Definition 2.17 (Polyhedron). *A set $P = \{x \in \mathbb{R}^n \mid Ax \geq b\}$ with $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$ is called a polyhedron.*

If $b_i = 0$ for all i , i.e., if the inequality system $Ax \geq b$ is homogeneous, then P is a polyhedron and a cone.

A cone Γ is *finitely generated* if there exist finite sets $\mathcal{G} \subset \Gamma_{\mathcal{N}}$ such that every flux vector $v \in \Gamma_{\mathcal{N}}$ can be obtained as a conical combination $v = \sum_{g \in \mathcal{G}} \lambda_g g$, $\lambda_g \geq 0$ of the elements $g \in \mathcal{G}$. By a theorem of Farkas-Minkowski-Weyl, a cone is polyhedral if and only if it has a finite set of generators.

Theorem 2.18 (Theorem 7.1a in [Schrijver, 1998], pp. 87). *A cone is polyhedral if and only if it is finitely generated.*

The lineality space of a polyhedral cone will play an important role:

Definition 2.19 (Lineality Space). *Given a polyhedral cone $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$, with $A \in \mathbb{R}^{m \times n}$, the linear subspace $\Lambda = \{x \in \mathbb{R}^n \mid Ax = 0\} = \Gamma \cap (-\Gamma)$ is called the lineality space of Γ .*

We also need the definition of a face:

Definition 2.20 (Face). *For a polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \geq b\}$, we call a set $F = \{x \in P \mid a^\top x = b\}$ a face of P , if $a^\top x \geq b$, $a \in \mathbb{R}^n \setminus \{0\}$ is a valid inequality for P , i.e., $\Gamma \subseteq \{x \in \mathbb{R}^n \mid a^\top x \geq b\}$.*

The *dimension* of a polyhedron P , and therefore also of a polyhedral cone, is the maximum number of affinely independent points in P minus one. A *minimal proper face* (MPF) is a face of the cone of dimension $t + 1$, where t is the dimension of the lineality space. A *vertex* is a face of dimension 0. Note that a face is a polyhedron too.

2. Basics

Definition 2.21 (Facet). *Given a polyhedron $P = \{x \in \mathbb{R}^n \mid Ax \geq b\}$, with $A \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$, a face F of P is called a facet if $\dim(F) = \dim(P) - 1$.*

We call all non-zero elements of a polyhedral cone rays:

Definition 2.22 (Ray). *Non-zero elements of a polyhedral cone $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$, with $A \in \mathbb{R}^{m \times n}$, are called rays of Γ .*

A polyhedral cone Γ is called *pointed* if $\Lambda = \{0\}$. This implies that whenever $x \in \Gamma \setminus \{0\}$, we have $-x \notin \Gamma$. In other words, Γ does not contain any line $\{\lambda x \mid \lambda \in \mathbb{R}\}$, for $x \neq 0$. By basic linear algebra, the cone Γ is pointed if and only if the matrix A has full column rank, i.e., $\rho(A) = n$. See Figure 2.2 for the illustration of a non-pointed polyhedral cone.

Two rays $x, x' \in \Gamma$ are considered identical if $x' = \lambda x$ for some $\lambda > 0$. The rays which will play an important role for our purposes are the extreme rays:

Definition 2.23 (Extreme ray (ER)). *Let $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$, with $A \in \mathbb{R}^{m \times n}$, be a polyhedral cone. A ray $y \in \Gamma \setminus \{0\}$ is called an extreme ray (ER) of Γ if there exist no two linearly independent rays $x^1, x^2 \in \Gamma$ such that $y = x^1 + x^2$.*

If Γ is pointed the minimal proper faces have dimension one. They form a unique minimum set of generators $\{g^1, \dots, g^l\} \subseteq \Gamma$ which correspond to the ERs of Γ . See Figure 2.2 for an illustration.

To have a different characterisation of ERs of a pointed polyhedral cone we can use the zero set:

Definition 2.24 (Zero set). *We denote with $Z(x) = \{i \mid x_i = 0\}$ the set of indices of entries of a given vector $x \in \mathbb{R}^n$ which are zero. We call this set the zero set.*

The characterisation of the ERs is then as follows:

Theorem 2.25 ([Fukuda and Prodon, 1996] and [Jevremović et al., 2008]). *A ray x of a pointed polyhedral cone $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$ is an extreme ray if and only if $\rho(A_{Z(Ax),*}) = n - 1$, where $A \in \mathbb{R}^{m \times n}$ (with $m \leq n$) has full rank (i.e., $\rho(A) = n$).*

A particular type of polyhedral cone we consider here is the *flux cone*:

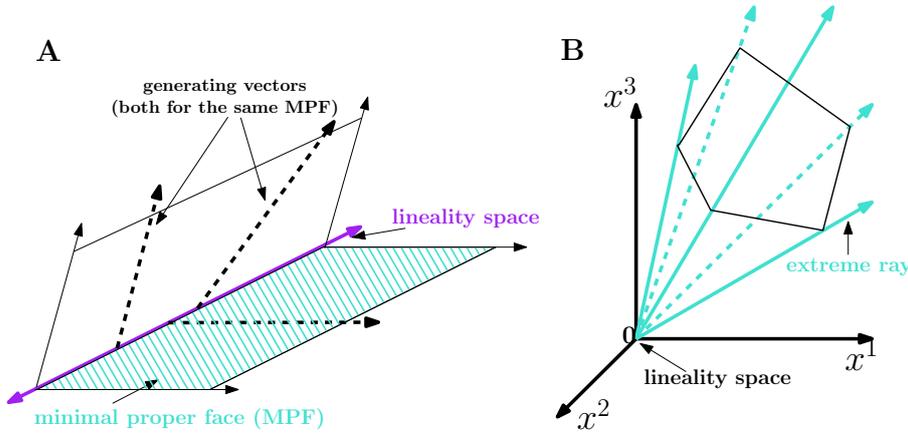


Figure 2.2: **(A)** A non-pointed polyhedral cone in 3-dimensional space with a lineality space of dimension 1 and 2 minimal proper faces (MPFs) of dimension 2. The dashed vectors in one of the MPFs illustrate that the set of generating vectors is not unique: one of the two vectors can be used as well as any other vector in the MPF. To represent the whole polyhedral cone, one vector of each MPF is needed and one basis vector of the lineality space. **(B)** A pointed polyhedral cone in 3-dimensional space with 5 extreme rays. The lineality space is $\{0\}$.

Definition 2.26 (Flux cone). *A polyhedral cone Γ is called a flux cone if there exists a matrix $B \in \mathbb{R}^{m \times n}$ and an index set $I \subseteq \{1, \dots, n\}$ such that*

$$\Gamma = \{x \in \mathbb{R}^n \mid \begin{pmatrix} B \\ -B \\ E_{I,*} \end{pmatrix} x \geq 0\} = \{x \in \mathbb{R}^n \mid Bx = 0, x_I \geq 0\}.$$

Either B or I may be empty, but not both.

Clearly, the flux cone $\Gamma_{\mathcal{N}}$ of a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$, see Definition 2.2, is a flux cone in the sense of Definition 2.26, with $B = S$ and $I = \text{Irr}$.

Proposition 2.27. *Let Γ be a flux cone. If $x \geq 0$ for all $x \in \Gamma$, then Γ is pointed and the extreme rays of Γ are exactly the rays in Γ of minimal support.*

In [Schuster and Hilgetag, 1994; Gagneur and Klamt, 2004], this result is proven for flux cones $\Gamma = \Gamma_{\mathcal{N}}$ originating from a metabolic network \mathcal{N} . In this case, Proposition 2.27 states that in a metabolic network where all reactions are irreversible the extreme rays of $\Gamma_{\mathcal{N}}$ are exactly the EFMs. The proof in [Schuster and Hilgetag, 1994; Gagneur and Klamt, 2004] directly

2. Basics

carries over to the more general case of flux cones in the sense of Definition 2.26 for which $I = \{1, \dots, n\}$.

In general, however, metabolic networks may contain reversible reactions together with reversible flux vectors $v \neq 0$ for which $v, -v \in \Gamma_{\mathcal{N}}$. For example in Figure 1.1, there is a reversible flux vector with support $\{3, 5, 9, 10\}$. In such networks the flux cone is not pointed and Proposition 2.27 does not apply.

Similar to the method of verifying extreme rays, we can verify if a given flux vector v is an EFM using the support of a vector:

Theorem 2.28 ([Urbanczik and Wagner, 2005a; Jevremović et al., 2008]). *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and its flux cone $\Gamma_{\mathcal{N}}$, a flux vector $v \in \Gamma_{\mathcal{N}}$ is an elementary flux mode of \mathcal{N} if and only if $\rho(S_{*, \text{supp}(v)}) = |\text{supp}(v)| - 1$, where $|\text{supp}(v)|$ is the number of non-zero entries of v .*

If there exist reversible flux vectors the minimal generating set of the flux cone is not unique anymore. One way to compute the whole set of EFMs is by splitting all reversible reactions which is explained in the following.

2.3.1.2 Splitting variables

In many cases it is desirable to have a cone $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$ which is pointed. If the system $Ax \geq 0$ includes constraints $x_i \geq 0$, for all $i \in \{1, \dots, n\}$, then the resulting cone will be pointed. In general, however, a variable $x_i \in \mathbb{R}$ can take negative values, and we cannot simply add the constraint $x_i \geq 0$.

To overcome this problem, a well-known method also used in linear programming [Schrijver, 1998] is to *split* variables. Splitting a variable $x_i \in \mathbb{R}$ means replacing it by two non-negative variables $\tilde{x}_i, \tilde{x}_{n+i} \geq 0$, such that $x_i = \tilde{x}_i - \tilde{x}_{n+i}$. Note that this will change the structure of the cone and increase the dimension of the underlying vector space by 1 for each split variable.

Definition 2.29 (π_I). *To describe the split variables transformation formally and for several indices, we use a map $\pi_I : \mathbb{R}^n \rightarrow \mathbb{R}^{n+|I|}$, where $I = \{i_1, \dots, i_s\}$ denotes the set of variables $x_{i_k}, i_k \in I, k \in \{1, \dots, s\}$ to be split.*

For $x \in \mathbb{R}^n$ we get $\pi_I(x) = \tilde{x}$ with $\tilde{x}_j = x_j$, for all $j \in \{1, \dots, n\} \setminus I$, and for each $i_k \in I$:

$$\begin{aligned} \tilde{x}_{i_k} &= x_{i_k} & \text{and} & & \tilde{x}_{n+k} &= 0 & \text{if } x_{i_k} &\geq 0, \\ \tilde{x}_{i_k} &= 0 & \text{and} & & \tilde{x}_{n+k} &= -x_{i_k} & \text{if } x_{i_k} &< 0. \end{aligned} \tag{2.1}$$

By applying the map π_I , a polyhedral cone $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$, with $A \in \mathbb{R}^{m \times n}$, is mapped to the *augmented* polyhedral cone

$$\tilde{\Gamma}^I = \left\{ \tilde{x} = (x, y) \in \mathbb{R}^{n+|I|} \mid Ax - A_{*,I}y \geq 0, x_I \geq 0, y \geq 0 \right\}, \quad (2.2)$$

with $x \in \mathbb{R}^n$, $y \in \mathbb{R}^{|I|}$.

Definition 2.30 (π_I^r , recombine). *The inverse transformation $\pi_I^r : \tilde{\Gamma}^I \rightarrow \Gamma$ maps each vector $\tilde{x} \in \tilde{\Gamma}^I$ to $x = \pi_I^r(\tilde{x})$ such that $x_j = \tilde{x}_j$, for all $j \in \{1, \dots, n\} \setminus I$, and $x_{i_k} = \tilde{x}_{i_k} - \tilde{x}_{n+k}$, for all $i_k \in I, k \in \{1, \dots, s\}$. If we apply π_I^r , we say that we recombine the variables that were split before.*

2.3.1.3 Splitting reversible reactions

In the case of metabolic networks, the variables corresponding to the irreversible reactions, by definition, can take only non-negative values. In order to obtain a pointed cone, we can split each reversible reaction into two irreversible ones.

Example 2.31 (Splitting all reversible reactions). *In Figure 2.3 all reversible reactions of the network in Figure 1.1 are split. After splitting, there exists no reversible flux vector anymore. For example, a flux vector of the original network with support $\{3, 5, 9, 10\}$ has now either the support $\{3^-, 5^+, 9^+, 10^+\}$ or $\{3^+, 5^-, 9^-, 10^-\}$ (the reverse direction).*

This leads to a new network $\tilde{\mathcal{N}}$ with the corresponding pointed flux cone $\tilde{\Gamma}_{\tilde{\mathcal{N}}}^{\text{Rev}}$. The uniquely determined extreme rays of this cone are called *extreme currents*:

Definition 2.32 (Extreme Currents (ECs), [Clarke, 1988; Schuster et al., 2000]). *Let $\Gamma_{\mathcal{N}}$ be the flux cone of a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and let $\tilde{\Gamma}_{\mathcal{N}}^{\text{Rev}}$ be the pointed polyhedral cone after splitting all reversible reactions Rev. The ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\text{Rev}}$ are called extreme currents (ECs) of \mathcal{N} .*

It can be shown that after recombination, the ECs correspond exactly to the EFMs of the metabolic network [Gagneur and Klamt, 2004]. In addition, for each split reaction $i_k \in I$, there exists a *2-cycle*:

Definition 2.33 (2-cycle). *Let $\Gamma_{\mathcal{N}}$ be the flux cone of a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and let $\tilde{\Gamma}_S^I$ be an augmented flux cone, where a subset $I \subseteq \text{Rev}$ of reversible reactions were split. All flux vectors $\bar{v} \in \tilde{\Gamma}_S^I$ can be recombined using the map π_I^r to become flux vectors in the cone $\Gamma_{\mathcal{N}}$. The set of 2-cycles $\Pi_I \subseteq \tilde{\Gamma}_S^I$ can be defined as follows:*

$$\Pi_I = \left\{ \bar{v} \in \tilde{\Gamma}_S^I \mid \bar{v}_j = 0 \forall j \in \{1, \dots, n + |I|\} \setminus \{i_k, n + k\}, \right. \\ \left. \text{and } \bar{v}_{i_k} = \bar{v}_{n+k} \neq 0 \text{ for } i_k \in I \right\}.$$

2. Basics

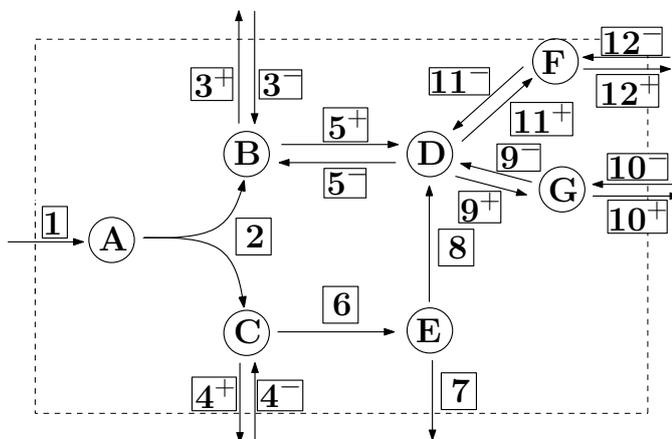


Figure 2.3: Network from Figure 1.1 where all reversible reactions are split. For example, reaction 3 is split into 3^- and 3^+ .

The set of 2-cycles are non-zero flux vectors in $\tilde{\Gamma}_S^I$ but after recombination they become zero-flux vectors. These cycles do not have a biological meaning and can be eliminated.

Splitting all reversible reactions will highly increase the number of variables and thus the dimension of the vector space where the augmented cone lives. Already for medium-sized networks, the number of extreme rays, which corresponds to the number of EFMs (up to the 2-cycles), will be huge and therefore computing the whole set may not be feasible or desirable. In Chapter 5 and Chapter 6 we discuss how to obtain pointed polyhedral cones by splitting only a subset of the reversible reactions. We show that the ERs of the augmented cones correspond to a subset of the EFMs of the underlying network which represents the whole metabolism.

2.3.1.4 Computing extreme rays and EFMs

Computing extreme rays is not easy. The most famous work regarding this challenge is the *Double Description method* (DD-method) [Fukuda and Prodon, 1996]. It is called the DD-method because for a matrix $A \in \mathbb{R}^{m \times n}$ of full rank, describing a pointed polyhedral cone $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$ there always exists a matrix $R \in \mathbb{R}^{n \times l}$ which describes the same cone $\Gamma = \{x \in \mathbb{R}^n \mid x = R \cdot \lambda \text{ for some } \lambda \in \mathbb{R}_{\geq 0}^l\}$. The pair (A, R) is called a *double-description pair*. The DD-method computes for a given $A \in \mathbb{R}^{m \times n}$ a matrix $R \in \mathbb{R}^{n \times l}$ such that the columns of R correspond to the extreme rays of Γ . There exist several tools which compute the ERs of a given pointed polyhedral cone. `polco` [Terzer, 2017b, 2009a] is nowadays the fastest tool

and is based on the DD-method, including many improvements. In this work we use `polco` whenever we compute the ERs of a pointed polyhedral cone.

There exist several methods for computing the set of all EFMs and most of them are based on the DD-method [Fukuda and Prodon, 1996]. For example `efmtool` [Terzer, 2017a] or `metatool` [Pfeiffer et al., 1999] split all reversible reactions and compute the ERs of the augmented flux cone. In [Gagneur and Klamt, 2004], a description of the DD-method for metabolic networks is given. Another technique based on the DD-method is the null-space algorithm in [Urbanczik and Wagner, 2005a]. Other approaches are based on graph theory like `gEFM` in [Ullah et al., 2016] and a depth-first algorithm in [Quek and Nielsen, 2014]. All these methods can compute the whole set of EFMs. In practice, however, they often do not terminate because the number of EFMs gets too large [Klamt and Stelling, 2002] and the program runs out of memory.

We will see in Chapter 6 how reaction splitting can be used to find a minimum set of EFMs which describes the whole metabolic network. To do so we also need some help from oriented matroid theory, for which we give some basics in Section 2.3.5.

More recent approaches to compute a subset of EFMs are based on mixed-integer programming which is introduced in the following section.

2.3.2 Linear Programming

As explained in Section 2.2.4, *in silico* experiments require bounds on the flux rates. The steady-state assumption ($Sv = 0$) and the restriction on the flux rates of the irreversible reactions ($v_{\text{irr}} \geq 0$) define a polyhedral cone. If the bounds on the flux rates ($l \leq v \leq u$) are included we do not have a polyhedral cone anymore but a polyhedron, see Definition 2.17. One method for analysing polyhedra is linear programming. The following subsection is based on the notations used in [Schrijver, 1998], which is a comprehensive introduction to the field of linear programming.

A general, finite dimensional, optimisation program can be defined as follows:

Definition 2.34 (Optimisation program).

$$\begin{aligned} \min_x \quad & f(x) \\ \text{s.t.} \quad & g_i(x) \leq 0 \\ & x \in D, \end{aligned}$$

with $D \subseteq \mathbb{R}^n$, $g_i : D \rightarrow \mathbb{R}$ for $i = 1, \dots, m$, and $f : D \rightarrow \mathbb{R}$.

2. Basics

Instead of minimising we could also maximise. We call $f(x)$ an *objective function* and $g_i(x) \leq 0$ (*side*) *constraints*. The definition of an optimisation program is very general. The objective can be any function and so can be the constraints. In this work we only consider *linear* programs (LP) which means that f and g_i are linear in x for all i . We can also define an LP using vectors and matrices instead of f and g_i which is convenient and which we do in the following as well.

Definition 2.35 (Linear program (LP)).

$$\begin{aligned} \min_x \quad & c^\top x \\ \text{s.t.} \quad & Ax \leq b \\ & x \in \mathbb{R}^n, \end{aligned}$$

with $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, where c^\top denotes the transpose of the vector c .

Note that every LP can be brought to the form in Definition 2.35. For example every equality constraint can be equally stated as two inequality constraints.

A vector $x^* \in \mathbb{R}^n$ is called a *feasible solution* for an LP if it fulfils all constraints, i.e., if $Ax^* \leq b$. The *feasible region* or the *solution space* is the set of all feasible solutions. A feasible solution $x^* \in \mathbb{R}^n$ is an *optimal solution* if $c^\top x^* \leq c^\top x$ for all $x \in \mathbb{R}^n$ with $Ax \leq b$. Note that a feasible solution does not have to exist. If a feasible solution exists, there does not have to be an optimal solution and if there is an optimal solution it does not have to be unique. The value $z_{opt} = c^\top x^*$ for an optimal solution x^* is called the *objective value*. If a finite optimal solution exists, this value is unique for every optimal solution x^* .

Every constraint of an LP can be considered as a half space in \mathbb{R}^n which cuts the solution space. Therefore $A_{i,*}x \leq b_i$ cuts all solutions x away for which it holds $A_{i,*}x > b_i$. The intersections of all half spaces define a polyhedron. See Figure 2.4 for an example.

The best known way to solve an LP is using the *simplex algorithm* [Dantzig, 1948]. Suppose the feasible region describes a pointed polyhedron, i.e. the feasible region does not contain any line. Roughly speaking, the algorithm starts by checking if a feasible solution exists. If this is the case, a feasible vertex of the polyhedron is computed. If such a vertex exists the algorithm tries iteratively to find a vertex next to the current position, while improving the objective value. The simplex algorithm stops, if either an optimal solution is found, the problem is unbounded or the problem is infeasible.

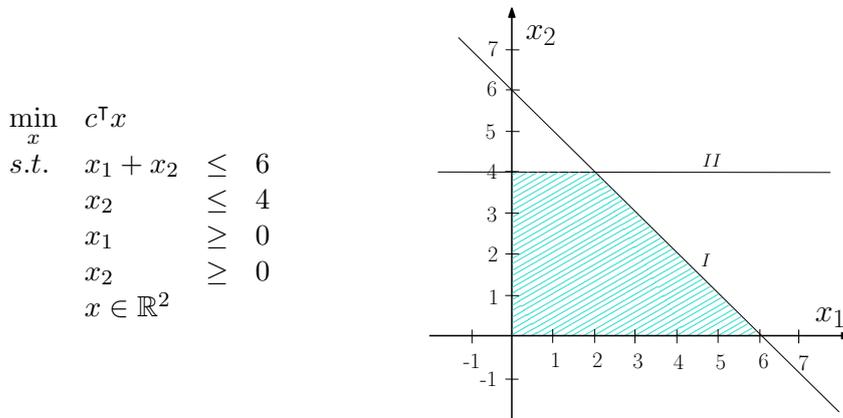


Figure 2.4: For each constraint, i.e., for each inequality, we have one half space cutting \mathbb{R}^2 . The turquoise area is the feasible region which is, in this case, bounded.

It has been shown that, if some rules are followed, the simplex will terminate and find an optimal solution. In the worst case the algorithm needs an exponential number of iterations [Klee and Minty, 1972]. Fortunately, the simplex method is very fast in practice [Borgwardt, 1982]. Another method, which is proven to have a run time that is polynomial in the size of the input is the ellipsoid method [Khachiyan, 1980]. For the first implementation the running time of the method was worse than for the simplex method but in 1984 [Karmarkar, 1984] introduced an efficient method. Due to numerical stability issues, most LPs are still solved using the simplex algorithm [IBM Knowledge Center, 2010].

2.3.2.1 Application for LP: Flux balance analysis

The best known application of an LP studying metabolic network reconstructions is *flux balance analysis* (FBA) [Varma and Palsson, 1994]. This method assumes the cell has evolved to maximise a certain biological objective, such as growing as fast as possible. Thus, in most applications, the objective is to maximise the biomass production, i.e., the growth yield. To do so, the artificial biomass reaction is added to the network. Since all constraints involved are linear, an LP can be formulated:

$$\begin{array}{ll}
 \max_v & v_{\text{bio}} \\
 \text{s.t.} & Sv = 0 \\
 & v \geq l \\
 & v \leq u \\
 & v \in \mathbb{R}^n,
 \end{array} \tag{2.3}$$

2. Basics

where $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$ is the stoichiometric matrix, and $l \in \mathbb{R}^{\mathfrak{R}}$, resp. $u \in \mathbb{R}^{\mathfrak{R}}$ are lower bounds, resp. upper bounds on the flux rates.

2.3.2.2 Application for LP: Flux variability analysis

The algorithmic solution of an FBA problem $\max\{c^T v \mid Sv = 0, l \leq v \leq u\}$ gives one optimal flux vector $v^* \in \Gamma_{\mathcal{N}}$, maximising the biomass production. This solution does not have to be unique. There can exist infinitely many vectors, all of them maximising the growth yield. Therefore, computing all optimal flux vectors cannot be done. To still analyse the optimal solution space, thus all possible optimal flux vectors, *flux variability analysis* (FVA) [Mahadevan and Schilling, 2003] can be applied. Here, all reactions rates are maximised and minimised, given that the biomass reaction has the maximum reaction rate v_{bio}^* , obtained by performing an FBA. To perform an FVA two LPs have to be solved for each reaction.

For all $i \in \mathfrak{R}$:

$$\begin{aligned} \max_v \quad & \pm v_i \\ \text{s.t.} \quad & Sv = 0 \\ & v \geq l \\ & v \leq u \\ & v_{\text{bio}} = v_{\text{bio}}^* \\ & v \in \mathbb{R}^{\mathfrak{R}} \end{aligned} \tag{2.4}$$

FVA gives the flux range for each reaction under the constraint that the biomass reaction rate is maximal.

2.3.2.3 Application for LP: Flux coupling analysis

One way to understand the topology and robustness of a metabolic network is *Flux Coupling Analysis* (FCA) [Burgard et al., 2004]. It can be performed quite fast [David et al., 2011; Larhlimi et al., 2012b] and recently it has been generalised via a lattice-theoretic framework [Reimers et al., 2015]. To detect which reactions are coupled, LPs can be used [Burgard et al., 2004; David et al., 2011; Larhlimi et al., 2012b].

Definition 2.36 (Blocked reaction). *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. A reaction $i \in \mathfrak{R}$ is called blocked if $v_i = 0$ for all $v \in \Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$.*

To detect blocked reactions, the following two LPs can be used:

A reaction $i \in \mathfrak{R}$ is blocked if and only if

$$\max_v \{\pm v_i \mid Sv = 0, v_{\text{Irr}} \geq 0\} = 0. \tag{2.5}$$

2.3 Mathematical basics

In a pre-processing step, blocked reactions can be removed from the network. Thus, we assume from now on that the network contains only unblocked reactions. For two unblocked reactions $i, j \in \mathfrak{R}$ we can define three different coupling relations [Burgard et al., 2004; Larhlimi et al., 2012b].

Definition 2.37 (Coupling relations). *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and its flux cone $\Gamma_{\mathcal{N}}$.*

$i \xrightarrow{=0} j$: j is directionally coupled to i if and only if $v_i = 0$ implies $v_j = 0$ for all $v \in \Gamma_{\mathcal{N}}$.

$i \xleftrightarrow{=0} j$: j is partially coupled to i if and only if $v_i = 0 \Leftrightarrow v_j = 0$ for all $v \in \Gamma_{\mathcal{N}}$.

$i \sim j$: j is fully coupled to i if and only if there exists $\lambda \in \mathbb{R} \setminus \{0\}$ such that $v_i = \lambda v_j$ for all $v \in \Gamma_{\mathcal{N}}$.

Remark 2.38. *If two reactions are fully coupled, then they are also partially coupled (but not necessarily the converse).*

The coupling relation $\xrightarrow{=0}$ is reflexive and transitive, and thus defines a *pre-order* [Schröder, 2003]. The relation $\xleftrightarrow{=0}$ is also symmetric and therefore an *equivalence relation*. This means that the set of reactions \mathfrak{R} can be partitioned into equivalence classes $[i] = [i]_{\xleftrightarrow{=0}} = \{j \in \mathfrak{R} \mid i \xleftrightarrow{=0} j\}$. We have $\mathfrak{R} = \bigcup_{[i] \in \overline{\mathfrak{R}}} [i]$, where $\overline{\mathfrak{R}} = \mathfrak{R} / \xleftrightarrow{=0}$ denotes the set of all equivalence classes.

An equivalence class can be represented by any of its elements. We say that i is a *representative* of $[i]$ or that $[i]$ is the *coupling class* of i . Note that $[i] = [j]$ if and only if $i \xleftrightarrow{=0} j$. Coupling classes are similar to the *enzyme subsets* introduced by [Pfeiffer et al., 1999]. Enzyme subsets are groups of reactions that are fully coupled. Here, we relax this condition and also consider reactions that are only partially coupled.

Definition 2.39 (Reaction coupling order $\prec_{\xrightarrow{=0}}$). *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. The partial ordering on the coupling classes $\prec_{\xrightarrow{=0}} \subseteq \overline{\mathfrak{R}} \times \overline{\mathfrak{R}}$ defined by*

$$[i] \prec_{\xrightarrow{=0}} [j] :\Leftrightarrow i \xrightarrow{=0} j$$

is called the reaction coupling order (RCorder) induced by the coupling relation $\xrightarrow{=0}$.

Note that this construction works, because $\xrightarrow{=0}$ is a preorder [Schröder, 2003]. Figure 2.5 shows an example network with its corresponding RCorder.

2. Basics

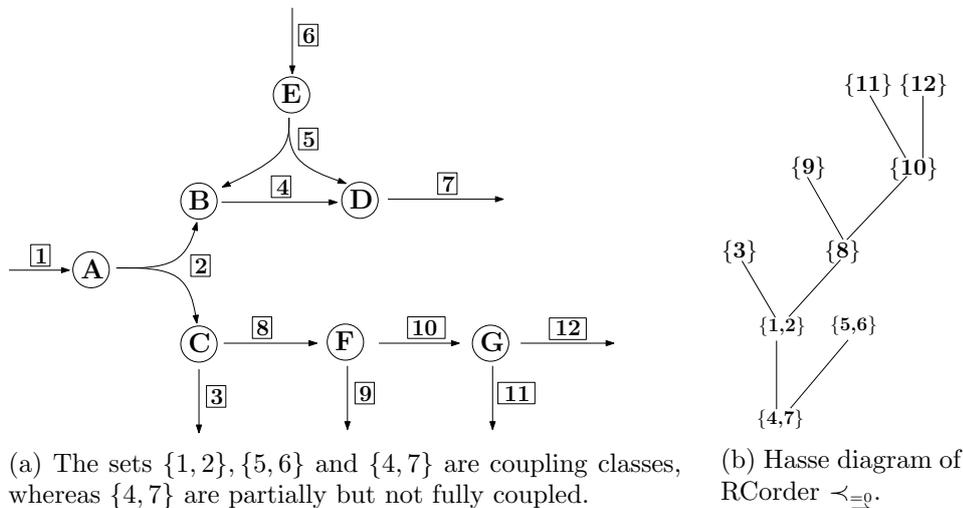


Figure 2.5: Reaction coupling $\xrightarrow{\Rightarrow 0}$ is a preorder of the reactions. It induces a partial order $\prec_{\Rightarrow 0}$. A knock-out of reaction 1 implies inactivity in reaction 3, 8 and others, but not in reaction 5. Thus, $1 \xrightarrow{\Rightarrow 0} 3$ and $1 \xrightarrow{\Rightarrow 0} 8$ but $\neg(1 \xrightarrow{\Rightarrow 0} 5)$. This results in $1 \prec_{\Rightarrow 0} 3$ and $1 \prec_{\Rightarrow 0} 8$ for the ROrder $\prec_{\Rightarrow 0}$, but 1 and 5 are incomparable. The Hasse diagram visualises this by having downward directed paths starting in the greater and ending in the smaller of each pair of coupled reaction representatives, in our example from 3 to 1 and from 8 to 1, but not from 5 to 1. A knock-out of reaction 4 implies inactivity in all reactions.

Example 2.40 (Coupling relations). *The network in Figure 2.5a contains two pairs of fully coupled reactions, namely $\{1, 2\}$ and $\{5, 6\}$. The reactions $\{4, 7\}$ are partially coupled. These three sets can be represented by 1, 5 and 4. Figure 2.5b shows the Hasse diagram of $\prec_{\Rightarrow 0}$, where the nodes represent reactions. If a reaction has zero flux, then exactly those reactions that are connected by a path going strictly upwards have zero flux, too. For example, reaction 1 is coupled to 3, i.e., $1 \xrightarrow{\Rightarrow 0} 3$, but 1 is uncoupled to reaction 5. More on Hasse diagrams can be found in [Schröder, 2003].*

FCA can be formulated using the following LPs:

Two unblocked reactions $i, j \in \mathfrak{R}$ are directionally coupled, if and only if

$$\max_v \{\pm v_j \mid Sv = 0, v_{\text{irr}} \geq 0, v_i = 0\} = 0. \quad (2.6)$$

To detect if i and j are partially coupled one checks if i is directionally coupled to j and vice versa. Detecting fully coupled reactions is slightly more complicated and described in [Larhlmi et al., 2012b].

We make use of flux coupling information to further improve efficiency of some algorithms presented in this work, see Chapters 3 and 4.

2.3.3 Mixed integer linear programming

In some applications for metabolic networks binary variables are used to indicate if a reaction carries flux or not. Thus, for every reaction $i \in \mathfrak{R}$, a binary variable $a_i \in \mathbb{B}$ is introduced and constraints are added which translate to $a_i = 0 \Leftrightarrow v_i = 0$. There exist algorithms which will find an optimal solution for LPs within polynomial time. If (some of) the variables are integers (as the binary variables a_i) we talk about a *mixed integer linear program* (MILP). Although at the first sight one might think that these extra requirements do not affect the problem that much, it has been shown that solving MILPs is NP-hard, see [Garey and Johnson, 2002].

2.3.3.1 Application for MILP: Minimum network

One application of an MILP for a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ is to find a minimum set of reactions which is capable of maintaining predefined growth rates. The corresponding MILP was introduced in [Burgard et al., 2001] and refined in [Jonnalagadda and Srinivasan, 2014]. The MILP for minimising the number of reactions needed to have a maximum biomass yield is as follows:

$$\min_{v, a, \bar{a}} \sum_{i \in \mathfrak{R}} a_i + \sum_{k \in \text{Rev}} \bar{a}_k \quad (2.7)$$

$$\text{s.t. } Sv = 0 \quad (2.8)$$

$$v_{\text{bio}} = v_{\text{bio}}^* \quad (2.9)$$

$$\delta a_i \leq v_i \leq M a_i \quad \forall i \in \text{Irr} \quad (2.10)$$

$$\delta a_i - M \bar{a}_i \leq v_i \leq M a_i - \delta \bar{a}_i \quad \forall i \in \text{Rev} \quad (2.11)$$

$$l \leq v \leq u \quad (2.12)$$

$$v \in \mathbb{R}^{\mathfrak{R}}, a \in \mathbb{B}^{\mathfrak{R}}, \bar{a} \in \mathbb{B}^{\text{Rev}}$$

The objective (2.7) is to minimise the number of active reactions, i.e., to minimise over the binary variables a_i and \bar{a}_k indicating if a reaction carries flux or not. Note that we have for each reversible reaction $i \in \text{Rev}$ two binary variables a_i and \bar{a}_i . Our system is in steady-state, which is ensured by constraint (2.8) and the biomass reaction has the maximum biomass yield (2.9). The relationship $a_i = 0$ if and only if $v_i = 0$ is ensured by

2. Basics

the constraints (2.10) for the irreversible reactions, and for the reversible reactions by the constraints (2.11). They are called *big M* constraints, where $M \gg 0$ is a sufficiently large constant, e.g. some upper bound on the flux rates. With $\delta > 0$ we denote a threshold indicating above which flux rate a reaction is considered to be active. In practice δ will be chosen between 10^{-6} and 10^{-4} to indicate if a reaction is active [Erdrich et al., 2015]. Finally, (2.12) constrains the flux rates due to their bounds.

2.3.3.2 Solving MILPs

The most common approach to solve MILPs [Land and Doig, 1960] is the *branch and bound* method [Schrijver, 1998]. The algorithm can be illustrated as a search tree. For example if we consider an MILP where the integer variables $a \in \mathbb{B}^n$ are binary, at step i the variable a_i is chosen. We branch now by considering two different cases: $a_i = 0$ and $a_i = 1$, see Figure 2.6 for an illustration. If this is done for every variable the search tree presents all possible cases for the values of a . Solving every case results in a combinatorial explosion, where in each branching step the resulting sub-problems are solved. Therefore, in each branching step the variables a are considered as being continuous, i.e., the *LP-Relaxation* is solved. After computing both objective values (*bounds*) of the two sub problems (for $a_i = 0$ and for $a_i = 1$) the branch is chosen which delivers the better bound. Here the next branching occurs, i.e., for variable a_{i+1} the two different bounds of the corresponding LP-relaxations are compared. If the better of the two bounds is worse than a bound computed in a different part of the tree, the search continues in the more promising branch. The algorithm stops if a solution is found where all variables which have to be integer are indeed integer variables and no better solution can be expected in any other branch of the search tree. Doing so, the size of the search space is decreased since most of the time only a part of the full search tree has to be investigated. For more details we refer to [Schrijver, 1998].

There exist a lot of different solvers which can be used to solve LPs or MILPs. Some of them are commercial, such as **Gurobi** [Gurobi Optimization, 2016] and **CPLEX** [IBM Knowledge Center, 2010], some are non-commercial, e.g. **SCIP** (solving MILPs) including **SoPlex** (solving LPs) of the Zuse Institute in Berlin [Maher et al., 2017]. **SCIP** can solve an MILP exactly over the rational numbers, and the underlying LP solver **SoPlex** performs iterative refinement [Gleixner et al., 2012, 2016].

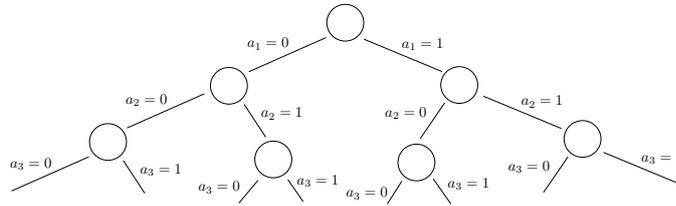


Figure 2.6: Illustration of the branch and bound method. In each node the current LP-Relaxation of the problem is solved, i.e., all constraints of the MILP are kept, but the integer variables are considered as continuous variables. In the first step this delivers a first bound for the given MILP. Then we branch, which is indicated by the arcs. In the first branch, the first binary variable a_1 is set to 0, resp. to 1, thus two new LP-Relaxations are solved now and the direction (arc) with the better bound is considered. If all bounds in the current branch are worse than bounds computed in a different part of the search tree, the search continues in the branch where the best bound was computed so far. If a solution is computed, where all variables which should be integer are indeed integers and the bound cannot be improved, the method stops and returns the solution. In the worst case all possible values of a are examined.

2.3.4 Numerical instability

In the following we discuss the reasons for numerical instability, for more information we refer to [Higham, 2002; Klotz, 2014]. The methods we introduce in this thesis are implemented in MATLAB. Therefore, the examples and functions used in the following are related to this language.

Numerical instabilities can be caused by the input data, the algorithm used to solve the problem, or rounding errors. The problems we are dealing with here are MILPs and therefore we need to solve systems of linear inequalities, e.g., $Ax \geq b$ with $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, and $x \in \mathbb{R}^n$, where some or all variables have to be integers. Thus the input values are given by the matrix A and the vector b . To solve the given MILPs we use MILP solvers like CPLEX [IBM Knowledge Center, 2010] or Gurobi [Gurobi Optimization, 2016]. Rounding errors occur all the time. They become relevant or have non-negligible impact especially for the *big M* formulations.

2.3.4.1 Condition

We start with discussing the effect of the input values on the accuracy of the solution. The condition of a problem, e.g. of a given LP, is the sensitivity of the solution w.r.t. the input.

2. Basics

Lets consider as an example an LP

$$\begin{aligned} \min_x \quad & c^\top x \\ \text{s.t.} \quad & Ax \leq b \\ & x \in \mathbb{R}^n, \end{aligned}$$

with $c \in \mathbb{R}^n$, $A \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$.

To solve the LP, the simplex algorithm searches in each step a subset B of rows of the matrix A such that $A_{B,*}$ is regular. We call such a sub matrix a *basis matrix*. The solution in this step is then given by

$$x_B = A_{B,*}^{-1} b.$$

The condition for such an LP characterises the effect of small perturbations in b on the solution x_B and is given by the condition number of the basis matrix $A_{B,*}$ in the last step of the simplex:

Definition 2.41 (Condition number). *The condition number κ of a regular matrix $D \in \mathbb{R}^{m \times m}$ is defined by $\kappa(D) = \|D\| \|D^{-1}\|$.*

If we consider an LP as above, $\kappa(A_{B,*})$ is the rate at which the solution x_B will change w.r.t. a change in b . For example if the condition number $\kappa(A_{B,*}) = 10^k$ then one may lose up to k digits of accuracy.

A problem with a low condition number is called *well-conditioned* and a problem with a high condition number is called *ill-conditioned*. If the condition number is infinite we say the problem is *ill-posed* which means that in general no algorithm can be expected to find a reliable solution. A problem is well-conditioned if small changes in the input lead to small changes in the results whereas the problem is ill-conditioned if small changes in the input may lead to large changes in the results.

There are in general several algorithms to solve a given problem. How much an algorithm is reliable is classified by the *stability*. The stability indicates the changes in the output value of the algorithm due to small changes in the input.

The condition of a problem is independent from the stability of algorithms for tackling it. But the choice of which algorithm to use is influenced by the condition of the problem. One should always use an algorithm with the highest possible stability given the accuracy requirements one needs. The condition of MILPs depends on the analytic properties of the constraint matrix.

The maximal condition numbers of regular submatrices of stoichiometric matrices for some metabolic networks used in this thesis, can be found in the Tables 2.1 and 2.2. We computed them using `CPLEX` and the integrated command `CPLEX.Solution.quality.kappamax.value`. According to [IBM, 2011; Klotz, 2014] a given problem is considered as well-conditioned if the condition number κ is smaller than 10^7 , suspicious if $10^7 \leq \kappa \leq 10^{10}$ and ill-conditioned if $10^{10} < \kappa \leq 10^{14}$. For $\kappa > 10^{14}$ the problem is considered ill-posed.

Using big M constraints increases in general the condition number of the problem.

2.3.4.2 Rounding errors

We recall that $M \gg 0$ is part of our constraint matrix, see Subsection 2.3.3.1. Since we are dealing with MILPs, we use solvers like `CPLEX` [IBM Knowledge Center, 2010] to find the optimal solution. The methods used in the solvers are based on well known MILP solving strategies such as *branch and bound*. Inside of these methods several LPs have to be solved. This is usually done by using the simplex method.

If we are using big M formulations our constraint matrix includes M . If the simplex-algorithm searches for a basis matrix $A_{B,*}$ it can happen that $A_{B,*}$ is considered as being regular although it is not. Or there is a loss of precision while computing $A_{B,*}^{-1}$. We consider an example how rounding errors can cause difficulties using `MATLAB`.

Example 2.42 (Rounding errors). *Assume that*

$$A_{B,*} = \begin{pmatrix} 1 & M & 1 \\ 0 & 0 & M \\ \frac{1}{M} & 1 & 0 \end{pmatrix}.$$

One can easily see that $A_{B,}$ is singular, since $\det(A_{B,*}) = 0$. In `MATLAB`, for $M = 10^{12}$, we obtain that $\det(A_{B,*}) = 0$, whereas for $M = 10^{12} + 1e^{-4}$ it holds that $\det(A_{B,*}) = -0.1110e^{-3}$. This means that small changes in the input influences the output enormously.*

To overcome this problem different approaches exist. One is to use values for M as small as possible and, if they exist, different values for M for different variables. For example in our case, big M values are introduced to bound reaction rates from above. Thus we can use the upper bounds of the reaction rates given by the description of the underlying metabolic network. But we

2. Basics

can still run into heavy numerical instabilities. The other approach is a pre-conditioning step of the matrix: Instead of solving $Ax = b$, one solves $(C_1AC_2)y = C_1b$, $x := C_2y$, where the matrices C_1, C_2 are chosen, such that $\kappa(C_1AC_2) \ll \kappa(A)$ [Golub and Van Loan, 2012].

We use big M formulations in the inequalities (2.10) and (2.11) to connect the flux rates with the binary variables. We give the condition numbers of the corresponding matrices in the Tables 2.1 and 2.2. As one can see, introducing big M constraints increases the condition number of the constraint matrix and therefore makes it more likely that numerical issues occur.

If numerical issues occur we can make use of *indicator variables* which are integrated in CPLEX [IBM Knowledge Center, 2010]. This improves the numerical stability significantly but increases the running time. CPLEX has two strategies to deal with indicator variables. The first one is to reformulate the constraints into big M constraints, but with values for M as small as possible. The second one is to branch on the indicator variables and to update bounds (and M) in the newly appearing branches. CPLEX chooses one or a mixture of both strategies, depending on the possible values for M [IBM Knowledge Center, 2010].

All further information needed for MILPs used in this thesis will be given in the corresponding sections.

2.3.5 Oriented Matroids

Matroids and oriented matroids are applied in several fields, e.g. for (directed) graphs [Finschi and Fukuda, 2002], vector subspaces or linear programming [Björner, 1999]. Here we use this concept to gain insights into metabolic networks. In the case of metabolic networks we always have *representable* oriented matroids (matroids which can be represented by a matrix) but we use the term oriented matroids.

As in most literature we use underlined notations for non-oriented matroids, i.e., we denote a matroid by $\underline{\mathcal{M}}$ and an oriented matroid by \mathcal{M} . A matroid $\underline{\mathcal{M}}$ can be defined in several ways. Here we use a set of elements U and a set $\underline{\mathcal{C}}$ of circuits. We start with three examples for non-oriented matroids before we give definitions.

Example 2.43 (Vector matroid). *The first example for a matroid is the vector matroid: Suppose U is a finite set of vectors of a vector space V . We say that the corresponding matroid $\underline{\mathcal{M}}$ is represented by U . A cycle is a linearly dependent subset of U and a circuit $\underline{\mathcal{C}}$ is a minimal cycle: If any vector is deleted from a circuit $\underline{\mathcal{C}} \in \underline{\mathcal{C}}$ the remaining vectors are not linearly dependent anymore.*

2.3 Mathematical basics

| model id | condition number | condition number big M |
|----------------|------------------|--------------------------|
| e_coli_core | 1.00E+02 | 2.06E+08 |
| iAB_RBC_283 | 1.33E+02 | 3.21E+08 |
| iIT341 | 7.65E+02 | 1.08E+11 |
| iLJ478 | 2.17E+03 | 2.73E+08 |
| iAF692 | 5.64E+03 | 1.62E+14 |
| iSB619 | 3.13E+03 | 4.88E+10 |
| iNF517 | 4.34E+03 | 1.51E+13 |
| iHN637 | 2.73E+03 | 3.03E+07 |
| iJB785 | 3.12E+07 | 1.01E+07 |
| iJN678 | 8.10E+03 | 5.29E+08 |
| iAT_PLT_636 | 2.24E+03 | 4.46E+06 |
| iNJ661 | 2.34E+04 | 5.13E+12 |
| iJN746 | 2.14E+03 | 6.04E+07 |
| iJR904 | 7.31E+03 | 4.18E+17 |
| iYO844 | 7.61E+03 | 1.31E+13 |
| iND750 | 1.79E+04 | 2.10E+09 |
| iAF987 | 1.90E+04 | 1.02E+13 |
| iMM904 | 1.72E+06 | 9.86E+15 |
| iPC815 | 8.55E+04 | 1.00E+17 |
| iRC1080 | 3.30E+05 | 8.13E+19 |
| iYL1228 | 1.42E+04 | 2.07E+14 |
| iAF1260 | 3.25E+04 | 8.99E+15 |
| iAF1260b | 3.79E+04 | 1.25E+17 |
| iSDY_1059 | 7.53E+05 | 5.89E+13 |
| STM_v1_0 | 2.55E+06 | 1.86E+13 |
| iJO1366 | 1.37E+04 | 6.47E+15 |
| iSbBS512_1146 | 1.58E+08 | 7.89E+12 |
| iSBO_1134 | 6.50E+05 | 5.07E+13 |
| iS_1188 | 4.94E+05 | 2.02E+13 |
| iSFV_1184 | 2.00E+08 | 2.88E+14 |
| iSF_1195 | 5.19E+05 | 2.18E+12 |
| iSFxv_1172 | 3.50E+05 | 3.53E+15 |
| iSSON_1240 | 1.26E+08 | 9.29E+12 |
| iECH74115_1262 | 1.05E+06 | 1.40E+14 |
| iE2348C_1286 | 1.03E+06 | 6.46E+13 |
| iG2583_1286 | 1.12E+06 | 6.09E+16 |
| iECED1_1282 | 5.19E+08 | 4.81E+15 |
| iECSP_1301 | 1.29E+09 | 9.49E+12 |
| iML1515 | 2.72E+04 | 1.85E+16 |
| iEC042_1314 | 1.25E+09 | 1.10E+13 |
| iECNA114_1301 | 1.22E+06 | 5.24E+16 |
| iECs_1301 | 1.24E+06 | 5.19E+17 |

Table 2.1: The condition number of regular submatrices from stoichiometric matrices of the first 42 models from the BiGG Models database, [King et al., 2016], computed using the command `CPLEX.Solution.quality.kappamax.value`. **model id:** the id of the model as it can be found in the BiGG Models database. **condition number:** the condition of a regular submatrix from the stoichiometric matrix. **condition number big M :** the condition of a regular submatrix from the matrix including the stoichiometric matrix and big M constraints connecting binary variables and continuous variables, as in the constraints (2.10) and (2.11). The value(s) for M are given by the upper bounds on the flux rates.

2. Basics

| model id | condition number | condition number big M |
|-------------------|------------------|--------------------------|
| iECIAI39_1322 | 2.02E+08 | 3.38E+15 |
| iZ_1308 | 1.20E+09 | 2.39E+16 |
| iUTI89_1310 | 1.13E+09 | 6.08E+16 |
| ic_1306 | 1.07E+06 | 4.39E+12 |
| iLF82_1304 | 1.32E+08 | 2.27E+14 |
| iECOK1_1307 | 1.14E+06 | 1.03E+14 |
| iECS88_1305 | 1.70E+08 | 1.71E+13 |
| iECABU_c1320 | 6.78E+07 | 1.75E+16 |
| iAPECO1_1312 | 1.29E+06 | 4.52E+14 |
| iNRG857_1313 | 1.42E+08 | 3.70E+14 |
| iUMN146_1321 | 1.12E+06 | 2.55E+17 |
| iECP_1309 | 1.23E+06 | 6.51E+13 |
| iECUMN_1333 | 1.18E+06 | 7.91E+13 |
| iB21_1397 | 1.41E+08 | 2.00E+14 |
| iBWG_1329 | 2.27E+04 | 3.28E+13 |
| iECD_1391 | 1.28E+06 | 1.11E+14 |
| iECDH10B_1368 | 3.66E+04 | 1.49E+14 |
| iECSF_1327 | 3.16E+04 | 8.67E+12 |
| iEcSMS35_1347 | 5.52E+08 | 1.68E+17 |
| iECB_1328 | 1.12E+06 | 7.54E+15 |
| iECBD_1354 | 1.13E+06 | 4.10E+14 |
| iEcDH1_1363 | 9.65E+05 | 1.71E+15 |
| iEcHS_1320 | 1.09E+06 | 1.02E+14 |
| iECDH1ME8569_1439 | 1.08E+09 | 1.02E+15 |
| iEC55989_1330 | 5.68E+08 | 1.14E+14 |
| iETEC_1333 | 1.21E+09 | 1.49E+16 |
| iECO103_1326 | 5.92E+08 | 1.30E+15 |
| iY75_1357 | 7.10E+07 | 3.01E+12 |
| iECO111_1330 | 1.23E+09 | 1.10E+14 |
| iEcE24377_1341 | 3.07E+05 | 3.56E+14 |
| iECIAI1_1343 | 1.81E+04 | 9.80E+12 |
| iEcolC_1368 | 6.49E+08 | 8.36E+14 |
| iECSE_1348 | 1.23E+09 | 5.42E+14 |
| iUMNK88_1353 | 1.27E+09 | 7.49E+16 |
| iEKO11_1354 | 5.66E+08 | 4.54E+13 |
| iECO26_1355 | 1.12E+09 | 5.26E+14 |
| iECW_1372 | 1.06E+06 | 5.10E+12 |
| iWFL_1372 | 1.27E+06 | 4.24E+17 |
| iMM1415 | 1.76E+05 | 7.70E+14 |
| RECON1 | 3.15E+06 | 1.73E+16 |
| iLB1027_lipid | 4.50E+06 | 2.18E+04 |
| iCHOv1 | 9.72E+04 | 3.07E+13 |

Table 2.2: The condition number of regular submatrices from stoichiometric matrices of the first 42 models from the **BiGG Models** database, [King et al., 2016], computed using the command `CPLEX.Solution.quality.kappamax.value`. **model id**: the id of the model as it can be found in the **BiGG Models** database. **condition number**: the condition of a regular submatrix from the stoichiometric matrix. **condition number big M** : the condition of a regular submatrix from the matrix including the stoichiometric matrix and big M constraints connecting binary variables and continuous variables, as in the constraints (2.10) and (2.11). The value(s) for M are given by the upper bounds on the flux rates.

Example 2.44 (Column matroid). *The second example is the column matroid. We consider a matrix $A \in \mathbb{R}^{m \times n}$, where the entries of A can be in general from any field. The corresponding matroid $\underline{\mathcal{M}}$ is represented by the columns of A . The cycles are the sets of linearly dependent columns of A , thus subsets of columns which are linearly dependent vectors. The circuits are the minimal sets of linearly dependent columns.*

Example 2.45 (Graphic matroid). *The last example is the graphic matroid. Suppose we have a given undirected graph $G = (V, U)$, where V is the set of vertices and U is the set of edges. Cycles in the matroid are cycles in the graph, where the circuits are the cycles of G which consist of a minimal set of edges.*

In the case of oriented matroids the elements U have an orientation. Therefore, for a graphic matroid the underlying graph is not undirected anymore but directed.

In the case of a metabolic network where *all reactions are reversible*, the matrix representing the oriented matroid is the stoichiometric matrix $S \in \mathbb{R}^{m \times n}$. The set of elements U are the columns of S , i.e., the reactions of the metabolic network. Oriented cycles are feasible flux vectors and oriented circuits correspond to elementary flux modes. Unfortunately, we cannot consider a metabolic network as a graphic matroid, since metabolic networks are usually hypergraphs.

A nice introduction to oriented matroids is the book [Björner, 1999], which is used as a basis for the following section. Based on this book, we now give a proper definition of oriented matroids.

2.3.5.1 Basics for oriented matroids

Let U be a set. A *signed subset* of U is a pair $X = (X^+, X^-)$ with a set of positive elements $X^+ \subseteq U$ and a set of negative elements $X^- \subseteq U$ such that $X^+ \cap X^- = \emptyset$. The *support* of X is the set $\text{supp}(X) = X^+ \cup X^-$. An *oriented matroid* is a pair $\mathcal{M} = (U, \mathcal{C})$ where \mathcal{C} is a family of signed subsets of U , called the *oriented circuits* of \mathcal{M} , which satisfy the circuit axioms:

Definition 2.46 (Oriented Matroid). *A tuple $\mathcal{M} = (U, \mathcal{C})$ with a set of elements U and oriented circuits $\mathcal{C} \in \{-, 0, +\}^U$ is called an oriented matroid if the following axioms are satisfied:*

1. $0 \notin \mathcal{C}$.

2. Basics

2. $X \in \mathcal{C} \Rightarrow -X \in \mathcal{C}$
3. For all $X, Y \in \mathcal{C}$ we have: $\text{supp}(X) \subseteq \text{supp}(Y) \Rightarrow X = Y$ or $X = -Y$.
4. $X, Y \in \mathcal{C}$, $X \neq -Y$ and $u \in (X^+ \cap Y^-) \Rightarrow$ there exists a $Z \in \mathcal{C}$ with $Z^+ \subseteq (X^+ \cup Y^+) \setminus \{u\}$ and $Z^- \subseteq (X^- \cup Y^-) \setminus \{u\}$.

Oriented circuits are a certain type of *oriented cycles*. To define oriented cycles we need the notion of a dependent set. The idea of a dependent set comes from linear algebra where linearly dependent sets are considered. This concept is generalised in oriented matroids. For example in a graphic matroid a dependent subset forms a directed cycle in the underlying graph.

Since we consider only oriented matroid represented by a matrix, we give the definition of an oriented cycle related to those matroids:

Definition 2.47 (Oriented cycle). *Let $\mathcal{M} = (U, \mathcal{C})$ be an oriented matroid, where U is the set of columns of a given matrix A . A cycle X of \mathcal{M} is a dependent subset E of U , thus a set of linearly dependent columns of A .*

Every oriented matroid \mathcal{M} has an *underlying* (non-oriented) matroid $\underline{\mathcal{M}}$. In the underlying matroid the cycles do not have an orientation. Thus the cycles are not a result of the sign function but only the support of it.

One important fact about oriented circuits is that every oriented cycle of the oriented matroid is a composition of these oriented circuits (Proposition 3.7.2 in [Björner, 1999]):

Definition 2.48 (Composition). *The composition $X \circ Y$ of two signed sets X, Y is the signed set defined by $(X \circ Y)^+ = X^+ \cup (Y^+ \setminus X^-)$ and $(X \circ Y)^- = X^- \cup (Y^- \setminus X^+)$.*

In this work, we consider oriented matroids which are based on metabolic networks. We give some short notions and definitions about the connection between oriented matroids and metabolic networks based on [Reimers, 2014].

Since the elements will have an orientation, the first definition we need is the *sign function* together with the *sign vector*:

Definition 2.49 (Sign function). *The sign function of a number $\lambda \in \mathbb{R}$ is defined as follows:*

$$\sigma(\lambda) = \begin{cases} -1 & \text{if } \lambda < 0 \\ 0 & \text{if } \lambda = 0 \\ 1 & \text{if } \lambda > 0 \end{cases}$$

Definition 2.50 (Sign vector). *The sign vector $\sigma(x) \in \{-, 0, +\}^n$ of a vector $x \in \mathbb{R}^n$ is the result of applying the sign function component-wise to x .*

We usually denote the sign vector of a vector x by using the corresponding capital letter, thus $X = \sigma(x)$.

Definition 2.51. *Let X be the sign vector of $x \in \mathbb{R}^n$. We denote with X^+ (resp. X^-) the indices of all non zero elements in the vector X where the corresponding entry is positive (resp. negative), thus: $X^+ := \{i \mid \sigma(x_i) = +\} = \{i \mid x_i > 0\}$ (resp. $X^- := \{i \mid \sigma(x_i) = -\} = \{i \mid x_i < 0\}$).*

We use the notion of a sign vector σ for a set of vectors too. For example $\sigma(A)$ for a matrix $A \in \mathbb{R}^{m \times n}$ means that σ is applied to every column of A . In combination with the minimal operator Min we denote non-empty signed sets with inclusion-minimal support. For example $\text{Min}(\sigma(A))$ is the set of non-empty signed columns of minimal support of A .

A metabolic network where all reactions are reversible can be considered as an oriented matroid:

Proposition 2.52 (Proposition 2.5.1 in [Reimers, 2014]). *Let $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$ denote a stoichiometric matrix. Then the tuple $(\mathfrak{R}, \mathcal{C})$ defines an oriented matroid, where $\mathcal{C} = \text{Min}(\mathcal{V})$ with $\mathcal{V} = \{\sigma(v) \mid Sv = 0, v \neq 0\}$.*

We can now define the flux mode matroid (Definition 2.5.2 in [Reimers, 2014]):

Definition 2.53 (Flux Mode Matroid). *Let $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$ be a stoichiometric matrix, where \mathfrak{M} is the set of (internal) metabolites and \mathfrak{R} the set of reactions. Then $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ denotes the oriented matroid obtained from S . This oriented matroid is called the flux mode matroid.*

In general there exist irreversible reactions in a metabolic network. Therefore not all oriented circuits of the flux mode matroid correspond to EFMs, resp. to feasible flux vectors. We say that a circuit X corresponds to a sign vector of an EFM if and only if all irreversible reactions have non-negative signs. Thus:

Proposition 2.54. *Consider a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. The sign vectors of the EFMs of \mathcal{N} are a subset of the circuits \mathcal{C} of the flux mode matroid \mathcal{M}_S :*

$$\sigma(\text{EFMs}) = \{X \in \mathcal{C} \mid X_i \in \{0, +\} \text{ for all } i \in \text{Irr}\}.$$

2. Basics

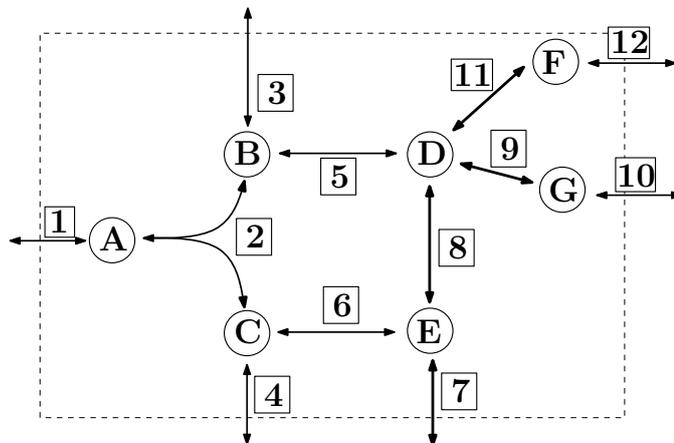


Figure 2.7: Illustration of the same metabolic network as in Figure 1.1, except that all reactions are reversible. The number of EFMs, resp. oriented circuits, is 24. All EFMs of the network in Figure 1.1, where the reactions $\{1, 2, 6, 7, 8\}$ are irreversible are contained in these 24 EFMs.

Proof. All signed vectors of feasible flux vectors of \mathcal{N} are oriented cycles of the flux mode matroid \mathcal{M}_S . The EFMs are defined as minimal sets of active reactions that can operate together in steady-state. Therefore they correspond to oriented circuits of the flux mode matroid, given that the elements corresponding to irreversible reactions are 0 or +. \square

As already mentioned the flux mode matroid contains oriented cycles which do not correspond to feasible flux vectors. But we can still use the idea of oriented matroids to get a different view of our network.

2.3.5.2 Fundamental Circuits

In the following we consider a specific subset of circuits of a matroid, the *fundamental circuits*. They can be computed in incremental polynomial time [Boros et al., 2003] and all other circuits and cycles of the underlying matroid can be generated with them. We make use of the idea of fundamental circuits in Chapter 5 which serves as a basis for Chapter 6 where we introduce a method to compute a minimum set of EFMs needed to represent all possible flux vectors of a metabolic network \mathcal{N} . To illustrate the following concepts we utilise the metabolic network in Figure 2.7, which is the same as in Figure 1.1, except that all reactions are reversible.

Definition 2.55 (Matroid bases). *Let U be a set of elements and $\underline{\mathcal{M}}$ be a matroid on U . Bases of $\underline{\mathcal{M}}$ are the maximal subsets of U which contain no circuit.*

Remark 2.56. *There exists in general more than one basis for a matroid.*

Example 2.57 (Basis of a vector matroid). *Consider for example a vector matroid, where U is a finite subset of a vector space V . Then a basis of $\underline{\mathcal{M}}$ is a basis of the vector space V .*

The Steinitz exchange theorem, see Theorem 7 on page 115 in [Kung, 1986], for bases of vector spaces demonstrates that there are several bases, that all of them have the same number of elements, and that no basis can be a proper subset of another basis. In the case of matroids we refer to this attribute as the *exchange property*.

Example 2.58 (Basis of a column matroid). *For a column matroid, where U is a set of columns of a matrix $A \in \mathbb{R}^{m \times n}$, a basis is a maximal set of linearly independent columns. If the rank of the underlying matrix is r then each basis has a cardinality of r . Thus, for the network in Figure 2.7 a basis would be a maximal subset of reactions such that the corresponding set of columns of the stoichiometric matrix S is a linearly independent set of vectors.*

Example 2.59 (Basis of a flux mode matroid). *Consider the network in Figure 2.7. The set of reactions $\mathcal{B} = \{1, 2, 3, 5, 6, 9, 11\}$ is a basis for the corresponding flux mode matroid, since the matrix $S_{*,\mathcal{B}}$ and the matrix S both have rank 7 and $|\mathcal{B}| = 7$. A different basis would be the set $\mathcal{B} = \{2, 5, 6, 8, 10, 11, 12\}$. In total there exist 268 different bases for the network in Figure 2.7.*

We discuss the number of different bases in Section 5.3 and give an upper bound.

Definition 2.60 (Oriented fundamental circuit (FC), [Björner, 1999] p. 115). *If \mathcal{M} is an oriented matroid on U , \mathcal{B} is a basis of $\underline{\mathcal{M}}$, and $u \in U \setminus \mathcal{B}$ then there exists a unique circuit $\underline{C}(u, \mathcal{B})$ of $\underline{\mathcal{M}}$ contained in $\mathcal{B} \cup u$. Furthermore $\underline{C}(u, \mathcal{B})$ supports a unique oriented circuit $C(u, \mathcal{B})$ up to sign reversal ($-C(u, \mathcal{B})$), thus there exists an oriented circuit $C(u, \mathcal{B})$ which has the same support as $\underline{C}(u, \mathcal{B})$. $C(u, \mathcal{B})$ denotes the basic or oriented fundamental circuit (FC) of u with respect to \mathcal{B} .*

Example 2.61 (Fundamental circuits of a flux mode matroid). *Consider again the network in Figure 2.7. As mentioned in Example 2.59 the set of reactions $\mathcal{B} = \{1, 2, 3, 5, 6, 9, 11\}$ forms a basis for the underlying flux mode matroid. It is not possible to create a feasible flux using only the reactions in \mathcal{B} . But for each reaction $i \in \mathfrak{R} \setminus \mathcal{B}$ we can find an EFM:*

$$i = 4 : \underline{C}(i, \mathcal{B}) = \{1, 2, 3, 4\} \text{ with } C = (+, +, +, +, 0, 0, 0, 0, 0, 0, 0).$$

2. Basics

$i = 7$: $\underline{C}(i, B) = \{1, 2, 3, 6, 7\}$ with $C = (+, +, +, 0, 0, +, +, 0, 0, 0, 0)$.

$i = 8$: $\underline{C}(i, B) = \{1, 2, 3, 5, 6, 8\}$ with $C = (+, +, +, 0, -, +, 0, +, 0, 0, 0)$.

$i = 10$: $\underline{C}(i, B) = \{3, 5, 9, 10\}$ with $C = (0, 0, -, 0, +, 0, 0, 0, +, +, 0, 0)$.

$i = 12$: $\underline{C}(i, B) = \{3, 5, 11, 12\}$ with $C = (0, 0, -, 0, +, 0, 0, 0, 0, 0, +, +)$.

Given a matroid \underline{M} on a set of elements U , computing a basis of \underline{M} and enumerating the corresponding FCs for the computed basis can be done in incremental polynomial time [Boros et al., 2003].

2.4 Handling metabolic networks

To use and exchange metabolic networks, in general the *Systems Biology Markup Language* (SBML) [Hucka et al., 2003] is used. Here, metabolic networks are represented in an XML machine-readable format. Most of the metabolic network reconstructions nowadays, and all models used in this thesis, are published in SBML-format. To read these models we used the COBRA-Toolbox, version 2.0.5 [Schellenberger et al., 2007].

For most of the models, there exist blocked reactions (see Definition 2.36) and dead-end metabolites [Mackie et al., 2013]:

Definition 2.62 (Dead-end metabolite). *Consider a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. A metabolite $m \in \mathfrak{M}$ is called a dead-end metabolite if there exists a reaction $i \in \mathfrak{R}$ which produces (resp. consumes) m but no reaction in \mathfrak{R} that consumes (resp. produces) m .*

The reversibility of the reactions involved in a metabolic network \mathcal{N} is given in the XML-file as well as the lower and upper bounds. Unfortunately not each reaction which is declared as being reversible can carry positive and negative flux. Those reactions are technically irreversible. See Figure 2.8 for an illustration of dead-end metabolites, blocked reactions, and reactions which are declared as reversible but are irreversible. To detect blocked reactions, dead-end metabolites, and truly reversible reactions we implemented an algorithm based on LP and used this algorithm as a pre-processing step for the models. Given the model, including the upper and lower bounds on the reaction rates, we maximise and minimise the flux through each reaction. If the minimum value is non-negative then the reaction is irreversible. If both the maximum and the minimum flux are zero, the considered reaction is blocked. After removing all columns corresponding to blocked reactions from the stoichiometric matrix S , zero-rows in S correspond to dead-end metabolites.

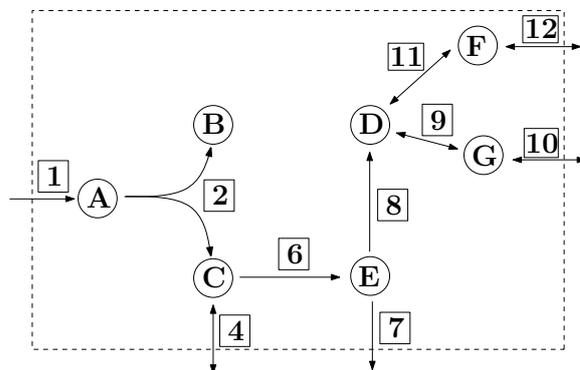


Figure 2.8: The network is a modified version of the network in Figure 1.1. Here, reactions 3 and 5 are removed. As a consequence, metabolite *B* is a dead-end metabolite. This implies that reaction 2 is blocked, thus metabolite *A* is a dead-end metabolite as well and reaction 1 is blocked. Finally, reaction 4 can carry flux in only one direction, thus 4 is not reversible as assumed.

2.4.0.1 Number of reactions vs. unblocked reactions

Blocked reactions will never carry flux in a metabolic network and therefore will never be part of an MCS or EFM. For all methods introduced in this thesis we computed, in a pre-processing step, all blocked reactions of the network using linear programming and removed them from the network.

We illustrate the number of blocked reactions in Figure 2.9 for all 84 networks of the **BiGG Models Database** [King et al., 2016]. For almost all networks the number of blocked reactions in relation to the total number of reactions is more or less the same (roughly 40% of the reactions are blocked). The figure gives a first impression on the result, whereas the explicit numbers can be found in the Tables 2.3 and 2.4. All computations were done on a desktop machine with four processors Intel(R) Core(TM) i5-6500, CPU 3.20GHZ, each with 1 thread. We used CPLEX [IBM Knowledge Center, 2010] as an LP solver.

2. Basics

| network id | mets | rxns | un- blocked rxns | non- dead-end mets | time |
|----------------|------|------|------------------------|--------------------------|------|
| e_coli_core | 72 | 95 | 87 | 68 | 1 |
| iAB_RBC_283 | 342 | 469 | 453 | 333 | 5 |
| iIT341 | 485 | 554 | 436 | 381 | 8 |
| iLJ478 | 570 | 652 | 385 | 331 | 12 |
| iAF692 | 628 | 690 | 484 | 417 | 12 |
| iSB619 | 655 | 743 | 450 | 381 | 13 |
| iNF517 | 650 | 754 | 513 | 435 | 12 |
| iHN637 | 698 | 785 | 524 | 448 | 15 |
| iJB785 | 768 | 849 | 741 | 671 | 25 |
| iJN678 | 795 | 863 | 675 | 597 | 23 |
| iAT_PLT_636 | 738 | 1008 | 1008 | 738 | 20 |
| iNJ661 | 825 | 1025 | 740 | 579 | 25 |
| iJN746 | 907 | 1054 | 652 | 539 | 23 |
| iJR904 | 761 | 1075 | 667 | 450 | 27 |
| iYO844 | 990 | 1250 | 657 | 500 | 34 |
| iND750 | 1059 | 1266 | 631 | 479 | 39 |
| iAF987 | 1109 | 1285 | 840 | 708 | 52 |
| iMM904 | 1226 | 1577 | 893 | 650 | 59 |
| iPC815 | 1552 | 1961 | 1065 | 761 | 76 |
| iRC1080 | 1706 | 2191 | 1583 | 1102 | 163 |
| iYL1228 | 1658 | 2262 | 1223 | 830 | 102 |
| iAF1260 | 1668 | 2382 | 1532 | 1032 | 138 |
| iAF1260b | 1668 | 2388 | 1554 | 1040 | 154 |
| iSDY_1059 | 1888 | 2539 | 1502 | 1026 | 165 |
| STM_v1_0 | 1802 | 2545 | 1597 | 1086 | 163 |
| iJO1366 | 1805 | 2583 | 1705 | 1155 | 196 |
| iSbBS512_1146 | 1910 | 2591 | 1540 | 1018 | 161 |
| iSBO_1134 | 1908 | 2591 | 1530 | 1022 | 155 |
| iS_1188 | 1914 | 2619 | 1504 | 1017 | 169 |
| iSFV_1184 | 1917 | 2621 | 1516 | 1026 | 157 |
| iSF_1195 | 1917 | 2630 | 1512 | 1022 | 170 |
| iSFxv_1172 | 1918 | 2638 | 1554 | 1045 | 166 |
| iSSON_1240 | 1936 | 2693 | 1601 | 1066 | 180 |
| iECH74115_1262 | 1918 | 2694 | 1636 | 1083 | 175 |
| iE2348C_1286 | 1919 | 2703 | 1641 | 1087 | 183 |
| iG2583_1286 | 1919 | 2704 | 1644 | 1087 | 195 |
| iECED1_1282 | 1929 | 2706 | 1644 | 1087 | 190 |
| iECSP_1301 | 1920 | 2712 | 1646 | 1087 | 194 |
| iML1515 | 1877 | 2712 | 1744 | 1147 | 244 |
| iEC042_1314 | 1926 | 2714 | 1644 | 1084 | 188 |
| iECNA114_1301 | 1927 | 2718 | 1656 | 1091 | 185 |
| iECs_1301 | 1923 | 2720 | 1646 | 1087 | 178 |

Table 2.3: Number of reactions and time for the preprocessing for the first 42 networks (according to their number of reactions) of the BiGG Models Database [King et al., 2016]. **network id**: The id of the network on the BiGG Models Database. **mets**: number of metabolites. **rxns**: number of reactions. **unblocked rxns**: number of unblocked reactions. **non-dead-end metabolites**: number of non-dead-end metabolites. **time**: time needed to compute all blocked reactions and dead-end metabolites in seconds.

2.4 Handling metabolic networks

| network id | mets | rxns | un- blocked rxns | non- dead-end mets | time |
|-------------------|------|------|------------------------|--------------------------|------|
| iECIAI39_1322 | 1953 | 2721 | 1569 | 1044 | 170 |
| iZ_1308 | 1923 | 2721 | 1646 | 1087 | 193 |
| iUTI89_1310 | 1940 | 2725 | 1662 | 1096 | 200 |
| ic_1306 | 1936 | 2726 | 1656 | 1090 | 186 |
| iLF82_1304 | 1938 | 2726 | 1650 | 1082 | 193 |
| iECOK1_1307 | 1941 | 2729 | 1670 | 1096 | 179 |
| iECS88_1305 | 1942 | 2729 | 1653 | 1088 | 180 |
| iECABU_c1320 | 1942 | 2731 | 1663 | 1094 | 185 |
| iAPECO1_1312 | 1942 | 2735 | 1668 | 1096 | 178 |
| iNRG857_1313 | 1943 | 2735 | 1675 | 1100 | 180 |
| iUMN146_1321 | 1942 | 2735 | 1670 | 1096 | 179 |
| iECP_1309 | 1941 | 2739 | 1668 | 1094 | 176 |
| iECUMN_1333 | 1935 | 2740 | 1657 | 1093 | 174 |
| iB21_1397 | 1943 | 2741 | 1650 | 1089 | 170 |
| iBWG_1329 | 1949 | 2741 | 1739 | 1164 | 197 |
| iECD_1391 | 1943 | 2741 | 1650 | 1089 | 171 |
| iECDH10B_1368 | 1947 | 2742 | 1736 | 1160 | 200 |
| iECSF_1327 | 1951 | 2742 | 1743 | 1162 | 204 |
| iEcSMS35_1347 | 1947 | 2746 | 1673 | 1102 | 187 |
| iECB_1328 | 1951 | 2748 | 1660 | 1096 | 172 |
| iECBD_1354 | 1952 | 2748 | 1651 | 1089 | 170 |
| iEcDH1_1363 | 1949 | 2750 | 1667 | 1099 | 186 |
| iEcHS_1320 | 1963 | 2753 | 1645 | 1094 | 173 |
| iECDH1ME8569_1439 | 1950 | 2755 | 1670 | 1101 | 190 |
| iEC55989_1330 | 1953 | 2756 | 1670 | 1103 | 185 |
| iETEC_1333 | 1962 | 2756 | 1658 | 1095 | 188 |
| iECO103_1326 | 1958 | 2758 | 1660 | 1096 | 178 |
| iY75_1357 | 1953 | 2759 | 1670 | 1101 | 195 |
| iECO111_1330 | 1959 | 2760 | 1651 | 1089 | 170 |
| iEcE24377_1341 | 1972 | 2763 | 1655 | 1092 | 180 |
| iECIAI1_1343 | 1968 | 2765 | 1638 | 1089 | 172 |
| iEcolC_1368 | 1969 | 2768 | 1653 | 1092 | 187 |
| iECSE_1348 | 1957 | 2768 | 1664 | 1098 | 176 |
| iUMNK88_1353 | 1969 | 2777 | 1665 | 1098 | 183 |
| iEKO11_1354 | 1972 | 2778 | 1655 | 1098 | 186 |
| iECO26_1355 | 1965 | 2780 | 1666 | 1098 | 185 |
| iECW_1372 | 1973 | 2782 | 1668 | 1102 | 190 |
| iWFL_1372 | 1973 | 2782 | 1668 | 1102 | 185 |
| iMM1415 | 2775 | 3726 | 2432 | 1665 | 343 |
| RECON1 | 2766 | 3741 | 2467 | 1586 | 250 |
| iLB1027_lipid | 2172 | 4456 | 4047 | 1814 | 870 |
| iCHOv1 | 4456 | 6663 | 4280 | 2213 | 1281 |

Table 2.4: Number of reactions and time for the preprocessing for the next 42 networks (by the number of reactions) of the BiGG Models Database [King et al., 2016]. **network id**: The id of the network on the BiGG Models Database. **mets**: number of metabolites. **rxns**: number of reactions. **un-blocked rxns**: number of unblocked reactions. **non-dead-end metabolites**: number of non-dead-end metabolites. **time**: time needed to compute all blocked reactions and dead-end metabolites, in seconds.

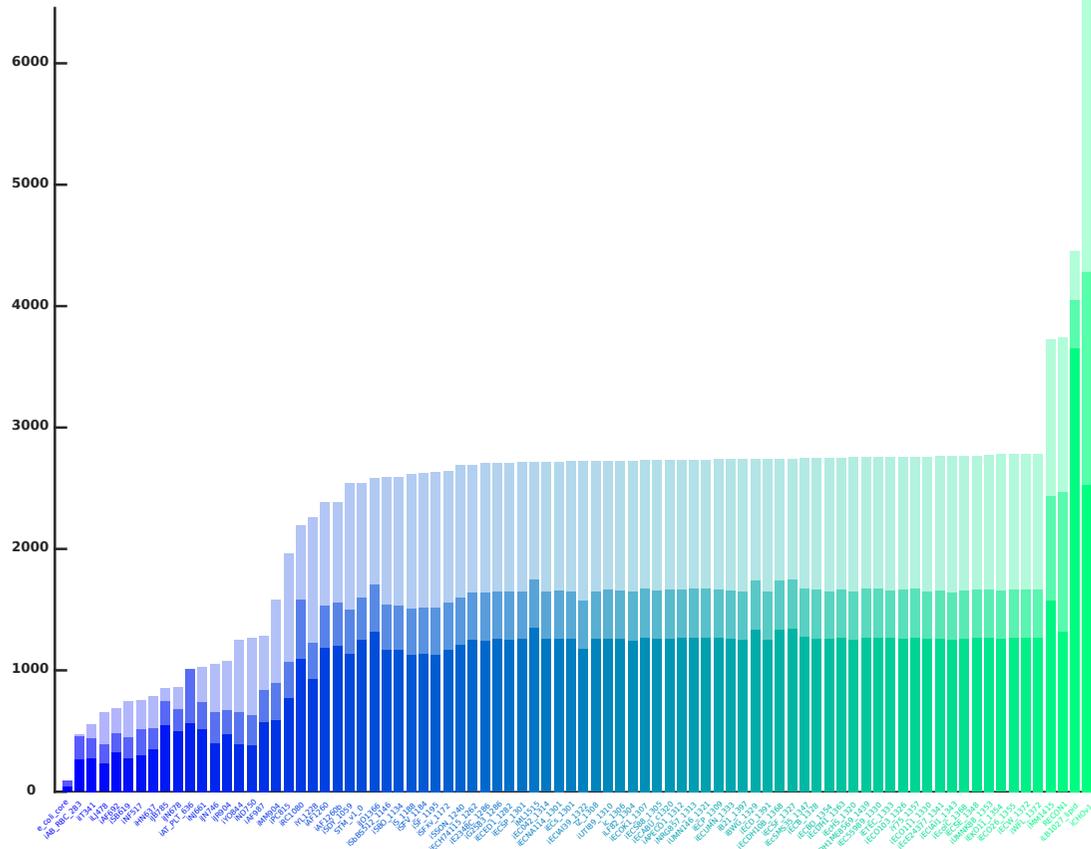


Figure 2.9: This figure illustrates the results from the Tables 2.3 and 2.4. We see here all 84 networks (ordered by their number of reactions from left to right, where the first network has 87 unblocked reactions and the last one 4280). The id of the network can be found at the bottom of the x -axis. For each network there are three overlaying bars. The bar in the back, or the most transparent one, corresponds to the number of all reactions in the network as it can be found at the **BiGG Models Database** [King et al., 2016]. The middle bar corresponds to the number of unblocked reactions and the smallest bar corresponds to the number of unblocked irreversible reactions. Let us consider for example the last network, `iCHOv1`: The network has 6663 reactions, where 4280 are unblocked and 2527 are unblocked irreversible reactions.

Chapter 3

Computing EFMs exploiting FCA

The work presented in this chapter has been done in collaboration with Yaron Goldstein and Alexander Bockmayr and is published in *Proceedings of the Strasbourg Spring School on advances in Systems and Synthetic Biology* under [Röhl et al., 2015], available at <https://assb.lri.fr/Proceedings/LivreStrasbourg-15.pdf>

EFMs play an important role in the analysis of metabolic networks. Although they are of great interest, it is still hard to compute them, since their number grows exponentially with the number of reactions involved in the network. Therefore, it is desirable to develop methods which simplify the computation of EFMs in a given network.

For the approaches presented in this chapter we use flux coupling relations in order to reduce the solution space for EFM computation. *Flux Coupling Analysis* (FCA) determines coupling relations between the reactions in a network, see Section 2.3.2.3. As a reminder, two reactions are partially coupled, if zero flux through one reaction implies zero flux through the other. Whenever there are several reactions that are partially coupled to each other, it is sufficient to take one reaction of this class as a *representative* for the activity of those reactions. There is no need to check if the other reactions in this class carry flux or not.

In this chapter we improve a given MILP, originally presented by [De Figueiredo et al., 2009], to compute shortest EFMs containing one target reaction. We reduce the number of binary variables and consequently the size of the search space significantly with the help of the representatives.

3. Computing EFMs exploiting FCA

Including additional information on directional coupling relations leads to further improvements.

3.1 Shortest elementary flux modes

In the following we assume that the metabolic networks consist of irreversible reactions only. If the original network contains reversible reactions, then these reactions can be split into two irreversible reactions (one for each direction), see Section 2.3.1.2.

3.1.1 Mixed integer linear program to compute the shortest elementary flux modes

In 2009 [De Figueiredo et al., 2009] introduced an algorithm that allows enumerating EFMs of a given metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ using an MILP:

$$\text{(shortestEFMs)} \quad \min_{v,a} \sum_{i \in \mathfrak{R}} a_i \quad (3.1)$$

$$\text{s.t.} \quad Sv = 0 \quad (3.2)$$

$$a_i \leq v_i \quad i \in \mathfrak{R} \quad (3.3)$$

$$v_i \leq M \cdot a_i \quad \forall i \in \mathfrak{R} \quad (3.4)$$

$$\sum_{i \in \mathfrak{R}} a_i \geq 1 \quad (3.5)$$

$$\sum_{i \in \mathfrak{R}} Z_i^k a_i \leq \left(\sum_{i \in \mathfrak{R}} Z_i^k \right) - 1 \quad k = \{1, \dots, K\} \quad (3.6)$$

$$a_i \in \{0, 1\}, v_i \geq 0, \quad \forall i \in \mathfrak{R} \quad (3.7)$$

where K is the total number of EFMs we want to compute. In each step $k \in \{1, \dots, K\}$, the MILP is solved and a new constraint (3.6) is added to exclude the EFMs computed so far.

The algorithm minimises (3.1) the number of active reactions in a steady-state flux vector (3.2). As before, S denotes the stoichiometric matrix and v is a flux vector. To determine the active reactions, binary variables are used such that $a_i = 1$ if and only if reaction i is carrying flux. If $a_i = 0$, then $v_i = 0$, see (3.4), thus reaction i is not allowed to carry flux. Here, $M \gg 0$ is some big constant (“Big M ”). Conversely, $a_i = 1$ implies $v_i \geq 1$, which is ensured by (3.3). To get a feasible flux vector different from the zero flux, (3.5) forces at least one reaction to be active. By definition, Z_i^k

3.1 Shortest elementary flux modes

equals 1 if reaction i carries flux in the EFM which was computed in the k -th step, otherwise Z_i^k is 0. Thus (3.6) guarantees that the EFMs which were computed in the previous steps are not enumerated again. For more details, we refer to [De Figueiredo et al., 2009]. Since $v \in \mathbb{R}_{\geq 0}^{\mathfrak{R}}$ are continuous variables, $a \in \mathbb{B}^{\mathfrak{R}}$ are binary variables, and all constraints involved are linear, **shortestEFMs** is a mixed integer linear program, see Section 2.3.3.

Based on the reaction coupling order, see Section 2.3.2.3, we can now use binary variables corresponding to the coupling classes $[i]$ instead of using binary variables for every individual reaction. Thus we can rewrite the algorithm of De Figueiredo et al. in the following way:

$$(\mathbf{shortestEFMs}_{\mathbf{rep}}) \min_{a,v} \sum_{[i] \in \overline{\mathfrak{R}}} |[i]| a_{[i]} \quad (3.8)$$

$$\text{s.t.} \quad Sv = 0 \quad (3.9)$$

$$a_{[i]} \leq v_j \quad \forall [i] \in \overline{\mathfrak{R}} \text{ and } \forall j \in [i] \quad (3.10)$$

$$v_j \leq M \cdot a_{[i]} \quad \forall [i] \in \overline{\mathfrak{R}} \text{ and } \forall j \in [i] \quad (3.11)$$

$$\sum_{[i] \in \overline{\mathfrak{R}}} a_{[i]} \geq 1 \quad \forall [i] \in \overline{\mathfrak{R}} \quad (3.12)$$

$$\sum_{[i] \in \overline{\mathfrak{R}}} Z_{[i]}^k a_{[i]} \leq \left(\sum_{[i] \in \overline{\mathfrak{R}}} Z_{[i]}^k \right) - 1 \quad \forall k = \{1, \dots, K\} \quad (3.13)$$

$$a_{[i]} \in \{0, 1\} \quad \forall [i] \in \overline{\mathfrak{R}} \quad (3.14)$$

$$v_i \geq 0 \quad \forall i \in \mathfrak{R} \quad (3.15)$$

$|[i]|$ in (3.8) denotes the cardinality of the coupling class $[i]$. Thus, we compute the *shortest* EFMs w.r.t. the number of reactions and not the number of representatives.

The main advantage of our method is that we need only $|\overline{\mathfrak{R}}|$ instead of $|\mathfrak{R}|$ binary variables. For many genome-wide networks, this reduces the number of 0-1 variables by about one half, as shown in Table 3.2.

To further improve our approach, we may add the coupling constraints (3.16) and explicitly help the solver to set coupled variables to their correct values:

$$a_{[i]} \leq v_j \text{ if } j \xrightarrow{0} l \text{ with } l \in [i], \forall [i] \in \overline{\mathfrak{R}}. \quad (3.16)$$

Here we use directional coupling properties of the representatives, i.e., if reaction $l \in \mathfrak{R}$ is directionally coupled to reaction $j \in \mathfrak{R}$ then all reactions in the same coupling class as l are directionally coupled to reaction j . Therefore if j is not active, none of the reactions of the coupling class of l are. With this additional information, we do not reduce the number of binary

3. Computing EFMs exploiting FCA

variables, but may speed up the running time of the algorithm. We call the MILP consisting of all constraints of `shortestEFMsrep` and containing the constraints (3.16) `shortestEFMscoup`.

3.1.2 Computational results

In a preprocessing step, we identified blocked and coupled reactions for different genome-scale metabolic network reconstructions, using the software F2FC [Larhlimi et al., 2012b,a]. The results are given in Table 3.1. From this, we created Table 3.2, which shows the effect of using coupling classes instead of the original set of (unblocked) reactions, regarding the number of binary variables used to solve the MILP. For most of the networks it is sufficient to work with as few as a third of the original number of reactions. Figure 3.1 gives a graphical representation of the results from Table 3.2 and relates them to the time needed to compute them using results from Table 3.1. The larger the network is, the longer it takes to compute the blocked reactions and the representatives.

Next we computed EFMs using `shortestEFMsrep` and `shortestEFMscoup` on different metabolic networks. All computations were done on a desktop machine with two processors Intel(R) Core(TM) i5-2400S, CPU 2.50GHZ, each with 2 threads. Table 3.3 shows how long it takes to calculate a desired number of EFMs, namely 10, 100 and 1000. In Table 3.4 the time ratio of the original algorithm compared to the ones introduced here is shown. Using coupling classes results in smaller MILPs (less binary variables), so we can expect shorter running times. The results in Table 3.3 meet these expectations especially for a large number of EFMs. In most cases, the method `shortestEFMscoup` combining coupling classes with directional coupling constraints yields the best results.

3.1 Shortest elementary flux modes

| model | unblocked reactions | fully coupled | partially coupled | time |
|-------------------------------|---------------------|---------------|-------------------|---------|
| <i>S. cerevisiae</i> iND750 | 744 | 791 | 84 | 20 sec |
| <i>M. tuberculosis</i> iNJ661 | 800 | 8567 | 7588 | 20 sec |
| <i>S. aureus</i> iSB619 | 583 | 1204 | 874 | 12 sec |
| <i>H. pylori</i> iT341 | 501 | 4167 | 5212 | 6 sec |
| <i>E. coli</i> textbook | 95 | 44 | 0 | 0.5 sec |

Table 3.1: Number of reaction couplings (computed with F2FC [Larhlimi et al., 2012b,a]) for different genome-wide metabolic networks and the corresponding running times. **model:** the name of the model, where all models were taken from the BiGG Models database, [King et al., 2016]. **unblocked reactions:** number of unblocked reactions. **fully coupled:** number of fully coupled pairs (where several pairs can form a representative resp. coupling class). **partially coupled:** number of partially coupled pairs (where several pairs can form a representative, resp. coupling class, also together with fully coupled pairs of reactions). **time:** time F2C2 needed to compute the shown results.

| model | reactions | unblocked | representatives |
|-------------------------------|-----------|-----------|-----------------|
| <i>S. cerevisiae</i> iND750 | 1266 | 744 | 446 |
| <i>M. tuberculosis</i> iNJ661 | 1025 | 800 | 412 |
| <i>S. aureus</i> iSB619 | 743 | 583 | 292 |
| <i>H. pylori</i> iT341 | 554 | 501 | 209 |
| <i>E. coli</i> textbook | 95 | 95 | 60 |

Table 3.2: Number of representatives for different genome-wide metabolic networks (computed with F2FC [Larhlimi et al., 2012b,a]). **reactions:** number of reactions of the given models. **model:** name of the model, where all models were taken from the BiGG Models database, [King et al., 2016]. **unblocked:** the number of unblocked reactions. **representatives:** the number of coupling classes.

3. Computing EFMs exploiting FCA

| Model | Nr. of EFMs | Method | | |
|-------------------------------|-------------|-----------|----------|----------|
| | | all | reps | coup |
| <i>E. coli</i> textbook | 10 | 16 sec | 2 sec | 0.6 sec |
| | 100 | 5.5 min | 23 sec | 8 sec |
| | 1000 | 5.75 hrs | 3.5 min | 1.5 min |
| <i>H. pylori</i> iT341 | 10 | 11 sec | 16.5 sec | 15.5 sec |
| | 100 | 18 min | 44 sec | 32 sec |
| | 1000 | 46.5 hrs | 15 min | 16.5 min |
| <i>S. aureus</i> iSB619 | 10 | 29 sec | 7 sec | 6 sec |
| | 100 | 55 min | 51 sec | 37 sec |
| | 1000 | 29.75 hrs | 10 min | 7.5 min |
| <i>M. tuberculosis</i> iNJ661 | 10 | 10 sec | 43 sec | 38 sec |
| | 100 | 3.5 min | 2.5 min | 1 min |
| | 1000 | 34 min | 1.5 hrs | 30 min |
| <i>S. cerevisiae</i> iND750 | 10 | 29 sec | 7 sec | 6 sec |
| | 100 | 1.75 hrs | 1.5 min | 1 min |

Table 3.3: Time needed to compute a given number of EFMs for different modelling approaches. **all**: Each unblocked reaction i has its own binary variable $a_i = 1 \Leftrightarrow v_i \geq 1$ (original MILP from [De Figueiredo et al., 2009]). **reps**: Only coupling class representatives $[i] \in \overline{\mathfrak{R}}$ have binary variables $a_{[i]} = 1 \Leftrightarrow v_j \geq 1$ for $j \xleftrightarrow{0} l$, with $l \in [i]$. **coup**: Same as **reps**, but with additional directional coupling constraints $a_{[i]} \leq v_j$, for all $[i] \in \overline{\mathfrak{R}}$ with $j \xleftrightarrow{0} l$, where $l \in [i]$. We were not able to compute 1000 EFMs for the network *S. cerevisiae* iND750 due to lack of memory.

3.1 Shortest elementary flux modes

| Model | EFMs | Time ratio | |
|-------------------------------|------|------------|-------|
| | | reps | coup |
| <i>E. coli</i> textbook | 10 | 8.7 | 27.7 |
| | 100 | 13.7 | 38.9 |
| | 1000 | 96.5 | 220.7 |
| <i>H. pylori</i> iT341 | 10 | 0.66 | 0.7 |
| | 100 | 23.7 | 32.6 |
| | 1000 | 187.1 | 167.7 |
| <i>S. aureus</i> iSB619 | 10 | 4.2 | 5 |
| | 100 | 64.1 | 88.6 |
| | 1000 | 178 | 239.4 |
| <i>M. tuberculosis</i> iNJ661 | 10 | 0.2 | 0.3 |
| | 100 | 1.4 | 2.9 |
| | 1000 | 0.4 | 1.1 |
| <i>S. cerevisiae</i> iND750 | 10 | 4 | 4.4 |
| | 100 | 64.9 | 98.7 |

Table 3.4: Speed up of the algorithms compared to the standard algorithm [De Figueiredo et al., 2009]. For example, 8.7 means that the method **reps** is 8.7 times faster than **all**. We were not able to compute 1000 EFMs for the network *S. cerevisiae* iND750 due to lack of memory.

3. Computing EFMs exploiting FCA

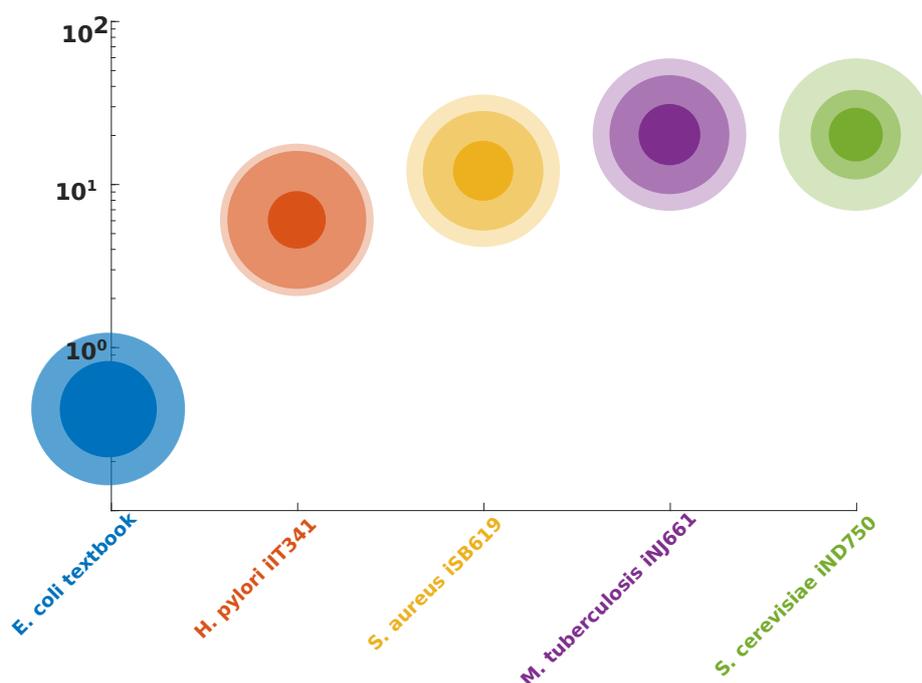


Figure 3.1: The number of reactions, unblocked reactions, and representatives of the five networks are compared. For each network there are three circles. The outer circle indicates the total number of reactions in the network, where the volume for all networks is normalised to 1. The volume of the next smaller circle relates to the number of unblocked reactions and the volume of the smallest circle corresponds to the number of representatives (in relation to the total number of reactions). On the y -axis in log scale the time (in seconds) needed to compute the blocked reactions and the representatives is shown.

3.2 Further work

Exploiting coupling information on reactions of a given metabolic network can be used to decrease the solution space whenever binary variables are used to indicate if a reaction carries flux or not.

For example [David and Bockmayr, 2014] introduce a method based on the MILP of the previous section. The MILP they propose enumerates EFMs where two given target reactions are involved. If only one target reaction is given, the MILP corresponds to **ShortestEFMs**. Thus, in a preprocessing step coupling classes can be determined and used to decrease the running time needed to compute shortest EFMs involving two target reactions.

There exist several methods based on MILP for computing MCSs of a given metabolic network \mathcal{N} [Li et al., 2009; von Kamp and Klamt, 2014; Tobalina et al., 2016; Apaolaza et al., 2017], where it is also feasible to apply the method introduced here.

In general, it is possible to exploit coupling relations between reactions, or variables, not only for metabolic networks but for other networks as well.

In the next chapter we introduce a method for finding a functional subnetwork of a given metabolic network such that given requirements are kept. The method is also based on MILP and we apply coupling information on the reactions in Section 4.3.4 in order to decrease the running time.

3. Computing EFMs exploiting FCA

Chapter 4

Reduction of Networks

The work presented in this chapter has been done in collaboration with Alexander Bockmayr and is published in *BMC Bioinformatics* under [Röhl and Bockmayr, 2017], available at <https://bmcbioinformatics.biomedcentral.com/articles/10.1186/s12859-016-1412-z>.

4.1 Introduction

To analyse genome-scale metabolic network reconstructions, a large variety of constraint-based methods has been developed over the years [Lewis et al., 2012]. Some methods can be applied to genome-scale network reconstructions with several thousands of reactions. Others are limited to small or medium-sized models, like for example the computation of elementary flux modes [Schuster and Hilgetag, 1994], minimal cut sets [Klamt and Gilles, 2004] or the dynamical version of FBA coupled with enzyme costs [Waldherr et al., 2015]. In such situations, a natural question is whether it is possible to reduce the given large network to a smaller one of practical size.

In this chapter we introduce an MILP which computes a minimum subnetwork such that predefined functionalities are kept. In order to decrease the running time we make use of the technique introduced in Chapter 3.

4.2 Background

[Erdrich et al., 2015] introduced a method called `NetworkReducer`, which reduces large metabolic networks to smaller subnetworks, while preserving

4. Reduction of Networks

relevant biological properties of interest. The algorithm in [Erdrich et al., 2015] is divided into two parts: network pruning and network compressing. In the *compressing* step, reactions belonging to the same enzyme subset [Pfeiffer et al., 1999] are lumped together. In the *pruning* step *removable* and *non-removable* reactions are identified such that the subnetwork consisting of the non-removable reactions fulfils four requirements, which can be specified by the user:

- (a) Set of *protected metabolites* $\Upsilon^{\mathfrak{M}}$: all metabolites in $\Upsilon^{\mathfrak{M}}$ must be retained in the subnetwork.
- (b) Set of *protected reactions* $\Upsilon^{\mathfrak{R}}$: all reactions in $\Upsilon^{\mathfrak{R}}$ must be retained in the reduced network.
- (c) Set \mathcal{F} of *protected functionalities* (or phenotypes) for the subnetwork. We assume that any protected functionality $f \in \mathcal{F}$ can be described by a corresponding system of linear inequalities: $D_f v \leq d_f$.
- (d) Minimum *degrees of freedom*: $dof \geq dof_{\min}$. Here, the degrees of freedom dof correspond to the dimension of the null space of the stoichiometric matrix S , i.e., $dof = |\mathfrak{R}| - \rho(S)$.

The overall goal of `NetworkReducer` is to find a subnetwork containing as few reactions as possible such that all requirements (a) – (d) can be satisfied by a suitable flux vector. An example is given in Figure 4.1.

The method of [Erdrich et al., 2015] searches for a suitable subnetwork by iterating over the reactions. In every iteration, the flux through one particular reaction is set to zero and a linear program (LP) is solved to check if the remaining reactions still form a *feasible* subnetwork. Feasibility means that there exist non-zero flux vectors satisfying the steady-state condition and the other requirements. To identify the reaction to be eliminated a flux variability analysis (FVA) [Mahadevan and Schilling, 2003], see Section 2.3.2.2, is done and a reaction with the smallest overall flux range is selected. Thus in every iteration, an LP is solved and an FVA is performed. Each FVA involves solving up to $2 \cdot |\mathfrak{R}|$ LPs, where $|\mathfrak{R}|$ is the number of reactions.

An important aspect of the method in [Erdrich et al., 2015] is that it does not necessarily compute a minimum subnetwork (with respect to the number of active reactions), see Figure 4.2 for an example. The method that we develop here always finds a feasible subnetwork with a minimum number of active reactions. A subnetwork satisfying the requirements (a) – (c) can be obtained by solving only one mixed-integer linear program (MILP). If

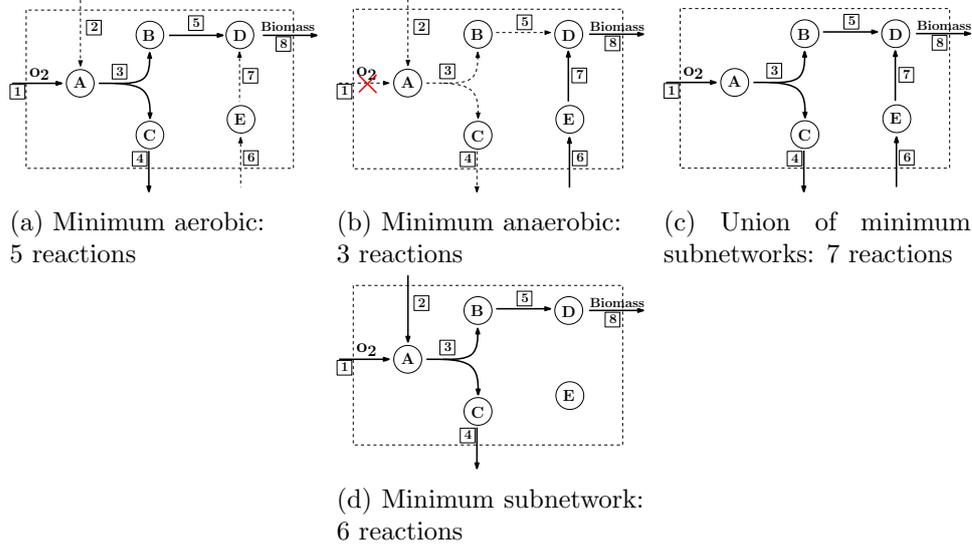


Figure 4.1: Solid arcs correspond to active reactions, dotted arcs to inactive reactions. In Figure 4.1a, the flux vector satisfies the functionality of carrying flux through the biomass reaction while having oxygen uptake. In Figure 4.1b, the functionality is carrying flux through the biomass reaction while there is no oxygen uptake. Combining the two flux vectors leads to the network in Figure 4.1c, which contains seven active reactions. A minimum subnetwork enabling both functionalities with only six reactions is given in Figure 4.1d. The corresponding binary variables for Figure 4.1d would have the following values: $a_1 = 1, a_2 = 1, a_3 = 1, a_4 = 1, a_5 = 1, a_6 = 0, a_7 = 0, a_8 = 1$, where a_i corresponds to reaction i .

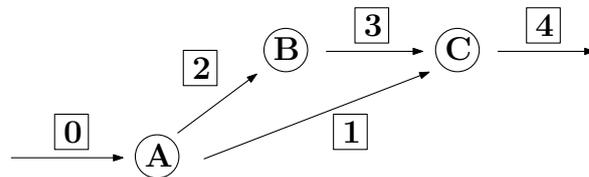


Figure 4.2: If in the first step of the pruning procedure the flux through reaction 1 is set to zero, reaction 1 is removable and reactions 2 and 3 are non-removable. If in the first step reaction 2 or 3 is set to zero, both of them would be removable and reaction 1 would be non-removable. The resulting subnetwork is then smaller than the first one.

4. Reduction of Networks

this subnetwork does not fulfil the *dof*-requirement (d), we exclude this subnetwork and compute a new subnetwork by solving the MILP again. This method turns out to be much faster than the algorithm introduced in [Erdrich et al., 2015]. More importantly, we are guaranteed to obtain a minimum subnetwork regarding the number of active reactions, which is not the case for **NetworkReducer**. However, due to the minimality condition, our method cannot preserve flux variability in the same way as **NetworkReducer** does.

A second related work is the **FASTCORE** algorithm of [Vlassis et al., 2014]. This method is also based on solving several LPs but without performing an FVA in between. Thus it is a very fast approach. However, the resulting subnetworks are not minimum and only protected reactions can be specified, but no protected metabolites, functionalities, or degrees of freedom.

An early approach for network reduction was introduced by [Burgard et al., 2001], see Section 2.3.3.1 and later improved in 2014 by [Jonnalagadda and Srinivasan, 2014]. This method also allows computing minimum subnetworks using an MILP approach. However, only one functionality can be formulated and not several ones like in **NetworkReducer**.

Altogether, our method can be seen as a network reduction algorithm that merges features from **NetworkReducer** and the method in [Burgard et al., 2001], such that we can specify biological requirements like in [Erdrich et al., 2015] and compute all minimum subnetworks using an MILP, similar to [Burgard et al., 2001].

The organisation of this chapter is as follows. In Section 4.3 we develop the underlying MILP methods. We start with the basic algorithm and then describe several improvements. In Section 4.4 we compare our MILP approach with the existing methods **NetworkReducer** and **FASTCORE**. Furthermore, we apply it to a collection of genome-scale network reconstructions and discuss the results. Section 4.5 presents the conclusions of this chapter.

A software tool implementing the algorithms described in this chapter is available at <https://sourceforge.net/projects/minimalnetwork/>.

4.3 Methods

4.3.1 Basic MILP to compute a minimum subnetwork

We always assume that our network is in steady-state, i.e., $Sv = 0$, with bounds on the reaction rates $l \leq v \leq u$. Each functionality $f \in \mathcal{F}$ is described by a system of linear inequalities: $D_f v \leq d_f$. For example, we may require that the biomass reaction has to carry at least 99% of its maximal rate: $v_{\text{Bio}} \geq 0.99 \cdot \max(v_{\text{Bio}})$.

We will use binary variables $a_i \in \{0, 1\}$ to indicate whether or not reaction i carries flux in the subnetwork. Thus we need the relationship $a_i = 0$ if and only if $v_i = 0$. For an irreversible reaction $i \in \text{Irr}$, this can be achieved using constraints of the form

$$\delta a_i \leq v_i \leq M a_i. \quad (4.1)$$

For reversible reactions, we use another binary variable \bar{a}_i and the constraints

$$\delta a_i - M \bar{a}_i \leq v_i \leq M a_i - \delta \bar{a}_i, \quad a_i + \bar{a}_i \leq 1. \quad (4.2)$$

To force protected irreversible reactions to carry flux, we use the constraints $a_i = 1$ for all $i \in \Upsilon^{\text{Irr}} = \Upsilon^{\mathfrak{R}} \cap \text{Irr}$. Enforcing flux through a protected reversible reaction can be realized in a similar way with the constraints $a_i + \bar{a}_i = 1$, for all $i \in \Upsilon^{\text{Rev}} = \Upsilon^{\mathfrak{R}} \cap \text{Rev}$.

For any protected metabolite $m \in \Upsilon^{\mathfrak{M}}$, let \mathfrak{R}_m be the set of reactions involving m . By Irr_m resp. Rev_m we denote the set of irreversible resp. reversible reactions in \mathfrak{R}_m . If \mathfrak{R}_m contains at least one protected reaction i , metabolite m will be protected by reaction i . However, if $\mathfrak{R}_m \cap \Upsilon^{\mathfrak{R}} = \emptyset$, further constraints are needed to protect m :

$$\sum_{i \in \mathfrak{R}_m} a_i + \sum_{i \in \text{Rev}_m} \bar{a}_i \geq 1, \quad \forall m \in \Upsilon_0^{\text{Met}}, \quad (4.3)$$

where $\Upsilon_0^{\text{Met}} = \{m \in \Upsilon^{\mathfrak{M}} \mid \mathfrak{R}_m \cap \Upsilon^{\mathfrak{R}} = \emptyset\}$.

In [Erdreich et al., 2015], an additional requirement is to have a minimum number of active reactions. Here we do not include this restriction for the following reasons. First, we will search for the minimum number of active reactions such that all the other requirements are fulfilled. Second, in [Erdreich et al., 2015] the minimum number of active reactions is always set to

4. Reduction of Networks

1. Since there exist reactions which are forced to carry flux, this constraint is redundant.

To find a subnetwork which contains the minimum number of active reactions, we minimize over the sum of the binary variables a_i, \bar{a}_k , which indicate whether a reaction carries flux. The resulting MILP is the following:

$$\begin{aligned}
 (\text{MinNW-0}) \min_{a, \bar{a}, v} & \sum_{i \in \mathfrak{R}} a_i + \sum_{k \in \text{Rev}} \bar{a}_k & (4.4) \\
 Sv = 0, & \quad l \leq v \leq u \\
 D_f v \leq d_f & \quad \forall f \in \mathcal{F} \\
 \delta a_i \leq v_i \leq M a_i & \quad \forall i \in \text{Irr} \\
 \delta a_i - M \bar{a}_i \leq v_i \leq M a_i - \delta \bar{a}_i & \quad \forall i \in \text{Rev} \\
 a_i + \bar{a}_i \leq 1 & \quad \forall i \in \text{Rev} \\
 a_i = 1, \quad a_k + \bar{a}_k = 1 & \quad \forall i \in \Upsilon^{\text{Irr}}, \forall k \in \Upsilon^{\text{Rev}} \\
 \sum_{i \in \mathfrak{R}_m} a_i + \sum_{i \in \text{Rev}_m} \bar{a}_i \geq 1, & \quad \forall m \in \Upsilon_0^{\text{Met}} \\
 v_i \in \mathbb{R}, \quad a_i \in \{0, 1\} & \quad \forall i \in \mathfrak{R} \\
 \bar{a}_k \in \{0, 1\} & \quad \forall k \in \text{Rev}
 \end{aligned}$$

4.3.2 Conflicting functionalities

In the case study considered in [Erdrich et al., 2015], the resulting subnetwork should keep two desired functionalities: at least 99.9% of the maximal growth rate under both aerobic and anaerobic conditions should be maintained. These two requirements cannot be realized with the same flux vector v because they imply two opposite states of the reaction o_2 that transports O_2 into the network. We would need a vector v with $v_{o_2} \geq \delta$ and $v_{o_2} = 0$ at the same time, which is not possible.

MinNW-0 computes one feasible flux vector v of the network. But, to get a subnetwork which fulfils the two functionalities we need one flux vector which fulfils the aerobic condition and another one for the anaerobic condition, see Figure 4.1. To realize this with a single MILP we have to modify MinNW-0. First, we search for a flux vector v^0 which contains the protected metabolites and protected reactions. Additionally, for each functionality $j \in \mathcal{F}$ we search for a flux vector v^j satisfying $D_j v^j \leq d_j$ and corresponding binary variables. For example, in Figure 4.1, we would have $a_1 = 1$ in case 1a) and $a_1 = 0$ in case 1b). Due to (4.1) and (4.2), this would imply $a_1 = 1$ and $a_1 = 0$ at the same time, which is not possible. Thus we have to use

different binary variables a_i^j for v_i^j . With this, the inequalities (4.1) and (4.2) become

$$\delta a_i^j \leq v_i^j \leq M a_i^j \quad \forall j \in \{0, \dots, |\mathcal{F}|\}, \forall i \in \text{Irr} \quad (4.5)$$

$$\delta a_i^j - M \bar{a}_i^j \leq v_i^j \leq M a_i^j - \delta \bar{a}_i^j, \quad \forall j \in \{0, \dots, |\mathcal{F}|\}, \forall i \in \text{Rev}, \quad (4.6)$$

$$a_i^j + \bar{a}_i^j \leq 1 \quad \forall j \in \{0, \dots, |\mathcal{F}|\}, \forall i \in \text{Rev}. \quad (4.7)$$

Using the new variables a_i^0 , we reformulate the constraints for the protected reactions: $a_i^0 = 1$, for all $i \in \Upsilon^{\text{Irr}}$. Enforcing flux through a reversible reaction is realized in a similar way by $a_i^0 + \bar{a}_i^0 = 1$, for all $i \in \Upsilon^{\text{Rev}}$. Finally, the constraints for the protected metabolites become

$$\sum_{i \in \mathfrak{R}_m} a_i^0 + \sum_{i \in \text{Rev}_m} \bar{a}_i^0 \geq 1, \quad \forall m \in \Upsilon_0^{\text{Met}}. \quad (4.8)$$

To obtain a minimum subnetwork, we have to minimize the total number of active reactions. Thus, we need binary variables a_i with the property

$$\begin{aligned} a_i &= 0 \text{ if and only if } a_i^j = 0 \text{ for all } j \in \{0, \dots, |\mathcal{F}|\}, \text{ or equivalently} \\ a_i &= 1 \text{ if and only if } a_i^j = 1 \text{ for some } j \in \{0, \dots, |\mathcal{F}|\}. \end{aligned}$$

For irreversible reactions, this can be encoded by the constraints

$$a_i \leq \sum_{j=0}^{|\mathcal{F}|} a_i^j \leq (1 + |\mathcal{F}|) \cdot a_i, \quad \forall i \in \text{Irr}, \quad (4.9)$$

and for reversible reactions we get

$$a_i \leq \sum_{j=0}^{|\mathcal{F}|} (a_i^j + \bar{a}_i^j) \leq (2 + 2|\mathcal{F}|) \cdot a_i, \quad \forall i \in \text{Rev}. \quad (4.10)$$

4. Reduction of Networks

The resulting MILP is the following:

$$\begin{aligned}
(\text{minNW}) \quad & \min_{a, \bar{a}, v} \sum_{i \in \mathfrak{R}} a_i & (4.11) \\
Sv^j = 0, \quad & l \leq v^j \leq u & \forall j \in \{0, \dots, |\mathcal{F}|\} \\
D_j v^j \leq d_j & & \forall j \in \{1, \dots, |\mathcal{F}|\} \\
\delta a_i^j \leq v_i^j \leq M a_i^j & & \forall j \in \{0, \dots, |\mathcal{F}|\}, \forall i \in \text{Irr} \\
\delta a_i^j - M \bar{a}_i^j \leq v_i^j \leq M a_i^j - \delta \bar{a}_i^j & & \forall j \in \{0, \dots, |\mathcal{F}|\}, \forall i \in \text{Rev} \\
a_i^j + \bar{a}_i^j \leq 1 & & \forall j \in \{0, \dots, |\mathcal{F}|\}, \forall i \in \text{Rev} \\
a_i^0 = 1, \quad a_k^0 + \bar{a}_k^0 = 1 & & \forall i \in \Upsilon^{\text{Irr}}, \forall k \in \Upsilon^{\text{Rev}} \\
\sum_{i \in \mathfrak{R}_m} a_i^0 + \sum_{i \in \text{Rev}_m} \bar{a}_i^0 \geq 1, & & \forall m \in \Upsilon_0^{\text{Met}} \\
a_i \leq \sum_{j=0}^{|\mathcal{F}|} a_i^j \leq (1 + |\mathcal{F}|) \cdot a_i, & & \forall i \in \text{Irr} \\
a_i \leq \sum_{j=0}^{|\mathcal{F}|} (a_i^j + \bar{a}_i^j) \leq (2 + 2|\mathcal{F}|) \cdot a_i, & & \forall i \in \text{Rev} \\
v_i^j \in \mathbb{R}, \quad a_i^j, a_i \in \{0, 1\}, & & \forall i \in \mathfrak{R} \\
\bar{a}_k^j \in \{0, 1\} & & \forall k \in \text{Rev}
\end{aligned}$$

`minNW` computes a subnetwork with minimum number of active reactions while satisfying all the requirements.

Figure 4.1 illustrates the following example:

Example 4.1 (Example for `minNW`). *The network in Figure 4.1a fulfils the functionality regarding the aerobic condition, while the network in Figure 4.1b fulfils the anaerobic condition. The combination of the minimum subnetworks corresponding to each functionality does not lead to a minimum subnetwork for both, see Figure 4.1c. The minimum subnetwork for this example is given in Figure 4.1d.*

4.3.3 Computing all minimum subnetworks

There are scenarios where we have to compute more than one subnetwork. For instance, consider the case where the minimal *dof* (requirement (d)) is larger than 1. If the subnetwork computed with `minNW` does not have the required *dof*, we have to compute a different subnetwork. Furthermore, the computed minimum subnetwork need not be unique. Thus there may exist

different subnetworks which all fulfil the requirements and have the same number of active reactions. So we may be interested in finding *all* minimum subnetworks. To compute different subnetworks we can use the MILP `minNW` in an iterative way. Whenever a minimum subnetwork is found, we formulate a constraint which excludes this subnetwork as a feasible solution and solve the (extended) MILP again. For that purpose we formulate the following constraints:

$$\sum_{i \in \mathfrak{R}} (1 - Z_i^k) a_i + \sum_{i \in \mathfrak{R}} Z_i^k (1 - a_i) \geq 1, \quad k = 1, 2, \dots \quad (4.12)$$

where $Z_i^k = 1$ if reaction i carries flux in the subnetwork which was computed in the k -th step, otherwise $Z_i^k = 0$. Thus, (4.12) guarantees that at least one inactive reaction will become active, or at least one active reaction will become inactive in the new solution.

Solving `minNW` iteratively and adding the constraints (4.12) in each step, we are now able to enumerate all minimum subnetworks.

4.3.4 Reducing the number of binary variables

To further improve efficiency, we will make use of flux coupling information, see Section 2.3.2.3, Chapter 3, and [Burgard et al., 2004; Larhlimi et al., 2012b; Goldstein and Bockmayr, 2015; Röhl et al., 2015].

The main advantage of introducing coupling classes is that, if one reaction in a class is not carrying flux, no other reaction in the class does, and vice versa. Therefore, in every approach where binary variables are used to indicate if a reaction appears or not, it suffices to consider one reaction from every coupling class instead of considering all of them. Depending on the number of reactions and associated coupling classes, this may significantly reduce the number of required variables, see Table 4.1. Based on the equivalence relation $\overset{0}{\leftrightarrow}$, we now use binary variables corresponding to the coupling classes $[r]$ instead of having binary variables for each individual reaction. Thus we can rewrite the algorithm `minNW` in the following way:

4. Reduction of Networks

$$\begin{aligned}
(\mathbf{minNW})_{\mathbf{rep}} \min_{a, \bar{a}, v} \sum_{[i] \in \overline{\mathfrak{R}}} |[i]| a_{[i]} & \quad (4.13) \\
Sv^j = 0, \quad l \leq v^j \leq u & \quad \forall j \in \{0, \dots, |\mathcal{F}|\} \\
D_j v^j \leq d_j & \quad \forall j \in \{1, \dots, |\mathcal{F}|\} \\
\delta a_{[i]}^j \leq v_s^j \leq M a_{[i]}^j & \quad \forall j \in \{0, \dots, |\mathcal{F}|\}, [i] \in \overline{\text{Irr}}, s \in [i] \\
\delta a_{[i]}^j - M \bar{a}_{[i]}^j \leq v_s^j \leq M a_{[i]}^j - \delta \bar{a}_{[i]}^j & \quad \forall j \in \{0, \dots, |\mathcal{F}|\}, [i] \in \overline{\text{Rev}}, s \in [i] \\
a_{[i]}^j + \bar{a}_{[i]}^j \leq 1 & \quad \forall j \in \{0, \dots, |\mathcal{F}|\}, [i] \in \overline{\text{Rev}} \\
a_{[i]}^0 = 1, \quad a_{[i']}^0 + \bar{a}_{[i']}^0 = 1 & \quad \forall [i] \in \overline{\Upsilon^{\text{Irr}}}, [i'] \in \overline{\Upsilon^{\text{Rev}}} \\
\sum_{[i] \in \overline{\mathfrak{R}_m}} a_{[i]}^0 + \sum_{[i] \in \overline{\text{Rev}_m}} \bar{a}_{[i]}^0 \geq 1, & \quad \forall m \in \overline{\Upsilon_0^{\text{Met}}} \\
a_{[i]} \leq \sum_{j=0}^{|\mathcal{F}|} a_{[i]}^j \leq (|\mathcal{F}| + 1) \cdot a_{[i]} & \quad \forall [i] \in \overline{\text{Irr}} \\
a_{[i]} \leq \sum_{j=0}^{|\mathcal{F}|} (a_{[i]}^j + \bar{a}_{[i]}^j) \leq (2 + 2|\mathcal{F}|) \cdot a_{[i]} & \quad \forall [i] \in \overline{\text{Rev}} \\
a_{[i]}^j, a_{[i]} \in \{0, 1\}, v_s^j \in \mathbb{R} & \quad \forall [i] \in \overline{\mathfrak{R}}, s \in \mathfrak{R} \\
\bar{a}_{[i']}^j \in \{0, 1\} & \quad \forall [i'] \in \overline{\text{Rev}}
\end{aligned}$$

Here, $|[i]|$ denotes the cardinality of the coupling class $[i]$. Thus, we compute the smallest subnetwork with respect to the number of active reactions and not with respect to the number of active representatives. $\overline{\text{Irr}}$ denotes the representatives of the irreversible reactions, and $\overline{\text{Rev}}$ those of the reversible reactions. Similarly, $\overline{\Upsilon^{\text{Irr}}}$ resp. $\overline{\Upsilon^{\text{Rev}}}$ is the set of representatives of protected irreversible resp. protected reversible reactions. With $\overline{\Upsilon_0^{\text{Met}}}$ we denote the representatives which include a protected metabolite.

To exclude previously enumerated subnetworks the constraints (4.12) can be adapted in the following way:

$$\sum_{[i] \in \overline{\mathfrak{R}}} (1 - Z_{[i]}^k) a_{[i]} + \sum_{[i] \in \overline{\mathfrak{R}}} Z_{[i]}^k (1 - a_{[i]}) \geq 1, \quad k = 1, 2, \dots \quad (4.14)$$

Using representatives we need only $|\overline{\mathfrak{R}}|$ instead of $|\mathfrak{R}|$ binary variables. For many genome-wide networks, this reduces the number of 0-1 variables by about 1/2, see the examples in Table 4.1 and Figure 4.3.

| model | reactions | unblocked | representatives |
|--------------------------------------|------------------|------------------|------------------------|
| <i>Mus musculus</i> | 3726 | 2436 | 1489 |
| <i>E. coli</i> iJO1366 | 2583 | 2369 | 1399 |
| <i>S. Typhimurium</i> LT2 | 2545 | 1620 | 1047 |
| <i>S. boydii</i> CDC 3083-94 | 2592 | 1546 | 1016 |
| <i>K. pneumoniae</i> MGH 78578 | 2262 | 1223 | 804 |
| <i>Y. pestis</i> CO92 | 1961 | 1065 | 639 |
| <i>S. cerevisiae</i> S288c | 1577 | 885 | 558 |
| <i>G. metallireducens</i> GS-15 | 1285 | 845 | 330 |
| <i>M. tuberculosis</i> iNJ661 | 1025 | 800 | 412 |
| <i>B. subtilis</i> 168 | 1250 | 658 | 342 |
| <i>P. putida</i> KT2440 | 1056 | 652 | 282 |
| <i>C. ljungdahlii</i> DSM 13528 | 785 | 526 | 215 |
| <i>H. pylori</i> iT341 | 554 | 501 | 209 |
| <i>M. barkeri</i> str. <i>Fusaro</i> | 690 | 484 | 174 |
| <i>S. aureus</i> iSB619 | 743 | 465 | 224 |
| <i>T. maritima</i> MSB8 | 652 | 385 | 148 |

Table 4.1: Number of representatives for different genome-wide metabolic networks (computed with F2FC [Larhlimi et al., 2012b,a]). All models are from BiGG Models database [King et al., 2016]. **model**: name of the model. **reactions**: number of reactions. **unblocked**: number of unblocked reactions. **representatives**: number of representatives.

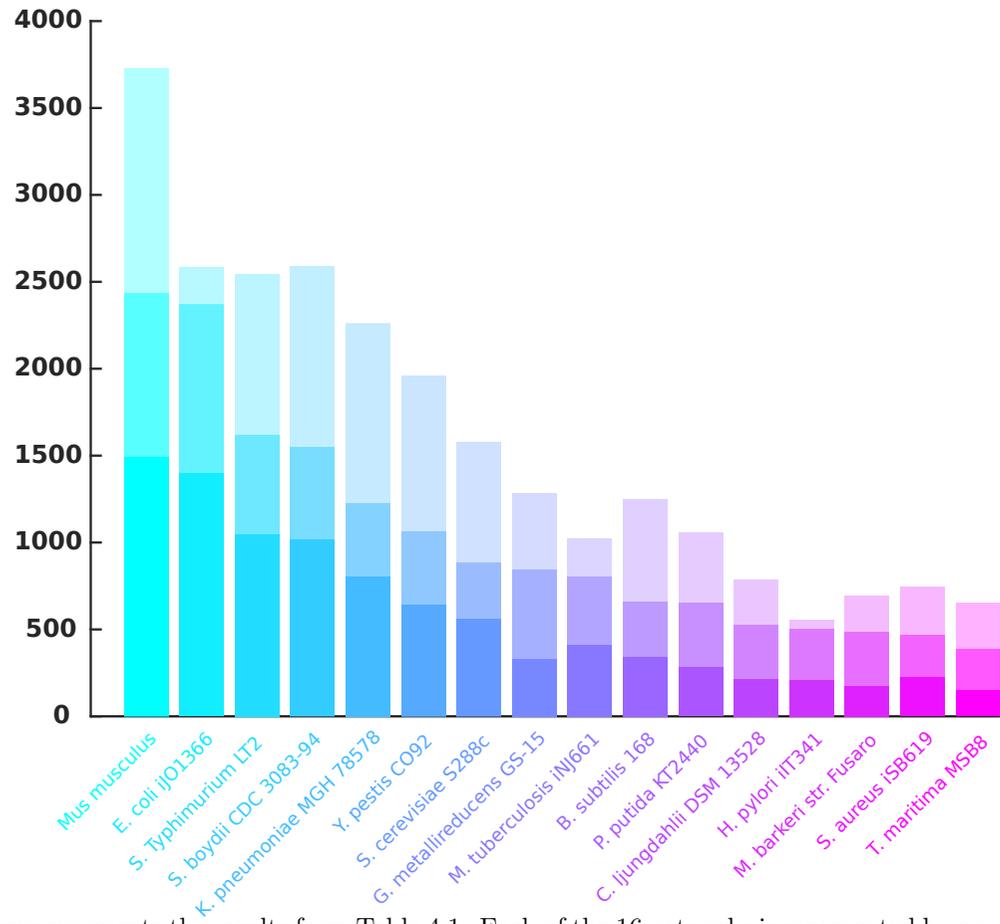


Figure 4.3: This figure represents the results from Table 4.1. Each of the 16 networks is represented by one colour. The largest and most transparent bar indicates the total number of reactions, written on the y -axis. The middle bar corresponds to the number of unblocked reactions and therefore to the number of binary variables needed if coupling of the reactions is not exploited. Finally, the smallest bar indicates the number of representatives of the network.

4.4 Results and discussion

We implemented our MILPs in `MATLAB` and used `CPLEX` [IBM Knowledge Center, 2010] as a solver like in [Erdrich et al., 2015]. For `NetworkReducer` resp. `FASTCORE` we used the implementation provided by the authors of [Erdrich et al., 2015] resp. [Vlassis et al., 2014]. All computations were done on a desktop machine with two processors Intel(R) Core(TM) i5-2400S, CPU 2.50GHZ, each 1 thread. For algorithm `minNWrep`, we computed the coupling classes for partially coupled reactions using the software `F2FC` [Larhlimi et al., 2012b,a]. Due to numerical instabilities, see Section 2.3.4, we decided to use indicator variables in `CPLEX`. The use of indicator variables is straightforward. For example, the *big M* constraint $\delta a \leq v \leq Ma$ is replaced by $a = 0 \Rightarrow v = 0$, $a = 1 \Rightarrow v \geq \delta$, where $a \in \{0, 1\}$ is the indicator variable, see Section 2.3.4.2 for more details. While indicator variables drastically increase the running time, we still outperform the algorithm in [Erdrich et al., 2015].

4.4.1 Comparison with `NetworkReducer`

In a first experiment, we ran our implementations on the two metabolic network reconstructions and functionalities considered in [Erdrich et al., 2015]. Table 4.2 shows the running time for calculating a subnetwork with the desired properties.

For *Synechocystis* sp. PCC 6803 [Knoop et al., 2013], the subnetwork computed by `NetworkReducer` [Erdrich et al., 2015] contains 462 reactions and thus 9 reactions more than the minimum subnetwork with 453 reactions obtained by our method. The two subnetworks have 413 reactions in common. 49 reactions in the larger subnetwork cannot be found in the minimum subnetwork, while 40 reactions in the minimum subnetwork do not appear in the larger one.

For the *E. coli* iAF1260 model [King et al., 2016] we get similar results. The subnetwork computed by `NetworkReducer` contains 35 reactions more than the minimum subnetwork obtained by our method. Both networks have 424 reactions in common. There are 51 reactions that can only be found in the subnetwork computed with `NetworkReducer`, while there are 12 reactions which appear only in the minimum subnetwork.

4. Reduction of Networks

| Algorithm | <i>Synechocystis</i> | | <i>E. coli</i> | |
|----------------|----------------------|-----------|----------------|-----------|
| | time | reactions | time | reactions |
| NetworkReducer | 5.5 min | 462 | 6 hrs | 455 |
| minNW | 31 sec | 453 | 1 hr | 416 |

Table 4.2: Time needed to compute a subnetwork with given requirements resp. constraints. **NetworkReducer**: The algorithm introduced in [Erdreich et al., 2015]. **minNW**: The MILP introduced here, using indicator variables. *Synechocystis* denotes the model *Synechocystis* sp. PCC 6803 from [Knoop et al., 2013]. *E. coli* is the model *Escherichia coli* iAF1260 from [King et al., 2016].

| model | rxns | rxns | time | rxns | time |
|-------------------------------|------|-----------|-----------|-------|---------|
| | | FAST-CORE | FAST-CORE | minNW | minNW |
| <i>M. tuberculosis</i> iNJ661 | 1025 | 134 | 0.12 sec | 62 | 2.5 hrs |
| <i>H. pylori</i> 26695 | 501 | 319 | 0.6 sec | 306 | 2 min |

Table 4.3: **model**: Name of the model. Both models were taken from **BiGG Models** database [King et al., 2016]. **rxns**: number of unblocked reactions in the original network. **rxns FASTCORE**: number of the reactions in the subnetwork computed with **FASTCORE**. **time FASTCORE**: running time of **FASTCORE**. **rxns minNW**: number of the reactions in the subnetwork computed with **minNW**. **time minNW**: running time for the algorithm **minNW** using indicator variables.

4.4.2 Comparison with FASTCORE

FASTCORE [Vlassis et al., 2014] is a heuristic algorithm which is much faster than our method. However, the computed subnetworks are not minimum as can be seen from Table 4.3. The subnetwork computed with our method is not contained in the subnetwork computed with **FASTCORE**. For *H. pylori* 26695 there are 22 reactions that appear only in the **FASTCORE** subnetwork and 9 reactions which can be found only in the minimum subnetwork. Similarly, for *M. tuberculosis* iNJ661, there are 78 reactions that appear only in the **FASTCORE** subnetwork and six reactions which can be found only in our subnetwork. The names of the reactions for both examples are given in the Appendix in Section 10.1.

4.4.3 Network reduction for genome-scale metabolic networks

As a proof of concept we applied our methods to compute minimum subnetworks for 16 metabolic network reconstructions taken from BiGG Models [King et al., 2016] under different scenarios. For each type of organism in BiGG we considered one model (except for human recon because there is no biomass reaction). An overview of the results is given in Table 4.4, the time relation between using indicator variables for each reaction and using indicator variables for representatives is shown in Figure 4.4. In some cases we had only one minimum subnetwork, while for some models and scenarios we found different ones. For example, in the case of *H. pylori* 26695, we get 16 distinct minimum subnetworks, which will be discussed in Section 4.4.4.

Following [Klamt and Gilles, 2004], we call a reaction *essential* if after removing this reaction it is no longer possible to achieve at least $p\%$ of the maximal biomass production rate, see Definition 2.10. Like in [Klamt and Gilles, 2004], we choose $p = 20$. A minimum subnetwork where it is possible to achieve a maximal biomass rate constitutes a subnetwork where all essential reactions can be active and so all essential reactions have to be included in the subnetwork. We will give the number of essential reactions for the different models to give an idea about how many reactions are additionally needed to have a functional minimum subnetwork including all essential reactions.

The scenarios for the different networks and some conclusions are given next, full details can be found in Section 10.1 of the Appendix. The bounds on the flux rates are the default rates given by the BiGG Models database [King et al., 2016].

For the networks *Mus musculus*, *E. coli* iJO1366, *S. Typhimurium* LT2, *S. boydii* CDC 3083-94, and *K. pneumoniae* MGH 78578 the requirements are that at least 99.9 % of the maximal biomass rates for the aerobic and anaerobic case can be realized by the subnetwork. For *Y. pestis* CO92 the requirements are that at least 99.9 % of the maximal growth rate with glycine uptake and the maximal growth rate without glycine uptake can be realized by the subnetwork. For *S. cerevisiae* S288c the maximal biomass rate with ethanol exchange and without ethanol exchanged has to be realized by the reduced subnetwork. For *G. metallireducens* GS-15, *C. ljungdahlii* DSM 13528, and *T. maritima* MSB8 the maximal biomass rate with H₂O uptake and without H₂O exchanged has to be realized by the reduced subnetwork. For *M. tuberculosis* iNJ661 one requirement is that at least 99.9 % of the maximal growth rate can be achieved. Additionally, we defined 36 protected reactions. For *B. subtilis* 168 the requirements are that at least

4. Reduction of Networks

99.9 % of the maximal growth rate with hydrogen uptake and the maximal growth rate without hydrogen uptake can be realized by the subnetwork. For *P. putida* KT2440 one requirement is that at least 99.9 % of the maximal growth rate can be achieved. Additionally, we defined protected reactions to keep the TCA cycle. For *H. pylori* 26695 one requirement is that at least 99.9 % of the maximal growth rate can be achieved. Additionally, we defined 28 protected reactions. A detailed discussion of this test case will be given in the next subsection. For *M. barkeri str. Fusaro* the requirements are that at least 99.9 % of the maximal growth rate with ammonia uptake and the maximal growth rate without ammonia uptake can be realized by the subnetwork. For *S. aureus* N315 the maximal biomass rate with glucose uptake and without glucose uptake has to be realized by the subnetwork.

| model | rxns | metS | ess rxns | rxns in SNW | metS in SNW | reps in SNW | time minNW | time minNW _{rep} | SNWs |
|--------------------------------------|------|------|-------------|-------------------|-------------------|-------------------|---------------|------------------------------|------|
| <i>Mus musculus</i> | 2436 | 1665 | 247 | 351 | 351 | 241 | 13.5 hrs | 50 min | 1 |
| <i>E. coli</i> iJO1366 | 2369 | 1159 | 363 | 562 | 601 | 262 | 12 min | 10 min | 1 |
| <i>S. Typhimurium</i> LT2 | 1620 | 1098 | 305 | 458 | 455 | 277 | 26 min | 25 min | 1 |
| <i>S. boydii</i> CDC 3083-94 | 1546 | 1019 | 441 | 445 | 450 | 209 | 4.5 hrs | 7.5 min | 1 |
| <i>K. pneumoniae</i> MGH 78578 | 1223 | 830 | 203 | 338 | 340 | 188 | 5 min | 3 min | 1 |
| <i>Y. pestis</i> CO92 | 1065 | 761 | 279 | 339 | 339 | 171 | 2 hrs | 1.75 hrs | 1 |
| <i>S. cerevisiae</i> S288c | 885 | 639 | 262 | 290 | 289 | 195 | 20 min | 12 min | 2 |
| <i>G. metallireducens</i> GS-15 | 845 | 710 | 544 | 557 | 567 | 153 | 4 min | 49 sec | 1 |
| <i>M. tuberculosis</i> iNJ661 | 1025 | 580 | 314 | 427 | 425 | 168 | 1 hr | 13 min | 1 |
| <i>B. subtilis</i> 168 | 658 | 500 | 270 | 296 | 300 | 134 | 4.75 hrs | 2.75 hrs | 1 |
| <i>P. putida</i> KT2440 | 652 | 539 | 300 | 344 | 348 | 116 | 1 hr | 14 min | 7 |
| <i>C. ljungdahlii</i> DSM 13528 | 526 | 448 | 369 | 383 | 389 | 118 | 26 sec | 7 sec | 44 |
| <i>H. pylori</i> 26695 | 501 | 381 | 265 | 321 | 323 | 89 | 9 sec | 9 sec | 16 |
| <i>M. barkeri</i> str. <i>Fusaro</i> | 484 | 417 | 289 | 364 | 369 | 90 | 25 sec | 24 sec | 20 |
| <i>S. aureus</i> iSB619 | 465 | 387 | 71 | 122 | 127 | 75 | 28 sec | 27 sec | 1 |
| <i>T. maritima</i> MSB8 | 385 | 331 | 267 | 282 | 280 | 87 | 14 sec | 5 sec | 28 |

Table 4.4: Computational results using indicator variables. **model**: name of the model. All models are from the BiGG Models database [King et al., 2016]. **rxns**: number of unblocked reactions in the original network. **metS**: number of metabolites in the original network after removing dead-end-metabolites. **ess rxns**: number of essential reactions in the original network. **rxns in SNW**: number of the reactions in the minimum subnetwork. **metS in SNW**: number of the metabolites in the minimum subnetwork. **reps in SNW**: number of the representatives in the minimum subnetwork. **time minNW**: running time for the algorithm minNW. **time minNW_{rep}**: running time for the algorithm minNW_{rep}. **SNWs**: number of minimum subnetworks which fulfil all the requirements. For detecting the running time, only one subnetwork was computed.

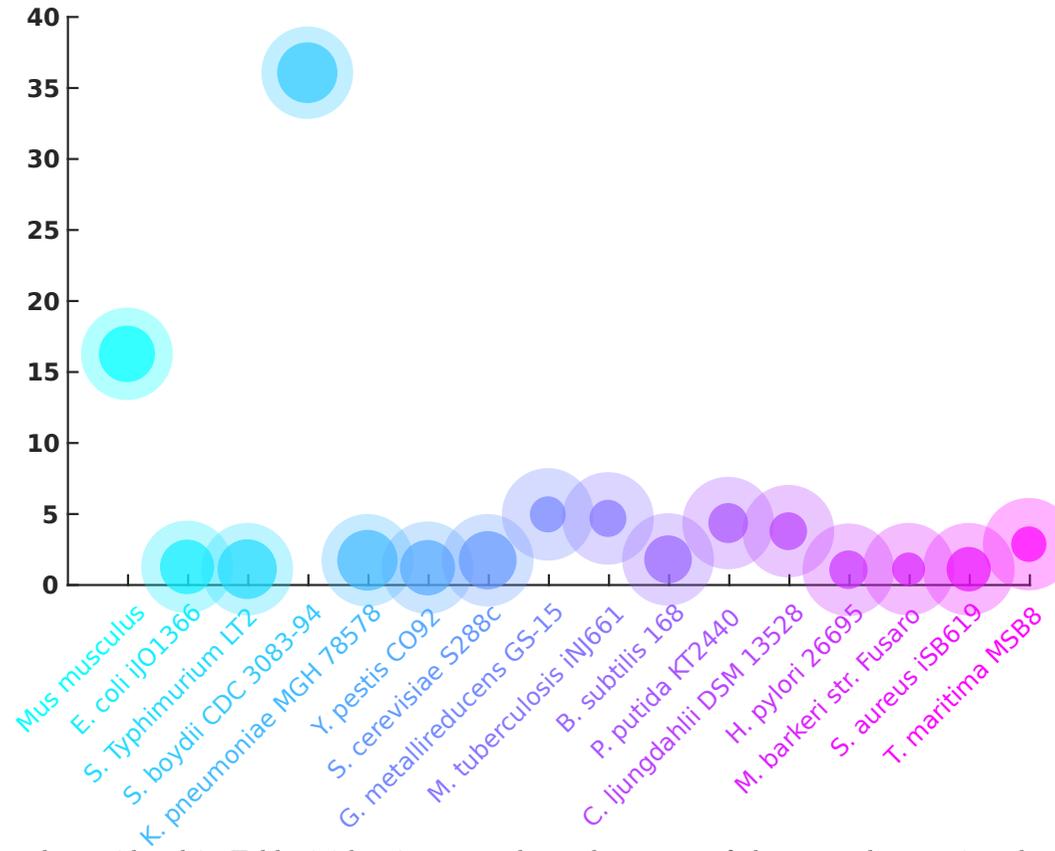


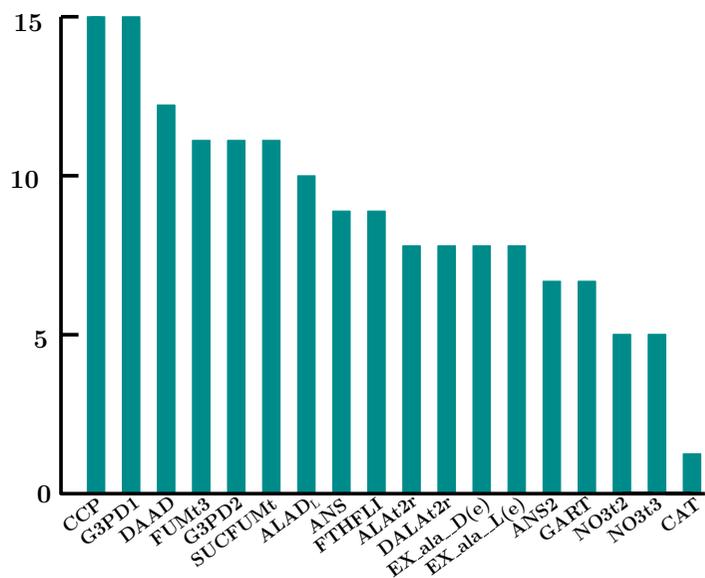
Figure 4.4: Each network considered in Table 4.4 has its own colour, the names of the networks are given by the x -axis. The y -axis shows the relative time needed to compute a subnetwork using indicator variables for each reaction or using indicator variables for each representative. Thus, since the centres of all circles are above 1 it is always faster to use representatives. The volume of the outer circle corresponds to the number of reactions in relation to the number of representatives, which is indicated by the volume of the inner circle: the volume of the outer circle is always one and the volume of the inner circle was computed correspondingly. For most networks it can be seen that the smaller the number of representatives is (the smaller the volume of the inner circle is), the faster it is to use representatives (the higher the circle is). The most advantage can be seen for the network *S. boydii* CDC 3083-94, where it is 36 times faster using representatives.

4.4.4 Case study: *Helicobacter pylori* 26695

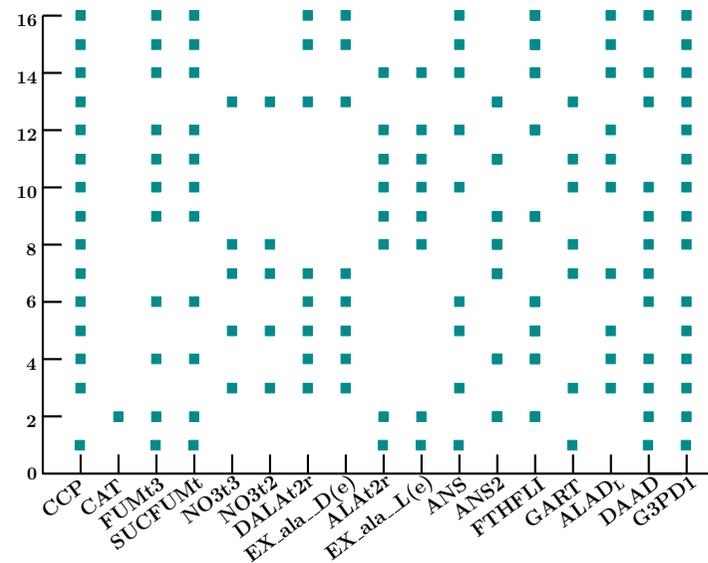
In this section we discuss the results for computing several minimum subnetworks for the metabolic network *H. pylori* 26695 using indicator variables. The requirements are the following:

1. There are 28 protected reactions.
2. The maximal biomass yield is 20.2606, and the subnetworks should be able to produce at least 99.9% of this yield.

In total we computed 16 subnetworks each containing 321 reactions, which is the minimum number needed to fulfil the requirements. The time needed to compute all these minimum subnetworks was 127 seconds with `minNW` and 33 seconds with `minNWrep`. Altogether the 16 minimum subnetworks use 329 different reactions, which can be found in the Appendix. 311 reactions are present in every subnetwork, among them all the 265 essential reactions of *H. pylori*. Only 18 reactions are not present in every subnetwork: CCP, G3PD1, D-Amino acid dehydrogenase, FUMt3, Glycerol-3-phosphate dehydrogenase (NADP), SUCFUMt, L-alanine dehydrogenase, Anthranilate synthase, Formate-tetrahydrofolate ligase, D-Alanine exchange, D-alanine transport via proton symport, L alanine reversible transport via proton symport, L-Alanine exchange, ANS2, GAR transformylase-T, NO3t2, NO3t3, Catalase. Figure 4.5 shows the distributions of these reactions in the 16 subnetworks.



(a) The x -axis gives the names of the reactions. The y -axis shows the number of subnetworks where the reaction is present.



(b) The x -axis again corresponds to the reactions. The y -axis corresponds to the subnetworks. A dot means that the reaction is present in the subnetwork.

Figure 4.5: The two illustrations show the distribution of the reactions which are not present in all subnetworks. In Figure 4.5a each reaction (x -axis) has a bar. The bar indicates in how many subnetworks the reaction can be found. For example, reaction CCP can be found in every subnetwork except 1 (there are in total 16 subnetworks) and reaction CAT can be found in only one subnetwork. Figure 4.5b illustrates where the reactions are found. Again the x -axis corresponds to the reactions. Thus a dot at (1, CCP) means that CCP appears in subnetwork 1. CCP can be found in every subnetwork except in the second one, whereas CAT can be found only in the second one.

Additional insight can be obtained by analysing co-occurrence patterns of the 18 non-essential reactions. Some of these reactions are mutually exclusive regarding the minimum subnetworks. For example, all subnetworks that contain reaction CCP do not contain CAT and vice versa. The same holds for the pair FTHFLi and GART, and the pair ANS and ANS2. Regarding the functionalities of these reaction pairs, one can easily check that the two reactions in each pair do basically the same. Therefore, it is sufficient if only one of them is present. In opposite to this, we can see that DALAt2r and EX_ala_D(e) form a cycle since they always appear in the same subnetworks. The same holds for ALAt2r and EX_ala_L(e). Both cycles also seem to be mutually exclusive, thus only one of them is present in the subnetworks. Similar observations can be made for the cycle formed by NO3t2 and NO3t3, which is mutually exclusive to the cycle formed by SUCFUMt and FUMt3.

One may ask whether the reactions that never appear together in the same subnetwork are also mutually exclusive regarding EFMs, i.e., whether or not there exists an EFM involving both reactions [Marashi and Bockmayr, 2011]. While this holds for the reaction pair FTHFLi and GART and the pair CCP and CAT, it is not true for the other reactions.

4.5 Conclusion

We presented in this chapter an MILP approach we developed to compute for a given large metabolic network one or more minimum subnetworks preserving biological requirements that can be specified by the user. Compared to previous work [Burgard et al., 2001; Vlassis et al., 2014; Erdrich et al., 2015], our method guarantees minimality of the subnetwork regarding the number of active reactions while preserving all the given requirements. In case there exist several minimal solutions, we are able to enumerate all of them. This may give additional insight into how the network is functioning and which reactions are really needed to satisfy the requirements. We applied our algorithms to several genome-scale metabolic networks and we always found the minimum subnetworks in reasonable time.

Once these subnetworks have been computed, further analysis becomes possible by using methods that are not applicable to the original network. For example, one may compute EFMs and MCSs. In addition, one can take a closer look at the reactions involved in one or all minimum subnetworks in order to get a better understanding of their role in the network.

4. Reduction of Networks

Chapter 5

Polyhedral Cones, Oriented Matroids, and Metabolic Networks

So far we only used MILP to analyse metabolic networks. MILP is a strong tool and can be applied to answer different questions regarding metabolic networks. However, for some questions MILP is not the right choice. Here, as in Chapter 3, we want to compute EFMs of a given network. Compared to the method in Chapter 3 we do not want to compute shortest EFMs but a subset of EFMs which can be used to represent every possible behaviour of the network. We start by considering only networks where all reactions are reversible. In Chapter 6 we extend our findings to metabolic networks, where irreversible reactions exist as well.

In this chapter, we recapitulate the connection between polyhedral cones, oriented matroids, and metabolic networks and present new results that strengthen this connection.

All definitions regarding metabolic networks, polyhedral cones and oriented matroids needed in the following can be found in the Sections 2.2, 2.3.1, and 2.3.5 respectively.

We give a first idea about the connection of metabolic networks, polyhedral cones, and oriented matroids: all three systems have minimum sets of generators. For a metabolic network a minimum subset of the elementary flux modes, see Definition 2.5, can be a generating system, for a polyhedral cone there exists a finite set of rays, see Definition 2.22, and for an oriented matroid the oriented fundamental circuits can be used, see Definition 2.60.

5. Cones, Matroids, Networks

We show the connection between these three sets and use new information to introduce a method on how to find a minimum set of elementary flux modes which fully describes the underlying metabolic network.

5.1 Oriented fundamental circuits and reaction splitting

To show the connections between oriented matroids, polyhedral cones and metabolic networks we consider here a network where all reactions are reversible. An illustration can be found in Figure 2.7. The flux cone $\Gamma_{\mathcal{N}}$ is described by the steady-state assumption only: $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0\}$, thus $\Gamma_{\mathcal{N}}$ is a linear subspace. The set of elements \mathcal{U} of the related flux mode matroid $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ is the set of columns of the stoichiometric matrix S , which we identify with the reactions \mathfrak{R} of the network \mathcal{N} , see Definition 2.53. Since all reactions are reversible it holds that $\Gamma_{\mathcal{N}} = \Lambda_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} = 0\}$.

According to Proposition 2.52, the tuple $(\mathfrak{R}, \mathcal{C})$ defines an oriented matroid, where $\mathcal{C} = \text{Min}(\mathcal{V})$ with $\mathcal{V} = \{\sigma(v) \mid Sv = 0, v \neq 0\}$. This oriented matroid is called the *flux mode matroid*, see Definition 2.53. Since the metabolic network considered in this chapter contains only reversible reactions there exists for each oriented cycle $X \in \mathcal{C}$ of \mathcal{M}_S a ray $v \in \Gamma_{\mathcal{N}}$ with $\sigma(v) = X$ and vice versa. The EFMs of \mathcal{N} are flux vectors consisting of a minimal set of active reactions, thus they are rays in $\Gamma_{\mathcal{N}}$ of minimal support. Oriented cycles in \mathcal{M}_S of minimal support are oriented circuits of \mathcal{M}_S , therefore there exists a bijection between the EFMs of \mathcal{N} and the oriented circuits of \mathcal{M}_S , see Table 5.1 for an overview. This connection is not true, if there exist irreversible reactions in \mathcal{N} , see Proposition 2.54. We discuss this issue in Chapter 6.

So far we have a connection between the flux vectors in a network \mathcal{N} , the rays in $\Gamma_{\mathcal{N}} = \Lambda_{\mathcal{N}}$, and the oriented cycles in \mathcal{M}_S . Furthermore we have seen the relationships between EFMs in \mathcal{N} , the rays of minimal support in $\Gamma_{\mathcal{N}}$, and the oriented circuits in \mathcal{M}_S .

| Metabolic Network | Flux cone | Oriented Matroid |
|-------------------|------------------------|------------------|
| Flux vector | Ray | Oriented cycle |
| EFM | Ray of minimal support | Oriented circuit |

Table 5.1: This table connects metabolic networks, polyhedral cones, and oriented matroids. The connections are only valid if all reactions in the metabolic network are reversible.

5.1 Oriented fundamental circuits and reaction splitting

In the following we consider a connection between the oriented fundamental circuits of \mathcal{M}_S and certain rays, resp. EFMs in \mathcal{N} . We use the network in Figure 2.7 for an illustration:

Example 5.1 (Flux cone and flux mode matroid if all reactions are reversible). *The stoichiometric matrix $S \in \mathbb{R}^{\mathfrak{m} \times \mathfrak{R}}$ of the network in Figure 2.7 has rank 7. As in Example 2.59, a basis for the corresponding flux mode matroid consists of seven reactions, e.g. $\mathcal{B} = \{1, 2, 3, 5, 6, 9, 11\}$. For each $i \in \mathfrak{R} \setminus \mathcal{B}$ exists one oriented fundamental circuit, see Example 2.61.*

To connect the oriented fundamental circuits to $\Gamma_{\mathcal{N}}$ we use reaction splitting, see Section 2.3.1.2 and 2.3.1.3. For this we first consider the lineality space of a polyhedral cone after splitting a subset of reactions. Generalizing a result from [Larhlimi and Bockmayr, 2008], we get:

Proposition 5.2. *Let $\Gamma \subseteq \mathbb{R}^n$ be a polyhedral cone with lineality space Λ . For a set of variables $I \subseteq \{1, \dots, n\}$, the lineality space $\tilde{\Lambda}^I$ of the reconfigured cone $\tilde{\Gamma}^I$ (see (2.2)) is given by:*

$$\tilde{\Lambda}^I = \left\{ \begin{pmatrix} x \\ 0 \end{pmatrix} \in \mathbb{R}^{n+|I|} \mid x \in \Lambda \text{ and } x_I = 0 \right\},$$

where $x_I = 0$ means $x_{i_k} = 0$ for all $i_k \in I$.

Proof. For the cone $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$ the lineality space is $\Lambda = \{x \in \mathbb{R}^n \mid Ax = 0\}$. Splitting the variables in I delivers the cone $\tilde{\Gamma}^I = \left\{ \begin{pmatrix} x \\ w \end{pmatrix} \in \mathbb{R}^{n+|I|} \mid (A, -A_{*,I}) \begin{pmatrix} x \\ w \end{pmatrix} \geq 0, x_I \geq 0, w \geq 0 \right\}$. For the lineality space we get:

$$\begin{aligned} \tilde{\Lambda}^I &= \left\{ \begin{pmatrix} x \\ w \end{pmatrix} \in \mathbb{R}^{n+|I|} \mid (A, -A_{*,I}) \begin{pmatrix} x \\ w \end{pmatrix} = 0, x_I = 0, w = 0 \right\} \\ &= \left\{ \begin{pmatrix} x \\ 0 \end{pmatrix} \in \mathbb{R}^{n+|I|} \mid (A, -A_{*,I}) \begin{pmatrix} x \\ 0 \end{pmatrix} = 0, x_I = 0 \right\} \\ &= \left\{ \begin{pmatrix} x \\ 0 \end{pmatrix} \in \mathbb{R}^{n+|I|} \mid Ax = 0, x_I = 0 \right\} \\ &= \left\{ \begin{pmatrix} x \\ 0 \end{pmatrix} \in \mathbb{R}^{n+|I|} \mid x \in \Lambda \text{ and } x_I = 0 \right\}. \end{aligned}$$

□

We can use this proposition to obtain a first result regarding the connection of the basis of a matroid and the corresponding polyhedral cone:

5. Cones, Matroids, Networks

Theorem 5.3. *Let $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ be a metabolic network where all reactions are reversible. Furthermore let $\Gamma_{\mathcal{N}}$ be the flux cone with $\Lambda_{\mathcal{N}} = \Gamma_{\mathcal{N}}$, where the dimension of $\Lambda_{\mathcal{N}}$ is t . Let \mathcal{M}_S be the corresponding flux mode matroid of \mathcal{N} and let \mathcal{B} be a basis of $\underline{\mathcal{M}}_S$. After splitting the reactions of the set $\mathfrak{R} \setminus \mathcal{B}$ the augmented polyhedral cone $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ is pointed.*

Proof. Let $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ be the flux mode matroid and $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0\}$ be the flux cone of \mathcal{N} with $\rho(S) = |\mathfrak{R}| - t$. Since all reactions are reversible it holds that $\Lambda_{\mathcal{N}} = \Gamma_{\mathcal{N}}$ with $\dim(\Lambda_{\mathcal{N}}) = t$.

Let \mathcal{B} be a basis of $\underline{\mathcal{M}}_S$. \mathcal{B} consists of $|\mathfrak{R}| - t$ linearly independent columns of S . Thus it holds that $\rho(S_{*,\mathcal{B}}) = |\mathcal{B}|$.

According to (2.2) we denote with $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ the flux cone derived from $\Gamma_{\mathcal{N}}$ by splitting the reactions of the set $\mathfrak{R} \setminus \mathcal{B}$. Since $\Lambda_{\mathcal{N}} = \Gamma_{\mathcal{N}}$ and by Proposition 5.2 the following holds:

$$\begin{aligned} \tilde{\Lambda}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}} &= \left\{ \begin{pmatrix} v \\ 0 \end{pmatrix} \in \mathbb{R}^{|\mathfrak{R}|+t} \mid v \in \Lambda_{\mathcal{N}}, v_{\mathfrak{R} \setminus \mathcal{B}} = 0 \right\} \\ &= \left\{ \begin{pmatrix} v \\ 0 \end{pmatrix} \in \mathbb{R}^{|\mathfrak{R}|+t} \mid v \in \Gamma_{\mathcal{N}}, v_{\mathfrak{R} \setminus \mathcal{B}} = 0 \right\} \\ &= \left\{ \begin{pmatrix} v \\ 0 \end{pmatrix} \in \mathbb{R}^{|\mathfrak{R}|+t} \mid Sv = 0, v_{\mathfrak{R} \setminus \mathcal{B}} = 0 \right\} \\ &= \left\{ \begin{pmatrix} v \\ 0 \end{pmatrix} \in \mathbb{R}^{|\mathfrak{R}|+t} \mid S_{*,\mathcal{B}}v_{\mathcal{B}} = 0, v_{\mathfrak{R} \setminus \mathcal{B}} = 0 \right\}. \end{aligned}$$

Because $\rho(S_{*,\mathcal{B}}) = |\mathcal{B}|$ it follows for all $v \in \Gamma_{\mathcal{N}}$ with $S_{*,\mathcal{B}}v_{\mathcal{B}} = 0$ that $v_{\mathcal{B}} = 0$ and therefore $\tilde{\Lambda}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}} = \{0\}$. Thus $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ is a pointed polyhedral cone. \square

Since $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ is pointed it can be completely described by its extreme rays. It is shown in [Larhlimi and Bockmayr, 2009] that there exists a 1-1-correspondence between the minimal proper faces (MPFs), see Definition 2.20, of a flux cone and the minimal metabolic behaviours (MMBs), see Definition 2.7. If a polyhedral cone is pointed, the MPFs are the set of ERs. From this we can deduce the following proposition:

Proposition 5.4. *Let $\Gamma_{\mathcal{N}}$ be a pointed flux cone of a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. The ERs of $\Gamma_{\mathcal{N}}$ are equivalent to the flux vectors of $\Gamma_{\mathcal{N}}$ with minimal support w.r.t. the irreversible reactions.*

We denote the support w.r.t the irreversible reactions with supp_{Irr} , thus for $v \in \Gamma_{\mathcal{N}}$ we have $\text{supp}_{\text{Irr}}(v) = \{i \mid i \in \text{Irr}, v_i \neq 0\}$. Note that $\text{supp}_{\text{Irr}}(v)$ is the metabolic behaviour of v .

5.1 Oriented fundamental circuits and reaction splitting

Using Proposition 5.4 we can consider the pointed polyhedral cone $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$, its ERs, and its connection to \mathcal{M}_S more closely:

Theorem 5.5. *Let $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ be a metabolic network where all reactions are reversible with the flux cone $\Gamma_{\mathcal{N}} = \Lambda_{\mathcal{N}}$ and $\dim(\Lambda_{\mathcal{N}}) = t$. Let \mathcal{M}_S be the corresponding flux mode matroid of \mathcal{N} and \mathcal{B} an arbitrary but fixed basis of $\underline{\mathcal{M}}_S$. Let $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ be the pointed polyhedral cone derived by splitting the reactions of the set $\mathfrak{R} \setminus \mathcal{B}$ in $\Gamma_{\mathcal{N}}$. The sign vectors of the ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ are, after recombination, the oriented fundamental circuits of \mathcal{M}_S (w.r.t. the set $\mathcal{B} \subseteq \mathfrak{R}$).*

Proof. Let \mathcal{B} be an arbitrary but fixed basis of $\underline{\mathcal{M}}_S$ and $t = |\mathfrak{R} \setminus \mathcal{B}|$. For each $i \in \mathfrak{R} \setminus \mathcal{B}$ there exists a fundamental circuit $X \in \underline{\mathcal{M}}_S$ with $\text{supp}(X) \subseteq \mathcal{B} \cup \{i\}$, see Section 2.3.5.2. It holds that $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}} = \left\{ \begin{pmatrix} v \\ x \end{pmatrix} \in \mathbb{R}^{|\mathfrak{R}|+t} \mid (S| - S_{*, \mathfrak{R} \setminus \mathcal{B}}) \begin{pmatrix} v \\ x \end{pmatrix} = 0, v_{\mathfrak{R} \setminus \mathcal{B}} \geq 0, x \geq 0 \right\}$. Since $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ is pointed, the ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ are the rays of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ which have minimal support w.r.t. the set of irreversible reactions, which is the set $\mathfrak{R} \setminus \mathcal{B}$ in v and all entries of x . For $v \in \Gamma_{\mathcal{N}}$ we have $\pi_{\mathfrak{R} \setminus \mathcal{B}}(v) = \tilde{v}$ with $\tilde{v}_j = v_j$ for all $j \in \mathcal{B}$ and for each $i_k \in \mathfrak{R} \setminus \mathcal{B}$ we have $\tilde{v}_{i_k} = v_{i_k}$ and $\tilde{v}_{n+k} = 0$, if $v_{i_k} \geq 0$ or $\tilde{v}_{i_k} = 0$ and $\tilde{v}_{n+k} = -v_{i_k}$ if $v_{i_k} < 0$, see Definition 2.29 and Equation (2.1).

What we need to show is:

- i) for each ER $\tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ it holds that $X = \sigma(v)$ for $v = \pi_{\mathfrak{R} \setminus \mathcal{B}}^r(\tilde{v})$ is an oriented fundamental circuit of \mathcal{M}_S w.r.t. \mathcal{B} and
- ii) for each oriented fundamental circuit X w.r.t. \mathcal{B} of \mathcal{M}_S with $X = \sigma(v)$ and $v \in \Gamma_{\mathcal{N}}$ the ray $\tilde{v} = \pi_{\mathfrak{R} \setminus \mathcal{B}}(v)$ is an ER of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$.

We denote with $C(i, \mathcal{B})$ the oriented fundamental circuit X of \mathcal{M}_S with $\text{supp}(X) \subseteq \mathcal{B} \cup \{i\}$, where $i \in \mathfrak{R} \setminus \mathcal{B}$.

- i) All ERs $\tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}} \setminus \{0\}$ have minimal support w.r.t. the irreversible reactions. In $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ the irreversible reactions are the split reactions thus the set $\mathfrak{R} \setminus \mathcal{B}$ and their split counterparts. It holds that $|\text{supp}_{\text{Irr}}(\tilde{v})| = 1$: Since \mathcal{B} is a basis of \mathcal{M} it holds that $\rho(S_{*, \mathcal{B}}) = |\mathcal{B}| = |\mathfrak{R}| - t = \rho(S)$. For each $i \in \mathfrak{R} \setminus \mathcal{B}$ the column $S_{*, i}$ is therefore linearly dependent on the columns of $S_{*, \mathcal{B}}$. This implies that for each $i \in \mathfrak{R} \setminus \mathcal{B}$ there exists a ray $v \in \Gamma_{\mathcal{N}}$ with $\tilde{v} = \pi_{\mathfrak{R} \setminus \mathcal{B}}(v)$ and $\text{supp}_{\text{Irr}}(\tilde{v}) = i$. The rays with this property are ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ since they have minimal support w.r.t. the irreversible reactions of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$. This implies that for each ER $\tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ it holds for $v = \pi_{\mathfrak{R} \setminus \mathcal{B}}^r(\tilde{v})$ that $\sigma(v) = C(i, \mathcal{B})$, thus $X = \sigma(v)$ is an oriented fundamental circuit of \mathcal{M}_S w.r.t. \mathcal{B} .

5. Cones, Matroids, Networks

ii) For all $i \in \mathfrak{R} \setminus \mathcal{B}$ there exists a ray $v \in \Gamma_{\mathcal{N}}$ with $\sigma(v) = C(i, \mathcal{B})$. According to Definition 2.29 we have that $\pi_{\mathfrak{R} \setminus \mathcal{B}}(v) = \tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$. It holds that $\text{supp}_{\text{Irr}}(\tilde{v}) = \{i\}$. Thus all rays $\tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}} \setminus \{0\}$ corresponding to a fundamental circuit of \mathcal{M}_S w.r.t. \mathcal{B} have minimal support w.r.t. the irreversible reactions in $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ and are therefore ER in $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$, see Proposition 5.4. \square

We give now an intuition to the meaning of Theorem 5.5. Note again that the cone $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$, and in particular the set of ERs, does not have to be unique. The set of ERs depends on $\mathcal{B} \subseteq \mathfrak{R}$, a basis of the matroid $\underline{\mathcal{M}}_S$. Depending on \mathcal{B} we have a set of oriented fundamental circuits of the oriented matroid \mathcal{M}_S which correspond to the sign vectors of the recombined ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$.

Example 5.6 (Oriented fundamental circuits correspond to extreme rays). *Lets illustrate Theorem 5.5 by the network in Figure 2.7, with $\mathfrak{M} = \{A, \dots, G\}$ and $\mathfrak{R} = \{1, \dots, 12\}$. From Example 2.59 we know that one basis for the network is given by $\mathcal{B} = \{1, 2, 3, 5, 6, 9, 11\}$, since $\rho(S_{*, \mathcal{B}}) = \rho(S) = 7$, where $S \in \mathbb{R}^{7 \times 12}$ is the stoichiometric matrix of the network. There exist five oriented fundamental circuits w.r.t. \mathcal{B} . The five non-basis elements are $\mathfrak{R} \setminus \mathcal{B} = \{4, 7, 8, 10, 12\}$. In Figure 5.1 the non-basis reactions are split and no reversible flux vector exists. Therefore the corresponding polyhedral cone $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ is pointed. The recombined ERs of the augmented polyhedral cone are $\{+1, +2, +3, +4\}$, $\{+1, +2, +3, +6, +7\}$, $\{+1, +2, +3, -5, +6, +8\}$, $\{-3, +5, +9, +10\}$, $\{-3, +5, +11, +12\}$ and the reverse of each ray. The sign vectors of the ERs of this cone are, after recombination, the oriented fundamental circuits of the flux mode matroid $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ w.r.t. the basis \mathcal{B} . Another basis for the network in Figure 2.7 is given by $\mathcal{B} = \{1, 2, 4, 7, 8, 10, 11\}$. The rank of $S_{*, \mathcal{B}}$ is again seven. There exist again five oriented fundamental circuits w.r.t. \mathcal{B} and splitting the reactions of $\mathfrak{R} \setminus \mathcal{B}$ delivers a pointed polyhedral cone $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ where the ERs correspond to the oriented fundamental circuits, see Figure 5.2. Note that the 2-cycles (see Definition 2.33), i.e., the vectors consisting of two directions of a split reaction, e.g. $\{+4, -4\}$, are feasible flux vectors in $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$, but no ERs. This can be seen by applying the rank test, i.e., Theorem 2.25.*

Note that the ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ describe $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ fully, thus they can be used to generate every ray of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$. After recombination they are EFMs of the network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. Since every ray $\tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ (except the 2-cycles) corresponds to a ray in $\Gamma_{\mathcal{N}}$ (and vice versa) and therefore to a flux vector of \mathcal{N} , the set of recombined ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ is a subset of EFMs of \mathcal{N} which can be used to generate every other flux vector in \mathcal{N} .

5.1 Oriented fundamental circuits and reaction splitting

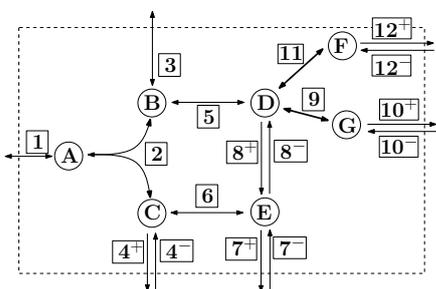


Figure 5.1: A basis of the underlying matroid \mathcal{M}_S is $\mathcal{B} = \{1, 2, 3, 5, 6, 9, 11\}$. In the network the reactions of $\mathfrak{R} \setminus \mathcal{B}$ are split. There exists no reversible flux vector anymore.

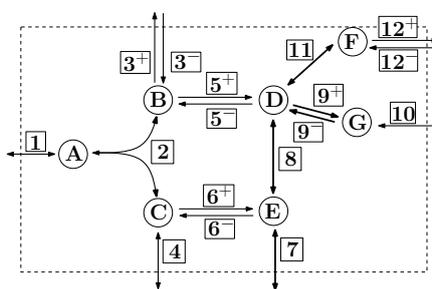


Figure 5.2: A different basis of the underlying matroid \mathcal{M}_S is $\mathcal{B} = \{1, 2, 4, 7, 8, 10, 11\}$. Again, the reactions of $\mathfrak{R} \setminus \mathcal{B}$ are split and there exists no reversible flux vector in the modified network.

Figure 5.3: Both networks are the same as in Figure 2.7 except that the non-basis elements were split w.r.t. the given bases. In both examples there exists no feasible reversible flux vector. Thus the corresponding polyhedral cones $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ are pointed. For both bases the cones $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ have ten different ERs which correspond, after recombination, to five EFMs (up to orientation) of the original network and which are the oriented fundamental circuits of \mathcal{M}_S w.r.t the given basis.

5.2 Computing a basis and oriented fundamental circuits

To compute a basis we need an important concept from matrix theory [Gilbert and Gilbert, 2014, p.85]. A matrix $B^{\text{rcef}} \in \mathbb{R}^{m \times n}$ is in *reduced column echelon form* if the following properties hold (see Figure 5.4):

1. The first non-zero element in column k is a 1 in row j_k , for $k = 1, 2, \dots, r$ (this 1 is called a *pivot*).
2. $1 \leq j_1 < j_2 < \dots < j_r \leq m$ (i.e., for each change in columns from left to right, the pivot appears in a lower row).
3. For $k = 1, \dots, r$, the pivot in column k is the only non-zero element in row j_k .
4. Each of the last $n - r$ columns consists entirely of zeros.

To compute a basis of a matrix A we can transform A to the reduced row echelon form (which is the transpose of the column reduced echelon form, see Figure 5.4). The pivot elements then refer to the columns of A which define the basis.

We give now the algorithm from [Khachiyan et al., 2005] to find fundamental circuits. We can use this algorithm to compute all *oriented* fundamental circuits, because every fundamental circuit \underline{X} has exactly two possible orien-

$$B^{\text{rcef}} = \begin{pmatrix} \textcircled{1} & 0 & 0 & 0 & 0 & 0 \\ * & 0 & 0 & 0 & 0 & 0 \\ 0 & \textcircled{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & \textcircled{1} & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 \\ * & * & * & 0 & 0 & 0 \\ 0 & 0 & 0 & \textcircled{1} & 0 & 0 \\ * & * & * & * & 0 & 0 \end{pmatrix} \begin{matrix} \leftarrow j_1 \\ \\ \leftarrow j_2 \\ \leftarrow j_3 \\ \vdots \\ \leftarrow j_r \end{matrix}$$

Figure 5.4: B^{rcef} is a matrix in reduced column echelon form, where $*$ are appropriate values in \mathbb{R} .

tations, X and $-X$, and leads therefore two oriented fundamental circuits.

Algorithm 1: Compute Oriented Fundamental Circuits

Input : Matrix A which represents the oriented matroid $\mathcal{M} = (U, ?)$
where U is the set of columns of A

Output: Fundamental circuits of \mathcal{M}

```

1  $B^{\text{rref}} \leftarrow \text{ReducedRowEchelonForm}(A)$  ;
  /* Compute the reduced row echelon form of  $A$  */
2  $\mathcal{B} \leftarrow$  basis of  $\underline{\mathcal{M}}$ ;
  /*  $\mathcal{B}$  are the pivot elements of  $B^{\text{rref}}$  */
3 for  $i \in U \setminus \mathcal{B}$  do
4    $X^i \leftarrow C(i, \mathcal{B})$ ;
   /* unique oriented circuit contained in  $\mathcal{B} \cup i$ ,
      e.g. computed solving  $A_{*, \mathcal{B} \cup \{i\}} x = 0$  (since  $\mathcal{B}$  is a
      basis there exists a unique solution), with
       $X^i = \sigma(x)$  */
5 end
6  $\mathcal{F} \leftarrow \bigcup_i \pm X^i$ ;
  /*  $\mathcal{F}$  is the set of oriented fundamental circuits of  $\mathcal{M}$ 
      w.r.t.  $\mathcal{B}$  */
7 return  $\mathcal{F}$ ;

```

If an oriented matroid \mathcal{M} is represented by a matrix $A \in \mathbb{R}^{m \times n}$, with $n \geq m$, and $\rho(A) = m$ then there exist $n - m$ oriented fundamental circuits w.r.t. a given basis. To compute all oriented fundamental circuits w.r.t. a given basis $n - m$ LPs have to be solved, thus it can be done in polynomial time.

5.3 Different bases

Let $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ be a metabolic network, where all reactions are reversible and let \mathcal{M}_S be its underlying flux mode matroid. Let furthermore $\Gamma_{\mathcal{N}}$ be the corresponding flux cone and $\Lambda_{\mathcal{N}}$ its lineality space, where the dimension of $\Lambda_{\mathcal{N}}$ is t . Note that it holds that $\Gamma_{\mathcal{N}} = \Lambda_{\mathcal{N}}$. A basis \mathcal{B} of \mathcal{M}_S consists always of t columns of S , but is not unique. So one question is, how many different bases exist for a given network? A trivial upper bound would be the number of every combination of t different reactions, thus $\binom{|\mathfrak{R}|}{t} = \frac{|\mathfrak{R}| (|\mathfrak{R}| - 1) \cdots (|\mathfrak{R}| - t + 1)}{t(t - 1) \cdots 1}$, where $|\mathfrak{R}|$ is the number of reactions of the network \mathcal{N} , where all reactions are reversible. But this bound is not exact as can be seen in Figure 5.5.

5. Cones, Matroids, Networks

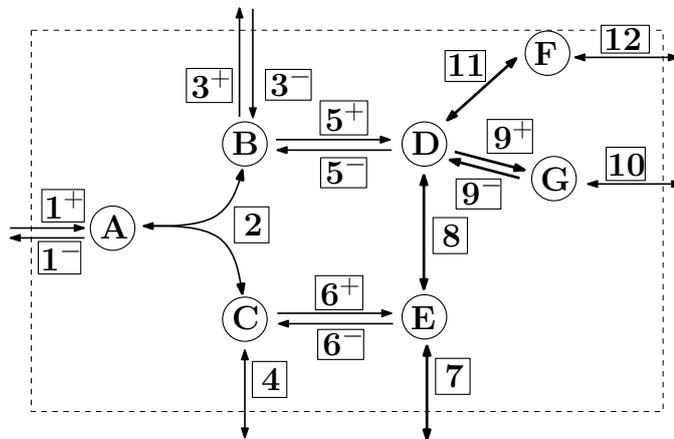


Figure 5.5: The stoichiometric matrix of the original network, where no reaction is split (see Figure 2.7) has a rank of seven. But not all sets of seven reactions form a basis \mathcal{B} of the underlying flux mode matroid \mathcal{M}_S . For example the reactions $\mathcal{B} = \{2, 4, 7, 8, 10, 11, 12\}$ are not a basis, since the reactions $\{7, 8, 11, 12\} \subset \mathcal{B}$ form a feasible reversible flux vector.

Example 5.7 (Not all sets of t reactions form a basis). The dimension of $\Lambda_{\mathcal{N}}$ for the network of Figure 2.7 is seven. We could choose $\mathcal{B} = \{2, 4, 7, 8, 10, 11, 12\}$, thus $\mathfrak{R} \setminus \mathcal{B} = \{1, 3, 5, 6, 9\}$. But the polyhedral cone $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$ is not pointed as can be seen in Figure 5.5, where the reactions of $\mathfrak{R} \setminus \mathcal{B}$ are split. The reactions $\{7, 8, 11, 12\}$ can still form a reversible flux vector. Furthermore $\rho(S_{*, \mathcal{B}}) = 6$ and $\rho(S) = 7$, thus \mathcal{B} is not a basis of $\underline{\mathcal{M}}_S$.

To compute an exact bound for a given network, we can consider every set of t reactions and check if they correspond to a basis \mathcal{B} . This can be done by computing the *Tutte-polynomial* [Welsh, 1999] of the matrix S .

Definition 5.8 (Tutte Polynomial, [Welsh, 1999]). Let $\underline{\mathcal{M}} = (U, \underline{\mathcal{C}})$ be a matroid represented by a matrix A . The Tutte polynomial for two variables x, y is

$$T(\underline{\mathcal{M}}; x, y) = \sum_{F \subseteq U} (x-1)^{\rho(U)-\rho(F)} (y-1)^{|F|-\rho(F)},$$

where $\rho(F)$ is the maximum size of a linearly independent set in F .

The number of different bases for a matroid $\underline{\mathcal{M}}$ is given by $T(\underline{\mathcal{M}}; 1, 1)$, [Jaeger et al., 1990]. For $x = 1 = y$ most of the terms of the Tutte-polynomial equal zero, except for those cases where the exponent is zero as well. This is the case for the terms $(x-1)^{\rho(U)-\rho(F)}$ if $\rho(U) = \rho(F)$ and for the term $(y-1)^{|F|-\rho(F)}$, if $|F| = \rho(F)$, thus if F is a basis for $\underline{\mathcal{M}}$. For example the Tutte-polynomial for the network in Figure 2.7 is:

$x^7 + 5x^6 + x^5y + 14x^5 + 11x^4y + 8x^3y^2 + 5x^2y^3 + 3xy^4 + y^5 + 23x^4 + 26x^3y + 21x^2y^2 + 13xy^3 + 4y^4 + 24x^3 + 31x^2y + 22xy^2 + 7y^3 + 15x^2 + 18xy + 7y^2 + 4x + 4y$.
 For $x = 1 = y$ this equals to 268 which is exactly the number of different bases.

We used the **SAGE** toolbox [The Sage Developers, 2016] to compute the Tutte-polynomial. We applied this to the reversible part of metabolic networks in the next chapter, see Table 6.2.

Using the Tutte-polynomial gives an exact bound for the number of different bases of the matroid and therefore for the number of different sets of $\mathfrak{R} \setminus \mathcal{B}$, a minimum set of reactions which can be split to obtain an augmented pointed polyhedral cone $\tilde{\Gamma}_{\mathcal{N}}^{\mathfrak{R} \setminus \mathcal{B}}$. The drawback is that computing the Tutte-polynomial to determine the number of different bases is computationally very expensive. To compute the Tutte-polynomial every subset of the columns of S is considered and the rank of those subsets is computed. To compute the number of different bases it would be sufficient to compute the rank of all subsets of S which have a cardinality of $\rho(S)$, thus $|\mathfrak{R}| - t$.

5.4 Conclusion

In this chapter we considered the relation between polyhedral cones and metabolic networks. The flux cone, a polyhedral cone, is an appropriate description of the underlying metabolic network. A polyhedral cone is always finitely generated and so are metabolic networks, i.e., the number of EFMs is finite. The elementary flux modes are a widely known generating set for metabolic networks, although they are a superset of flux vectors needed to generate every possible behaviour of a network.

If a polyhedral cone is pointed the minimum generating set is unique: the extreme rays. The flux cone is pointed if no reversible flux vectors exist.

To get an augmented pointed flux cone one can split the whole set of reversible reactions. The extreme rays are, after recombination, the whole set of elementary flux modes. But it is in general sufficient to split only a subset of the reversible reactions. If the dimension of the lineality space of the non-pointed flux cone is t , it is sufficient to split a certain subset of t reactions.

In this chapter we considered the case where all reactions are reversible. What elementary flux modes are for metabolic networks, the oriented circuits are for oriented matroids. A minimum set of oriented circuits which

5. Cones, Matroids, Networks

can be used to compute all remaining oriented circuits are the oriented fundamental circuits. These oriented circuits are not unique, but depend on the corresponding matroid bases. But there exists a bijection between these oriented fundamental circuits and the extreme rays of the augmented pointed cone which is derived by splitting a minimum set of reversible reactions.

How the results of this chapter can be generalised to metabolic networks including irreversible reactions will be discussed in the following chapter.

Chapter 6

Finding MEMo

A preliminary version of this chapter has been done in collaboration with Alexander Bockmayr and appeared as a conference paper in *Proceedings of the Lyon Spring School on advances in Systems and Synthetic Biology* under [Röhl and Bockmayr, 2017], available at <https://assb.lri.fr/Proceedings/LivreLyon-17.pdf>. A long journal version is currently under preparation.

As emphasised in the last chapter, if all reactions of a metabolic network $\mathcal{N} = (\mathcal{M}, \mathfrak{R}, S, \text{Irr})$ are reversible, there is a direct connection between the flux cone $\Gamma_{\mathcal{N}}$ and the flux mode matroid \mathcal{M}_S . The topic of this chapter will be the general case, where irreversible reactions are involved.

The flux cone $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ is still a complete and suitable description of the underlying network, whereas the flux mode matroid $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ is a superset of the stoichiometric network: all circuits $X \in \mathcal{C}$ with $X^i = -$ for some $i \in \text{Irr}$ are sign vectors of non-feasible vectors regarding the metabolic network. But we can use the knowledge gained in Chapter 5: We will present a method based on reaction splitting to compute minimum sets of EFMs (MEMo) that generate the whole flux space.

It is often impossible to compute the set of all EFMs, due to their huge number. From a practical point of view, this might also not be necessary because a subset of EFMs may already be sufficient to answer relevant biological questions. The number of EFMs in a MEMo may be by several orders of magnitude smaller than the total number of EFMs. Using MEMos, we can compute generating sets of EFMs in metabolic networks where the whole set of EFMs is too large to be enumerated.

6. Finding MEMo

6.1 Introduction

While the EFMs are of great theoretical and practical interest, already for medium-sized metabolic networks it is often not possible to enumerate all of them, since their number grows exponentially with the size of the network. Thus, over the last 20 years, various methods have been developed to compute some or all EFMs in a given metabolic network, see e.g. [Fukuda and Prodon, 1996; Pfeiffer et al., 1999; Gagneur and Klamt, 2004; Urbanczik and Wagner, 2005a; De Figueiredo et al., 2009; Terzer, 2009b; Rezola et al., 2011; David and Bockmayr, 2014; Pey and Planes, 2014].

In this chapter, we study minimum sets of EFMs that generate the full flux cone $\Gamma_{\mathcal{N}}$. We study their mathematical properties and develop an algorithm to compute them. We start with a formal definition.

Definition 6.1 (MEMo). *Let $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ be a metabolic network with flux cone $\Gamma_{\mathcal{N}}$ and set of elementary flux modes $\mathcal{E}_{\mathcal{N}}$. A **Minimum set of Elementary Modes** or MEMo is a minimum set $\mathcal{U} \subseteq \mathcal{E}_{\mathcal{N}}$ such that every $v \in \Gamma_{\mathcal{N}}$ can be represented as a linear combination*

$$v = \sum_{e \in \mathcal{U} \cap \mathcal{E}_{\mathcal{N}}^{\text{Rev}}} \lambda_e e + \sum_{f \in \mathcal{U} \cap \mathcal{E}_{\mathcal{N}}^{\text{Irr}}} \lambda_f f,$$

for some $\lambda_e, \lambda_f \in \mathbb{R}$, with $\lambda_f \geq 0$, for all irreversible $f \in \mathcal{U} \cap \mathcal{E}_{\mathcal{N}}^{\text{Irr}}$.

A MEMo \mathcal{U} can be used to represent every other flux vector as a conical combination, in particular all EFMs not belonging to \mathcal{U} . The minimum number of EFMs needed to represent the whole flux cone is in general much smaller than the number of all EFMs. Therefore, it may be possible to compute a MEMo in a reasonable amount of time even for large networks where the set of all EFMs can not be enumerated.

6.1.1 Intuition

We begin by providing some intuition on how to compute MEMos using the example in Figure 1.1. As emphasised in Section 2.3.1 the flux cone $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ is a *finitely generated* polyhedral cone. If the polyhedral cone is *pointed* there exists a unique minimal set of generators, the *extreme rays* (ERs). In general, flux cones of metabolic networks are *non-pointed* because they contain *reversible flux vectors* $v \neq 0$ for which both $v, -v \in \Gamma_{\mathcal{N}}$.

Following [Larhlimi and Bockmayr, 2009] we call reversible reactions, which can carry flux in a reversible flux vector, *fully reversible reactions*.

Definition 6.2 (Fully reversible reaction). *Let $\Gamma_{\mathcal{N}}$ be the flux cone for a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. A reversible reaction $i \in \text{Rev}$ is called fully reversible if there exists a flux vector $v \in \Gamma_{\mathcal{N}}$ with $v_i \neq 0$ and $v_{\text{Irr}} = 0$, thus if i is active in a reversible flux vector. Otherwise i is called pseudo-irreversible. We will denote the set of fully reversible reactions of a network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ by $\text{Frev} \subseteq \text{Rev}$.*

Example 6.3 (Reversible flux vectors and fully reversible reactions). *The network in Figure 1.1 contains flux vectors that consist of reversible reactions only. For example, $v^\top = (0, 0, -1, 0, 1, 0, 0, 0, 1, 1, 0, 0)$ is a vector with support $\{3, 5, 9, 10\}$ and can operate in both directions. Thus, the reactions 3, 5, 9, and 10 are fully reversible and v^\top is a reversible flux vector. In addition, reactions 11 and 12 of the network in Figure 1.1 are fully reversible as well. On the other hand, reaction 4 is reversible, but not fully reversible: Whenever reaction 4 is active at least one irreversible reaction is active as well.*

As mentioned before, the set $\mathcal{E}_{\mathcal{N}}$ of EFMs is a finite generating set of the flux cone $\Gamma_{\mathcal{N}}$. However, $\mathcal{E}_{\mathcal{N}}$ need not to be minimal. A well-known method to compute the set of all EFMs consists in reducing the lineality space $\Lambda_{\mathcal{N}}$ to $\{0\}$ by splitting all reversible reactions in \mathcal{N} , leading to a new network $\tilde{\mathcal{N}}$ and a pointed flux cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{Rev}}$, see Section 2.3.1.3 for more details and examples. The ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\text{Rev}}$ are the extreme currents (EC) of \mathcal{N} and are the whole set of EFMs of \mathcal{N} after recombination.

However, the set of $\mathcal{E}_{\mathcal{N}}$ of EFMs need not be minimal for $\Gamma_{\mathcal{N}}$.

Example 6.4 (Not all EFMs are needed). *For the network in Figure 1.1 the set of reactions $\{1, 2, 3, 4\}$ is the support of the EFM $v^\top = (1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0)$. The reactions $\{3, 5, 11, 12\}$ are the support of an EFM as well $w^\top = (0, 0, -1, 0, 1, 0, 0, 0, 0, 0, 1, 1)$. Note that reaction 3 has a positive flux of 1 in v and a negative flux of -1 in w . The sum of v and w generates another EFM, $u^\top = (1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1)$, consisting of the reactions $\{1, 2, 4, 5, 11, 12\}$. Thus, the EFM with the support $\{1, 2, 4, 5, 11, 12\}$ is redundant and not required in a minimum generating set of EFMs for the network in Figure 1.1.*

In this chapter, we present a method for finding a minimum number of reversible reactions such that after splitting these reactions no feasible reversible flux vector exists in the modified network $\tilde{\mathcal{N}}$. Therefore, the modified flux cone $\Gamma_{\tilde{\mathcal{N}}}$ is pointed. We will show that the ERs of $\Gamma_{\tilde{\mathcal{N}}}$ correspond to a MEMo of the original network \mathcal{N} , c.f. Definition 6.1. The method is based on the idea presented in the last Chapter 5, where the special case was considered that all reactions of \mathcal{N} are reversible. The method introduced in

6. Finding MEMo

Chapter 5 is applied to the lineality space $\Lambda_{\mathcal{N}}$ in order to transfer the idea to metabolic networks including irreversible reactions.

Example 6.5 (Splitting a minimum set of reversible reactions). *For the network in Figure 1.1, it is sufficient to split only two reversible reactions, for example 3 and 9, see Figure 6.1, to obtain a modified flux cone $\Gamma_{\tilde{\mathcal{N}}}$ which is pointed. There are 5 ERs of $\Gamma_{\tilde{\mathcal{N}}}$, which correspond to 5 EFMs in \mathcal{N} . Thus, only 5 instead of 18 EFMs are needed to generate $\Gamma_{\mathcal{N}}$. The minimum set of reversible reactions to split is not unique. Instead of splitting 3 and 9 as in Figure 6.1 one could also split the reactions 10 and 12, as in Figure 6.2. Again, we obtain a MEMo containing 5 different EFMs, which however are different from the 5 EFMs in the MEMo obtained by splitting the reactions 3 and 9. Finally, not any set of two reversible reactions can be split to obtain a pointed cone. For example, splitting reactions 11 and 12 will not eliminate all reversible vectors, as shown in Figure 6.3.*

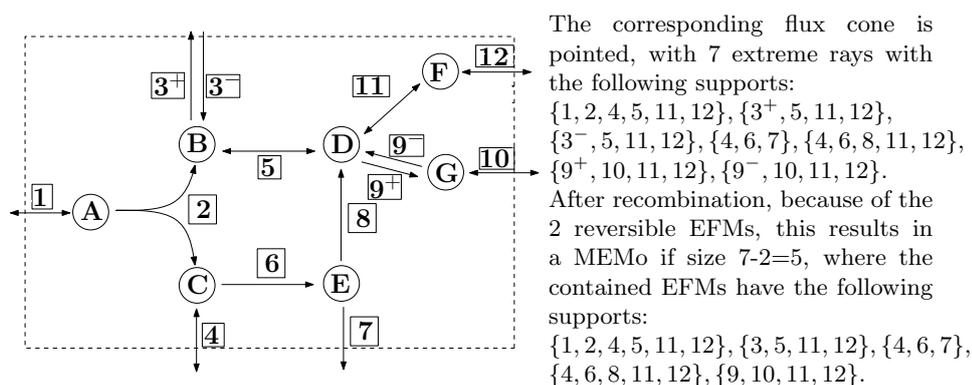


Figure 6.1: The fully reversible reactions 3 and 9 are split, leading to a pointed flux cone for the new network. The transformed network does not contain any reversible flux vectors.

6.1.2 Contributions

The contributions of this chapter are the following:

- We prove that whenever a (sub-)set of reversible reactions is split and the corresponding augmented flux cone is pointed, the extreme rays of the pointed augmented flux cone correspond to EFMs of the original network (Theorem 6.7).
- We introduce a method and provide an implementation for finding a minimum number of reversible reactions such that after splitting these

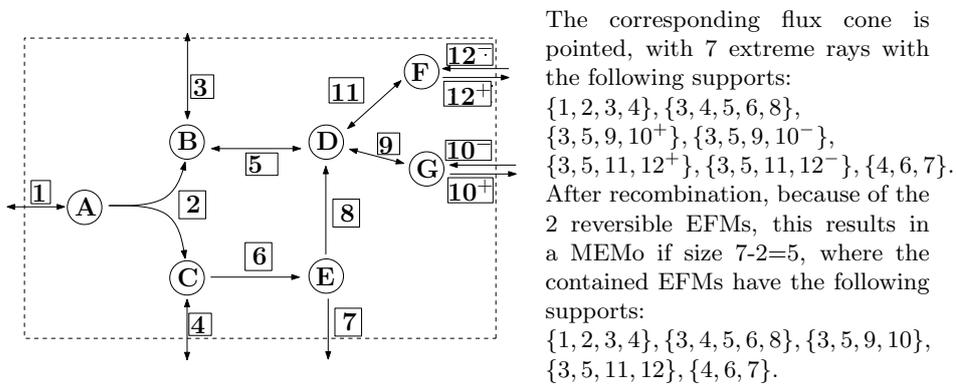


Figure 6.2: The fully reversible reactions 10 and 12 are split, leading to a pointed flux cone for the new network. The transformed network does not contain any reversible flux vectors.

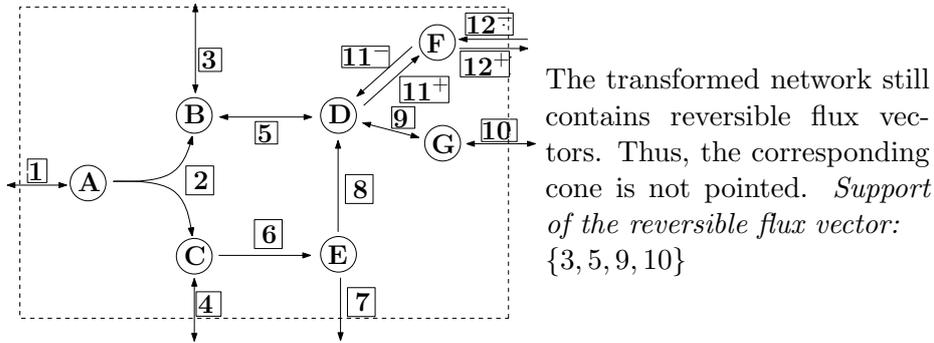


Figure 6.3: The fully reversible reactions 11 and 12 are split.

reactions the resulting augmented flux cone is pointed. Additionally, we prove the correctness of the method (Theorem 6.9).

- We prove that the extreme rays of the pointed augmented flux cone (after splitting a minimum set of reversible reactions) correspond to a MEMo, needed to fully describe the underlying metabolic network (Theorem 6.15).
- We apply this method to several metabolic networks and the results show that the number of EFMs in a MEMo is by several orders of magnitude smaller than the number of all EFMs (Section 6.6).

The chapter is organised as follows: We start in Section 6.3 by proving that extreme rays of pointed augmented flux cones correspond to EFMs of the underlying metabolic network. In Section 6.4 we show how to find a minimum set of reversible reactions to split such that the resulting augmented

6. Finding MEMo

flux cone is pointed. We show in Section 6.5 that the extreme rays of the pointed augmented flux cone (after splitting a minimum set of reversible reactions) correspond to a MEMo. Finally, in Section 6.6 we provide computational results for various genome-scale metabolic networks that were obtained by the software that we implemented. Related and future work are discussed in Section 6.7 and Section 6.8.

6.2 ERs, EFMs, and MMBs

The following proposition summarises what is known about the relationship between extreme rays, EFMs, and MMBs. The detailed mathematical background on polyhedral cones can be found Section 2.3.1 and [Schrijver, 1998].

Proposition 6.6. *Let $\Gamma_{\mathcal{N}}$ be the flux cone of a metabolic network \mathcal{N} .*

- 1. If all reactions are irreversible, then $\Gamma_{\mathcal{N}}$ is pointed and the extreme rays of $\Gamma_{\mathcal{N}}$ are exactly the EFMs of \mathcal{N} [Schuster and Hilgetag, 1994; Gagneur and Klamt, 2004].*
- 2. If there are reversible reactions and $\Gamma_{\mathcal{N}}$ is pointed, then the extreme rays of $\Gamma_{\mathcal{N}}$ form a subset of the set of EFMs in \mathcal{N} . However, not every EFM needs to be an extreme ray [Jevremović et al., 2010]. The extreme rays are exactly the vectors in $\Gamma_{\mathcal{N}}$ with inclusion-minimal support in Irr .*
- 3. If $\Gamma_{\mathcal{N}}$ is not pointed, then there is a 1-1 correspondence between the MMBs of \mathcal{N} and the minimal proper faces of $\Gamma_{\mathcal{N}}$ [Larhlimi and Bockmayr, 2009].*

Proof. We have only to prove the last part of 2). By [Larhlimi and Bockmayr, 2009] (or 3)) there is a 1-1 correspondence between the minimal proper faces of a flux cone and the MMBs. All vectors in a minimal proper face F have the same support in Irr , which is exactly the MMB corresponding to F . If a polyhedral cone is pointed, the minimal proper faces are the extreme rays. \square

In general, a metabolic network may contain reversible reactions together with reversible flux vectors $v \neq 0$, for which $v, -v \in \Gamma_{\mathcal{N}}$, and the flux cone is non-pointed. For example, in Figure 1.1, there is a reversible flux vector with the support $\{3, 5, 9, 10\}$. If there exist reversible flux vectors, a minimal

6.3 Extreme rays of pointed augmented flux cones are EFMs

generating set of the flux cone is not unique anymore. It may consist of an arbitrary flux vector for each minimal proper face of the flux cone, together with a vector space basis of the lineality space, see Section 8.8 in [Schrijver, 1998]. Computing a minimal generating set by one of the standard software tools usually does not deliver a set of EFMs. In order to obtain a minimal generating set consisting of EFMs, one can transform the flux cone into a pointed cone.

6.3 Extreme rays of pointed augmented flux cones are EFMs

We give a short reminder about notation regarding reaction splitting. All details can be found in Section 2.3.1.3.

In the case of metabolic networks, the variables corresponding to the irreversible reactions, by definition, can take only non-negative values. In order to obtain a pointed cone, we can split all reversible reactions into two irreversible ones, see Figure 2.3 and Example 2.31 in Section 2.3.1.3. This leads to the pointed augmented flux cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{Rev}}$. The uniquely determined ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\text{Rev}}$ are called *extreme currents* [Clarke, 1988].

It can be shown that after recombination, they correspond exactly to the EFMs of the metabolic network [Gagneur and Klamt, 2004]. In addition, for each split reaction $i_k \in \text{Rev}$, there exists a *2-cycle* $\tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\text{Rev}}$ with $\tilde{v}_{i_k} = -\tilde{v}_{n+k}$, and $v_i = 0$ otherwise, see Definition 2.33. In general we denote with $\tilde{\Gamma}_{\mathcal{N}}^I$ the augmented flux cone obtained by splitting the set of reactions $I \subseteq \text{Rev}$ in $\Gamma_{\mathcal{N}}$ and with Π_I the corresponding 2-cycles.

In this section, we prove that whenever an augmented flux cone is pointed after splitting a set of reversible reactions, the ERs are EFMs of the underlying metabolic network after recombination.

Already in [Schilling et al., 2000], it has been shown that splitting only the *internal* reversible reactions delivers an augmented flux cone which is pointed (assuming that there is at most one exchange reaction per internal metabolite). After recombination, the extreme rays of this cone are called *extreme pathways* (EPs), see Section 6.7.1 for further discussion.

Extending the results in [Schilling et al., 2000], we show next that if splitting a subset $I \subseteq \text{Rev}$ of reversible reactions results in an pointed augmented flux cone $\tilde{\Gamma}_{\mathcal{N}}^I$ then the ERs of $\tilde{\Gamma}_{\mathcal{N}}^I$ after reconfiguration define a subset of EFMs in the metabolic network \mathcal{N} .

6. Finding MEMo

Theorem 6.7. *Let $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ be the flux cone of a metabolic network \mathcal{N} . Let $I \subseteq \text{Rev}$ be a set of reversible reactions such that the augmented cone $\tilde{\Gamma}_{\mathcal{N}}^I$ obtained by splitting the reactions in I is pointed. With exception of the 2-cycles, the extreme rays of $\tilde{\Gamma}_{\mathcal{N}}^I$ after recombination are elementary flux modes in \mathcal{N} .*

Proof. By splitting the reactions in I we get the metabolic network $\tilde{\mathcal{N}} = (\tilde{\mathfrak{M}}, \tilde{\mathfrak{R}}, \tilde{S}, \tilde{\text{Irr}})$ with $\tilde{\mathfrak{R}} = \mathfrak{R} \cup I_{\text{split}}$, $\tilde{S} = (S \mid -S_{*,I})$ and $\tilde{\text{Irr}} = \text{Irr} \cup I \cup I_{\text{split}}$. Here, I_{split} is the set of additional irreversible reactions obtained by splitting the reactions in I . The corresponding flux cone is

$$\Gamma_{\tilde{\mathcal{N}}} = \tilde{\Gamma}_{\mathcal{N}}^I = \left\{ \begin{pmatrix} v \\ w \end{pmatrix} \in \mathbb{R}^{|\mathfrak{R}|+|I|} \mid \tilde{S} \begin{pmatrix} v \\ w \end{pmatrix} = 0, v_{\text{Irr}} \geq 0, v_I \geq 0, w \geq 0 \right\}.$$

Let \tilde{e} being an ER of $\Gamma_{\tilde{\mathcal{N}}}$ and $e = \pi_I^r(\tilde{e})$ the corresponding flux vector in $\Gamma_{\mathcal{N}}$. By Proposition 6.6, the ERs of the pointed cone $\Gamma_{\tilde{\mathcal{N}}}$ are EFMs in $\tilde{\mathcal{N}}$. By the rank test for EFMs, this implies $\rho(\tilde{S}_{*,\text{supp}(\tilde{e})}) = |\text{supp}(\tilde{e})| - 1$.

Suppose first that for all $i_k \in I$ we have $\tilde{e}_{i_k} = 0$ or $\tilde{e}_{n+k} = 0$, i.e., there is no non-zero flux through both directions of a former reversible reaction. Then $|\text{supp}(\tilde{e})| = |\text{supp}(e)|$ and $\rho(\tilde{S}_{*,\text{supp}(\tilde{e})}) = \rho(S_{*,\text{supp}(e)})$ (by the definition of $\tilde{S} = (S \mid -S_{*,I})$). It follows that $\rho(S_{*,\text{supp}(e)}) = |\text{supp}(e)| - 1$. Using again the rank test, this shows that e is an EFM in $\Gamma_{\mathcal{N}}$.

Suppose now that there exists $i_k \in I$ with $\tilde{e}_{i_k} \neq 0$ and $\tilde{e}_{n+k} \neq 0$. Since \tilde{e} is an EFM of $\tilde{\mathcal{N}}$, it follows $\tilde{e}_{i_k} = \tilde{e}_{n+i}$ and $\tilde{e}_j = 0$, for all $j \in \tilde{\mathfrak{R}} \setminus \{i_k, n+k\}$. Hence, \tilde{e} is a 2-cycle, which we excluded from our considerations. \square

Thus, whenever the augmented flux cone $\tilde{\Gamma}_{\mathcal{N}}^I$ is pointed, the recombined ERs of $\tilde{\Gamma}_{\mathcal{N}}^I$ correspond to a subset of EFMs in the original metabolic network \mathcal{N} . The remaining EFMs of \mathcal{N} correspond to rays inside of $\tilde{\Gamma}_{\mathcal{N}}^I$ and can be obtained from the ERs by conical combinations. In the following we develop a method for determining a minimum set of reversible reactions that have to be split in order to obtain a pointed cone.

6.4 Splitting a minimum number of reversible reactions

We show in this section that there exist minimum sets of reversible reactions such that, after splitting these reactions, the augmented flux cone becomes

6.4 Splitting a minimum number of reversible reactions

pointed. These sets always contain t fully reversible reactions, where t is the dimension of the lineality space. We note that splitting one reaction reduces the dimension of the lineality space by at most 1, see [Larhlmi and Bockmayr, 2008]. Therefore at least t reversible reactions have to be split in order to obtain a pointed cone.

Definition 6.8 (minFrev). *Let $\Gamma_{\mathcal{N}}$ be the flux cone of a metabolic network \mathcal{N} . Let t be the dimension of the lineality space $\Lambda_{\mathcal{N}}$. A minimum set of fully reversible reactions is a set minFrev of t fully reversible reactions such that the flux cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ obtained by splitting the reactions in minFrev is pointed. We denote by minFrev_{split} the set of the t additional irreversible reactions obtained by splitting. Thus*

$$\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}} = \{\tilde{v} \in \mathbb{R}^{|\mathfrak{R}|+t} \mid (S \mid -S_{*,\text{minFrev}})\tilde{v} = 0, \tilde{v}_{\text{Irr} \cup \text{minFrev} \cup \text{minFrev}_{\text{split}}} \geq 0\}.$$

The next theorem states that for any metabolic network \mathcal{N} there exists a minimum set of fully reversible reactions minFrev.

Theorem 6.9. *Let $\Gamma_{\mathcal{N}}$ be the flux cone of a metabolic network \mathcal{N} . If t is the dimension of the lineality space $\Lambda_{\mathcal{N}}$, then there exists a set minFrev of t fully reversible reactions such that the cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ obtained by splitting the reactions in minFrev is pointed.*

Proof. The idea of the proof is the following. As a vector space, the lineality space $\Lambda_{\mathcal{N}}$ has a basis consisting of t linearly independent vectors. In order to obtain a pointed flux cone, we intuitively have to destroy $\Lambda_{\mathcal{N}}$, i.e., all vectors in the basis. We do this by splitting one reaction in each vector of the basis. To make sure that each of these reactions is present in exactly one basis vector, we make use of the reduced column echelon form, cf. Figure 5.4.

We may assume that $\Gamma_{\mathcal{N}}$ is not pointed, hence $\Lambda_{\mathcal{N}} \neq \{0\}$. Let $B \in \mathbb{R}^{|\mathfrak{R}| \times t}$ be a matrix whose columns $(\check{b}^1, \dots, \check{b}^t)$ form a basis of the lineality space $\Lambda_{\mathcal{N}}$, where $t \geq 1$ is the dimension of $\Lambda_{\mathcal{N}}$. By applying elementary column operations, we can obtain the reduced column echelon form B^{rcef} of B , see Figure 5.4, which is uniquely determined by B [Gilbert and Gilbert, 2014, p.85]. The columns (b^1, \dots, b^t) of B^{rcef} define again a basis of $\Lambda_{\mathcal{N}}$. The row indices j_1, \dots, j_r for the pivot 1's in B^{rcef} are exactly the indices of the fully reversible reactions we are looking for. To see this, define minFrev = $\{j_1, \dots, j_r\}$. From $t = \rho(B) = \rho(B^{\text{rcef}}) = r$, we get $r = t$. Since $b_{j_k}^k = 1 \neq 0$, for $k = 1, \dots, r = t$, we have minFrev \subseteq Frev.

After splitting the reactions in minFrev we get the augmented flux cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$, which by Proposition 5.2 has the lineality space

$$\tilde{\Lambda}_{\mathcal{N}}^{\text{minFrev}} = \left\{ \begin{pmatrix} v \\ 0 \end{pmatrix} \in \mathbb{R}^{|\mathfrak{R}|+|\text{minFrev}|} \mid v \in \Lambda_{\mathcal{N}} \text{ and } v_{\text{minFrev}} = 0 \right\}.$$

6. Finding MEMo

Since (b^1, \dots, b^t) defines a basis of $\Lambda_{\mathcal{N}}$, any $v \in \Lambda_{\mathcal{N}}$ can be written as $v = \sum_{k=1}^t \lambda_k b^k$, for some $\lambda_k \in \mathbb{R}$. The matrix B^{rcef} is in reduced column-echelon form. This means that for each $j \in \text{minFrev}$ there is exactly one b^{k_j} with $b_j^{k_j} = 1$, and for all $k \neq k_j$, it holds $b_j^k = 0$. Since for $\begin{pmatrix} v \\ 0 \end{pmatrix} \in \tilde{\Lambda}_{\mathcal{N}}^{\text{minFrev}}$ we have $v_j = \sum_{k=1}^t \lambda_k b_j^k = \lambda_{k_j} b_j^{k_j} = \lambda_{k_j} = 0$, for all $j \in \text{minFrev}$, it follows that $\lambda_k = 0$, for all $k \in \{1, \dots, t\}$. This implies $v = 0$ and thus $\tilde{\Lambda}_{\mathcal{N}}^{\text{minFrev}} = \{0\}$, which proves that $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ is pointed. \square

Based on the previous proof, Algorithm 2 summarizes how to find a minimum set minFrev such that $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ is pointed. First we compute a basis of the lineality space $\Lambda_{\mathcal{N}}$ of $\Gamma_{\mathcal{N}}$. Let B be the matrix whose columns are the vectors of this basis. Next we transform B in the reduced column echelon form (e.g. in MATLAB using `rref`). The row indices of the pivots then correspond to the indices of the reactions that form the set minFrev .

Algorithm 2: Finding a minimum set of fully reversible reactions

Input : Stoichiometric matrix S

Set of irreversible reactions Irr

Output: Minimum set of fully reversible reactions minFrev

- 1 $B \leftarrow \text{basis}(\Lambda_{\mathcal{N}})$;
/* columns of B are vectors of the basis of the lineality space $\Lambda_{\mathcal{N}}$ */
 - 2 $B^{\text{rcef}} \leftarrow \text{ReducedColumnEchelonForm}(B)$;
 - 3 $\text{minFrev} \leftarrow$ row indices of the pivots of B^{rcef} ;
 - 4 **return** minFrev ;
-

We note that the set minFrev computed by Algorithm 2 does not depend on the particular basis B of $\Lambda_{\mathcal{N}}$ because two matrices B, B' with the same column space have the same reduced column-echelon form [Gilbert and Gilbert, 2014].

Example 6.10 (Compute minFrev). *The flux cone for the network in Figure 1.1 has a lineality space of dimension 2. Therefore any set minFrev contains two reactions. The lineality space can be described by the matrix*

$$B^{\top} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ 0 & 0 & 1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \end{pmatrix}.$$

We can see that only the fully reversible reactions 3,5,9,10,11, and 12 are active in the lineality space. Furthermore, B is already in reduced column

6.5 MEMo: Minimum set of Elementary Modes

echelon form and thus $B = B^{\text{rcf}}$. The pivot elements are 3 and 9, therefore $\text{minFrev} = \{3, 9\}$. These are the reactions which are split in Example 6.5. Splitting reactions 3 and 9 delivers an augmented flux cone which is pointed. This can be seen from the modified network in Figure 6.1, since there exists no feasible flux vector that is reversible.

6.5 MEMo: Minimum set of Elementary Modes

The main result of this section is that the extreme rays of an augmented flux cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$, where minFrev is a minimum set of fully reversible reactions, after recombination form a Minimum set of Elementary Modes or MEMo, cf. Def. 6.1. We start by using Proposition 6.6 to prove the following result:

Proposition 6.11. *Let minFrev be a minimum set of fully reversible reactions in a metabolic network \mathcal{N} such that $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ is pointed. Then the set of extreme rays of the augmented cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ does not contain any 2-cycle.*

Proof. By the definition of the lineality space $\Lambda_{\mathcal{N}}$ and the set Frev , we have $\Lambda_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} = 0\} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid S_{*,\text{Rev}} v_{\text{Rev}} = 0, v_{\text{Irr}} = 0\} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid S_{*,\text{Frev}} v_{\text{Frev}} = 0, v_{\text{Irr} \cup (\text{Rev} \setminus \text{Frev})} = 0\}$. From $\dim(\Lambda_{\mathcal{N}}) = t$, we get $\rho(S_{*,\text{Frev}}) = |\text{Frev}| - t$.

By Proposition 5.2, it follows that $\tilde{\Lambda}_{\mathcal{N}}^{\text{minFrev}} = \{(v \mid 0)^{\top} \in \mathbb{R}^{|\mathfrak{R}|+|\text{minFrev}|} \mid S_{*,\text{Frev} \setminus \text{minFrev}} v_{\text{Frev} \setminus \text{minFrev}} = 0, v_{\text{Irr} \cup (\text{Rev} \setminus \text{Frev}) \cup \text{minFrev}} = 0\}$. Since $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ is pointed, this implies $\rho(S_{*,\text{Frev} \setminus \text{minFrev}}) = |\text{Frev}| - |\text{minFrev}| = |\text{Frev}| - t = \rho(S_{*,\text{Frev}})$. For each $i_k \in \text{minFrev}$, the column S_{*,i_k} is therefore linearly dependent on the columns of $S_{*,\text{Frev} \setminus \text{minFrev}}$. Thus, there exists a flux vector $v \in \Gamma_{\mathcal{N}} \setminus \{0\}$ with $v_{i_k} = 1$ and $\text{supp}(v) \subseteq (\text{Frev} \setminus \text{minFrev}) \cup \{i_k\}$. It follows that $\pi_{\text{minFrev}}(v) = \tilde{v} \in \tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ and $\text{supp}_{\widetilde{\text{Irr}}}(\tilde{v}) = \{i_k\}$, with $\widetilde{\text{Irr}} = \text{Irr} \cup \text{minFrev} \cup \text{minFrev}_{\text{split}}$. This implies that \tilde{v} has minimal support w.r.t irreversible reactions in $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$. According to Proposition 6.6, \tilde{v} is an ER of $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$.

Suppose there exists an ER $\tilde{w} \in \tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ with $\tilde{w}_{i_k} \neq 0 \neq \tilde{w}_{n+k}$. This would imply that $\text{supp}_{\widetilde{\text{Irr}}}(\tilde{v}) \subsetneq \text{supp}_{\widetilde{\text{Irr}}}(\tilde{w})$, which is a contradiction. It follows that the set of ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ does not contain any 2-cycle. \square

From the proof of Proposition 6.11 we can deduce the following corollaries:

6. Finding MEMo

Corollary 6.12. *Let \tilde{e} be an extreme ray in $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$. Then there is at most one reaction $j \in \min\text{Frev} \cup \min\text{Frev}_{\text{split}}$ with $\tilde{e}_j \neq 0$. If $\tilde{e}_j \neq 0$, for some $j \in \min\text{Frev} \cup \min\text{Frev}_{\text{split}}$, then $\tilde{e}_i = 0$, for all $i \in \text{Irr}$. Moreover, $\pi_{\min\text{Frev}}^r(\tilde{e})$ is a reversible EFM of the original network \mathcal{N} .*

Corollary 6.13. *For each $i_k \in \min\text{Frev}$ there exist exactly two extreme rays $\tilde{e}, \tilde{f} \in \tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ such that $\tilde{e}_{i_k} \neq 0$ and $\tilde{f}_{n+k} \neq 0$. After recombination, both \tilde{e} and \tilde{f} define a single reversible EFM e in $\Gamma_{\mathcal{N}}$ with $\text{supp}(e) = \text{supp}(\pi_{\min\text{Frev}}^r(\tilde{e})) = \text{supp}(\pi_{\min\text{Frev}}^r(\tilde{f}))$. Altogether, the t reactions in $\min\text{Frev}$ define $2t$ extreme rays in $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ and t linearly independent reversible EFMs in \mathcal{N} .*

The following proposition relates the supports of the extreme rays of $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ to the minimal metabolic behaviours (MMBs) of \mathcal{N} .

Proposition 6.14. *Let $\min\text{Frev}$ be a minimum set of fully reversible reactions in a metabolic network \mathcal{N} such that $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ is pointed. The non-empty supports $\text{supp}_{\text{Irr}}(\tilde{e})$ of the extreme rays \tilde{e} of $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ are exactly the minimal metabolic behaviours of \mathcal{N} .*

Proof. Let D be an MMB in \mathcal{N} and let $v \in \Gamma_{\mathcal{N}}$ such that $\text{supp}_{\text{Irr}}(v) = D$. By Lemma 2 in [Jevremović and Boley, 2013], there is a decomposition $v = e + y$, with $e, y \in \Gamma_{\mathcal{N}}$, $e_{\min\text{Frev}} = 0$ and $y_{\text{Irr}} = 0$. For $\tilde{e} = \pi_{\min\text{Frev}}(e)$ and $\widetilde{\text{Irr}} = \text{Irr} \cup \min\text{Frev} \cup \min\text{Frev}_{\text{split}}$ it follows that $\text{supp}_{\widetilde{\text{Irr}}}(\tilde{e}) = \text{supp}_{\text{Irr}}(\tilde{e}) = \text{supp}_{\text{Irr}}(e) = \text{supp}_{\text{Irr}}(v) = D$. Suppose \tilde{e} is not an extreme ray in $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$. By Proposition 6.6, there exists $\tilde{w} \in \tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ with $\emptyset \neq \text{supp}_{\widetilde{\text{Irr}}}(\tilde{w}) \subsetneq \text{supp}_{\widetilde{\text{Irr}}}(\tilde{e})$. For $w = \pi_{\min\text{Frev}}^r(\tilde{w}) \in \Gamma_{\mathcal{N}}$ this implies $\text{supp}_{\text{Irr}}(w) = \text{supp}_{\widetilde{\text{Irr}}}(\tilde{w}) \subsetneq D$, in contradiction to D being an MMB in $\Gamma_{\mathcal{N}}$.

Conversely, let \tilde{e} be an extreme ray in $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ with $\text{supp}_{\text{Irr}}(\tilde{e}) \neq \emptyset$. By Corollary 6.12, we have $\text{supp}_{\widetilde{\text{Irr}}}(\tilde{e}) = \text{supp}_{\text{Irr}}(\tilde{e})$. Assume $D = \text{supp}_{\text{Irr}}(\pi_{\min\text{Frev}}^r(\tilde{e})) = \text{supp}_{\text{Irr}}(\tilde{e})$ is not an MMB in \mathcal{N} . Then there exists $v \in \Gamma_{\mathcal{N}}$ with $\emptyset \neq \text{supp}_{\text{Irr}}(v) \subsetneq D$. As before, we consider a decomposition $v = u + y$, with $u, y \in \Gamma_{\mathcal{N}}$, $u_{\min\text{Frev}} = 0$ and $y_{\text{Irr}} = 0$. For $\tilde{u} = \pi_{\min\text{Frev}}(u)$ we get $\text{supp}_{\widetilde{\text{Irr}}}(\tilde{u}) = \text{supp}_{\text{Irr}}(u) = \text{supp}_{\text{Irr}}(v) \subsetneq D = \text{supp}_{\widetilde{\text{Irr}}}(\tilde{e})$, in contradiction to \tilde{e} being an extreme ray in $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$. \square

We prove now one of our main theorems, stating that for any minimum set of fully reversible reactions $\min\text{Frev}$, the extreme rays of $\tilde{\Gamma}_{\mathcal{N}}^{\min\text{Frev}}$ after recombination define a MEMo of \mathcal{N} .

6.5 MEMo: Minimum set of Elementary Modes

Theorem 6.15. *Let minFrev be a minimum set of fully reversible reactions in a metabolic network \mathcal{N} such that $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ is pointed. After recombination, the extreme rays of the pointed cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ define a minimum set of EFMs or MEMo in \mathcal{N} . Any set of EFMs of smaller cardinality cannot be a generating set for the flux cone $\Gamma_{\mathcal{N}}$.*

Proof. According to [Schrijver, 1998, Sect. 8.8], a minimum set of generating vectors for $\Gamma_{\mathcal{N}}$ always consists of one vector in each minimal proper face of $\Gamma_{\mathcal{N}}$ and a vector space basis of $\Lambda_{\mathcal{N}}$. According to Corollary 6.12, there are two types of extreme rays $\tilde{e} \in \tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$:

Case 1: $\text{supp}_{\text{Irr}}(\tilde{e}) \neq \emptyset$ and $\tilde{e}_{\text{minFrev} \cup \text{minFrev}_{\text{split}}} = 0$. Let $e = \pi_{\text{minFrev}}^r(\tilde{e})$ be the recombined vector. By Proposition 6.14, $\text{supp}_{\text{Irr}}(e) = \text{supp}_{\text{Irr}}(\tilde{e})$ is an MMB D in \mathcal{N} . By [Larhlmi and Bockmayr, 2009] this implies that e belongs to the minimal proper face defined by D . Since $\text{supp}_{\text{Irr}}(e) \neq \emptyset$, we also have $e \notin \Lambda_{\mathcal{N}}$. By Proposition 6.6, the extreme ray \tilde{e} is an EFM in $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$. Thus, $\text{supp}(\tilde{e})$ is inclusion-minimal in $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$. Since $\tilde{e}_{\text{minFrev} \cup \text{minFrev}_{\text{split}}} = 0$, we have $\text{supp}(e) = \text{supp}(\tilde{e})$, which implies that e is an EFM in $\Gamma_{\mathcal{N}}$ (otherwise, there would exist $v \in \Gamma_{\mathcal{N}} \setminus \{0\}$ with $\text{supp}(v) \subsetneq \text{supp}(e)$, and for $\tilde{v} = \text{supp}(\pi_{\text{minFrev}}(v))$ we would get $\text{supp}(\tilde{v}) = \text{supp}(v) \subsetneq \text{supp}(e) = \text{supp}(\tilde{e})$, which is a contradiction).

Case 2: $\tilde{e}_j \neq 0$ for some $j \in \text{minFrev} \cup \text{minFrev}_{\text{split}}$ and $\tilde{e}_{\text{Irr}} = 0$. According to Corollary 6.12, all ERs with this property lead to t reversible EFMs in \mathcal{N} , which after recombination generate $\Lambda_{\mathcal{N}}$.

In conclusion, after recombination, the ERs with the first property represent the MPFs of $\Gamma_{\mathcal{N}}$ and the ERs with the second property form a basis of $\Lambda_{\mathcal{N}}$. Altogether, the recombined ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ form a minimum set of EFMs generating $\Gamma_{\mathcal{N}}$, and therefore define a MEMo. \square

Corollary 6.16. *For a given metabolic network \mathcal{N} the number of EFMs in a MEMo is always $s + t$, where s is the number of minimal proper faces in $\Gamma_{\mathcal{N}}$ and t is the dimension of $\Lambda_{\mathcal{N}}$.*

Algorithm 3 describes how to compute a MEMo given a minimum set of fully reversible reactions minFrev , which can be obtained by Algorithm 2. First, the reactions in minFrev are split in order to get the pointed augmented flux cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$. Next, we compute the ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ using an existing tool based on the double description method [Fukuda and Prodon, 1996], e.g., `polco` [Terzer, 2017b] or `cdd` [Fukuda, 2005].

6. Finding MEMo

Algorithm 3: Finding MEMo

Input : Stoichiometric matrix S
Set of irreversible reactions Irr
A minimum set of fully reversible reactions minFrev

Output: MEMo

```

/* A minimum set of EFMs, describing the underlying
metabolic network */
1  $\tilde{S} \leftarrow (S| - S_{*,\text{minFrev}})$ ;
/*  $\tilde{S}$  consists of the columns of  $S$  and the negative of the
columns corresponding to the reactions of  $\text{minFrev}$  */
2  $I \leftarrow (\text{Irr} \cup \text{minFrev} \cup \text{minFrev}_{\text{split}})$ ;
/*  $H$  contains the indices of the irreversible reactions
of the pointed augmented cone  $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ , including the
split reactions of the set  $\text{minFrev}$  */
3  $\text{ER} \leftarrow$  extreme rays of  $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}} = \{\tilde{v} \in \mathbb{R}^{|\mathcal{R}|+t} \mid \tilde{S}\tilde{v} = 0, \tilde{v}_I \geq 0\}$ ;
/* use polco to compute the ERs of  $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$  */
4  $\text{MEMo} \leftarrow$  recombination( $\text{ER}$ );
/* apply  $\pi_{\text{minFrev}}^r$  to the ERs of  $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$  */
5 return MEMo;

```

The MEMos that can be obtained by Algorithm 3 are not unique. Their elements depend on the set minFrev of reactions that are split. As the following example shows, different minimum sets minFrev of fully reversible reactions may result in the same MEMo. Furthermore, there exist MEMos that cannot be obtained by Algorithm 3.

Example 6.17 (Different MEMos). *For the network in Figure 1.1 there exist three different reversible EFMs with the supports $\{3, 5, 9, 10\}$, $\{3, 5, 11, 12\}$, $\{9, 10, 11, 12\}$. Each two of them form a basis of the lineality space $\Lambda_{\mathcal{N}}$, thus there are 3 bases. There exist three MMBs $\{2\}$, $\{6, 7\}$, $\{6, 8\}$, see Example 2.8, which correspond to three minimal proper faces of the flux cone $\Gamma_{\mathcal{N}}$. The minimal proper face for the MMB $\{6, 7\}$ contains only one EFM with support $\{4, 6, 7\}$. The other two minimal proper faces each contain three EFMs, namely $\{1, 2, 3, 4\}$, $\{1, 2, 4, 5, 9, 10\}$, $\{1, 2, 4, 5, 11, 12\}$ for the MMB $\{2\}$ and $\{4, 6, 8, 9, 10\}$, $\{3, 4, 5, 6, 8\}$, $\{4, 6, 8, 11, 12\}$ for the MMB $\{6, 8\}$. The remaining 8 EFMs lie in the interior of the cone. By choosing one basis for the lineality space and one EFM for each minimal proper face, we get in total $3 \cdot 1 \cdot 3 \cdot 3 = 27$ possible MEMos.*

By enumeration, we can determine that there exist 12 different minimum sets of fully reversible reactions minFrev for the network in Figure 1.1, which give rise to 5 different MEMos. For example, the two

6.5 MEMo: Minimum set of Elementary Modes

minFrev sets $\{3, 9\}$ and $\{3, 10\}$ result in the same MEMo. A MEMo that cannot be obtained by choosing a minFrev set and computing the extreme rays of the cone $\Gamma_{\mathcal{N}}^{\text{minFrev}}$ is the set of EFMs with the supports $\{1, 2, 3, 4\}$, $\{4, 6, 7\}$, $\{4, 6, 8, 9, 10\}$, $\{3, 5, 9, 10\}$, $\{3, 5, 11, 12\}$.

Since there exist different MEMos a question which arises is if the computed MEMo using Algorithms 2 and 3 together is unique for the same metabolic network. The answer is yes and is given in the next theorem. For the proof of the theorem in this section we need the notion of column space:

Definition 6.18 (Column space and column equivalent, [Artin, 1998]). Consider a matrix $B \in \mathbb{R}^{m \times n}$. The column space of B are all linear combinations of the columns of B : $B\lambda = \sum_{i=1}^n \lambda_i b^i$, with b^i being the i -th column of B and $\lambda_i \in \mathbb{R}$. Matrices are column equivalent if and only if they have the same column space. If two matrices are column equivalent they always have the same reduced column echelon form (elementary column operations do not affect the column space of a matrix).

Theorem 6.19. Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and its corresponding flux cone $\Gamma_{\mathcal{N}}$. If the order of the columns of the stoichiometric matrix S is not changed, the set $\text{minFrev} \subseteq \text{Frev}$, found using Algorithm 2, is always the same and therefore unique.

Proof. Let $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ be the flux cone of $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and $\Lambda_{\mathcal{N}}$ its lineality space. Algorithm 2 computes a set minFrev by first computing a basis for $\Lambda_{\mathcal{N}}$. Let B denotes a matrix consisting of basis vectors for $\Lambda_{\mathcal{N}}$. All matrices B representing a basis for $\Lambda_{\mathcal{N}}$ have, per definition, the same column space and therefore, they are column equivalent and have the same reduced column echelon form.

Consequently, the reduced column form is the same for all bases for $\Lambda_{\mathcal{N}}$ and the set minFrev as defined in Definition 6.8 is always the same for a given stoichiometric matrix S , when using Algorithm 2. In particular, the ERs of $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$, and therefore the computed MEMo, using Algorithm 3, for a given stoichiometric matrix S is always the same. \square

In order to find a different set of minFrev we have to change the column space of B . Since B represents the null space of S changing the column space of B can be done by changing the order of the columns of $S_{*, \text{Frev}}$.

6.6 Computational results and discussion

We implemented our method in `MATLAB` using `POLCO` [Terzer, 2017b] to compute the extreme rays of a pointed cone. Our software is available at <https://sourceforge.net/projects/findingmemo/>. In the following, we present computational results for various metabolic network reconstructions taken from `BiGG Models` [King et al., 2016], `KEGG` [Kanehisa and Goto, 2000] (together with `KEGGtranslator` [Wrzodek et al., 2013]), and the `BioModels Database` [Li et al., 2010]. Table 6.1 summarises the main characteristics of the metabolic networks studied. In Table 6.2, we study the size of the MEMos compared to the number of extreme pathways (EPs) and EFMs, which were computed with `POLCO` [Terzer, 2017b] resp. `EFMTOOL` [Terzer, 2017a]. In Figure 6.4, we compare the number of flux vectors in a MEMo to the number of EPs and EFMs for those networks for which we were able to compute the EPs and EFMs. Since the number of EFMs of a MEMo is much smaller than the number of all EFMs, it is faster to compute them see Table 6.3.

6.6 Computational results and discussion

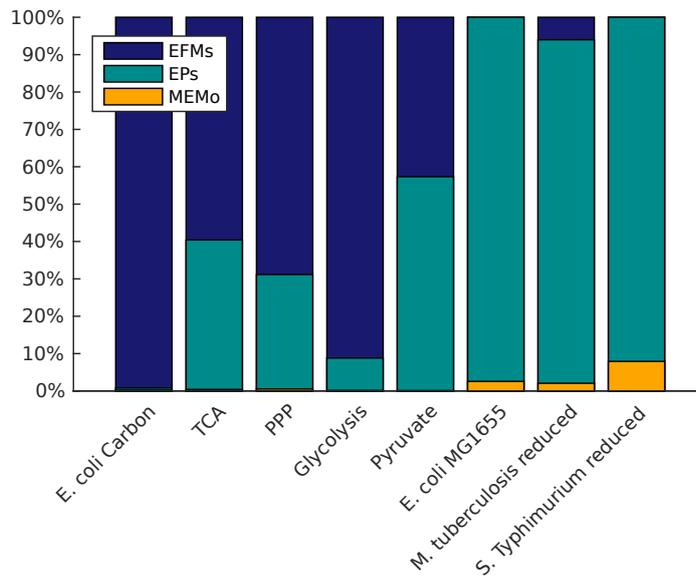


Figure 6.4: Comparison of the number of flux vectors of the three different generating sets we discussed in this article. We set the number of all EFMs to 100% and compare the vectors of the other sets accordingly. The bars overlay and the largest number is always on the top. Hence, if there is no blue bar for the EFMs *Escherichia coli* MG1655 the number of EPs and the number of EFMs are the same. Note that there is always a yellow bar indicating the size of the MEMo even it is hard to see. This shows how small the number of flux vectors in a MEMo is, compared to the whole set of EFMs.

| network | rxns | mets | rev | intrev | frev | t |
|---|-------------|-------------|------------|---------------|-------------|----------|
| <i>Escherichia coli</i> carbon metabolism ([Chassagnole et al., 2002]) | 34 | 18 | 34 | 18 | 32 | 15 |
| Citrate cycle (TCA) [Kanehisa and Goto, 2000] | 35 | 22 | 31 | 25 | 30 | 12 |
| Pentose phosphate pathway [Kanehisa and Goto, 2000; Wrzodek et al., 2013] | 57 | 34 | 31 | 18 | 27 | 8 |
| Glycolysis / Gluconeogenesis [Kanehisa and Goto, 2000; Wrzodek et al., 2013] | 61 | 32 | 42 | 28 | 33 | 13 |
| Pyruvate metabolism [Kanehisa and Goto, 2000; Wrzodek et al., 2013] | 81 | 28 | 34 | 29 | 24 | 12 |
| <i>Escherichia coli</i> MG1655 [King et al., 2016] | 87 | 68 | 34 | 28 | 0 | 0 |
| <i>Rhizobium etli</i> iOR363 [Resendis-Antonio et al., 2007; Li et al., 2010] | 194 | 371 | 71 | 60 | 0 | 0 |
| <i>Buchnera</i> iSM197 [Li et al., 2010; MacDonald et al., 2011] | 261 | 255 | 17 | 11 | 0 | 0 |
| <i>Mycoplasma pneumoniae</i> iJW145 [Li et al., 2010; Wodke et al., 2013] | 306 | 266 | 73 | 47 | 0 | 0 |
| <i>Blattabacterium cuenoti</i> iCG238 [Li et al., 2010; González-Domenech et al., 2012] | 350 | 364 | 73 | 60 | 5 | 2 |
| <i>Mycoplasma genitalium</i> iPS189 [Suthers et al., 2009a; Li et al., 2010] | 351 | 346 | 107 | 26 | 3 | 1 |
| <i>Mannheimia succiniciproducens</i> iSH335 [Hong et al., 2004; Li et al., 2010] | 374 | 316 | 58 | 49 | 3 | 1 |
| <i>Blattabacterium</i> iCG230 [Li et al., 2010; González-Domenech et al., 2012] | 412 | 358 | 71 | 59 | 5 | 2 |
| <i>Helicobacter pylori</i> 26695 [Schilling et al., 2002] | 452 | 396 | 94 | 82 | 31 | 6 |
| <i>Human Erythrocyte</i> iAB-RBC-283 [Li et al., 2010; Bordbar et al., 2011] | 469 | 342 | 123 | 68 | 10 | 2 |
| <i>Helicobacter pylori</i> iCS291 [Schilling et al., 2002; Li et al., 2010] | 493 | 396 | 94 | 67 | 31 | 6 |
| <i>Clostridium thermocellum</i> iSR432 [Li et al., 2010; Roberts et al., 2010] | 581 | 564 | 95 | 95 | 32 | 20 |

Table 6.1: Characteristics of different metabolic networks used for the comparison in Tab. 6.1 **network**: name of the network. **rxns**: number of unblocked reactions. **mets**: number of metabolites. **rev**: number of unblocked reversible reactions. **intrev**: number of unblocked internal reversible reactions. **frev**: number of unblocked fully reversible reactions. **t**: dimension of the lineality space of the flux cone.

| network | rxns | MEMo | EPs | EFMs |
|--|-------------|-------------|------------|-------------|
| <i>Escherichia coli carbon metabolism</i> [Chassagnole et al., 2002] | 34 | 15 | 52 | 6,421 |
| Citrate cycle (TCA) [Kanehisa and Goto, 2000] | 35 | 16 | 1,564 | 3,870 |
| Pentose phosphate pathway [Kanehisa and Goto, 2000] | 57 | 27 | 1,607 | 5,155 |
| Glycolysis / Gluconeogenesis [Kanehisa and Goto, 2000] | 61 | 29 | 1,716 | 19,464 |
| Pyruvate metabolism [Kanehisa and Goto, 2000] | 81 | 49 | 27,361 | 47,708 |
| <i>Escherichia coli</i> MG1655 [King et al., 2016] | 87 | 2,572 | 100,274 | 100,274 |
| <i>Rhizobium etli</i> iOR363 [Resendis-Antonio et al., 2007; Li et al., 2010] | 194 | 6,147 | N/A | N/A |
| <i>Buchnera</i> iSM197 [Li et al., 2010; MacDonald et al., 2011] | 261 | 245 | 1,863 | N/A |
| <i>Mycoplasma pneumoniae</i> iJW145 [Li et al., 2010; Wodke et al., 2013] | 306 | 1,593,880 | N/A | N/A |
| <i>Blattabacterium cuenoti</i> iCG238 [González-Domenech et al., 2012] | 350 | 376 | 1,863 | N/A |
| <i>Mycoplasma genitalium</i> iPS189 [Suthers et al., 2009a; Li et al., 2010] | 351 | 190,471 | N/A | N/A |
| <i>Mannheimia succiniciproducens</i> iSH335 [Hong et al., 2004; Li et al., 2010] | 374 | 2,064,760 | N/A | N/A |
| <i>Blattabacterium</i> iCG230 [Li et al., 2010; González-Domenech et al., 2012] | 412 | 84 | N/A | N/A |
| <i>Helicobacter pylori</i> 26695 [Schilling et al., 2002] | 452 | 150,138 | N/A | N/A |
| <i>Human Erythrocyte</i> iAB-RBC-283 [Li et al., 2010; Bordbar et al., 2011] | 469 | 6,179 | N/A | N/A |
| <i>Helicobacter pylori</i> iCS291 [Schilling et al., 2002; Li et al., 2010] | 493 | 150,138 | N/A | N/A |
| <i>Clostridium thermocellum</i> iSR432 [Li et al., 2010; Roberts et al., 2010] | 581 | 1,039,267 | N/A | N/A |

Table 6.2: Size of MEMos, number of EPs and EFMs for different networks. **rxns**: number of unblocked reactions of the network. **network**: name of the network. **MEMo**: size of a minimum set of elementary modes. **EPs**: number of the extreme pathways. **EFMs**: number of the elementary flux modes. **N/A**: The programs used here (POLCO [Terzer, 2017b] and EFMT00L [Terzer, 2017a]) could not handle the size of the models resp. the number of EPs or EFMs.

| network | rxns | MEMo | EPs | EFMs |
|--|-------------|-------------|------------|-------------|
| <i>Escherichia coli</i> carbon metabolism [Chassagnole et al., 2002] | 34 | 0.62 sec | 6 sec | 3 sec |
| Citrate cycle (TCA) [Kanehisa and Goto, 2000] | 35 | 0.34 s | 0.6 sec | 0.3 sec |
| Pentose phosphate pathway [Kanehisa and Goto, 2000] | 57 | 0.5 sec | 0.4 sec | 0.5 sec |
| Glycolysis / Gluconeogenesis [Kanehisa and Goto, 2000] | 61 | 0.5 sec | 0.3 sec | 0.9 sec |
| Pyruvate metabolism [Kanehisa and Goto, 2000] | 81 | 0.5 sec | 1.4 sec | 2 sec |
| <i>Escherichia coli</i> MG1655 [King et al., 2016] | 87 | 1.6 sec | 17 sec | 9 sec |
| <i>Rhizobium etli</i> iOR363 [Resendis-Antonio et al., 2007; Li et al., 2010] | 194 | 5 sec | N/A | N/A |
| <i>Buchnera</i> iSM197 [Li et al., 2010; MacDonald et al., 2011] | 261 | 2.5 sec | 2 sec | N/A |
| <i>Mycoplasma pneumoniae</i> iJW145 [Li et al., 2010; Wodke et al., 2013] | 306 | 5 min | N/A | N/A |
| <i>Blattabacterium cuenoti</i> iCG238 [González-Domenech et al., 2012] | 350 | 10.6 sec | 2 sec | N/A |
| <i>Mycoplasma genitalium</i> iPS189 [Suthers et al., 2009a; Li et al., 2010] | 351 | 3 min | N/A | N/A |
| <i>Mannheimia succiniciproducens</i> iSH335 [Hong et al., 2004; Li et al., 2010] | 374 | 3 min | N/A | N/A |
| <i>Blattabacterium</i> iCG230 [Li et al., 2010; González-Domenech et al., 2012] | 412 | 7.8 sec | N/A | N/A |
| <i>Helicobacter pylori</i> 26695 [Schilling et al., 2002] | 452 | 1 min | N/A | N/A |
| <i>Human Erythrocyte</i> iAB-RBC-283 [Li et al., 2010; Bordbar et al., 2011] | 469 | 15 sec | N/A | N/A |
| <i>Helicobacter pylori</i> iCS291 [Schilling et al., 2002; Li et al., 2010] | 493 | 1 min 5 sec | N/A | N/A |
| <i>Clostridium thermocellum</i> iSR432 [Li et al., 2010; Roberts et al., 2010] | 581 | 6.5 min | N/A | N/A |

Table 6.3: Time to compute a MEMo, EPs and EFMs for different networks. **rxns**: number of unblocked reactions of the network. **network**: name of the network. **MEMo**: time to compute a minimum set of elementary modes. **EPs**: time to compute the extreme pathways. **EFMs**: time to compute the elementary flux modes. **N/A**: The programs used here (`POLCO` [Terzer, 2017b] and `EFMTOOL` [Terzer, 2017a]) could not handle the size of the models resp. the number of EPs or EFMs.

6.7 Related work

In all networks that we considered the number of fully reversible reactions and the number of internal reversible reactions is greater than or equal to the dimension t of the lineality space $\Lambda_{\mathcal{N}}$. While there always exist internal reversible reactions, it can happen that there are no fully reversible reactions. In this case, $t = 0$ and the original flux cone is already pointed, i.e., there is no need to split any reaction before computing a MEMo. This holds for the networks *Escherichia coli* MG1655 [King et al., 2016], *Rhizobium etli* iOR363 [Resendis-Antonio et al., 2007; Li et al., 2010], *Buchnera* iSM197 [Li et al., 2010; MacDonald et al., 2011], and *Mycoplasma pneumoniae* iJW145 [Li et al., 2010; Wodke et al., 2013]. For all these networks, there exist internal reversible reactions which are split to compute the extreme pathways (EPs). This implies that the number of EPs is larger than the number of flux vectors in a MEMo. For example, instead, of 2,572 EFMs in a MEMo for the network *Escherichia coli* MG1655 [King et al., 2016] there exist 100,274 EPs. For this network, the number of EFMs and EPs is the same, although not all reversible reactions are internal.

For the networks *Mannheimia succiniciproducens* iSH335 [Hong et al., 2004; Li et al., 2010], *Helicobacter pylori* 26695 [Schilling et al., 2002], *Helicobacter pylori* iCS291 [Schilling et al., 2002; Li et al., 2010], and *Clostridium thermocellum* iSR432 [Li et al., 2010; Roberts et al., 2010] we were not able to compute the EPs because `polco` was running out of memory, while we were always able to compute a MEMo.

We compared the size of the different generating sets (MEMo, EPs, and EFMs) in Figure 6.4. There, we defined the number of EFMs as 100%. The number of EPs or the number of flux vectors in a MEMo are related accordingly. Again, we use only the networks for which we were able to compute the EPs and the whole set of EFMs.

6.7 Related work

In the following, we discuss a number of papers that are closely related to our work.

6.7.1 Extreme pathways

In our approach, we compute a MEMo by splitting a minimum subset of reversible reactions. A closely related concept is the set of *extreme pathways* (EPs) [Schilling et al., 2000; Papin et al., 2004], which are computed by splitting all the internal reversible reactions. As shown in [Papin et al.,

6. Finding MEMo

2004; Schilling et al., 2000], splitting only the *internal* reversible reactions always delivers an augmented flux cone which is pointed (given that there is only one exchange reaction per internal metabolite). This cone is unique and so are its extreme rays. After recombination, these extreme rays are called extreme pathways. The assumption that there exists only one reversible exchange reaction per metabolite is not true for all metabolic networks. Examples where this condition does not hold are the *Escherichia coli* carbon metabolism [Chassagnole et al., 2002] or RECON1 [King et al., 2016; Duarte et al., 2007]. For computing a MEMo we do not need any condition on the network. However, the main advantage of our approach is that the number of EFMs in a MEMo is typically much smaller than the number of EPs, see Table 6.2.

6.7.2 Minimal metabolic behaviours

Larhlimi and Bockmayr [Larhlimi and Bockmayr, 2009] introduce a unique minimum outer description of the flux cone $\Gamma_{\mathcal{N}}$ based on *minimal metabolic behaviours* (MMBs) and the lineality space $\Lambda_{\mathcal{N}}$. Each MMB corresponds to a minimal proper face of the flux cone or to set of flux vectors with the same minimum set of active irreversible reactions. Larhlimi and Bockmayr show that for each MMB there exists a corresponding EFM, but they do not consider reaction splitting and do not present a method for computing a minimal generating set consisting of EFMs only.

6.7.3 Minimal generating set

Jevremovic and Boley [Jevremović and Boley, 2013] introduce a method for finding a minimum generating set of flux vectors which corresponds to a set of EFMs and therefore is a MEMo. However, their approach is different from the one introduced here. We exploit splitting reversible reactions, while they use a method based on generating a basis of the null space of $\Gamma_{\mathcal{N}}$. Additionally, they only prove that the reversible generating flux vectors are EFMs, but do not consider the irreversible flux vectors. We prove that all generating flux vectors obtained by our approach are EFMs. The authors of [Jevremović and Boley, 2013] only mention that the minimum set of EFMs is not unique. Here we discuss this issue in a systematic way and show how to obtain different MEMo's by splitting different sets of reversible reactions. Furthermore, we provide a MATLAB tool to compute MEMos and illustrate the impact of our method by computational results for various genome-scale metabolic networks.

6.7.4 Subsets of EFMs

When computing only a subset of EFMs and not the whole set, several issues arise. One drawback of computing a MEMo is that the resulting set of EFMs is not unique. This problem was already addressed by Schuster et al. in [Schuster et al., 2002]. We prove in Section 6.5 that there exists a bijection between the irreversible EFMs in a MEMo and the MMBs of the network. Thus, the set of active irreversible reactions is the same for all EFMs in all different MEMos, while the set of active reversible reactions is not, therefore giving rise to the non-uniqueness of MEMos. As the authors of [Schuster et al., 2002] point out, not all EFMs of biological relevance may be contained in a MEMo. This is a valid argument for computing the whole set of EFMs. However, for genome-scale metabolic networks this is often not feasible, or the number of EFMs is too large to search for relevant EFMs. In such cases, a MEMo is a good alternative for studying the underlying metabolic network because it is smaller but still represents the whole network.

A typical application for EFMs is the decomposition of a given flux vector into a set of EFMs. If the whole set of EFMs cannot be computed but a MEMo is available, we can decompose the flux vector into EFMs using this MEMo. The EFMs of a MEMo can provide new insight on the importance of certain pathways or reactions for the whole network, especially since their irreversible part is unique. In fact, investigating which EFMs are exchangeable and why, for each metabolic network individually is an interesting task. Doing so, it could be possible to study robustness of the underlying metabolic network. The EFMs of a MEMo could also be used to discover similarities to other networks by detecting subsets of EFMs which can be found in other networks as well.

6.8 Conclusion and further work

In this chapter we introduced the concept of a minimum set of elementary modes (MEMo) necessary to generate the flux cone of a metabolic network. We presented a method to compute these MEMos. We implemented our algorithm using MATLAB and showed that the size of MEMos often is by several orders of magnitude smaller than the number of EPs or EFMs. One drawback of the MEMos is that they are not unique because they depend on the set of fully reversible reactions which have to be split. However, one can show that the set of irreversible reactions involved in the MEMos is unique and that they will always be part of the MEMos independently of the reactions that are split. The biological relevance of the MEMos has to be further investigated. The need of having smaller sets of generating flux

6. Finding MEMo

vectors has been addressed by various papers [De Figueiredo et al., 2009; Kaleta et al., 2009; Schilling et al., 2000; Rezola et al., 2011; Jevremović and Boley, 2013; Arabzadeh et al., 2018]. We expect that MEMos give rise to new opportunities for analysing large genome-scale metabolic networks.

Chapter 7

Projections of flux cones

This chapter is based on a joint article with Alexander Bockmayr, which is currently under revision.

From EFMs of a metabolic network \mathcal{N} we come now to the *dual* of EFMs: to *minimal cut sets* (MCSs). EFMs are generators of the network and can be used to construct every pathway and every behaviour of the underlying network. MCSs for metabolic networks are sets of reactions which, if they are removed, prevent a target reaction from carrying flux, see Section 2.2.3 for a detailed introduction.

To compute MCSs different methods exist. Due to combinatorial explosion, most of them fail to compute MCSs of higher cardinality or fail to compute MCSs for genome-scale metabolic networks when they use EFMs.

We introduce in this chapter *irreversible minimal cut sets* (iMCSs). These are MCSs that consist of irreversible reactions only. The advantage of iMCSs is that instead of using EFMs one can use *minimal metabolic behaviours* (MMBs) to compute iMCSs. Since the number of MMBs can be by several orders of magnitude smaller than the number of EFMs, computing iMCSs of higher cardinality becomes possible for genome-scale metabolic networks. We present a new method to compute MMBs by projecting the original flux cone on the set of irreversible reactions, which reduces the problem size. Using oriented matroid theory, we give an efficient and computationally easy way for performing the projection.

In Chapter 8 we will use the projected flux cone and apply other methods besides using MMBs to compute iMCSs.

7.1 Introduction

MCSs were first introduced in [Klamt and Gilles, 2004] and have many applications [Wilhelm et al., 2004; Klamt, 2006; Imielinski and Belta, 2008; Clark and Verwoerd, 2012; Gruchattka et al., 2013; Machado and Herrgård, 2015; Harder et al., 2016; Gerstl et al., 2015; von Kamp and Klamt, 2017; Apalaza et al., 2017]. Various methods exist to compute MCSs [Jungreuthmayer et al., 2013; Nair et al., 2017; Haus et al., 2008; Goldstein and Bockmayr, 2015; Ballerstein et al., 2012; von Kamp and Klamt, 2014; Tobalina et al., 2016]. The method that we consider in this chapter [Klamt and Gilles, 2004; Klamt, 2006] is based on EFMs and an illustration for this procedure can be found in Example 2.12. Computing EFMs for genome-scale metabolic networks is often not possible, since the number of EFMs grows exponentially with the size of the network. Therefore the method described in Example 2.12 may not be applicable.

To address this problem, we propose here to use *projections* in order to reduce the size of the underlying polyhedral cones. The projection of the flux cone $\Gamma_{\mathcal{N}}$ onto the set of irreversible reactions Irr leads to a smaller cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$, which is again a flux cone, but where only irreversible reactions are active. To compute MCSs, we can apply the methods mentioned before to $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ instead of $\Gamma_{\mathcal{N}}$ in order to compute MCSs. The resulting cuts are MCSs of \mathcal{N} that consist of irreversible reactions only. Therefore, we call them *irreversible minimal cut sets* (iMCSs).

Definition 7.1 (Irreversible Minimal Cut Set). *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$, a target reaction $\text{tar} \in \mathfrak{R}$, and a flux rate $v_{\text{tar}}^* \in \mathbb{R}$. A set of reactions $\xi \subseteq \mathfrak{R}$ is called a cut set (with respect to the defined target reaction tar) if after the removal of ξ from the network \mathcal{N} it is no longer possible to achieve a flux rate of more than v_{tar}^* for tar . Thus there exists no $v \in \Gamma_{\mathcal{N}}$ with $v_{\text{tar}} > v_{\text{tar}}^*$ and $v_{\xi} = 0$. An irreversible minimal cut set (iMCS) is an (inclusion-)minimal cut set $\zeta \subseteq \text{Irr}$ consisting of irreversible reactions only.*

Note that an iMCS is always an MCS while the converse is not true. We will show that the EFMs of the projection $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ are the *minimal metabolic behaviours* (MMBs) of the network \mathcal{N} . We can use MMBs in the same way as EFMs (see Example 2.12) in order to compute iMCSs instead of MCSs.

Suppose the whole set of MMBs is known and an irreversible target is chosen. The set MMB_{tar} consists of all MMBs which contain the target reaction. Within MMB_{tar} minimal hitting sets are searched which are MCSs. Since they consist of irreversible reactions only, they are iMCSs. Note that iMCSs

found with this technique do not allow any flux through the target reaction, i.e., $v_{\text{tar}} = 0$ for all $v \in \Gamma_{\mathcal{N}}$ with $v_{\xi} = 0$.

Example 7.2 (Using MMBs to compute iMCSs). *The network in Figure 1.1 has three different MMBs: $\{1, 2\}$, $\{6, 7\}$, and $\{6, 8\}$. We apply the method from Example 2.12 to compute MCSs, but instead of EFMs we now use MMBs. The target reaction is reaction 7. The only MMB containing reaction 7 is $\{6, 7\}$. Therefore, the iMCSs for reaction 7 are $\{6\}$ and $\{7\}$.*

The classical method to perform projections is Fourier-Motzkin elimination, see e.g. [Ziegler, 1995]. In general, Fourier-Motzkin elimination leads to a combinatorial explosion in the number of linear inequalities needed to describe the projected cone. However, in the special case of flux cones, we can use the theory of oriented matroids to avoid this if we project along reversible reactions on the set of irreversible reactions. This enables us to perform the projection efficiently also on genome-scale metabolic networks.

7.2 Methods

Most of the basic definitions and theorems utilised in the following can be found in Section 2.3.1. We consider here a special class of polyhedral cones, the *flux cones*, see Definition 2.26.

7.2.1 Computing minimal metabolic behaviours via projection

Up to now there are three ways for enumerating all MMBs in a metabolic network. The first one is to compute all EFMs and to search for the MMBs contained in them. Since this method involves enumerating the whole set of EFMs, it is usually not applicable to genome-scale networks. A closely related approach is to compute a minimum set of generators for the flux cone, e.g., by using the double description method [Fukuda and Prodon, 1996]. The supports of the generators that do not belong to the lineality space then correspond to the MMBs [Larhlimi and Bockmayr, 2009]. A third method is using mixed integer linear programming [Rezola et al., 2011] to compute so-called generating flux modes (GFMs), from which again the MMBs can be obtained.

In the following, we introduce a new method to compute the set of all MMBs using projection of polyhedral cones. We show that the set of MMBs can be obtained as the set of supports of the extreme rays of the projected cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$.

7. Projections of flux cones

Definition 7.3 (Projection). For $x \in \mathbb{R}^n$ and $H \subseteq \{1, \dots, n\}$ the projection of x on H is defined as $\text{proj}_H(x) = (x_H, 0)$. Similarly, for a polyhedral cone $\Gamma \subseteq \mathbb{R}^n$, we define $\text{proj}_H(\Gamma) = \{\text{proj}_H(x) \mid x \in \Gamma\}$.

It is well-known that $\text{proj}_H(\Gamma)$ is again a polyhedral cone [Ziegler, 1995]. Projections of the flux cone of a metabolic network have been considered before in the literature [Wiback and Palsson, 2002; Covert and Palsson, 2003; Urbanczik and Wagner, 2005b; Wagner and Urbanczik, 2005; Urbanczik, 2006; Marashi et al., 2012]. Here we show that the projection $\text{proj}_H(\Gamma_{\mathcal{N}})$ of the flux cone $\Gamma_{\mathcal{N}}$ of a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ will be again a flux cone in the sense of Definition 2.26, if we project along one or more reversible reactions on a superset $H \supseteq \text{Irr}$ of the set of irreversible reactions.

Theorem 7.4. Let $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$ be a flux cone with $A = \begin{pmatrix} B \\ -B \\ E_{I,*} \end{pmatrix}$.

For any $H \supseteq I$ the projection $\text{proj}_H(\Gamma)$ is again a flux cone.

Proof. For any $k \in \{1, \dots, n\}$ the projection in direction of k on the set $H_k = \{1, \dots, n\} \setminus \{k\}$ can be realised by Fourier-Motzkin elimination, see e.g. [Ziegler, 1995]. Here a matrix $A^{/k}$ is constructed such that $\text{proj}_{H_k}(\Gamma) = \{x \in \mathbb{R}^n \mid A^{/k}x \geq 0, x_k = 0\}$. The matrix $A^{/k}$ contains the following rows:

- the rows $A_{i,*}$ from A , for all i with $a_{i,k} = 0$.
- the rows $a_{i,k}A_{j,*} + (-a_{j,k})A_{i,*}$, for all i, j with $a_{i,k} > 0, a_{j,k} < 0$.

For the sake of convenience we divide the row indices of A into three sets:

$$J^0 = \{i \mid a_{i,k} = 0\}, \quad J^+ = \{i \mid a_{i,k} > 0\}, \quad J^- = \{i \mid a_{i,k} < 0\}.$$

If $k \notin I$ then J^0 will contain all rows corresponding to $E_{I,*}$. J^0 can contain rows corresponding to B as well. Furthermore we have $A_{J^+,*} = -A_{J^-,*}$ because A describes a flux cone. Following from this, the rows of $A^{/k}$ can

be ordered such that $A^{/k} = \begin{pmatrix} C \\ -C \\ E_{I,*} \end{pmatrix}$ and therefore $\text{proj}_{H_k}(\Gamma)$ is a flux cone

as well. Repeating this construction for all $k \notin H \supseteq I$, we conclude that $\text{proj}_H(\Gamma)$ is again a flux cone. \square

In the following, we consider the special case of flux cones $\Gamma = \Gamma_{\mathcal{N}}$ and $H = \text{Irr}$. This means that we project along all the reversible reactions on the set of all irreversible reactions. By Theorem 7.4, the projection $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ is again a flux cone in the sense of Definition 2.26. Every vector $(v_{\text{Irr}}, 0) \in \text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ can be lifted to a vector $v = (v_{\text{Irr}}, v_{\text{Rev}}) \in \Gamma_{\mathcal{N}}$ of the

original network by adding suitable components for the reversible reactions. The flux cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ is pointed since only irreversible reactions can be active. However, $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ does not necessarily describe a metabolic network. An example is given in Figure 7.1, which shows that projecting the flux cone along a reversible reaction is different from deleting this reaction from the network.

We next show that the MMBs of a metabolic network \mathcal{N} can be obtained from the extreme rays of the pointed cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$.

Theorem 7.5. $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ be the flux cone of a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$. The supports of the extreme rays of the pointed cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ are exactly the minimal metabolic behaviours of the network \mathcal{N} .

Proof. By Theorem 7.4 the cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ is a flux cone. Furthermore, we have $v \geq 0$ for all $v \in \text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$, because $v_{\text{Irr}} \geq 0$ and $v_{\text{Rev}} = 0$. Thus $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ is pointed and we may apply Proposition 2.27 to conclude that the extreme rays of $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ are the rays in $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ with minimal support. Since the supports of the rays in $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ are just the metabolic behaviours in \mathcal{N} , the result follows. \square

Theorem 7.5 leads to a new approach for computing the set of MMBs of a metabolic network \mathcal{N} . First compute an inequality description of the projection $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$, then compute the extreme rays of $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ and their supports.

7. Projections of flux cones

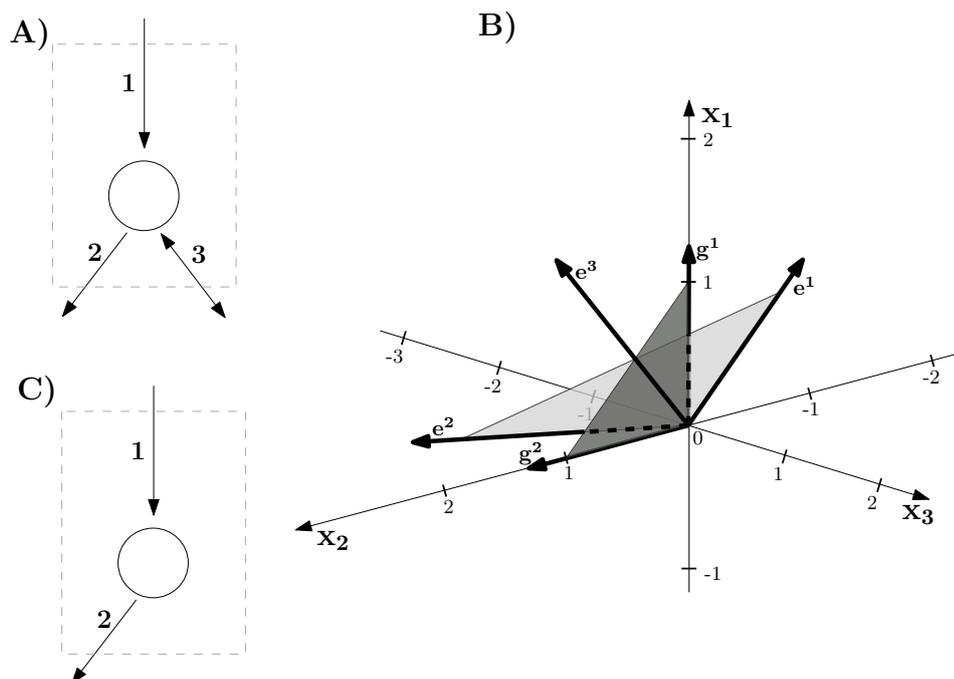


Figure 7.1: **A)** Metabolic network \mathcal{N} involving three reactions, where the third one is reversible. The network has three EFMs: $e^1 = (1, 0, 1)$, $e^2 = (0, 1, -1)$, and $e^3 = (1, 1, 0)$. **B)** The corresponding flux cone $\Gamma = \{x \in \mathbb{R}^3 \mid x_1 - x_2 - x_3 = 0, x_1, x_2 \geq 0\}$, shown in light grey, is a pointed cone and spanned by the EFMs e^1 and e^2 , which are the extreme rays of the cone. The third EFM e^3 lies inside Γ and is a conic combination of e^1 and e^2 . The lineality space is $\{0\}$. Projecting Γ in direction of the reversible reaction 3 results in the projected cone $\text{proj}_{\{1,2\}}(\Gamma) = \{x \in \mathbb{R}^3 \mid x_3 = 0, x_1, x_2 \geq 0\}$, shown in dark grey. It is again a pointed cone and is generated by the vectors $g^1 = (1, 0, 0)$ and $g^2 = (0, 1, 0)$, which correspond to the two MMBs $\{1\}$ and $\{2\}$ of \mathcal{N} . **C)** Removing reaction 3 results in the network in C), which has only one EFM, namely e^1 .

| network | rxns | irr | MMBs | EFMs | Time MMBs | Target MMBs | iMCSs | Time iMCSs |
|--|------|-----|---------|---------|-----------|-------------|-------|------------|
| <i>Escherichia coli</i> MG1655 [Orth et al., 2010; King et al., 2016] | 87 | 41 | 2,572 | 100,274 | 5.7s | 1151 | 257 | 113s |
| <i>Rhizobium etli</i> iOR363 [Resendis-Antonio et al., 2007; Li et al., 2010] | 194 | 104 | 6147 | N/A | 26s | 651 | 60 | 60s |
| <i>Buchnera</i> iSM197 [Li et al., 2010; MacDonald et al., 2011] | 244 | 170 | 245 | N/A | 7s | 146 | 200 | 130s |
| <i>Blattabacterium cuenoti</i> iCG238 [González-Domenech et al., 2012] | 308 | 197 | 374 | N/A | 7.7s | 128 | 184 | 138s |
| <i>Blattabacterium</i> iCG230 [Li et al., 2010; González-Domenech et al., 2012] | 400 | 192 | 82 | N/A | 6.7s | 32 | 159 | 115s |
| <i>Mycobacterium tuberculosis</i> iNJ661 [Jamshidi and Palsson, 2007; Li et al., 2010] | 427 | 296 | 501 | N/A | 13s | 350 | 381 | 449s |
| <i>Salmonella Typhimurium</i> STM.v1.0 [Li et al., 2010; Thiele et al., 2011] | 458 | 316 | 97 | N/A | 11.8s | 96 | 321 | 386s |
| <i>Helicobacter pylori</i> iCS291 [Schilling et al., 2002; Li et al., 2010] | 444 | 271 | 150,132 | N/A | 325s | 148,608 | 187 | 5h44min1s |

Table 7.1: Comparison of the number of EFMs, number of MMBs, number of target MMBs, number of iMCSs, and time to compute these sets for different networks. **network**: name of the metabolic network. **rxns**: number of unblocked reactions of the network. **irr**: number of unblocked irreversible reactions of the network. **MMBs**: number of minimal metabolic behaviours. **EFMs**: number of the elementary flux modes. **N/A**: The programs used here (`POLCO` [Terzer, 2017b] and `EFMTOOL` [Terzer, 2017a]) could not handle the size of the models resp. the number of EFMs. **Time MMBs**: Time to compute MMBs using projection. **Target MMBs**: number of MMBs involving the target reaction (biomass reaction). **iMCS**: number of irreversible minimal cut sets. **Time iMCSs**: Time to compute iMCSs.

7. Projections of flux cones

7.2.2 Projection and contraction

In this section, we apply the theory of oriented matroids to give an efficient procedure for computing an inequality description of the cone $\text{proj}_{\text{IR}}(\Gamma_{\mathcal{N}})$.

The classical method for projection of polyhedral cones is Fourier-Motzkin elimination [Ziegler, 1995]. In general, this can lead to a combinatorial explosion in the number of linear inequalities needed to describe the projected cone. In our case we are concerned with the special case flux cones of the

form $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$ with $A = \begin{pmatrix} B \\ -B \\ E_{I,*} \end{pmatrix}$, and we always project

along components $k \notin I$. As we have seen in the proof of Theorem 7.4, this implies that all changes in the row structure of A during projection will occur in the equational part $Bx = 0$, while the matrix $E_{I,*}$ representing the linear inequalities is not be changed. Since the homogeneous linear equation system $Bx = 0$ describes an oriented matroid [Björner, 1999], see Definition 2.46, the projection of flux cones can be performed efficiently via contraction of oriented matroids.

We denote an oriented matroid as $\mathcal{M} = (U, \mathcal{C})$ where U is a set and \mathcal{C} is the family of oriented circuits of \mathcal{M} . The equivalent of projection for oriented matroids is called *contraction*:

Proposition 7.6 (Proposition 3.3.2 in [Ziegler, 1995] on page 110). *Let $\mathcal{M} = (U, \mathcal{C})$ be an oriented matroid. For $X \in \mathcal{C}$ and $H \subseteq U$ we denote by $X|_H$ the signed set $(X^+ \cap H, X^- \cap H)$. The family $\text{Min}(\{X|_H \mid X \in \mathcal{C}\})$ of non-empty intersections of circuits of \mathcal{M} with H that have inclusion-minimal support is the set of circuits of an oriented matroid on H . This matroid is called the contraction of \mathcal{M} to H and will be denoted by $\text{contr}_H(\mathcal{M})$.*

In our case, we consider a special class of matroids which are called representable. Let $B \in \mathbb{R}^{m \times U}$ be a matrix and let U be the set of columns of B . The *oriented matroid* $\mathcal{M}_B = (U, \mathcal{C})$ represented by B is given by $\mathcal{C} = \text{Min}(\mathcal{V})$, where $\mathcal{V} = \{\sigma(x) \mid Bx = 0, x \in \mathbb{R}^U\}$. Here $\sigma(x) \in \{-, 0, +\}^U$ denotes the signed set obtained by applying the sign function component-wise to the vector x , and $\text{Min}(\mathcal{V})$ is the set of non-empty signed sets in \mathcal{V} with inclusion-minimal support.

We next state one lemma and three propositions needed to prove Theorem 7.11.

Lemma 7.7. *Let $\mathcal{A} \subseteq \mathcal{B} \subseteq \{+, 0, -\}^U$ be families of signed subsets of a set U . If $X \in \text{Min}(\mathcal{B})$ and $X \in \mathcal{A}$, then $X \in \text{Min}(\mathcal{A})$.*

Proof. The proof is obvious. \square

Proposition 7.8. *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$, let $\text{contr}_{\text{Irr}}(\mathcal{M}_S) = (\text{Irr}, \mathcal{C}_{\text{Irr}})$ be the contraction of the oriented matroid $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ to Irr . For families of signed subsets $\mathcal{D} \subseteq \{+, 0, -\}^{\mathfrak{R}}$ and $\mathcal{T} \subseteq \{+, 0, -\}^{\mathfrak{R}}$ let $\mathcal{D}|_{\text{Irr}} = \{Y|_{\text{Irr}} \in \mathcal{D}\}$ and $\mathcal{T}^{\text{pos}} = \{X \in \mathcal{T} \mid X^- = \emptyset\}$. Then*

$$(\text{Min}(\mathcal{C}|_{\text{Irr}}))^{\text{pos}} = \text{Min}((\mathcal{C}|_{\text{Irr}})^{\text{pos}}).$$

Proof. \subseteq : $(\mathcal{C}|_{\text{Irr}})^{\text{pos}} \subseteq \mathcal{C}|_{\text{Irr}}$ and Lemma 7.7.

\supseteq : Let $X \in \text{Min}((\mathcal{C}|_{\text{Irr}})^{\text{pos}})$. Then $X \in (\mathcal{C}|_{\text{Irr}})^{\text{pos}} \subseteq \mathcal{C}|_{\text{Irr}}$ and $X^- = \emptyset$. Suppose X is not minimal in $\mathcal{C}|_{\text{Irr}}$. Then there exists $\tilde{X} \in \mathcal{C}|_{\text{Irr}}$ with $\text{supp}(\tilde{X}) = \tilde{X}^+ \cup \tilde{X}^- \subsetneq \text{supp}(X) = X^+ \cup X^- = X^+$ and in particular $\tilde{X} \neq \pm X$.

Without loss of generality, we can choose $\tilde{X} \in \{W \in \mathcal{C}|_{\text{Irr}} \mid \text{supp}(W) \subsetneq \text{supp}(X)\}$ such that \tilde{X}^- is inclusion-wise minimal.

Case 1: $\tilde{X} \in (\mathcal{C}|_{\text{Irr}})^{\text{pos}}$. This implies that X is not minimal in $(\mathcal{C}|_{\text{Irr}})^{\text{pos}}$, which is a contradiction.

Case 2: $\tilde{X} \notin (\mathcal{C}|_{\text{Irr}})^{\text{pos}}$. Then $\tilde{X}^- \neq \emptyset$ while $X^- = \emptyset$.

Let $Y, \tilde{Y} \in \mathcal{C}$ such that $Y|_{\text{Irr}} = X$ and $\tilde{Y}|_{\text{Irr}} = \tilde{X}$.

Since $\tilde{X}^- \neq \emptyset$ and $\tilde{X}^+ \cup \tilde{X}^- \subsetneq X^+$, there exists $u \in X^+ \cap \tilde{X}^- \subseteq Y^+ \cap \tilde{Y}^-$. From $\tilde{X} \neq X$ we get $\tilde{Y} \neq \pm Y$.

By matroid axiom (4), there exists $Z \in \mathcal{C}$ with $Z^+ \subset (Y^+ \cup \tilde{Y}^+) \setminus \{u\}$ and $Z^- \subset (Y^- \cup \tilde{Y}^-) \setminus \{u\}$. It follows

- $Z_{\text{Irr}}^+ \subset (X^+ \cup \tilde{X}^+) \setminus \{u\} = (X^+ \setminus \{u\}) \cup \tilde{X}^+$
- $Z_{\text{Irr}}^- \subset (X^- \cup \tilde{X}^-) \setminus \{u\} = \tilde{X}^- \setminus \{u\}$
- $\text{supp}(Z_{\text{Irr}}) = (Z_{\text{Irr}}^+ \cup Z_{\text{Irr}}^-) \subseteq [(X^+ \cup \tilde{X}^+) \cup (X^- \cup \tilde{X}^-)] \setminus \{u\} = [(X^+ \cup X^-) \cup (\tilde{X}^+ \cup \tilde{X}^-)] \setminus \{u\} = X^+ \setminus \{u\} \subsetneq X^+$.

Thus we found $W = Z_{\text{Irr}} \in \mathcal{C}|_{\text{Irr}}$ with $\text{supp}(W) \subseteq \text{supp}(X)$ and $W^- \subsetneq \tilde{X}^-$, contradicting the choice of \tilde{X}^- . \square

Proposition 7.9. *Given a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and a signed set $Y \in \{+, 0, -\}^{\mathfrak{R}}$ we have $Y \in \text{Min}(\{\sigma(v) \mid Sv = 0, v \in \mathbb{R}^{\mathfrak{R}}\}) \wedge Y^- \cap \text{Irr} = \emptyset$ if and only if $Y \in \text{Min}(\{\sigma(v) \mid Sv = 0, v_{\text{Irr}} \geq 0\})$.*

Proof. \Rightarrow : Lemma 7.7

\Leftarrow : Let $Y \in \text{Min}(\{\sigma(v) \mid Sv = 0, v_{\text{Irr}} \geq 0\})$. Suppose $Y \notin \text{Min}(\{\sigma(v) \mid Sv = 0, v \in \mathbb{R}^{\mathfrak{R}}\})$.

Then there exists $\tilde{Y} \in \{\sigma(v) \mid Sv = 0, v \in \mathbb{R}^{\mathfrak{R}}\}$ with $\text{supp}(\tilde{Y}) \subsetneq \text{supp}(Y)$

7. Projections of flux cones

and in particular $\tilde{Y} \neq \pm Y$.

Without loss of generality, choose \tilde{Y} such that $\tilde{Y}^- \cap \text{Irr}$ is inclusion-minimal. If $\tilde{Y}^- \cap \text{Irr} = \emptyset$, then Y would not be minimal in $\{\sigma(v) \mid Sv = 0, v_{\text{Irr}} \geq 0\}$, which is a contradiction.

Thus $\tilde{Y}^- \cap \text{Irr} \neq \emptyset$ and $(\tilde{Y}^+ \cup \tilde{Y}^-) \cap \text{Irr} \subseteq (Y^+ \cap Y^-) \cap \text{Irr} = Y^+ \cap \text{Irr}$. It follows that there exists $u \in \text{Irr}$ with $u \in Y^+ \cap Y^-$.

By matroid axiom (4), there exists $Z \in \{\sigma(v) \mid Sv = 0, v \in \mathbb{R}^{\mathfrak{A}}\}$ with $Z^+ \subseteq (Y^+ \cup \tilde{Y}^+) \setminus \{u\}$ and $Z^- \subseteq (Y^- \cup \tilde{Y}^-) \setminus \{u\}$.

Then $\text{supp}(Z) \subseteq (\text{supp}(Y) \cup \text{supp}(\tilde{Y})) \setminus \{u\} \subseteq \text{supp}(Y) \setminus \{u\}$ and $Z^- \cap \text{Irr} \subseteq ((Y^- \cap \text{Irr}) \cup (\tilde{Y}^- \cap \text{Irr})) \setminus \{u\} = (\tilde{Y}^- \cap \text{Irr}) \setminus \{u\} \subsetneq \tilde{Y}^- \cap \text{Irr}$, contradicting the minimality assumption in the choice of \tilde{Y} . \square

Proposition 7.10. *In a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{A}, S, \text{Irr})$ we have $\text{Min}(\{Y|_{\text{Irr}} \mid Y \in \text{Min}(\{\sigma(v) \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{A}}\})\}) = \text{Min}(\{\sigma(v)|_{\text{Irr}} \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{A}}\})$.*

Proof. \subseteq : Let $X \in \text{Min}(\{Y|_{\text{Irr}} \mid Y \in \text{Min}(\{\sigma(v) \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{A}}\})\})$ and suppose $X \notin \text{Min}(\{\sigma(v)|_{\text{Irr}} \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{A}}\})$. Then there exists $v \in \Gamma_{\mathcal{N}}$ with $\text{supp}(\sigma(v)|_{\text{Irr}}) = \text{supp}(v) \cap \text{Irr} \subseteq X$. Furthermore, there exists $w \in \Gamma_{\mathcal{N}}$ with $\text{supp}(w) \subseteq \text{supp}(v)$ such that $Y = \text{supp}(w) \in \text{Min}(\{\sigma(v)|_{\text{Irr}} \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{A}}\})$ is inclusion-minimal.

It follows that $Y|_{\text{Irr}} = \text{supp}(w) \cap \text{Irr} \subseteq \text{supp}(v) \cap \text{Irr} \subsetneq X$, contradicting the minimality of X .

\supseteq : Let $X \in \text{Min}(\{\sigma(v)|_{\text{Irr}} \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{A}}\})$.

Let $v \in \Gamma_{\mathcal{N}}$ with $X = \sigma(v)|_{\text{Irr}}$.

There exists $w \in \Gamma_{\mathcal{N}}$ with $\text{supp}(w) \subseteq \text{supp}(v)$ and $Y = \text{supp}(w) \in \text{Min}(\{\sigma(v)|_{\text{Irr}} \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{A}}\})$ is inclusion-minimal.

Then $Y|_{\text{Irr}} = \text{supp}(w) \cap \text{Irr} \subseteq \text{supp}(v) \cap \text{Irr} \subseteq X$ and by the minimality of X it follows $Y|_{\text{Irr}} = X$. \square

Using the previous lemma and propositions we can now prove our main theorem:

Theorem 7.11. *The minimal metabolic behaviours of a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{A}, S, \text{Irr})$ with flux cone $\Gamma_{\mathcal{N}}$ are exactly the oriented circuits X of the contraction $\text{contr}_{\text{Irr}}(\mathcal{M}_S)$ for which $X^- = \emptyset$. If $T \in \mathbb{R}^{k \times \text{Irr}}$ is a matrix representing $\text{contr}_{\text{Irr}}(\mathcal{M}_S)$ then*

$$\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}}) = \{v \in \mathbb{R}^{\mathfrak{A}} \mid Tv_{\text{Irr}} = 0, v_{\text{Irr}} \geq 0, v_{\text{Rev}} = 0\}.$$

Proof. Let $\text{contr}_{\text{Irr}}(\mathcal{M}_S) = (\text{Irr}, \mathcal{C}_{\text{Irr}})$ be the contraction of $\mathcal{M}_S = (\mathfrak{A}, \mathcal{C})$ to

Irr. Let $\mathcal{C}_{\text{Irr}}^{\text{pos}}$ be the family of circuits $X \in \mathcal{C}_{\text{Irr}}$ with $X^- = \emptyset$. We have

$$\begin{aligned}
 \mathcal{C}_{\text{Irr}}^{\text{pos}} &= \text{Min}(\{Y|_{\text{Irr}} \mid Y \in \mathcal{C}\}) \cap \{X \mid X^- = \emptyset\} \\
 &\stackrel{\text{Prop.7.8}}{=} \text{Min}(\{Y|_{\text{Irr}} \mid Y \in \mathcal{C}, Y^- \cap \text{Irr} = \emptyset\}) \\
 &= \text{Min}(\{Y|_{\text{Irr}} \mid Y \in \text{Min}(\{\sigma(v) \mid Sv = 0, v \in \mathbb{R}^{\mathfrak{R}}\}), Y^- \cap \text{Irr} = \emptyset\}) \\
 &\stackrel{\text{Prop.7.9}}{=} \text{Min}(\{Y|_{\text{Irr}} \mid Y \in \text{Min}(\{\sigma(v) \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{R}}\})\}) \\
 &\stackrel{\text{Prop.7.10}}{=} \text{Min}(\{\sigma(v)|_{\text{Irr}} \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{R}}\})
 \end{aligned}$$

The last set just defines the MMBs in \mathcal{N} . If T represents $\text{contr}_{\text{Irr}}(\mathcal{M}_S)$, then by definition $\mathcal{C}_{\text{Irr}} = \text{Min}(\{\sigma(x) \mid Tx = 0, x \in \mathbb{R}^{\text{Irr}}\})$. It follows that

$$\begin{aligned}
 \mathcal{C}_{\text{Irr}}^{\text{pos}} &= \text{Min}(\{\sigma(x) \mid Tx = 0, x \in \mathbb{R}^{\text{Irr}}\}) \cap \{X \in \mathcal{C}_{\text{Irr}} \mid X^- = \emptyset\} \\
 &\stackrel{\text{Prop.7.9}}{=} \text{Min}(\{\sigma(x) \mid Tx = 0, x \geq 0, x \in \mathbb{R}^{\text{Irr}}\}) \\
 &= \text{Min}(\{\sigma(v) \mid Tv_{\text{Irr}} = 0, v_{\text{Irr}} \geq 0, v_{\text{Rev}} = 0, v \in \mathbb{R}^{\mathfrak{R}}\}) \\
 &= \text{Min}(\{\text{supp}(v) \mid Tv_{\text{Irr}} = 0, v_{\text{Irr}} \geq 0, v_{\text{Rev}} = 0, v \in \mathbb{R}^{\mathfrak{R}}\}) \\
 &= \text{Min}(\{\sigma(v)|_{\text{Irr}} \mid Sv = 0, v_{\text{Irr}} \geq 0, v \in \mathbb{R}^{\mathfrak{R}}\}) \\
 &= \text{Min}(\{\text{supp}(\text{proj}_{\text{Irr}}(v)) \mid v \in \Gamma_{\mathcal{N}}\}) \\
 &= \text{Min}(\{\text{supp}(w) \mid w \in \text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})\})
 \end{aligned}$$

Thus in $\mathbb{R}^{\mathfrak{R}}$, the two pointed cones $\tilde{\Gamma} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Tv_{\text{Irr}} = 0, v_{\text{Irr}} \geq 0, v_{\text{Rev}} = 0\}$ and $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ have the same set of minimal support vectors. By Proposition 2.27 they have the same set of ERs and therefore are identical. \square

Example 7.12 (Compute MMBs using the contracted matrix). Consider again the network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ in Figure 1.1. Let $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ be the oriented matroid represented by the stoichiometric matrix $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$. This matroid contains 24 oriented circuits, where we identify two oriented circuits X and Y if $X = -Y$. Among those, 18 correspond to EFMs, for example $\{+1, +2, +3, +4\}$ or $\{-4, +6, +7\}$. The remaining 6 oriented circuits $\{-7, +8, +9, +10\}$, $\{+1, +2, +4, +5, +7, -8\}$, $\{-3, +5, +7, -8\}$, $\{+1, +2, +5, +6, +7, -8\}$, $\{+1, +2, +4, +5, -6, -8\}$, $\{-7, +8, +11, +12\}$ do not correspond to a feasible flux vector. In each of these, there is an irreversible reaction with negative flux. Contracting \mathcal{M}_S on Irr results in the oriented matroid $\text{contr}_{\text{Irr}}(\mathcal{M}_S)$ represented by the matrix

$$T = \begin{matrix} & \begin{matrix} 1 & 2 & 6 & 7 & 8 \end{matrix} \\ \begin{matrix} A \\ E \end{matrix} & \begin{pmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 \end{pmatrix} \end{matrix}.$$

By Theorem 7.11, the matrix T also describes the flux cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}}) = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Tv_{\text{Irr}} = 0, v_{\text{Irr}} \geq 0, v_{\text{Rev}} = 0\}$, in which all active reactions are irreversible.

By Proposition 2.27, the extreme rays of $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ are exactly the EFMs of the network represented by $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$, which is shown in Figure 7.2.

7. Projections of flux cones

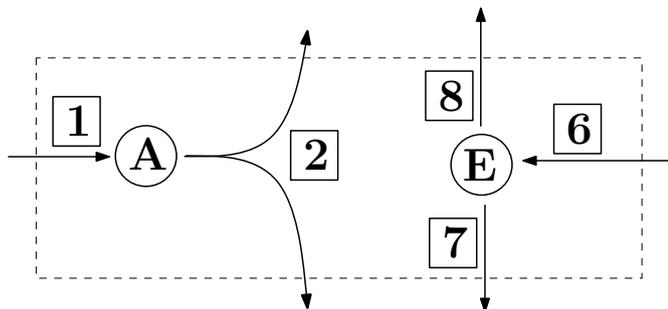


Figure 7.2: Projection of the network in Figure 1.1 on the set of irreversible reactions. Reactions 1, 2 are independent from reactions 6, 7, 8. The metabolites that were connected to the reversible reactions have been removed.

The supports of these extreme rays are $\{1, 2\}$, $\{6, 7\}$, $\{6, 8\}$, which are the MMBs of the original network from Figure 1.1. The oriented circuits of $\text{contr}_{\text{Irr}}(\mathcal{M}_S)$ include these MMBs (as signed sets) and one additional oriented circuit $\{-7, +8\}$.

Thus, in order to compute the matrix T in Theorem 7.11 that describes the projected flux cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$, we may use contraction of oriented matroids. For an oriented matroid \mathcal{M}_B represented by the matrix $B \in \mathbb{R}^{m \times n}$, computing the matrix representing the contraction can be done by basic linear algebra as we emphasise in the following.

7.2.2.1 Contraction via deletion in the dual matroid

Proposition 7.13 (Proposition 3.3.1 in [Björner, 1999] on page 110). *Let $\mathcal{M} = (U, \mathcal{C})$ be an oriented matroid and let $A \subseteq U$. The family $\mathcal{C} \setminus A = \{X \in \mathcal{C} \mid \text{supp}(X) \subseteq U \setminus A\}$ is the set of circuits of an oriented matroid $\mathcal{M} \setminus A = (U \setminus A, \mathcal{C} \setminus A)$, which is called the deletion of A from \mathcal{M} .*

An alternative notation for the contraction $\text{contr}_H(\mathcal{M})$ of $\mathcal{M} = (U, \mathcal{C})$ on $H \subseteq U$ is to write \mathcal{M}/A with $A = U \setminus H$ [Björner, 1999, p. 111].

Proposition 7.14 (Proposition 3.4.9 in [Björner, 1999] on page 123). *Let $\mathcal{M} = (U, \mathcal{C})$ be an oriented matroid on a set U and let A be a subset of U . Then*

$$\begin{aligned} (\mathcal{M} \setminus A)^* &= \mathcal{M}^*/A \\ (\mathcal{M}/A)^* &= \mathcal{M}^* \setminus A, \end{aligned}$$

where \mathcal{M}^* is the dual oriented matroid of \mathcal{M} .

Since $\mathcal{M}^{**} = \mathcal{M}$, the contraction of \mathcal{M} on $H = U \setminus A$ can be realised by the deletion of A in the dual matroid \mathcal{M}^* .

If an oriented matroid $\mathcal{M}_B = (U, \mathcal{C})$ is represented by a matrix $B \in \mathbb{R}^{m \times U}$, then the dual matroid \mathcal{M}_B^* can be obtained in the following way [Ziegler, 1995, p.166]: Suppose the matrix B representing \mathcal{M}_B is given as $B = (E_r \mid Q)$, where E_r is the identity matrix of r elements. Then the dual matroid \mathcal{M}_B^* is represented by the matrix $C = (-Q^\top \mid E_{n-r})$. Thus to compute the dual of the oriented matroid \mathcal{M}_B we have to bring B to the form $B = (E_r \mid Q)$, which is computationally easy. To do so, we can use Gaussian elimination [Artin, 1998; Gilbert and Gilbert, 2014]. By this, the row space is not changed, thus $B = (E_r \mid Q)$ describes the same matroid as before Gaussian elimination was applied.

To perform the contraction of \mathcal{M}_B on $H = U \setminus A$, we delete A in \mathcal{M}_B^* . This is done by removing in the matrix $C \in \mathbb{R}^{U \times m}$ that represents \mathcal{M}_B^* the columns corresponding to $A \subset U$. Finally, we compute $(\mathcal{M}^* \setminus A)^* = (\mathcal{M}/A)$. With this approach, no combinatorial explosion will occur like in Fourier-Motzkin elimination.

Example 7.15 (Contraction via deletion in the dual matroid). *The oriented matroid $\mathcal{M}_S = (\mathfrak{R}, \mathcal{C})$ we consider here is given by the set of reactions $\mathfrak{R} = \{1, \dots, 12\}$ of the network in Figure 1.1, thus \mathcal{M}_S is a flux-mode matroid, see Definition 2.53. The stoichiometric matrix $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$ is the following:*

$$S = \begin{matrix} & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} & \boxed{9} & \boxed{10} & \boxed{11} & \boxed{12} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \\ G \end{matrix} & \begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \end{pmatrix} \end{matrix}.$$

To compute the dual matroid \mathcal{M}_S^ we first have to bring S to the form $(E_r \mid Q)$ which can be done by first bringing the matrix into the row reduced echelon form e.g. using Gaussian elimination [Artin, 1998; Gilbert and Gilbert, 2014]. The result is the following matrix:*

$$S^{\text{rref}} = \begin{matrix} & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} & \boxed{5} & \boxed{6} & \boxed{7} & \boxed{8} & \boxed{9} & \boxed{10} & \boxed{11} & \boxed{12} \\ & \begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & -1 & -2 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & -1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \end{pmatrix} \end{matrix}.$$

7. Projections of flux cones

Since we want to have an identity matrix in the front we have to change the order of the columns of S^{rref} :

$$B = \begin{pmatrix} \boxed{1} & \boxed{2} & \boxed{3} & \boxed{5} & \boxed{6} & \boxed{9} & \boxed{11} & \boxed{4} & \boxed{7} & \boxed{8} & \boxed{10} & \boxed{12} \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & -1 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & -1 & -2 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}$$

$\underbrace{\hspace{10em}}_{E_r} \qquad \underbrace{\hspace{10em}}_Q$

The dual matroid \mathcal{M}_S^* is represented by the matrix $C = (-Q^\top \mid E_{n-r})$, where $B = (E_r \mid Q)$. Thus

$$C = \begin{pmatrix} \boxed{1} & \boxed{2} & \boxed{3} & \boxed{5} & \boxed{6} & \boxed{9} & \boxed{11} & \boxed{4} & \boxed{7} & \boxed{8} & \boxed{10} & \boxed{12} \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 2 & -1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

$\underbrace{\hspace{10em}}_{-Q^\top} \qquad \underbrace{\hspace{10em}}_{E_{n-r}}$

In order to contract \mathcal{M}_S on $\text{Irr} = \mathfrak{R} \setminus \text{Rev} = \{3, 4, 5, 9, 10, 11, 12\}$ in \mathcal{M}_S^* , resp. the corresponding columns in C :

$$C' = \begin{pmatrix} \boxed{1} & \boxed{2} & \boxed{6} & \boxed{7} & \boxed{8} \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

C' represents the matroid $\mathcal{M}_S^* \setminus \text{Rev}$. Since it holds $(\mathcal{M}_S^* \setminus \text{Rev})^* = (\mathcal{M}_S/\text{Rev})$ we have to compute the dual of $\mathcal{M}_S^* \setminus \text{Rev}$ as we did for \mathcal{M}_S . We first compute the reduced row echelon form:

$$C'^{\text{rref}} = \begin{pmatrix} \boxed{1} & \boxed{2} & \boxed{6} & \boxed{7} & \boxed{8} \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & -1 \end{pmatrix}$$

The last two rows are zero so we can omit them. The matrix Q consists of the columns corresponding to 2 and 8. Therefore the matrix representing the

matroid \mathcal{M}_S/Rev is

$$T = \begin{pmatrix} \boxed{2} & \boxed{8} & \boxed{1} & \boxed{6} & \boxed{7} \\ -1 & 0 & 1 & 0 & 0 \\ 0 & -1 & 0 & 1 & -1 \end{pmatrix}.$$

Which is T in Example 7.12 after reordering the columns.

In our computational experiments, we used for this operation the software **SAGE** [The Sage Developers, 2016]. Once the matrix T has been obtained, we can compute the extreme rays of the cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$. By Theorem 7.5 their supports are exactly the MMBs of \mathcal{N} .

Given the set of MMBs, computing iMCSs becomes a hitting set problem, see the following subsection and [Klamt, 2006].

7.2.2.2 Computing iMCSs as hitting sets

Given the set of MMBs of a network \mathcal{N} , computing the iMCSs of \mathcal{N} is a *hitting set problem*. In general, this has the following form: Given a set of elements Ω and a family Υ of subsets $L \subseteq \Omega$, find the smallest subsets of Ω that contain at least one element in each set of Υ . In our case, $\Omega = \text{Irr}$ and $\Upsilon = \text{MMBs}_{\text{tar}}$ is the family of MMBs involving the target reaction tar . We assume that there are no blocked reactions in the network, i.e., reactions which always have zero flux.

We solve the hitting set problem using MILP. For each subset $I \subseteq \text{Irr}$ we associate a vector $x \in \{0, 1\}^{\text{Irr}}$ such that $x_j = 1$ if reaction $j \in I$ and $x_j = 0$ if reaction $j \notin I$. Then the formulation to enumerate iMCSs of minimum cardinality is

$$\begin{aligned} (P) \text{ minimize } & \sum_{j \in \text{Irr}} x_j \\ \text{subject to } & \sum_{j \in D} x_j \geq 1, \quad \forall D \in \text{MMBs}_{\text{tar}}, \\ & x_j \in \{0, 1\}. \end{aligned}$$

Our goal is to enumerate all iMCSs and not only those of minimum cardinality. So whenever we find a new solution x^* at iteration i , we add a linear inequality to reject this solution at iteration $i+1$. If x is a candidate solution at iteration $i+1$ we require $\text{supp}(x^*) \not\subseteq \text{supp}(x)$, which can be formulated as $\sum_{j \in \text{supp}(x^*)} x_j \leq |\text{supp}(x^*)| - 1$.

7.3 Results and discussion

We implemented our method in MATLAB. We used the software SAGE (<http://www.sagemath.org>) [The Sage Developers, 2016] for computing the projection via contraction and the software polco [Terzer, 2009b] for enumerating the extreme rays. All computations were done on a desktop machine with two processors Intel(R) Core(TM) i5-2400S, CPU 2.50GHZ, each 1 thread. To evaluate our method, we considered a selection of medium-sized metabolic networks from BioModels Database [Li et al., 2010]. The number of unblocked reactions ranges from 87 up to 444.

While EFMs could be computed for only one network, the set of MMBs could be obtained for all these networks in a relatively short amount of time see Table 7.1.

All these networks contain a biomass reaction, which became the target reaction for computing the iMCSs. The hitting set approach allowed us to enumerate *all* iMCSs for these networks, where the iMCSs lead to a zero flux through the target reaction. For several networks, the maximal cardinality of the iMCSs did not exceed 5, and for these networks, most iMCSs have even cardinality 1. In *Blattabacterium* iCG230, we only have iMCSs of cardinality 1 (152 iMCSs) and cardinality 2 (7 iMCSs). *Salmonella Typhimurium* STM_v1.0 is another example with 296 iMCSs of cardinality 1, 19 of cardinality 2, and 6 of cardinality 3, cf. Table 7.2.

| network | rxns | irr | iMCSs | cardinality | | | | | | | | | | | | | | | |
|--|------|-----|-------|-------------|-----|----|----|----|---|----|----|----|----|----|----|----|----|----|----|
| | | | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
| <i>Escherichia coli</i> MG1655 [Orth et al., 2010] | 87 | 41 | 257 | 4 | 15 | 17 | 4 | 23 | 9 | 8 | 13 | 23 | 48 | 13 | 16 | 29 | 6 | 13 | 16 |
| <i>Rhizobium etli</i> iOR363 [Resendis-Antonio et al., 2007] | 194 | 104 | 60 | 29 | 18 | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <i>Buchnera</i> iSM197 [MacDonald et al., 2011] | 244 | 170 | 200 | 165 | 18 | 11 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <i>Blattabacterium cuenoti</i> iCG238 [González-Domenech et al., 2012] | 308 | 197 | 184 | 165 | 6 | 9 | 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <i>Blattabacterium</i> iCG230 [González-Domenech et al., 2012] | 400 | 192 | 159 | 152 | 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <i>Mycobacterium tuberculosis</i> iNJ661 [Jamshidi and Palsson, 2007] | 427 | 296 | 381 | 233 | 102 | 20 | 26 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <i>Salmonella Typhimurium</i> STM.v1.0 [Thiele et al., 2011] | 458 | 316 | 321 | 296 | 19 | 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| <i>Helicobacter pylori</i> iCS291 [Schilling et al., 2002] | 444 | 271 | 187 | 95 | 37 | 14 | 19 | 2 | 7 | 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Table 7.2: Comparison of the number of EFMs, number of MMBs, number of target MMBs, number of all iMCSs and time to compute these sets for different networks. The description of *Escherichia coli* MG1655 [Orth et al., 2010] was taken from the BiGG Models Database [King et al., 2016], whereas the remaining ones are all taken from BioModels Database [Li et al., 2010]. **network**: name of the metabolic network. **rxns**: number of unblocked reactions. **irr**: number of unblocked irreversible reactions. **iMCS**: number of irreversible minimal cut sets. **cardinality**: The number of all iMCSs of the corresponding cardinality.

7. Projections of flux cones

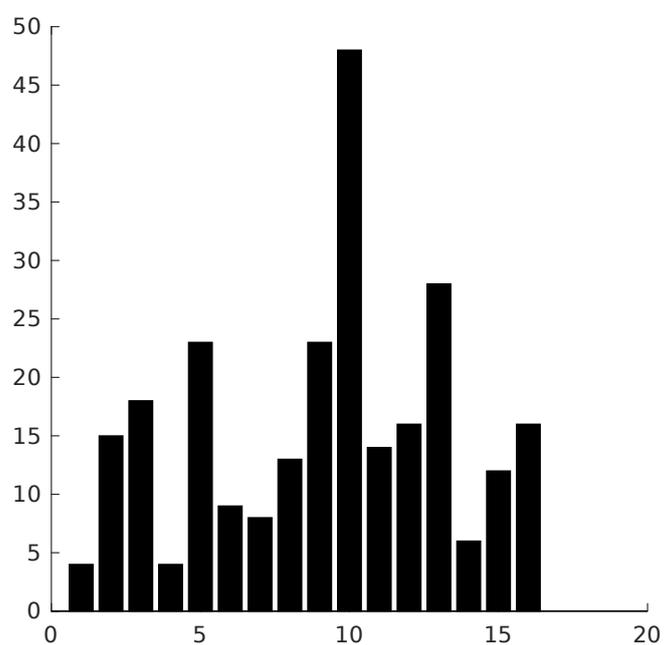


Figure 7.3: Cardinality of iMCSs for *Escherichia coli* MG1655 [King et al., 2016]. Each bar illustrates the number of iMCSs of the cardinality given on the x -axis. The number of iMCSs can be found on the y -axis.

7.4 Conclusion

We introduced the new concept of irreversible minimal cut sets, which can be computed using minimal metabolic behaviours. To find the MMBs of a metabolic network \mathcal{N} , we project the flux cone $\Gamma_{\mathcal{N}}$ on the set of irreversible reactions $\text{Irr} \subseteq \mathfrak{R}$. The supports of the extreme rays of the pointed cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ are the MMBs of \mathcal{N} .

One direction of future research is to project $\Gamma_{\mathcal{N}}$ on a set $H \supseteq \text{Irr}$ containing not only irreversible, but also some reversible reactions. These reversible reactions could be some promising candidates for MCSs. For example, it might be known that it is easy to knock out these reactions experimentally.

A second line of research is applying linear programming techniques and duality to compute MCSs on the projected cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$. Such MCSs are iMCSs for the original network. It was shown in [Ballerstein et al., 2012] that the MCSs of a flux cone $\Gamma_{\mathcal{N}}$ correspond to a subset of EFMs in a dual cone $\Gamma_{\mathcal{N}}^*$. Therefore computing MCSs in $\Gamma_{\mathcal{N}}$ can be done by computing EFMs in $\Gamma_{\mathcal{N}}^*$. Since $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ is a flux cone, it is possible to compute EFMs in the dual cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})^*$, which gives iMCSs in $\Gamma_{\mathcal{N}}$. In Chapter 8, we further explore this approach.

7. Projections of flux cones

Chapter 8

Computing iMCSs using the dual approach

This chapter is based on a joint article with Alexander Bockmayr, which is currently under revision.

In the last chapter we introduced a method for computing iMCSs, see Definition 7.1. Considering a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ we project the corresponding flux cone $\Gamma_{\mathcal{N}} = \{v \in \mathbb{R}^{\mathfrak{R}} \mid Sv = 0, v_{\text{Irr}} \geq 0\}$ onto the set of irreversible reactions along the reversible reactions. Every vector $(v_{\text{Irr}}, 0)$ in $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ can be lifted to a feasible flux vector $v \in \Gamma_{\mathcal{N}}$ of the original network by adding components corresponding to the reversible reactions and suitable flux rates. The polyhedral cone $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ is pointed since only irreversible reactions can be active and the supports of the extreme rays of $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ correspond exactly to the MMBs of the original metabolic network \mathcal{N} . Furthermore, since we project only along the reversible reactions, the projection can be computed in an efficient way because no inequalities are involved. In Chapter 7 we used the MMBs to compute iMCSs, based on the method explained in Section 2.2.3 where EFMs are used to compute MCSs.

The network sizes for which we are able to compute the MMBs are still limited. The bottleneck is computing the extreme rays of $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$ by `polco` [Terzer, 2009b], while doing the projection is feasible even for very large networks. In this chapter we make use of the *dual approach* [Burgard et al., 2003; Ballerstein et al., 2012] for computing MCSs of \mathcal{N} but do not apply it to $\Gamma_{\mathcal{N}}$ but to the projection $\text{proj}_{\text{Irr}}(\Gamma_{\mathcal{N}})$.

8. Computing iMCSs using the dual approach

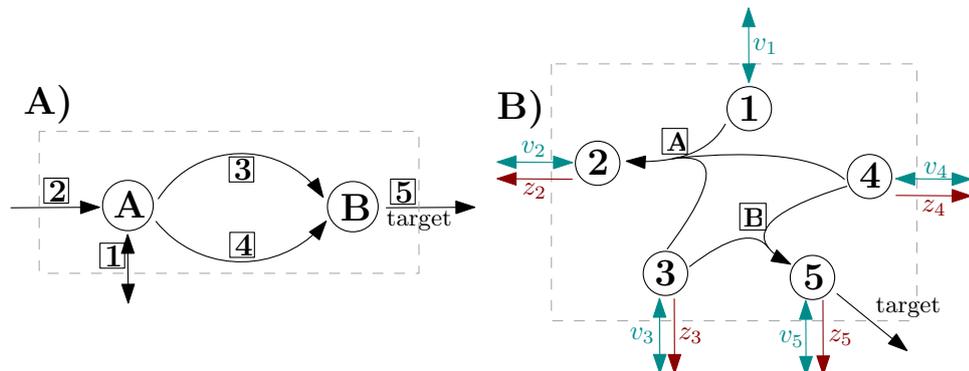


Figure 8.1: **A)** A metabolic network with 5 reactions and 2 metabolites. The dual network can be found in **B)**. Here we have 5 metabolites and 2 internal reactions. The exchange reactions are independent from the primal network [Ballerstein et al., 2012]. See Example 8.1 for a detailed explanation.

8.1 The dual approach

The dual approach is based on the Farkas' lemma and linear programming duality [Larhlmi and Bockmayr, 2007; Ballerstein et al., 2012; von Kamp and Klamt, 2014]. As shown in [Ballerstein et al., 2012], the MCSs in a network \mathcal{N} for a given target reaction $\text{tar} \in \mathfrak{R}$ correspond to EFMs in a *dual network* $\mathcal{N}_{\text{tar}}^*$. Intuitively speaking, the metabolites in $\mathcal{N}_{\text{tar}}^*$ correspond to the reactions in \mathcal{N} and vice versa, see Figure 8.1 for illustration. The stoichiometric matrix of $\mathcal{N}_{\text{tar}}^*$ is the transposed stoichiometric matrix of the primal network \mathcal{N} , where some columns have to be added. Exploiting this duality, one can use MILP to enumerate MCSs of increasing size [Larhlmi and Bockmayr, 2007; von Kamp and Klamt, 2014]. Further extending these methods, [Tobalina et al., 2016] compute MCSs that contain a predefined knockout reaction $\text{tar}' \neq \text{tar}$.

8.1.1 Dual network

We explain in the following the formulation of the dual problem step by step based on [Ballerstein et al., 2012; von Kamp and Klamt, 2014; Tobalina et al., 2016].

Assume a metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$ and its flux cone $\Gamma_{\mathcal{N}}$ is given. We consider the network at steady-state:

$$Sv = 0 \quad (8.1)$$

$$v_{\text{Irr}} \geq 0. \quad (8.2)$$

8.1 The dual approach

MCSs are defined w.r.t. a target reaction $\text{tar} \in \mathfrak{R}$ and a certain value v_{tar}^* . For example tar is the biomass reaction and v_{tar}^* is 20% of the maximal biomass rate. We call a minimal set of reactions $\xi \subseteq \mathfrak{R}$ an MCS if the maximum value of v_{tar} is less than or equal v_{tar}^* for all $v \in \Gamma_{\mathcal{N}}$ with $v_{\xi} = 0$. See Section 2.2.3 for details.

Thus, we want to prevent with an MCS that there exists a flux vector $v \in \Gamma_{\mathcal{N}}$ with

$$y^{\top} \cdot v \geq v_{\text{tar}}^*, \quad (8.3)$$

where $y \in \mathbb{B}^{\mathfrak{R}}$ with $y_i = 0$ for $i \in \mathfrak{R} \setminus \text{tar}$ and $y_{\text{tar}} = 1$. Note that for several target reactions the formulations are accordingly.

An MCS $\xi \subseteq \mathfrak{R}$ is a minimal set of reactions that would make the set of constraints (8.1), (8.2), and (8.3) infeasible. A set of possible constraints which can be added to make them infeasible is:

$$v_{\mathfrak{R}} = 0. \quad (8.4)$$

An MCS corresponds to a minimal subset of the constraints (8.4), such that (8.1), (8.2), and (8.3) become infeasible.

Altogether, the constraints (8.1), (8.2), (8.3), and (8.4) define the infeasible *primal* problem:

$$\begin{aligned} Sv &= 0 \\ v_{\text{Irr}} &\geq 0 \\ y^{\top} \cdot v &\geq v_{\text{tar}}^* \\ v_{\mathfrak{R}} &= 0 \\ v \in \mathbb{R}^{\mathfrak{R}}, \quad y &\in \mathbb{B}^{\mathfrak{R}} \end{aligned}$$

In order to search for a subset of constraints from the set (8.4) such that the constraints (8.1), (8.2), and (8.3) become infeasible one can consider the corresponding dual:

$$(S^{\top}|E| - \bar{E}_{\text{Irr}}|y) \cdot \begin{pmatrix} u \\ vp \\ vn \\ w \end{pmatrix} = 0 \quad (8.5)$$

$$v_{\text{tar}}^* \cdot w \geq \delta \quad (8.6)$$

$$vp \geq 0, vn \geq 0, w \geq 0 \quad (8.7)$$

$$u \in \mathbb{R}^{\mathfrak{M}}, vp \in \mathbb{R}^{\mathfrak{R}}, vn \in \mathbb{R}^{\mathfrak{R}}, w \in \mathbb{R} \quad (8.8)$$

where $\delta > 0$ denotes a threshold, indicating above which flux rate a reaction is considered to be active. Practically δ is chosen to be between 10^{-6}

8. Computing iMCSs using the dual approach

and 10^{-4} . E denotes the identity matrix and \bar{E}_{Irr} the identity matrix corresponding to the irreversible reactions where $|\text{Rev}|$ zero-rows are added at the positions of the reversible reactions. We denote in the following the concatenated matrix $(S^\top | E| - \bar{E}_{\text{Irr}} | y)$ with N .

The constraints (8.5), (8.6), (8.7), and (8.8) define the dual network $\mathcal{N}_{\text{tar}}^* = (\mathfrak{M}^*, \mathfrak{R}^*, N, \text{Irr}^*)$, where \mathfrak{M}^* denotes the metabolites in the dual network, \mathfrak{R}^* the reactions and Irr^* the irreversible reactions. N is the stoichiometric matrix of $\mathcal{N}_{\text{tar}}^*$ with $(u, vp, vn, w)^\top$ being the dual flux vector. Thus the number of reactions \mathfrak{R}^* of the dual network corresponds to the size of (u, vp, vn, w) . The irreversible reactions Irr^* are related to vp, vn , and w . The variables of u are related to the steady-state constraints in (8.1), vp , resp. vn , variables are linked to (8.4), resp. (8.2), and w corresponds to (8.3). An MCS of \mathcal{N} w.r.t. $\text{tar} \in \mathfrak{R}$ is an EFM in this dual problem that contains w and has minimal support in vp : if $vp_i \neq 0$ then i is part of an MCS for \mathcal{N} w.r.t. tar [Ballerstein et al., 2012; von Kamp and Klamt, 2014; Tobalina et al., 2016]. The constraint (8.6) rules out a trivial solution.

Example 8.1 (Dual network). *We consider the metabolic network given in Figure 8.1. The primal network \mathcal{N} has five reactions and two metabolites, where reaction 1 is the only reversible reaction. We choose reaction 5 as a target reaction. The stoichiometric matrix is*

$$S = \begin{matrix} & \boxed{1} & \boxed{2} & \boxed{3} & \boxed{4} & \boxed{5} \\ \begin{matrix} A \\ B \end{matrix} & \begin{pmatrix} -1 & 1 & -1 & -1 & 0 \\ 0 & 0 & 1 & 1 & -1 \end{pmatrix} \end{matrix}.$$

Let the maximal rate of reaction 5 be 10, so that $v_{\text{tar}}^ = 2$ (20% of the maximum value). The primal (infeasible) problem is then the following:*

$$\begin{aligned} \begin{pmatrix} -1 & 1 & -1 & -1 & 0 \\ 0 & 0 & 1 & 1 & -1 \end{pmatrix} v &= 0 \\ v_{2,3,4,5} &\geq 0 \\ (0, 0, 0, 0, 1) \cdot v &\geq 2 \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} v &= 0 \end{aligned} \tag{8.9}$$

The corresponding dual is:

$$\begin{pmatrix}
 -1 & 0 & | & 1 & 0 & 0 & 0 & 0 & | & 0 & 0 & 0 & 0 & 0 & | & 0 \\
 1 & 0 & | & 0 & 1 & 0 & 0 & 0 & | & 0 & -1 & 0 & 0 & 0 & | & 0 \\
 -1 & 1 & | & 0 & 0 & 1 & 0 & 0 & | & 0 & 0 & -1 & 0 & 0 & | & 0 \\
 -1 & 1 & | & 0 & 0 & 0 & 1 & 0 & | & 0 & 0 & 0 & -1 & 0 & | & 0 \\
 0 & -1 & | & 0 & 0 & 0 & 0 & 1 & | & 0 & 0 & 0 & 0 & -1 & | & 1
 \end{pmatrix}
 \begin{pmatrix}
 u \\
 vp \\
 vn \\
 w
 \end{pmatrix}
 = 0$$

$\underbrace{\hspace{1.5cm}}_{S^\top} \quad \underbrace{\hspace{2.5cm}}_E \quad \underbrace{\hspace{2.5cm}}_{-E_{\text{irr}}} \quad \underbrace{\hspace{1.5cm}}_y$

$$2 \cdot w \geq \delta.$$

For computing MCSs of \mathcal{N} w.r.t tar we can compute EFMs in $\mathcal{N}_{\text{tar}}^*$ using the constraints (8.5), (8.6), (8.7), and (8.8). We can make use of the MILP in [De Figueiredo et al., 2009], see Chapter 3, where the k -shortest EFMs are computed. Thus we compute the k -shortest MCSs. The binary variables, indicating which reactions are part of the computed MCS, are zp and zv and it holds that $zp_i = 0 \Leftrightarrow vp_i = 0$ and $zn_i = 0 \Leftrightarrow vn_i = 0$. For more details we refer to [Ballerstein et al., 2012; von Kamp and Klamt, 2014; Tobalina et al., 2016].

Building on this method [Tobalina et al., 2016] compute MCSs which always contain a specific, predefined reaction knock-out. Thus, in the dual network $\mathcal{N}_{\text{tar}}^*$ all EFMs are computed including the predefined reaction knock-out so this method is also based on MILP.

The definition of the dual network $\mathcal{N}_{\text{tar}}^*$ as above is based on Farkas' Lemma [Schrijver, 1998], which can be applied to any matrix, not only stoichiometric matrices. Therefore we can define a dual flux cone Γ^* for a general flux cone Γ as well, where $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$, with $A = \begin{pmatrix} B \\ -B \\ E_{H,*} \end{pmatrix}$. The dual flux cone is then given by $\Gamma^* = \{u \in \mathbb{R}^m \mid (B_{*,H})^\top u \geq 0, (B_{*,\mathfrak{A} \setminus H})^\top x = 0\}$.

8.2 Results

All computations were done on a desktop computer with eight processors Intel(R) Core(TM) i7-2600, CPU 3.40GHZ, each with 2 threads.

8.2.1 Projection

We used the software **SAGE** [The Sage Developers, 2016] for computing the projection via contraction. We performed the projection on all 84 networks

8. Computing iMCSs using the dual approach

of the BiGG Models Database [King et al., 2016], which took between 32 seconds (for a network of 87 unblocked reactions) and 35 minutes (for a network of 4047 unblocked reactions), see Table 8.1, Table 8.2, and Figure 8.2.

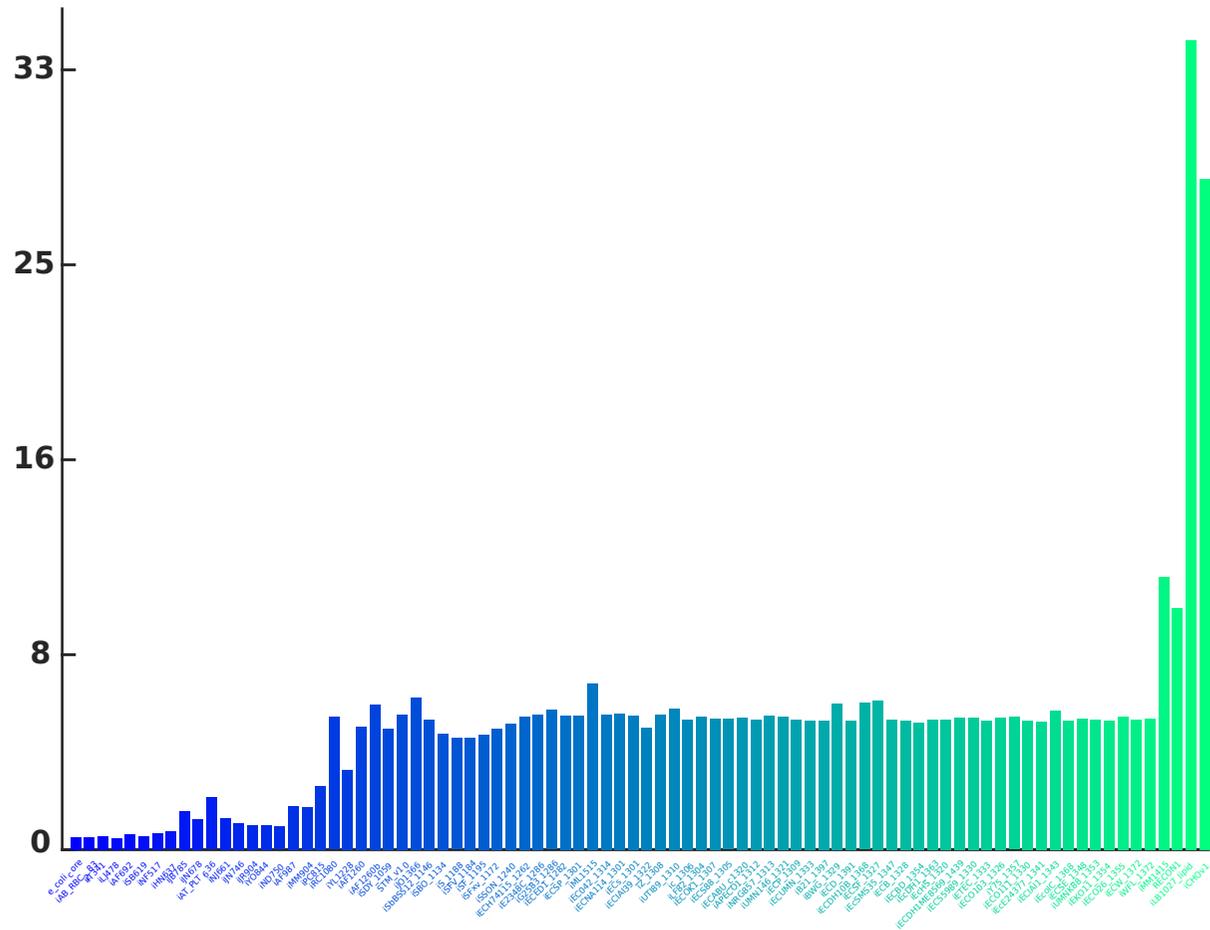


Figure 8.2: The 84 networks from the BiGG Models Database ordered by the number of reactions. For each network, the bar indicates the time in minutes for computing the projected flux cone via contraction.

8. Computing iMCSs using the dual approach

8.2.2 Computing shortest MCSs

To compute k -shortest MCSs and iMCSs we used the already existing implementation by [Tobalina et al., 2016], who provide a `MATLAB` script, using `cplex` [IBM Knowledge Center, 2010] as an MILP solver. We used the resulting matrices describing $\text{proj}_{\text{irr}}(\Gamma_{\mathcal{N}})$ and the description of the original flux cone $\Gamma_{\mathcal{N}}$ and compared the results.

8.2.2.1 Time differences

We computed 10, 100, and 300 iMCSs using both, the original and the projected flux cone for the 77 different networks including a biomass reaction which became the target reaction. For both, the original flux cone $\Gamma_{\mathcal{N}}$ and the projected flux cone, only iMCSs were computed. Figure 8.3 illustrates the relative time of the computations, where the details can be found in the Appendix in Section 10.2.1. Computing 10 iMCSs is sometimes faster using the original flux cone and sometimes faster using the projected flux cone. For computing 100 or 300 iMCSs it is almost always faster using the projected flux cone.

8.2.2.2 Cardinality

We used the 300 iMCSs to compare their cardinality with 300 MCSs we computed using the original flux cone $\Gamma_{\mathcal{N}}$. We end up with iMCSs of larger cardinality than computing MCSs (which include all the iMCSs), see the Tables in the Appendix in Section 10.2.2.

8.2 Results

| network id | size normal cone | size projected cone | time |
|----------------|------------------|---------------------|------|
| e_coli_core | 68 × 87 | 16 × 40 | 32 |
| iAB_RBC_283 | 333 × 453 | 136 × 264 | 32 |
| iIT341 | 381 × 436 | 213 × 276 | 34 |
| iLJ478 | 331 × 385 | 171 × 228 | 30 |
| iAF692 | 417 × 484 | 245 × 321 | 39 |
| iSB619 | 381 × 450 | 192 × 275 | 35 |
| iNF517 | 435 × 513 | 203 × 296 | 42 |
| iHN637 | 448 × 524 | 266 × 351 | 46 |
| iJB785 | 671 × 741 | 440 × 543 | 100 |
| iJN678 | 597 × 675 | 405 × 497 | 79 |
| iAT_PLT_636 | 738 × 1008 | 316 × 559 | 134 |
| iNJ661 | 579 × 740 | 335 × 515 | 82 |
| iJN746 | 539 × 652 | 283 × 401 | 69 |
| iJR904 | 450 × 667 | 243 × 475 | 63 |
| iYO844 | 500 × 657 | 220 × 385 | 63 |
| iND750 | 479 × 631 | 210 × 381 | 59 |
| iAF987 | 708 × 840 | 429 × 574 | 111 |
| iMM904 | 650 × 893 | 307 × 586 | 109 |
| iPC815 | 761 × 1065 | 450 × 774 | 163 |
| iRC1080 | 1102 × 1583 | 1102 × 1091 | 340 |
| iYL1228 | 830 × 1223 | 495 × 925 | 205 |
| iAF1260 | 1032 × 1532 | 661 × 1185 | 316 |
| iAF1260b | 1040 × 1554 | 662 × 1200 | 373 |
| iSDY_1059 | 1026 × 1502 | 627 × 1133 | 311 |
| STM_v1.0 | 1086 × 1597 | 711 × 1249 | 346 |
| iJO1366 | 1155 × 1705 | 732 × 1312 | 391 |
| iSbBS512_1146 | 1018 × 1540 | 622 × 1169 | 334 |
| iSBO_1134 | 1022 × 1530 | 630 × 1168 | 297 |
| iS_1188 | 1017 × 1504 | 604 × 1127 | 286 |
| iSFV_1184 | 1026 × 1516 | 605 × 1136 | 288 |
| iSF_1195 | 1022 × 1512 | 601 × 1129 | 294 |
| iSF×v_1172 | 1045 × 1554 | 627 × 1171 | 311 |
| iSSON_1240 | 1066 × 1601 | 638 × 1206 | 323 |
| iECH74115_1262 | 1083 × 1636 | 658 × 1246 | 342 |
| iE2348C_1286 | 1087 × 1641 | 657 × 1243 | 347 |
| iG2583_1286 | 1087 × 1644 | 662 × 1254 | 358 |
| iCED1_1282 | 1087 × 1644 | 657 × 1249 | 344 |
| iECSP_1301 | 1087 × 1646 | 662 × 1256 | 344 |
| iML1515 | 1147 × 1744 | 719 × 1350 | 427 |
| iEC042_1314 | 1084 × 1644 | 662 × 1257 | 347 |
| iECNA114_1301 | 1091 × 1656 | 660 × 1260 | 348 |
| iECs_1301 | 1087 × 1646 | 662 × 1256 | 343 |

Table 8.1: Sizes of the flux cones and time for the projection for the first 42 networks (w.r.t. the number of reactions) of the **BiGG Models Database** [King et al., 2016]. **network id**: The id of the network in the **BiGG Models Database**. **size normal cone**: size of the flux cone of the original network: unblocked reactions and non-dead end metabolites. **size projected cone**: size of the projected flux cone of the original network: number of columns and rows of the matrix describing the projected flux cone. **time**: time in seconds needed to project onto the irreversible reactions (using the network without the blocked reactions).

8. Computing iMCSs using the dual approach

| network id | size normal cone | size projected cone | time |
|-------------------|------------------|---------------------|------|
| iECIAI39_1322 | 1044 × 1569 | 613 × 1177 | 313 |
| iZ_1308 | 1087 × 1646 | 662 × 1256 | 347 |
| iUTI89_1310 | 1096 × 1662 | 660 × 1261 | 363 |
| ic_1306 | 1090 × 1656 | 654 × 1254 | 334 |
| iLF82_1304 | 1082 × 1650 | 645 × 1243 | 342 |
| iECOK1_1307 | 1096 × 1670 | 660 × 1269 | 337 |
| iECS88_1305 | 1088 × 1653 | 660 × 1260 | 335 |
| iECABU_c1320 | 1094 × 1663 | 659 × 1262 | 339 |
| iAPECO1_1312 | 1096 × 1668 | 660 × 1267 | 333 |
| iNRG857_1313 | 1100 × 1675 | 660 × 1268 | 344 |
| iUMN146_1321 | 1096 × 1670 | 660 × 1269 | 341 |
| iECP_1309 | 1094 × 1668 | 659 × 1267 | 334 |
| iECUMN_1333 | 1093 × 1657 | 655 × 1255 | 330 |
| iB21_1397 | 1089 × 1650 | 658 × 1253 | 332 |
| iBWG_1329 | 1164 × 1739 | 726 × 1335 | 375 |
| iECD_1391 | 1089 × 1650 | 658 × 1253 | 330 |
| iECDH10B_1368 | 1160 × 1736 | 721 × 1331 | 376 |
| iECSF_1327 | 1162 × 1743 | 726 × 1338 | 382 |
| iEcSMS35_1347 | 1102 × 1673 | 664 × 1271 | 334 |
| iECB_1328 | 1096 × 1660 | 662 × 1262 | 331 |
| iECBD_1354 | 1089 × 1651 | 658 × 1254 | 326 |
| iEcDH1_1363 | 1099 × 1667 | 663 × 1266 | 333 |
| iEcHS_1320 | 1094 × 1645 | 662 × 1251 | 334 |
| iECDH1ME8569_1439 | 1101 × 1670 | 663 × 1268 | 338 |
| iEC55989_1330 | 1103 × 1670 | 664 × 1268 | 339 |
| iETEC_1333 | 1095 × 1658 | 664 × 1263 | 332 |
| iECO103_1326 | 1096 × 1660 | 661 × 1262 | 338 |
| iY75_1357 | 1101 × 1670 | 663 × 1268 | 340 |
| iECO111_1330 | 1089 × 1651 | 660 × 1259 | 330 |
| iEcE24377_1341 | 1092 × 1655 | 663 × 1260 | 329 |
| iECIAI1_1343 | 1089 × 1638 | 663 × 1251 | 356 |
| iEcolC_1368 | 1092 × 1653 | 663 × 1261 | 331 |
| iECSE_1348 | 1098 × 1664 | 663 × 1266 | 337 |
| iUMNK88_1353 | 1098 × 1665 | 664 × 1268 | 334 |
| iEKO11_1354 | 1098 × 1655 | 663 × 1257 | 332 |
| iECO26_1355 | 1098 × 1666 | 663 × 1268 | 342 |
| iECW_1372 | 1102 × 1668 | 665 × 1269 | 334 |
| iWFL_1372 | 1102 × 1668 | 665 × 1269 | 337 |
| iMM1415 | 1665 × 2432 | 855 × 1576 | 700 |
| RECON1 | 1586 × 2467 | 525 × 1314 | 620 |
| iLB1027_lipid | 1814 × 4047 | 1355 × 3653 | 2075 |
| iCHOv1 | 2213 × 4280 | 943 × 2527 | 1720 |

Table 8.2: Sizes of the flux cones and time for the projection for the second 42 networks (w.r.t. the number of reactions) in the **BiGG Models Database** [King et al., 2016]. **network id**: The id of the network on the **BiGG Models Database**. **size normal cone**: size of the flux cone of the original network: unblocked reactions and non-dead end metabolites. **size projected cone**: size of the projected flux cone of the original network: number of columns and rows of the matrix describing the projected flux cone. **time**: time in seconds needed to project onto the irreversible reactions (using the network without the blocked reactions).

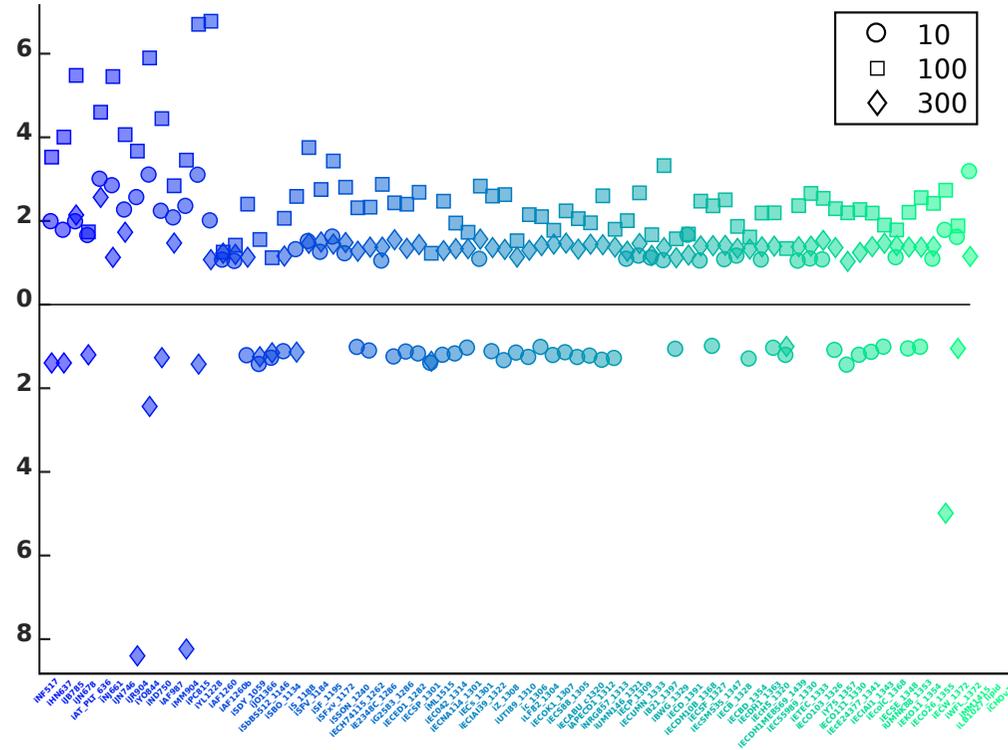


Figure 8.3: We consider here 76 networks the **BiGG Models Database**. The ids of the networks can be found on the x -axis. The networks are ordered according to their number of reactions. For each number of computed iMCSs (10,100, or 300) there exists a differently shaped marker: a circle for 10 iMCSs, a square for 100 and a diamond for 300 computed iMCSs. Their positions illustrates the relative time (y -axis) needed to compute the given number of iMCSs using the original flux cone vs. the projected flux cone. If the marker is above the line it means that using the projected cone is faster. For example computing 100 iMCSs is always faster, for the network **is_1188** it is roughly 3.7 times faster using the projected cone. If the marker is below one, which is almost always the case when 10 iMCSs are computed, it is faster to use the original cone. For example computing 300 iMCSs in the original flux cone is 1.4 times faster in the original flux cone. As one can see, for computing a small number of iMCSs it does not make a huge difference, using the original or the projected flux cone. But for larger numbers of iMCSs especially in bigger networks the benefit increases.

8. Computing iMCSs using the dual approach

8.2.3 Computing MCSs including a knock-out reaction

Finally we considered the computation of iMCSs containing a specific reaction knock-out. We applied the software of [Tobalina et al., 2016] to the original stoichiometric matrices as well as to the reduced matrices after projection, and compared the results for both cases. Like in [Tobalina et al., 2016], we used the biomass as target reaction. For each irreversible reaction, we set a time limit of 1 minute to compute a single iMCS that contains this reaction. If no MCS can be found within 1 minute, the computation was stopped and we moved on to the next irreversible reaction. We report on the 39 networks in the BiGG database for which the total computation time for this experiment did not exceed 3 days. Except for one network, we were able to compute more iMCSs using the projection, see Figure 8.4 and Table 8.3.

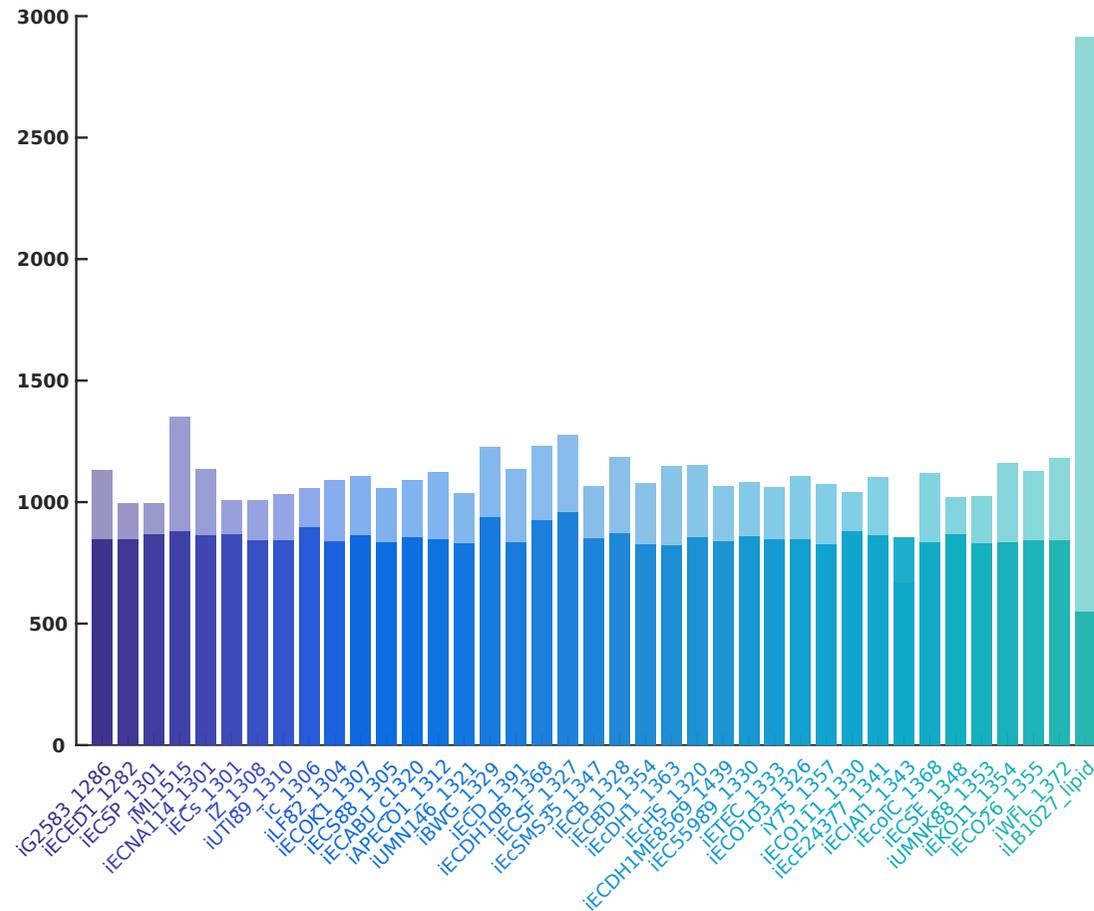


Figure 8.4: We consider in this figure all networks from the BiGG Models Database for which the total computation time took less than 3 days, using the method of [Tobalina et al., 2016]. The biomass reaction is the target reaction for computing the MCSs and one additional irreversible reaction is defined as knock-out reaction. We loop over all irreversible reactions of the network. In step i , reaction i is defined as a knock-out reaction and the program tries to compute an iMCS for the target reaction containing i . After 1 minute, or if an MCS has been found, the next step starts. The y -axis indicates the number of MCSs that were found. The transparent bar (which includes the non-transparent bar) indicates the number of MCSs computed using the projected network, the non-transparent bar indicates the number of MCSs computed for the original cone. For all networks, we were able to compute more MCSs using the projected network.

8. Computing iMCSs using the dual approach

| network id | Nr. normal | Nr. projected | difference |
|-------------------|------------|---------------|------------|
| iG2583_1286 | 847 | 1133 | 286 |
| iECED1_1282 | 845 | 994 | 149 |
| iECSP_1301 | 868 | 996 | 128 |
| iML1515 | 877 | 1350 | 473 |
| iECNA114_1301 | 862 | 1135 | 273 |
| iECs_1301 | 866 | 1007 | 141 |
| iZ_1308 | 841 | 1008 | 167 |
| iUTI89_1310 | 841 | 1032 | 191 |
| ic_1306 | 896 | 1055 | 159 |
| iLF82_1304 | 836 | 1088 | 252 |
| iECOK1_1307 | 861 | 1106 | 245 |
| iECS88_1305 | 835 | 1055 | 220 |
| iECABU_c1320 | 855 | 1090 | 235 |
| iAPECO1_1312 | 845 | 1121 | 276 |
| iUMN146_1321 | 831 | 1037 | 206 |
| iBWG_1329 | 937 | 1226 | 289 |
| iECD_1391 | 833 | 1135 | 302 |
| iECDH10B_1368 | 924 | 1231 | 307 |
| iECSF_1327 | 959 | 1277 | 318 |
| iEcSMS35_1347 | 851 | 1066 | 215 |
| iECB_1328 | 871 | 1183 | 312 |
| iECBD_1354 | 824 | 1078 | 254 |
| iEcDH1_1363 | 819 | 1148 | 329 |
| iEcHS_1320 | 854 | 1150 | 296 |
| iECDH1ME8569_1439 | 836 | 1065 | 229 |
| iEC55989_1330 | 858 | 1080 | 222 |
| iETEC_1333 | 844 | 1061 | 217 |
| iECO103_1326 | 846 | 1108 | 262 |
| iY75_1357 | 826 | 1074 | 248 |
| iECO111_1330 | 879 | 1042 | 163 |
| iEcE24377_1341 | 863 | 1101 | 238 |
| iECIAI1_1343 | 854 | 666 | -188 |
| iEcolC_1368 | 832 | 1117 | 285 |
| iECSE_1348 | 865 | 1018 | 153 |
| iUMNK88_1353 | 830 | 1023 | 193 |
| iEKO11_1354 | 835 | 1162 | 327 |
| iECO26_1355 | 842 | 1129 | 287 |
| iWFL_1372 | 840 | 1179 | 339 |
| iLB1027_lipid | 546 | 2916 | 2370 |

Table 8.3: We applied the program of [Tobalina et al., 2016] to the original stoichiometric matrices and to the matrices after the projection was performed. As a target reaction we had for all cases the biomass reaction. We looped over all irreversible reactions and compute an iMCS. After one minute, or after an iMCS was found, the next step started with a new knock-out reaction. **network id:** the id of the network as it can be found in the BiGG Models Database. **Nr. normal:** the number of iMCSs which were found using the original flux cone. **Nr. projected:** the number of iMCSs which were found using the projected flux cone. **difference:** the number of iMCSs we found more when using the projected flux cone instead the original.

8.3 Conclusion

As for the previous chapter, a direction of future research is to project $\Gamma_{\mathcal{N}}$ on a set $H \supseteq \text{Irr}$ containing not only irreversible, but also some reversible reactions.

In general, computing iMCSs using projection in order to decrease the size of the dual problem is a good start for analysing the structure of the metabolic network when the MMBs cannot be computed.

Altogether, searching for iMCSs in projected flux cones is a promising new method for computing MCSs in genome-scale metabolic networks

8. Computing iMCSs using the dual approach

Chapter 9

Conclusion

This thesis is a mathematical contribution to the study of metabolic networks. We tackled the unfavourable combination of large genome-scale metabolic networks and algorithms which are NP-hard and therefore very sensitive to the size of the input. We developed different techniques to significantly reduce the amount of information needed about the network, in order to analyse certain behaviours or to compute more efficiently properties of the underlying network.

The focus of the thesis was the analysis of metabolic networks in steady-state. The mathematical foundation for our algorithms is mixed integer linear programming in Chapters 3 and 4, and a combination of linear algebra and oriented matroid theory in Chapters 5, 6, and 7.

One contribution in the field of MILP is that we showed that coupling information on reactions of the metabolic network can substantially reduce the number of binary variables in an MILP and thus speed up the average time to enumerate shortest EFMs by a factor of 40. The developed method can be applied to any MILP where binary variables are used to indicate the activity of reactions. Furthermore, we introduced a novel MILP to compute a subnetwork while preserving predefined functionalities. The existing methods were either not able to cope with more than one functionality [Burgard et al., 2001] or could not compute the smallest subnetwork [Erdrich et al., 2014]. The MILP we introduced accounts for different functionalities as well as for a minimum subnetwork. Thus, it combines the advantages of the previous methods and eliminates their drawbacks. Additionally, the running time of the computation was reduced by an average factor of 8 compared to the former approach [Erdrich et al., 2014]. Using coupling information on reactions again reduces the search space and decreases the running time by a

9. Conclusion

factor of 4. The method for reducing a given network while predefined functionalities are kept is a promising tool for analysing genome-scale metabolic networks in more detail, thus focusing on certain properties or behaviours. In this thesis we only considered very prominent scenarios, such as aerobic and anaerobic growth, but any scenario which can be formulated using linear constraints is a valid input. Using this approach can help to find (sets of) reactions and metabolites which are important or necessary for certain functionalities. The method was applied within the *Mathematics in Life Sciences* research group to the metabolic network of Yeast 6 [Heavner et al., 2013] in order to reduce the network to a size at which dynamic modelling becomes feasible [Reimers, 2017]. After the reduction, the behaviours of the reduced network were well correlated to results from wet lab experiments. Thus, this tool is an important contribution to the already existing linear programming tools, such as FBA, FCA, and FVA.

To conclude on MILPs: Since LPs are solvable in polynomial time, they are the current standard for the analysis of metabolic networks, but they cannot model the discrete constraints arising in some problems. A lot of open questions regarding metabolic networks can be answered using MILPs where using LPs is neither sufficient nor satisfactory. Modern MILP solvers are fast enough to handle large metabolic networks and are, for some problems, a better choice than comparable LP approaches. MILPs cannot be solved more efficiently than LPs, but can be used to describe constraints which cannot be formulated using continuous variables only.

However, not all questions can or have to be formulated as MILPs. In the last three chapters of this thesis no MILPs were used to analyse metabolic networks. Instead, we used linear algebra and oriented matroids.

The main contribution of Chapters 5 and 6 was the introduction of a novel minimal description for metabolic networks called MEMo and an algorithm to compute it. A MEMo is a minimum set of EFMs such that every flux vector of the network can be generated using a conical combination of the EFMs included in this MEMo. To introduce and prove the correctness of this method, we made use of linear algebra, polyhedral cones, and oriented matroids. We applied the method to several metabolic networks and the size of the whole set of EFMs is on average 65 times larger than the sizes of the MEMos. EFMs give important insights into the metabolic network [Stelling et al., 2002; Carlson and Sreenc, 2004b; Chan and Ji, 2011]. Unfortunately, computing the whole set of EFMs for a given metabolic network is not feasible for genome-scale networks using a standard computer. Even if the whole set would be known, exploring all EFMs is in its own a challenge since the number of EFMs grows exponentially with the size of the network. We were able to compute MEMos for networks where computing the whole set of

EFMs failed because the program ran out of memory. Thus, such a MEMo can be used to further analyse flux vectors and the importance of reactions of a metabolic network if computing the whole set of EFMs is not possible. However, computing a MEMo for any given genome-scale metabolic network was not possible using the given computing capacity and the capability of the enumeration tool `polco` [Terzer, 2017b]. Computing only a subset of EFMs is not a new idea [De Figueiredo et al., 2009; Machado et al., 2012] and also computing a subset of EFMs which can still be used to generate every behaviour of the metabolic network was already introduced by computing the extreme pathways (EPs) [Schilling et al., 2000]. However, EPs are not a minimum set and the number of EFMs contained in EPs can be very large. A typical application for EFMs is for example the decomposition of a given flux vector into a set of EFMs. If the whole set of EFMs cannot be computed but a MEMo, we can decompose the flux vector into EFMs using this MEMo. Although EFMs contained in a MEMo are not unique, they can provide new insight into the importance of certain pathways or reactions for the whole network. In fact, investigating which EFMs are exchangeable and why, for each metabolic network individually is an interesting task. Doing so, it could be possible to study robustness of the underlying metabolic network. Another idea is to figure out how the network was assembled or to discover similarities to other networks by detecting subsets of EFMs which can be found in other networks as well. We conclude that the concept of a MEMo is a promising start for a new sort of examination of metabolism.

The contribution of the last two chapters was a novel method to efficiently reduce the representation of a network while keeping the overall topology. Using this method allowed us to compute MCSs faster or of larger cardinality than using the original representation of a network. We projected the flux cone which is typically done by applying the Fourier-Motzkin elimination [Urbanczik and Wagner, 2005b; Marashi et al., 2012] resulting in a combinatorial explosion regarding the number of constraints describing the projected cone. Since we project onto the irreversible reactions, we can make use of oriented matroids. This enables us to do the projection in $\mathcal{O}(n^3)$, where n is the number of reactions, while the number of constraints does not increase. Using oriented matroids, we proved that the extreme rays of the projected flux cone are the minimal metabolic behaviours (MMBs) [Larhlimi and Bockmayr, 2009] of the original metabolic network. These MMBs can be used in the same fashion as the whole set of EFMs for computing MCSs, but the resulting MCSs consist of irreversible reactions only and are therefore called irreversible MCSs (iMCSs). Since the number of MMBs is several orders of magnitude smaller than the number of EFMs, we were able to apply the method to larger networks than the method using EFMs. Not all MCSs are iMCSs, therefore not all MCSs can be found with the technique introduced here. But if it is not possible to compute the whole

9. Conclusion

set of EFMs we can use this technique in order to compute all iMCSs instead of no MCSs at all. Additionally, we used two already existing methods for computing MCSs and applied them to the projected flux cone in Chapter 8. We were able to compute iMCSs of larger cardinality than when using the original flux cone. Using the projected cone is most of the time faster than using the original one. This approach can be used for analysing the robustness or fragility of a metabolic network or of a given reaction. A possible extension would be to also project onto reversible reactions in addition to the irreversible reactions. Then one can compute EFMs and MCSs of the projected flux cone to analyse the impact of those reversible reactions as well. Furthermore, the projection of flux cones could be further extended in order to analyse the connectivity of metabolic networks, the importance of certain reactions, or certain metabolites.

The amount of metabolic network models available is steadily increasing. Analysis of these models shows great potential in industry and personalised medicine. However, to truly unleash the potential of network analysis, these methods must scale up to genome-scale models. Currently a lot of methods like computing EFMs cannot be applied to large networks. This thesis showed the potential of analysing and exploiting the mathematical structure underlying these networks, especially by exploiting the connections between metabolic networks, mixed integer linear optimisation, linear algebra and oriented matroid theory. The results of this thesis give us hope that existing analysis methods can be scaled up to larger problems and that new methods can be introduced which are applicable to genome-scale metabolic networks.

Bibliography

- V. Acuña, F. Chierichetti, V. Lacroix, A. Marchetti-Spaccamela, M.-F. Sagot, and L. Stougie. Modes and cuts in metabolic networks: Complexity and algorithms. *Biosystems*, 95(1):51–60, 2009. 17
- T. Aittokallio and B. Schwikowski. Graph-based methods for analysing networks in cell biology. *Briefings in Bioinformatics*, 7(3):243–255, 2006. 2
- I. Apaolaza, E. San José-Eneriz, L. Tobalina, E. Miranda, L. Garate, X. Agirre, F. Prósper, and F. J. Planes. An in-silico approach to predict and exploit synthetic lethality in cancer metabolism. *Nature Communications*, 8:459, 2017. 17, 57, 118
- M. Arabzadeh, M. S. Zamani, M. Sedighi, and S.-A. Marashi. A graph-based approach to analyze flux-balanced pathways in metabolic networks. *BioSystems*, 2018. 116
- M. Artin. *Algebra*. Birkhäuser, 1998. ISBN 3764359382 978-3764359386. 107, 129
- J. E. Bailey. Complex biology with no parameters. *Nature Biotechnology*, 19:503–504, 2001. 5
- K. Ballerstein, A. von Kamp, S. Klamt, and U.-U. Haus. Minimal cut sets in a metabolic network are elementary modes in a dual network. *Bioinformatics*, 28(3):381–387, 2012. 17, 118, 135, 137, 138, 140, 141
- S. Bazzani, A. Hoppe, and H.-G. Holzhütter. Network-based assessment of the selectivity of metabolic drug targets in *Plasmodium falciparum* with respect to human liver metabolism. *BMC Systems Biology*, 6:118, 2012. 1
- J. Behre, T. Wilhelm, A. von Kamp, E. Ruppin, and S. Schuster. Structural robustness of metabolic networks with respect to multiple knockouts. *Journal of Theoretical Biology*, 252(3):433–441, 2008. 14, 15

Bibliography

- D. R. Bentley, S. Balasubramanian, H. P. Swerdlow, G. P. Smith, J. Milton, C. G. Brown, K. P. Hall, D. J. Evers, C. L. Barnes, H. R. Bignell, et al. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature*, 456:53–59, 2008. 1
- C. Berge. *Hypergraphs: Combinatorics of Finite Sets*, volume 45. Elsevier, 1984. 17
- A. Björner. *Oriented matroids*, volume 46. Cambridge University Press, 1999. 36, 39, 40, 43, 124, 128
- A. Bordbar, N. Jamshidi, and B. Ø. Palsson. iAB-RBC-283: A proteomically derived knowledge-base of erythrocyte metabolism that can be used to simulate its physiological and patho-physiological states. *BMC Systems Biology*, 5:110, 2011. 110, 111, 112
- A. Bordbar, J. M. Monk, Z. A. King, and B. Ø. Palsson. Constraint-based models predict metabolic and associated cellular functions. *Nature Reviews Genetics*, 15(2):107–120, 2014. 2
- K.-H. Borgwardt. The average number of pivot steps required by the Simplex-Method is polynomial. *Mathematical Methods of Operations Research*, 26(1):157–177, 1982. 27
- E. Boros, K. Elbassioni, V. Gurvich, and L. Khachiyan. Algorithms for Enumerating Circuits in Matroids. In *International Symposium on Algorithms and Computation*, pages 485–494. Springer, 2003. 42, 44
- A. P. Burgard, S. Vaidyaraman, and C. D. Maranas. Minimal Reaction Sets for *Escherichia coli* Metabolism under Different Growth Requirements and Uptake Environments. *Biotechnology Progress*, 17(5):791–797, 2001. 31, 62, 79, 153
- A. P. Burgard, P. Pharkya, and C. D. Maranas. Optknock: A Bilevel Programming Framework for Identifying Gene Knockout Strategies for Microbial Strain Optimization. *Biotechnology and Bioengineering*, 84(6):647–657, 2003. 2, 17, 137
- A. P. Burgard, E. V. Nikolaev, C. H. Schilling, and C. D. Maranas. Flux Coupling Analysis of Genome-Scale Metabolic Network Reconstructions. *Genome Research*, 14:301–312, 2004. 3, 8, 28, 29, 67
- R. Carlson and F. Sreenc. Fundamental *Escherichia coli* Biochemical Pathways for Biomass and Energy Production: Identification of Reactions. *Biotechnology and Bioengineering*, 85(1):1–19, 2004a. 14
- R. Carlson and F. Sreenc. Fundamental *Escherichia coli* Biochemical Pathways for Biomass and Energy Production: Creation of Overall Flux States. *Biotechnology and Bioengineering*, 86(2):149–162, 2004b. 14, 154

- S. H. J. Chan and P. Ji. Decomposing flux distributions into elementary flux modes in genome-scale metabolic networks. *Bioinformatics*, 27(16):2256–2262, 2011. 14, 154
- C. Chassagnole, N. Noisommit-Rizzi, J. W. Schmid, K. Mauch, and M. Reuss. Dynamic Modeling of the Central Carbon Metabolism of *Escherichia coli*. *Biotechnology and Bioengineering*, 79(1):53–73, 2002. 110, 111, 112, 114
- S. T. Clark and W. S. Verwoerd. Minimal Cut Sets and the Use of Failure Modes in Metabolic Networks. *metabolites*, 2(3):567–595, 2012. 118
- B. L. Clarke. Stoichiometric Network Analysis. *Cell Biochemistry and Biophysics*, 12(1):237–253, 1988. 23, 99
- M. W. Covert and B. Ø. Palsson. Constraints-based models: regulation of gene expression reduces the steady-state solution space. *Journal of theoretical biology*, 221(3):309–325, 2003. 120
- M. W. Covert, C. H. Schilling, I. Famili, J. S. Edwards, I. I. Goryanin, E. Selkov, and B. Ø. Palsson. Metabolic modeling of microbial strains *in silico*. *Trends in Biochemical Sciences*, 26(3):179–186, 2001a. 5
- M. W. Covert, C. H. Schilling, and B. Ø. Palsson. Regulation of Gene Expression in Flux Balance Models of Metabolism. *Journal of Theoretical Biology*, 213(1):73–88, 2001b. 5
- G. B. Dantzig. Linear programming. *Proceedings of the Symposium on Modern Calculating Machinery and Numerical Methods*, 15:18–21, 1948. 26
- L. David and A. Bockmayr. Computing Elementary Flux Modes Involving a Set of Target Reactions. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 11(6):1099–1107, 2014. 57, 94
- L. David, S.-A. Marashi, A. Larhlimi, B. Mieth, and A. Bockmayr. FFCA: a feasibility-based method for flux coupling analysis of metabolic networks. *BMC Bioinformatics*, 12(1):236, 2011. 28
- L. F. De Figueiredo, A. Podhorski, A. Rubio, C. Kaleta, J. E. Beasley, S. Schuster, and F. J. Planes. Computing the shortest elementary flux modes in genome-scale metabolic networks. *Bioinformatics*, 25(23):3158–3165, 2009. 3, 49, 50, 51, 54, 55, 94, 116, 141, 155
- D. Deutscher, I. Meilijson, M. Kupiec, and E. Ruppin. Multiple knockout analysis of genetic robustness in the yeast metabolic network. *Nature Genetics*, 38(9):993–998, 2006. 15

Bibliography

- N. C. Duarte, S. A. Becker, N. Jamshidi, I. Thiele, M. L. Mo, T. D. Vo, R. Srivas, and B. Ø. Palsson. Global reconstruction of the human metabolic network based on genomic and bibliomic data. *Proceedings of the National Academy of Sciences*, 104(6):1777–1782, 2007. 2, 114
- M. Durot, P.-Y. Bourguignon, and V. Schachter. Genome-scale models of bacterial metabolism: reconstruction and applications. *FEMS Microbiology Reviews*, 33(1):164–190, 2009. 2
- J. S. Edwards, M. Covert, and B. Ø. Palsson. Metabolic modelling of microbes: the flux-balance approach. *Environmental Microbiology*, 4(3):133–140, 2002. 5
- P. Erdrich, H. Knoop, R. Steuer, and S. Klamt. Cyanobacterial biofuels: new insights and strain design strategies revealed by computational modeling. *Microbial Cell Factories*, 13:128, 2014. 1, 153
- P. Erdrich, R. Steuer, and S. Klamt. An algorithm for the reduction of genome-scale metabolic network models to meaningful core models. *BMC Systems Biology*, 9:48, 2015. 32, 59, 60, 62, 63, 64, 71, 72, 79
- A. M. Feist and B. Ø. Palsson. The Growing Scope of Applications of Genome-scale Metabolic Reconstructions: the case of *E. coli*. *Nature Biotechnology*, 26(6):659–667, 2008. 1
- L. Finschi and K. Fukuda. Generation of Oriented Matroids—A Graph Theoretical Approach. *Discrete & Computational Geometry*, 27(1):117–136, 2002. 36
- S. S. Fong, A. P. Burgard, C. D. Herring, E. M. Knight, F. R. Blattner, C. D. Maranas, and B. O. Palsson. In silico design and adaptive evolution of escherichia coli for production of lactic acid. *Biotechnology and bioengineering*, 91(5):643–648, 2005. 3
- C. Francke, R. J. Siezen, and B. Teusink. Reconstructing the metabolic network of a bacterium from its genome. *Trends in Microbiology*, 13(11):550–558, 2005. 2
- K. Fukuda. cdd, 2005. https://www.inf.ethz.ch/personal/fukudak/cdd_home/cdd.html. 105
- K. Fukuda and A. Prodon. Double description method revisited. In *Combinatorics and Computer Science*, pages 91–111. Springer, 1996. 20, 24, 25, 94, 105, 119
- J. Gagneur and S. Klamt. Computation of elementary modes: a unifying framework and the new binary approach. *BMC Bioinformatics*, 5:175, 2004. 21, 23, 25, 94, 98, 99

-
- M. R. Garey and D. S. Johnson. *Computers and Intractability*, volume 29. W.H. Freeman and Company, New York, 2002. 31
- P. Gawand, P. Hyland, A. Ekins, V. J. Martin, and R. Mahadevan. Novel approach to engineer strains for simultaneous sugar utilization. *Metabolic engineering*, 20:63–72, 2013. 3
- M. P. Gerstl, S. Klamt, C. Jungreuthmayer, and J. Zanghellini. Exact quantification of cellular robustness in genome-scale metabolic networks. *Bioinformatics*, 32(5):730–737, 2015. 118
- J. Gilbert and L. Gilbert. *Linear Algebra and Matrix Theory*. Academic Press, 2014. 88, 101, 102, 129
- D. T. Gillespie, A. Hellander, and L. R. Petzold. Perspective: Stochastic algorithms for chemical kinetics. *The Journal of Chemical Physics*, 138(17):170901 1–14, 2013. 2
- A. M. Gleixner, D. E. Steffy, and K. Wolter. Improving the Accuracy of Linear Programming Solvers with Iterative Refinement. In *Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation*, pages 187–194. ISAAC’12, 2012. 32
- A. M. Gleixner, D. E. Steffy, and K. Wolter. Iterative Refinement for Linear Programming. *INFORMS Journal on Computing*, 28(3):449–464, 2016. 32
- S. A. Goldansaz, A. C. Guo, T. Sajed, M. A. Steele, G. S. Plastow, and D. S. Wishart. Livestock metabolomics and the livestock metabolome: A systematic review. *PLOS ONE*, 12(5):e0177675, 2017. 5
- Y. A. B. Goldstein and A. Bockmayr. Double and multiple knockout simulations for genome-scale metabolic network reconstructions. *Algorithms for Molecular Biology*, 10:1, 2015. 17, 67, 118
- G. H. Golub and C. F. Van Loan. *Matrix computations*, volume 3. JHU Press, 2012. 36
- C. M. González-Domenech, E. Belda, R. Patiño-Navarrete, A. Moya, J. Peretó, and A. Latorre. Metabolic stasis in an ancient symbiosis: genome-scale metabolic networks from two *Blattabacterium cuenoti* strains, primary endosymbionts of cockroaches. *BMC Microbiology*, 12(1):S5, 2012. 110, 111, 112, 123, 133
- E. Gruchattka, O. Hädicke, S. Klamt, V. Schütz, and O. Kayser. In silico profiling of *Escherichia coli* and *Saccharomyces cerevisiae* as terpenoid factories. *Microbial Cell Factories*, 12:84, 2013. 118

Bibliography

- I. Gurobi Optimization. Gurobi optimizer reference manual, 2016. <http://www.gurobi.com>. 32, 33
- O. Hädicke and S. Klamt. Computing complex metabolic intervention strategies using constrained minimal cut sets. *Metabolic Engineering*, 13(2):204–213, 2011. 17
- B.-J. Harder, K. Bettenbrock, and S. Klamt. Model-based metabolic engineering enables high yield itaconic acid production by *Escherichia coli*. *Metabolic Engineering*, 38:29–37, 2016. 118
- U.-U. Haus, S. Klamt, and T. Stephen. Computing Knock-Out Strategies in Metabolic Networks. *Journal of Computational Biology*, 15(3):259–268, 2008. 15, 118
- B. D. Heavner, K. Smallbone, N. D. Price, and L. P. Walker. Version 6 of the consensus yeast metabolic network refines biochemical coverage and improves model performance. *Database*, 2013, 2013. 154
- C. S. Henry, M. DeJongh, A. A. Best, P. M. Frybarger, B. Linsay, and R. L. Stevens. High-throughput generation, optimization and analysis of genome-scale metabolic models. *Nature Biotechnology*, 28(9):977, 2010. 2
- N. J. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, 2002. 33
- S. H. Hong, J. S. Kim, S. Y. Lee, Y. H. In, S. S. Choi, J.-K. Rih, C. H. Kim, H. Jeong, C. G. Hur, and J. J. Kim. The genome sequence of the capnophilic rumen bacterium *Mannheimia succiniciproducens*. *Nature Biotechnology*, 22(10):1275–1281, 2004. 110, 111, 112, 113
- M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, et al. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, 2003. 44
- IBM. *IBM ILOG CPLEX Optimization Studio CPLEX User's Manual*, 2011. 35
- IBM Knowledge Center. IBM ILOG CPLEX Optimizer, 2010. <http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer/>. 27, 32, 33, 35, 36, 45, 71, 144
- M. Imielinski and C. Belta. Exploiting the pathway structure of metabolism to reveal high-order epistasis. *BMC systems biology*, 2(1):40, 2008. 118

- F. Jaeger, D. L. Vertigan, and D. J. Welsh. On the computational complexity of the jones and tutte polynomials. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 108, pages 35–53. Cambridge University Press, 1990. 90
- N. Jamshidi and B. Ø. Palsson. Investigating the metabolic capabilities of *Mycobacterium tuberculosis* H37Rv using the *in silico* strain *iNJ661* and proposing alternative drug targets. *BMC Systems Biology*, 1(26), 2007. 123, 133
- D. Jevremović and D. Boley. Finding minimal generating set for metabolic network with reversible pathways. *BioSystems*, 112:31–36, 2013. 15, 104, 114, 116
- D. Jevremović, C. T. Trinh, F. Srienc, and D. Boley. A Simple Rank Test to Distinguish Extreme Pathways from Elementary Modes in Metabolic Networks. *Univ. of Minnesota, Computer Science and Eng. Dept. Tech. Rep*, 2008. 20, 22
- D. Jevremović, C. T. Trinh, F. Srienc, and D. Boley. On algebraic properties of extreme pathways in metabolic networks. *J. Comput. Biol.*, 17(2):107–119, 2010. 98
- S. Jonnalagadda and R. Srinivasan. An efficient graph theory based method to identify every minimal reaction set in a metabolic network. *BMC Systems Biology*, 8(28), 2014. 2, 31, 62
- C. Jungreuthmayer, M. Beurton-Aimar, and J. Zanghellini. Fast Computation of Minimal Cut Sets in Metabolic Networks with a Berge Algorithm that Utilizes Binary Bit Pattern Trees. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(5):1329–1333, 2013. 17, 118
- C. Kaleta, L. F. de Figueiredo, and S. Schuster. Can the whole be less than the sum of its parts? pathway analysis in genome-scale metabolic networks using elementary flux patterns. *Genome Research*, 19(10):1872–1883, 2009. 116
- M. Kanehisa and S. Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28(1):27–30, 2000. 108, 110, 111, 112
- N. Karmarkar. A new polynomial-time algorithm for linear programming. In *Proceedings of the sixteenth annual ACM symposium on Theory of computing*, pages 302–311. ACM, 1984. 27

Bibliography

- L. G. Khachiyan. Polynomial algorithms in linear programming. *USSR Computational Mathematics and Mathematical Physics*, 20(1):53–72, 1980. 27
- L. G. Khachiyan, E. Boros, K. Elbassioni, V. Gurvich, and K. Makino. On the Complexity of Some Enumeration Problems for Matroids. *SIAM Journal on Discrete Mathematics*, 19(4):966–984, 2005. 88
- Z. A. King, J. Lu, A. Dräger, P. Miller, S. Federowicz, J. A. Lerman, A. Ebrahim, B. Ø. Palsson, and N. E. Lewis. BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Research*, 44(D1):D515–D522, 2016. 37, 38, 45, 46, 47, 48, 53, 69, 71, 72, 73, 75, 108, 110, 111, 112, 113, 114, 123, 133, 134, 142, 145, 146, 194, 201
- H. Kitano. Systems Biology: A Brief Overview. *Science*, 295(5560):1662–1664, 2002. 1
- S. Klamt. Generalized concept of minimal cut sets in biochemical networks. *Biosystems*, 83(2):233–247, 2006. 9, 16, 118, 131
- S. Klamt and E. D. Gilles. Minimal cut sets in biochemical reaction networks. *Bioinformatics*, 20(2):226–234, 2004. 3, 8, 9, 15, 16, 59, 73, 118
- S. Klamt and J. Stelling. Combinatorial Complexity of Pathway Analysis in Metabolic Networks. *Molecular Biology Reports*, 29(1-2):233–236, 2002. 25
- S. Klamt and J. Stelling. Two approaches for metabolic pathway analysis? *Trends in Biotechnology*, 21(2):64–69, 2003. 16, 17
- S. Klamt, S. Müller, G. Regensburger, and J. Zanghellini. A mathematical framework for yield (*vs.* rate) optimization in constraint-based modeling and applications in metabolic engineering. *Metabolic Engineering*, 47:153–169, 2018. 2
- V. Klee and G. J. Minty. How good is the simplex algorithm? *Inequalities III*, pages 159–175, 1972. 27
- E. Klotz. Identification, Assessment, and Correction of Ill-Conditioning and Numerical Instability in Linear and Integer Programs. pages 54–108, 2014. 33, 35
- H. Knoop, M. Gründel, Y. Zilliges, R. Lehmann, S. Hoffmann, W. Lockau, and R. Steuer. Flux Balance Analysis of Cyanobacterial Metabolism: The Metabolic Network of *Synechocystis* sp. PCC 6803. *PLOS Computational Biology*, 9(6), 2013. 71, 72

- J. P. S. Kung. *A Source Book in Matroid Theory*. Springer, 1986. 43
- A. H. Land and A. G. Doig. An Automatic Method of Solving Discrete Programming Problems. *Econometrica*, 28(3):497–520, 1960. 32
- A. Larhlimi and A. Bockmayr. Minimal direction cuts in metabolic networks. *AIP Conference Proceedings*, 940:73–86, 2007. 138
- A. Larhlimi and A. Bockmayr. On Inner and Outer Descriptions of the Steady-State Flux Cone of a Metabolic Network. In *International Conference on Computational Methods in Systems Biology*, volume 6, pages 308–327. Springer, 2008. 83, 101
- A. Larhlimi and A. Bockmayr. A new constraint-based description of the steady-state flux cone of metabolic networks. *Discrete Applied Mathematics*, 157(10):2257–2266, 2009. 3, 14, 15, 17, 84, 94, 98, 105, 114, 119, 155
- A. Larhlimi, L. David, J. Selbig, and A. Bockmayr. F2C2, 2012a. <http://sourceforge.net/projects/f2c2/?source=pdf>. 52, 53, 69, 71
- A. Larhlimi, L. David, J. Selbig, and A. Bockmayr. F2C2: a fast tool for the computation of flux coupling in genome-scale metabolic networks. *BMC Bioinformatics*, 13:57, 2012b. 8, 28, 29, 30, 52, 53, 67, 69, 71
- N. E. Lewis, H. Nagarajan, and B. Ø. Palsson. Constraining the metabolic genotype-phenotype relationship using a phylogeny of *in silico* methods. *Nature Review Microbiology*, 10(4):291–305, 2012. 59
- C. Li, M. Donizelli, N. Rodriguez, H. Dharuri, L. Endler, V. Chelliah, L. Li, E. He, A. Henry, M. I. Stefan, et al. BioModels Database: An enhanced, curated and annotated resource for published quantitative kinetic models. *BMC Systems Biology*, 4:92, 2010. 108, 110, 111, 112, 113, 123, 132, 133
- Z. Li, R.-S. Wang, X.-S. Zhang, and L. Chen. Detecting drug targets with minimum side effects in metabolic networks. *IET Systems Biology*, 3(6): 523–533, 2009. 17, 57
- S. J. MacDonald, G. H. Thomas, and A. E. Douglas. Genetic and metabolic determinants of nutritional phenotype in an insect–bacterial symbiosis. *Molecular Ecology*, 20(10):2073–2084, 2011. 110, 111, 112, 113, 123, 133
- D. Machado and M. J. Herrgård. Co-evolution of strain design methods based on flux balance and elementary mode analysis. *Metabolic Engineering Communications*, 2:85–92, 2015. 118
- D. Machado, Z. Soons, K. R. Patil, E. C. Ferreira, and I. Rocha. Random sampling of elementary flux modes in large-scale metabolic networks. *Bioinformatics*, 28(18):i515–i521, 2012. 155

Bibliography

- A. Mackie, I. M. Keseler, L. Nolan, P. D. Karp, and I. T. Paulsen. Dead End Metabolites-Defining the Known Unknowns of the *E. coli* Metabolic Network. *PLOS ONE*, 8(9):e75210, 2013. 44
- S. Magnúsdóttir, A. Heinken, L. Kutt, D. A. Ravcheev, E. Bauer, A. Noronha, K. Greenhalgh, C. Jäger, J. Baginska, P. Wilmes, R. M. T. Fleming, and I. Thiele. Generation of genome-scale metabolic reconstructions for 773 members of the human gut microbiota. *Nature Biotechnology*, 2016. 2
- R. Mahadevan and C. H. Schilling. The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic Engineering*, 5(4):264–276, 2003. 7, 28, 60
- S. J. Maher, T. Fischer, T. Gally, G. Gamrath, A. Gleixner, R. L. Gottwald, G. Hendel, T. Koch, M. E. Lübbecke, M. Miltenberger, B. Müller, M. E. Pfetsch, C. Puchert, D. Rehfeldt, S. Schenker, R. Schwarz, F. Serrano, Y. Shinano, D. Weninger, J. T. Witt, and J. Witzig. The SCIP Optimization Suite 4.0. Technical Report 17-12, ZIB, Berlin, Germany, 2017. 32
- S.-A. Marashi and A. Bockmayr. Flux coupling analysis of metabolic networks is sensitive to missing reactions. *BioSystems*, 103(1):57–66, 2011. 79
- S.-A. Marashi, L. David, and A. Bockmayr. Analysis of metabolic subnetworks by flux cone projection. *Algorithms for Molecular Biology*, 7(1):17, 2012. 120, 155
- M. Margulies, M. Egholm, W. E. Altman, S. Attiya, J. S. Bader, L. A. Bembien, J. Berka, M. S. Braverman, Y.-J. Chen, Z. Chen, et al. Genome Sequencing in Open Microfabricated High Density Picolitre Reactors. *Nature*, 437(7057):376–380, 2005. 1
- G. Melzer, M. E. Esfandabadi, E. Franco-Lara, and C. Wittmann. Flux Design: In silico design of cell factories based on correlation of pathway fluxes to desired properties. *BMC Systems Biology*, 3:120, 2009. 9, 14
- L. Menten and M. Michaelis. Die Kinetik der Invertinwirkung. *Biochem Z*, 49:333–369, 1913. 5
- G. Nair, C. Jungreuthmayer, and J. Zanghellini. Optimal knockout strategies in genome-scale metabolic networks using particle swarm optimization. *BMC Bioinformatics*, 18(1):78, 2017. 17, 118
- J. Nielsen and S. Oliver. The next wave in metabolome analysis. *TRENDS in Biotechnology*, 23(11):544–546, 2005. 5

- R. A. Notabaart, F. H. J. Van Enkevort, C. Francke, R. J. Siezen, and B. Teusink. Accelerating the reconstruction of genome-scale metabolic networks. *BMC Bioinformatics*, 7:296, 2006. 2
- M. A. Oberhardt, B. Ø. Palsson, and J. A. Papin. Applications of genome-scale metabolic reconstructions. *Molecular Systems Biology*, 5:320, 2009. 1
- J. D. Orth, R. M. Fleming, and B. Ø. Palsson. Reconstruction and Use of Microbial Metabolic Networks: the Core *Escherichia coli* Metabolic Model as an Educational Guide, 2010. 123, 133
- J. A. Papin, J. Stelling, N. D. Price, S. Klamt, S. Schuster, and B. Ø. Palsson. Comparison of network-based pathway analysis methods. *Trends in Biotechnology*, 22(8):400–405, 2004. 3, 113
- J. Pey and F. J. Planes. Direct calculation of elementary flux modes satisfying several biological constraints in genome-scale metabolic networks. *Bioinformatics*, 30(15):2197–2203, 2014. 94
- T. Pfeiffer, I. Sánchez-Valdenebro, J. C. Nuño, F. Montero, and S. Schuster. METATOOL: for studying metabolic networks. *Bioinformatics*, 15(3):251–257, 1999. 25, 29, 60, 94
- P. Pharkya and C. D. Maranas. An optimization framework for identifying reaction activation/inhibition or elimination candidates for overproduction in microbial systems. *Metabolic Engineering*, 8(1):1–13, 2006. 8, 9, 17
- N. D. Price, J. L. Reed, and B. Ø. Palsson. Genome-scale models of microbial cells: evaluating the consequences of constraints. *Nature Reviews Microbiology*, 2:886–897, 2004. 2, 5
- L.-E. Quek and L. K. Nielsen. A depth-first search algorithm to compute elementary flux modes by linear programming. *BMC Systems Biology*, 8:94, 2014. 25
- A. Reimers. *Metabolic Networks, Thermodynamic Constraints, and Matroid Theory*. PhD thesis, Freie Universität Berlin, 2014. 9, 40, 41
- A. C. Reimers, Y. Goldstein, and A. Bockmayr. Generic flux coupling analysis. *Mathematical Biosciences*, 262:28–35, 2015. 28
- A.-M. Reimers. *Understanding metabolic regulation and cellular resource allocation through optimization*. PhD thesis, Freie Universität Berlin, 2017. 154

Bibliography

- O. Resendis-Antonio, J. L. Reed, S. Encarnación, J. Collado-Vides, and B. Ø. Palsson. Metabolic Reconstruction and Modeling of Nitrogen Fixation in *Rhizobium etli*. *PLoS Computational Biology*, 3(10):e192, 2007. 110, 111, 112, 113, 123, 133
- A. Rezola, L. F. de Figueiredo, M. Brock, J. Pey, A. Podhorski, C. Wittmann, S. Schuster, A. Bockmayr, and F. J. Planes. Exploring metabolic pathways in genome-scale networks via generating flux modes. *Bioinformatics*, 27(4):534–540, 2011. 94, 116, 119
- S. B. Roberts, C. M. Gowen, J. P. Brooks, and S. S. Fong. Genome-scale metabolic analysis of *Clostridium thermocellum* for bioethanol production. *BMC Systems Biology*, 4:31, 2010. 110, 111, 112, 113
- A. Röhl and A. Bockmayr. A mixed-integer linear programming approach to the reduction of genome-scale metabolic networks. *BMC Bioinformatics*, 18:2, 2017. 59
- A. Röhl and A. Bockmayr. Reaction Splitting and Minimum Sets of Elementary Flux Modes. *Proceedings of the Lyon Spring School on advances in Systems and Synthetic Biology*, 16, 2017. 93
- A. Röhl, Y. A. B. Goldstein, and A. Bockmayr. EFM-Recorder - Faster Elementary Mode Enumeration via Reaction Coupling Order. *Proceedings of the Strasbourg Spring School on advances in Systems and Synthetic Biology*, 14:91–100, 2015. 49, 67
- D. E. Ruckerbauer, C. Jungreuthmayer, and J. Zanghellini. Predicting genetic engineering targets with Elementary Flux Mode Analysis: a review of four current methods. *New Biotechnology*, 32(6):534–546, 2015. 14
- M. Rügen, A. Bockmayr, and R. Steuer. Elucidating temporal resource allocation and diurnal dynamics in phototrophic metabolism using conditional FBA. *Scientific Reports*, 5(15247), 2015. 2
- E. Ruppin, J. A. Papin, L. F. De Figueiredo, and S. Schuster. Metabolic reconstruction, constraint-based analysis and game theory to probe genome-scale metabolic networks. *Current Opinion in Biotechnology*, 21(4):502–510, 2010. 2
- J. Schellenberger, R. Que, R. M. Fleming, I. Thiele, J. D. Orth, A. M. Feist, D. C. Zielinski, A. Bordbar, N. E. Lewis, S. Rahmanian, et al. Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2. 0. *Nature Protocols*, 2(3):727–738, 2007. 44
- C. H. Schilling, D. Letscher, and B. Ø. Palsson. Theory for the Systemic Definition of Metabolic Pathways and their use in Interpreting Metabolic

- Function from a Pathway-Oriented Perspective. *Journal of Theoretical Biology*, 203(3):229–248, 2000. 99, 113, 114, 116, 155
- C. H. Schilling, M. W. Covert, I. Famili, G. M. Church, J. S. Edwards, and B. Ø. Palsson. Genome-Scale Metabolic Model of *Helicobacter pylori* 26695. *Journal of Bacteriology*, 184(16):4582–4593, 2002. 110, 111, 112, 113, 123, 133
- A. Schrijver. *Theory of linear and integer programming*. Wiley, Chichester; New York, 1998. 7, 8, 18, 19, 22, 25, 32, 98, 99, 105, 141
- B. S. W. Schröder. *Ordered Sets*. Springer, 2003. 29, 30
- S. Schuster and C. Hilgetag. On elementary flux modes in biochemical reaction systems at steady state. *Journal of Biological Systems*, 2(02):165–182, 1994. 3, 8, 13, 14, 21, 59, 98
- S. Schuster, D. A. Fell, and T. Dandekar. A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. *Nature Biotechnology*, 18(3):326–332, 2000. 14, 23
- S. Schuster, C. Hilgetag, J. H. Woods, and D. A. Fell. Reaction routes in biochemical reaction systems: Algebraic properties, validated calculation procedure and example from nucleotide metabolism. *Journal of Mathematical Biology*, 45(2):153–181, 2002. 115
- J.-M. Schwartz and M. Kanehisa. A quadratic programming approach for decomposing steady-state metabolic flux distributions onto elementary modes. *Bioinformatics*, 21(suppl 2):ii204–ii205, 2005. 14
- J.-M. Schwartz and M. Kanehisa. Quantitative elementary mode analysis of metabolic pathways: the example of yeast glycolysis. *BMC Bioinformatics*, 7:186, 2006. 14
- D. Segre, D. Vitkup, and G. M. Church. Analysis of optimality in natural and perturbed metabolic networks. *Proceedings of the National Academy of Sciences*, 99(23):15112–15117, 2002. 2
- T. Shlomi, T. Benyamini, E. Gottlieb, R. Sharan, and E. Ruppin. Genome-Scale Metabolic Modeling Elucidates the Role of Proliferative Adaptation in Causing the Warburg Effect. *PLoS Computational Biology*, 7(3):e1002018, 2011. 1
- J. Stelling, S. Klamt, K. Bettenbrock, S. Schuster, and E. D. Gilles. Metabolic network structure determines key aspects of functionality and regulation. *Nature*, 420:190–193, 2002. 8, 14, 154

Bibliography

- P. F. Suthers, M. S. Dasika, V. S. Kumar, G. Denisov, J. I. Glass, and C. D. Maranas. A Genome-Scale Metabolic Reconstruction of *Mycoplasma genitalium*, iPS189. *PLoS Computational Biology*, 5(2):e1000285, 2009a. 110, 111, 112
- P. F. Suthers, A. Zomorodi, and C. D. Maranas. Genome-scale gene/reaction essentiality and synthetic lethality analysis. *Molecular Systems Biology*, 5(1):301, 2009b. 8, 17
- M. Terzer. *Large scale methods to enumerate extreme rays and elementary modes*. PhD thesis, Eidgenössische Technische Hochschule ETH Zürich, 2009a. 8, 24
- M. Terzer. *Large scale methods to enumerate extreme rays and elementary modes*. PhD thesis, ETH Zurich, 2009b. 94, 132, 137
- M. Terzer. efmtool, 2017a. <http://www.csb.ethz.ch/tools/software/efmtool.html>. 13, 25, 108, 111, 112, 123
- M. Terzer. polco, 2017b. <http://www.csb.ethz.ch/tools/software/polco.html>. 24, 105, 108, 111, 112, 123, 155
- M. Terzer, N. D. Maynard, M. W. Covert, and J. Stelling. Genome-scale metabolic networks. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 1(3):285–297, 2009. 2
- The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 7.4.1)*, 2016. <http://www.sagemath.org>. 91, 131, 132, 141
- I. Thiele and B. Ø. Palsson. A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nature Protocols*, 5(1):93–121, 2010. 2
- I. Thiele, D. R. Hyduke, B. Steeb, G. Fankam, D. K. Allen, S. Bazzani, P. Charusanti, F.-C. Chen, R. M. T. Fleming, C. A. Hsiung, et al. A community effort towards a knowledge-base and mathematical model of the human pathogen *Salmonella* Typhimurium LT2. *BMC Systems Biology*, 5(1):8, 2011. 123, 133
- L. Tobalina, J. Pey, and F. J. Planes. Direct calculation of minimal cut sets involving a specific reaction knock-out. *Bioinformatics*, 32(13):2001–2007, 2016. 57, 118, 138, 140, 141, 144, 148, 149, 150, 194, 201
- C. T. Trinh and F. Sreenc. Metabolic Engineering of *Escherichia coli* for Efficient Conversion of Glycerol to Ethanol. *Applied and Environmental Microbiology*, 75(21):6696–6705, 2009. 14
- C. T. Trinh, R. Carlson, A. Wlaschin, and F. Sreenc. Design, construction and performance of the most efficient biomass producing *E. coli* bacterium. *Metabolic Engineering*, 8(6):628–638, 2006. 14, 15

- C. T. Trinh, P. Unrean, and F. Sreenc. Minimal *Escherichia coli* Cell for the Most Efficient Production of Ethanol from Hexoses and Pentoses. *Applied and Environmental Microbiology*, 74(12):3634–3643, 2008. 14
- C. T. Trinh, J. Li, H. W. Blanch, and D. S. Clark. Redesigning *Escherichia coli* Metabolism for Anaerobic Production of Isobutanol. *Applied and Environmental Microbiology*, 77(14):4894–4904, 2011. 9, 14
- J. J. Tyson, K. Chen, and B. Novak. Network dynamics and cell physiology. *Nature Reviews Molecular Cell Biology*, 2(12):908–916, 2001. 2
- E. Ullah, S. Aeron, and S. Hassoun. gEFM: An Algorithm for Computing Elementary Flux Modes Using Graph Traversal. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 13(1):122–134, 2016. 25
- P. Unrean, C. T. Trinh, and F. Sreenc. Rational design and construction of an efficient *E. coli* for production of diapolycopendioic acid. *Metabolic Engineering*, 12(2):112–122, 2010. 14
- R. Urbanczik. SNA—a toolbox for the stoichiometric analysis of metabolic networks. *BMC Bioinformatics*, 7(129), 2006. 120
- R. Urbanczik and C. Wagner. An improved algorithm for stoichiometric network analysis: theory and applications. *Bioinformatics*, 21(7):1203–1210, 2005a. 22, 25, 94
- R. Urbanczik and C. Wagner. Functional stoichiometric analysis of metabolic networks. *Bioinformatics*, 21(22):4176–4180, 2005b. 120, 155
- J. H. van Heerden, M. T. Wortel, F. J. Bruggeman, J. J. Heijnen, Y. J. Bollen, R. Planqué, J. Hulshof, T. G. O’Toole, S. A. Wahl, and B. Teusink. Lost in transition: start-up of glycolysis yields subpopulations of nongrowing cells. *Science*, 343(6174):1245114, 2014. 2
- A. Varma and B. Ø. Palsson. Metabolic Flux Balancing: Basic Concepts, Scientific and Practical Use. *Bio/technology*, 12, 1994. 2, 3, 7, 27
- S. G. Villas-Bôas and P. Bruheim. Cold glycerol–saline: The promising quenching solution for accurate intracellular metabolite analysis of microbial cells. *Analytical Biochemistry*, 370(1):87–97, 2007. 5
- N. Vlassis, M. P. Pacheco, and T. Sauter. Fast Reconstruction of Compact Context-Specific Metabolic Network Models. *PLoS Computational Biology*, 10(1):e1003424, 2014. 62, 71, 72, 79
- A. von Kamp and S. Klamt. Enumeration of Smallest Intervention Strategies in Genome-Scale Metabolic Networks. *PLoS Computational Biology*, 10(1):e1003378, 2014. 15, 17, 57, 118, 138, 140, 141

- A. von Kamp and S. Klamt. Growth-coupled overproduction is feasible for almost all metabolites in five major production organisms. *Nature Communications*, 8:15956, 2017. 118
- P. Waage and C. M. Gulberg. Studies concerning affinity. *Videnskabs-Selskabet i Christiania*, 1864. 5
- C. Wagner and R. Urbanczik. The Geometry of the Flux Cone of a Metabolic Network. *Biophysical Journal*, 89(6):3837–3845, 2005. 120
- S. Waldherr, D. A. Oyarzún, and A. Bockmayr. Dynamic optimization of metabolic networks coupled with gene expression. *Journal of Theoretical Biology*, 365:469–485, 2015. 2, 59
- M. Walhout, M. Vidal, and J. Dekker. *Handbook of systems biology: concepts and insights*. Academic Press, 2012. 2
- D. Welsh. The Tutte Polynomial. *Random Structures and Algorithms*, 15(3-4):210–228, 1999. 90
- S. J. Wiback and B. Ø. Palsson. Extreme Pathway Analysis of Human Red Blood Cell Metabolism. *Biophysical Journal*, 83(2):808–818, 2002. 120
- T. Wilhelm, J. Behre, and S. Schuster. Analysis of structural robustness of metabolic networks. *Systems biology*, 1(1):114–120, 2004. 118
- J. A. H. Wodke, J. Puchałka, M. Lluch-Senar, J. Marcos, E. Yus, M. Godinho, R. Gutiérrez-Gallego, V. A. P. M. Dos Santos, L. Serrano, E. Klipp, and T. Maier. Dissecting the energy metabolism in *Mycoplasma pneumoniae* through genome-scale metabolic modeling. *Molecular Systems Biology*, 9(1):653, 2013. 110, 111, 112, 113
- C. Wrzodek, F. Büchel, M. Ruff, A. Dräger, and A. Zell. Precise generation of systems biology models from KEGG pathways. *BMC Systems Biology*, 7:15, 2013. 108, 110
- J. Zanghellini, D. E. Ruckerbauer, M. Hanscho, and C. Jungreuthmayer. Elementary flux modes in a nutshell: Properties, calculation and applications. *Biotechnology Journal*, 8(9):1009–1016, 2013. 8, 14
- G. M. Ziegler. *Lectures on Polytopes*, volume 152. Springer Science & Business Media, 1995. 119, 120, 124, 129

Notation

All notations used in the thesis:

- \mathbb{B} : set of binary numbers
- \mathbb{N} : set of natural numbers
- \mathbb{Q} : set of rational numbers
- \mathbb{R} : set of real numbers
- $\mathbb{R}_{\geq 0}$: set of non-negative real numbers
- x_i : i -th element of the vector $x \in \mathbb{R}^n$
- x_H : vector containing the elements of the indices of H
- $x \geq 0$: all elements of $x \in \mathbb{R}^n$ are non-negative, hence $x_i \geq 0$ for all i
- $A_{i,*}$: i -th row of a matrix $A \in \mathbb{R}^{m \times n}$
- $A_{*,j}$: j -th column of a matrix $A \in \mathbb{R}^{m \times n}$
- $A_{H,*}$, resp. $A_{*,H}$: rows, resp. columns of a matrix $A \in \mathbb{R}^{m \times n}$ corresponding to the indices of the set H
- $\rho(A)$: the rank of a matrix $A \in \mathbb{R}^{m \times n}$, the maximum number of linear independent columns of A
- $\text{supp}(x)$: support of a vector x .
- Min : minimal subsets of a set: $\text{Min}(\mathcal{V}) := \{v \in \mathcal{V} \mid \nexists w \in \mathcal{V} \setminus \{v\} \text{ s.t. } w \subset v\}$.
- $(A|B)$ or $\begin{pmatrix} A \\ B \end{pmatrix}$: concatenation of two matrices A and B : columns and rows wise

- $\sigma(x) \in \{-, 0, +\}^n$: sign vector of a vector $x \in \mathbb{R}^n$
- X^+ resp. X^- : all positive, resp. negative non-zero elements of X , hence $X^+ = \{i \mid x_i > 0\} = \{i \mid \sigma(x_i) = +\}$
- $X \circ Y$: composition of two signed vectors: $(X \circ Y)^+ = X^+ \cup (Y^+ \setminus X^-)$ and $(X \circ Y)^- = X^- \cup (Y^- \setminus X^+)$
- $i \stackrel{=0}{\leftrightarrow} j$: the reactions i and j are partially coupled: if $v_i = 0 \Leftrightarrow v_j = 0$ for all $v \in \Gamma_{\mathcal{N}}$
- $X|_F$: contraction of \mathcal{M} to F denoted by \mathcal{M}/A where $A = U \setminus F$, where $F \subseteq U$.
- $\text{proj}_H(\Gamma)$: projection of Γ onto the variables $i \in H$
- $\tilde{\Gamma}^I$: polyhedral cone, related to $\Gamma = \{x \in \mathbb{R}^n \mid Ax \geq 0\}$, where the variables $i \in I$ are split: $\tilde{\Gamma}^I = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^{n+|I|} \mid (A \mid -A_{*,I}) \begin{pmatrix} x \\ y \end{pmatrix} \geq 0, x_I \geq 0, y \geq 0 \right\}$.

Names of variables:

- \mathfrak{R} : set of reactions
- Irr : set of irreversible reactions
- Rev : set of reversible reactions
- Int_{Rev} : set of internal reversible reactions
- \mathfrak{M} : set of metabolites
- S : stoichiometric matrix
- $\underline{\mathcal{M}}$: matroid
- \mathcal{M} : oriented matroid
- \mathcal{B} : basis of a matroid $\underline{\mathcal{M}}$
- \mathcal{M}_S : flux mode matroid corresponding to the metabolic network described by the stoichiometric matrix $S \in \mathbb{R}^{\mathfrak{M} \times \mathfrak{R}}$
- $\underline{\mathcal{C}}$ resp. \mathcal{C} : circuits resp. oriented circuits
- Γ : polyhedral cone
- $\Gamma_{\mathcal{N}}$: flux cone related to the metabolic network $\mathcal{N} = (\mathfrak{M}, \mathfrak{R}, S, \text{Irr})$

- $v \in \Gamma_{\mathcal{N}}$: feasible flux vector
- P : polyhedron
- Λ : lineality space
- $\Lambda_{\mathcal{N}}$: lineality space of the flux cone $\Gamma_{\mathcal{N}}$, thus $\Lambda_{\mathcal{N}} = \Gamma_{network} \cap (-\Gamma_{\mathcal{N}})$
- π : map
- Π : 2 cycles
- Frev: fully reversible reactions
- minFrev: set of minimum number of fully reversible reactions, s.t. splitting them the augmented cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$ is pointed
- \hat{e}_k : k -th unit vector

Abbreviations

- DD: Double Description
- EFM: Elementary Flux Mode
- EP: Extreme Pathway
- ER: Extreme Ray
- FBA: Flux Balance Analysis
- FC: Fundamental oriented Circuit
- FCA: Flux Coupling Analysis
- FrevEFM: Recombined ERs of the pointed cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{Frev}}$. Unique set of EFMs. Can describe whole flux space $\Gamma_{\mathcal{N}}$, but is not a minimum set of EFMs
- FVA: Flux Variability Analysis
- GFM: Generating Flux Modes
- iMCS: irreversible Minimal Cut Set
- LP: Linear Programming
- MCS: Minimal Cut Set

- MEMo: Recombined ERs of the pointed cone $\tilde{\Gamma}_{\mathcal{N}}^{\text{minFrev}}$. **M**inimum set of **E**lementary flux **M**odes
- MILP: Mixed Integer Linear Programming
- MMB: Minimal Metabolic Behaviour
- MPF: Minimal Proper Face
- RMS: Reversible Metabolic Space
- SBML: Systems Biology Markup Language

Index

- Escherichia coli* iAF1260, 72
Helicobacter pylori 26695, 72
Mycobacterium tuberculosis
iNJ661, 72
Synechocystis sp. PCC 6803, 72
Bacillus subtilis subsp. *subtilis*
str. 168, 74, 75
Blattabacterium cuenoti iCG238,
110–112
Blattabacterium iCG230, 110–112
Buchnera iSM197, 110–113
Clostridium ljungdahlii DSM
13528, 74, 75
Clostridium thermocellum
iSR432, 110–113
Escherichia coli MG1655,
110–113
Escherichia coli iJO1366, 74, 75
Escherichia coli textbook, 53–55
Geobacter metallireducens GS-15,
74, 75
Helicobacter pylori 26695, 74, 75,
77, 110–113
Helicobacter pylori iCS291,
110–113
Helicobacter pylori iIT341, 53–55
Human Erythrocyte
iAB-RBC-283, 110–112
Klebsiella pneumoniae
subsp. *pneumoniae*
pneumoniae MGH 78578,
74, 75
Mannheimia succiniciproducens
iSH335, 110–113
Methanosarcina barkeri
str. *Fusaro*, 74, 75
Mus musculus, 74, 75
Mycobacterium tuberculosis
iNJ661, 53–55, 74, 75
Mycoplasma genitalium iPS189,
110–112
Mycoplasma pneumoniae
iJW145, 110–113
Pseudomonas putida KT2440, 74,
75
Rhizobium etli iOR363, 110–113
Saccharomyces cerevisiae S288c,
74, 75
Saccharomyces cerevisiae
iND750, 53–55
Salmonella enterica
subsp. *enterica* serovar
Typhimurium LT2, 74, 75
Shigella boydii CDC 3083-94, 74,
75
Staphylococcus aureus
subsp. *aureus* N315, 74,
75
Staphylococcus aureus
subsp. *aureus* iSB619,
53–55
Thermotoga maritima MSB8, 74,
75
Yersinia pestis CO92, 74, 75

2-cycle, 23
 basis, 42, 83, 88–91
 Big M, 35–36, 50
 binary variable, 31, 50
 biomass reaction, 4, 7, 27, 63, 73, 74, 77
 branch and bound, 32
 chemical reaction, 4
 circuits, *see* oriented circuits
 column matroid, 39
 column space, 107
 composition, 40
 condition number, 37, 38
 cone, 19
 coupled, 29
 directionally, 29
 fully, 29
 partially, 29
 CPLEX, 32
 cplex, 144
 Double Description method, 24
 efmtool, 13, 108
 elementary flux modes, 8, 13–14, 16, 22, 41, 49–52, 95, 99, 108
 equivalence class, 29
 extreme currents, 23
 extreme pathways, 108
 extreme ray, 20, 24, 85, 99
 F2C2, 52, 53, 71
 face, 19
 facet, 20
 FASTCORE, 62, 71–72
 feasible solution, 26
 flux balance analysis, 7, 27–28
 flux cone, 8, 12
 flux coupling analysis, 8, 28–31
 application, 50–52
 flux mode matroid, 41
 flux variability analysis, 7, 28
 flux vector, 7, 12
 fundamental circuits, *see* oriented fundamental circuits
 graphic matroid, 39
 growth rate, 7
 growth yield, 7, 27
 Gurobi, 32
 hypergraph, 5
 indicator variables, 36, 71
 irreversible minimal cut set, 117–135, 137–151
 lineality space, 19, 83
 linear program, 7–8, 25–31
 MEMo, 93–116
 metabolic network, 4–7, 12, 81
 metabolite, 4
 concentration, 5
 external and internal, 5
 minFrev, 101
 minimal cut set, 9, 15–17, 117, 137
 minimal metabolic behaviour, 14–15, 117
 mixed integer linear program, 8, 31–32, 50–52, 63–71
 NetworkReducer, 59, 62, 71
 numerical instability, 33–35
 optimal solution, 26
 oriented circuits, 39, 41, 82
 oriented cycle, 40, 82
 oriented fundamental circuits, 42–44, 83, 85, 88–89
 oriented matroid, 9, 36–44, 81
 polco, 24, 108
 polyhedral cone, 8–9, 19, 81
 pointed, 20
 polyhedron, 19
 ray, 20, 82

- reaction, 4
 - active, 5, 12
 - blocked, 28
 - essential, 8, 16, 73, 77
 - exchange and internal, 4
 - fully reversible, 95
 - irreversible, 4, 7, 12
 - pseudo-irreversible, 95
 - rate, 5
 - reversible, 4, 12
- reaction rate, 12
- reduced column echelon form, 88
- reduced row echelon form, *see*
 - reduced column echelon form
- representative, 49, 53
- SAGE, 91
- SCIP, 32
- sign vector, 41
- simplex algorithm, 26
- solution space, 7, 26
- SoPlex, 32
- split variable, 22–25
- steady-state, 5, 12
- stoichiometric matrix, 4, 12
 - stoichiometric coefficient, 4, 12
- support, 13
- Tutte Polynomial, 90
- uptake rates, 7
- vector matroid, 36
- zero set, 20

Chapter 10

Appendix

10.1 Appendix for RedNet

10.1.1 Comparison with NetworkReducer

For the reduction of *Synechocystis* sp. PCC 6803:

Reactions which were in the subnetwork computed with `NetworkReducer` but not in the subnetwork computed with the MILP-approach `minNW`:

PP0007, GS0008, PP0014, PP0009, PT0009, GS0010, GS0004, GS0018, PP0010, PP0013, PT0001, PP0001, TR0050, PP0011, GE0001, PP0012, GS0013, TR0051, GS0014, PP0003, GS0011, TR0052, PR0001, PDC, EtOH_ex, Bio_T, PR0028, PY0016, PR0016, PR0029, PP0020, PR0032, PR0033, PT0026, PR0030, TE0008, PR0031, PR0051, PT0021, PR0004, GE0002, PR0035, PR0049, PR0036, PR0050, TR0036, TR0003, TR0006, TR0002.

Reactions which were in the subnetwork computed with the MILP-approach `minNW` but not in the subnetwork computed with `NetworkReducer`:

TR0050, TR0023, GL0003, RI0004, PU0011, PG0008, RI0002, GE0015, CP0007, LI0009, QT0001, TP0002, QT0013, CA0008, LI0058, CP0021, QT0014, AG0020, PY0020, GL0001, LI0004, GL0004, CA0004, PU0010, FO0017, TR0015, LI0033, PY0007, LI0057, GL0005, LI0002, BM0007,

GE0004, PR0023, PR0003, PR0027, PR0035, PR0042, TR0055, PR0046.

For the reduction of *E. coli* iAF1260:

Reactions which were in the subnetwork computed with NetworkReducer but not in the subnetwork computed with the MILP-approach minNW:

R_ACALDt_{tex}, R_ACALDt_{pp}, R_ADNK1, R_AKGt_{2rpp}, R_AKGt_{tex}, R_CYTBD_{pp}, R_D_LACt_{2pp}, R_D_LACt_{tex}, R_EX_acald_{e-}, R_EX_akg_{e-}, R_EX_fe2_{e-}, R_EX_for_{e-}, R_EX_glu_{L_e-}, R_EX_lac_{D_e-}, R_EX_pyr_{e-}, R_FBP, R_FE2_{tex}, R_FE2_{tp}, R_FORt_{2pp}, R_FORt_{tex}, R_FORt_{ppi}, R_FRD3, R_GART, R_GLUN, R_GLUSy, R_GLUt_{2rpp}, R_GLUt_{tex}, R_GLYCTO4, R_GRXR, R_GTHOr, R_ICL, R_LDH_D, R_ME1, R_ME2, R_NADTRHD, R_NDPK1, R_NDPK6, R_PAPSR2, R_PPCK, R_PPM, R_PPS, R_PUNP1, R_PYRt_{2rpp}, R_PYRt_{tex}, R_SUCc_{2_2pp}, R_THRt_{2rpp}, R_THRt_{4pp}, R_TRPS2, R_TRPS3, R_URIDK2r, R_RFD_{new}.

Reactions which were in the subnetwork computed with the MILP-approach minNW but not in the subnetwork computed with NetworkReducer:

R_ACOTA, R_ADK1, R_ADNK1, R_CYTBD_{pp}, R_DPR, R_EX_mn2_{e-}, R_EX_o2_{e-}, R_GAPD, R_GARFT, R_MTHFD, R_RNTR3c, R_RNTR4c.

10.1.2 Comparison with FASTCORE

For the reduction of *M. tuberculosis* iNJ661 :

Reaction which were in the subnetwork computed with FASTCORE but not in the subnetwork computed with the MILP-approach minNW:

TALA, SUCc_{2r}, RPI, RPE, PSP_L, PSERT, PRPPS, PREPTTA, PREPTHs, PPA_{tr}, PHTHS, PHTHDLS, PGCD, PDIMAT, PDIMAS, MMM2, MME, MMCD, METS, METAT, UMPK_{copy2}, TKT2, TKT1, MALS, L_LACD3, ICL, H2O_t, GLYCL, GLUD_{xi}, GLNS, GHMT_{2r}, FRDO3r, FRDO2r, EX_{h_e}, EX_{h2o_e}, EX_{cytd_e}, ORPT, OMPDC,

OCOAT1, NTPP4, NTD4, NH4t, NDPK1, MTHFR2, DHORTS, CYTDt2, CYTBD2, CTPS1, CITL_copy1, CBPS, ASPTA, ASPCT, ARGSS, ARGSL, ARGDr, ARACHTA, AHCi, ADNK1, FASm280, FASm260, FASm240, FASm220, FAS80_L, FAS200, FAS180, FAS160, FAS140, FAS120, FAS100, FACOALPREPH, FACOAL200, EX_succ_e, EX_ppa_e, EX_pdima_e, EX_nh4_e, ADK1, ACCOAC, ACACCT.

Reactions which were in the subnetwork computed with the MILP-approach `minNW` but not in the subnetwork computed with `FASTCORE`:

TALA, PFK, MALS, LLACD3, GLYCL, GLUDxi.

For the reduction of *H. pylori* **26695**:

Reactions which were in the subnetwork computed with `FASTCORE` but not in the subnetwork computed with the MILP-approach `minNW`:

ACt2r, ALAt2r, ANS2, APRAUR, CO2t, DAAD, DB4PS, DHPPDA, EX_ac(e), EX_ala_L(e), EX_co2(e), EX_h2co3(e), EX_urea(e), GTPCII, H2CO3D, H2CO3TP, PMDPHT, RBFSa, RBFSb, RNDR4, UREAt, URIDK2r.

Reactions which were in the subnetwork computed with the MILP-approach `minNW` but not in the subnetwork computed with `FASTCORE`:

ACt2r, ALAt2r, AMAOTr, AOXSr, DASYN_HP, DPCOAK, EX_ala_D(e), FLDO, UAGDP.

10.1.3 New experiments

Mus musculus: The maximal biomass rate with oxygen uptake is 1.3634 h^{-1} . Without oxygen, the rate is 0.0971 h^{-1} . Per default the lower bound of oxygen is -100 mmol/gDW/h and -1 mmol/gDW/h of glucose. The upper bounds of both are $100000 \text{ mmol/gDW/h}$. The requirements are that at

least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 2436 non-blocked reactions. All the requirements can be fulfilled by using 338 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 274.

***E. coli* iJO1366:** The requirements are that at least 99.9 % of the maximal growth rate under aerobic conditions (37.9623 h^{-1}) and at least 99.9 % of the maximal growth rate under anaerobic conditions (28.6643 h^{-1}) can be realized by the subnetwork. The original network consists of 2369 non-blocked reactions. Per default the lower bounds of oxygen and glucose are both -1000 mmol/gDW/h . The upper bounds of both are 1000 mmol/gDW/h . All the requirements can be fulfilled by using 562 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 363.

***S. Typhimurium* LT2:** The requirements are that at least 99.9 % of the maximal growth rate under aerobic conditions (34.4108 h^{-1}) and under anaerobic conditions (18.3065 h^{-1}) can be realized by the subnetwork. The original network consists of 1620 non-blocked reactions. Per default the lower bounds of oxygen and glucose are both -1000 mmol/gDW/h . The upper bounds of both are 1000 mmol/gDW/h . All the requirements can be fulfilled by using 458 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 305.

***S. boydii* CDC 3083-94:** The requirements are that at least 99.9 % of the maximal growth rate under aerobic conditions (0.9828 h^{-1}) and at least 99.9 % of the maximal growth rate under anaerobic conditions (0.2418 h^{-1}) can be realized by the subnetwork. The original network consists of 1546 non-blocked reactions. Per default the lower bound of oxygen is -1000 mmol/gDW/h and the one of glucose is -10 mmol/gDW/h . The upper bounds of both are 1000 mmol/gDW/h . All the requirements can be fulfilled by using 445 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 441.

***K. pneumoniae* MGH 78578:** The requirements are that at least 99.9 % of the maximal growth rate under aerobic conditions (32.3593 h^{-1}) and under anaerobic conditions (16.8005 h^{-1}) can be realized by the subnetwork. The original network consists of 1223 non-blocked reactions. Per default the lower bounds of oxygen and glucose are both -1000 mmol/gDW/h . The upper bounds of both are 1000 mmol/gDW/h too. All the requirements can be fulfilled by using 338 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 203.

***Y. pestis* CO92:** The maximal biomass rate with glycine uptake is

0.2836 h⁻¹. Without glycine, the rate is 0.0886 h⁻¹. Per default the lower bound of oxygen is -20 mmol/gDW/h, -2 mmol/gDW/h of glucose and -2.2 mmol/gDW/h of glycine. The upper bounds of oxygen and glucose are 100000 mmol/gDW/h, the one for glycine is 1000 mmol/gDW/h. The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 1065 non-blocked reactions. All the requirements can be fulfilled by using 339 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 279.

***S. cerevisiae* S288c:** The maximal biomass rate with ethanol exchange is 0.2879 h⁻¹. Without ethanol, the rate is 0.2311 h⁻¹. Per default the lower bound of oxygen is -2 mmol/gDW/h, -10 mmol/gDW/h of glucose and 0 mmol/gDW/h of ethanol. The upper bounds of all three reactions are 9999999 mmol/gDW/h. The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 885 non-blocked reactions. All the requirements can be fulfilled by using 290 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 262.

***G. metallireducens* GS-15:** The maximal biomass rate with H₂O uptake is 5.8178 h⁻¹. Without H₂O, the rate is 2.2028 h⁻¹. Per default the lower bound of H₂O is -1000 mmol/gDW/h and of glucose 0. The upper bounds of both are 1000 mmol/gDW/h. The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 710 non-blocked reactions. All the requirements can be fulfilled by using 557 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 544.

***M. tuberculosis* iNJ661:** One requirement is that at least 99.9 % of the maximal growth rate (0.0525 h⁻¹) can be achieved. Additionally we defined 36 protected reactions. The original network consists of 1025 non-blocked reactions. Per default the lower bound of oxygen and glucose are both -1 mmol/gDW/h. The upper bounds of both are 999999 mmol/gDW/h. All the requirements can be fulfilled by using 427 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 314.

Protected reactions for the network ***M. tuberculosis* iNJ661:**

Succinyl-CoA synthetase (ADP-forming), Succinate dehydrogenase, Pyruvate decarboxylase, Pyruvate kinase, PPGKr, Propanoyl-CoA: succinate CoA-transferase, Phosphoenolpyruvate carboxykinase, Phosphoglucosmutase, Phosphoglycerate mutase, Phosphoglycerate kinase, Glucose-6-phosphate isomerase, Phosphofructokinase, PEPCK re, PDHbr, PDHa,

Pyruvate dehydrogenase, Pyruvate carboxylase, Malate dehydrogenase, Triose-phosphate isomerase, L-lactate dehydrogenase, Isocitrate dehydrogenase (NADP), Hexokinase (D-glucose:ATP), Glycogen synthase (ADPGlc), Glyceraldehyde-3-phosphate dehydrogenase, Fumarase, OXGDC, Enolase, Citrate synthase, Aldehyde dehydrogenase (acetaldehyde, NAD), FRD5, FRD, Fructose-bisphosphatase, Fructose-bisphosphate aldolase, Acetyl-CoA synthetase, Aconitase.

***B. subtilis* 168:** The maximal biomass rate with hydrogen uptake is 0.1247 h^{-1} . Without hydrogen, the rate is 0.1186 h^{-1} . Per default the lower bounds of oxygen and hydrogen are $-9999999 \text{ mmol/gDW/h}$ and of glucose -1.7 mmol/gDW/h . The upper bound of oxygen is 0 and the one of glucose and hydrogen are $9999999 \text{ mmol/gDW/h}$. The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 658 non-blocked reactions. All the requirements can be fulfilled by using 296 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 270.

***P. putida* KT2440:** One requirement is that at least 99.9 % of the maximal growth rate (1.4000 h^{-1}) can be achieved. Additionally we defined protected reactions to keep the TCA cycle. Per default the lower bounds of oxygen and glucose are both -20 mmol/gDW/h . The upper bounds are $9999999 \text{ mmol/gDW/h}$. The original network consists of 652 non-blocked reactions. All the requirements can be fulfilled by using 344 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 300. Protected reactions for the network ***P. putida* KT2440:**

Acetate kinase, Aconitase, Aconitase (half-reaction A, Citrate hydro-lyase), Aconitase (half-reaction B, Isocitrate hydro-lyase), Oxoglutarate dehydrogenase (lipoamide), Oxoglutarate dehydrogenase (dihydrolipoamide S-succinyltransferase), Citrate synthase.

***C. ljungdahlii* DSM 13528:** The maximal biomass rate with H_2O uptake is 0.2245 h^{-1} . Without H_2O , the rate is 0.2027 h^{-1} . Per default the lower bound of H_2O is -1000 mmol/gDW/h and of glucose 0. The upper bounds of both are 1000 mmol/gDW/h . The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 526 non-blocked reactions. All the requirements can be fulfilled by using 383 reactions. The number of essential reactions for

having at least 20% of the biomass reaction is 369.

***H. pylori* 26695:** One requirement is that at least 99.9 % of the maximal growth rate (20.2606 h^{-1}) can be achieved. Additionally we defined 28 protected reactions. Per default the lower bound of oxygen is -1000 mmol/gDW/h and the one of glucose is 0. The upper bounds of both are 1000 mmol/gDW/h.

Protected reactions for the network ***H. pylori* 26695:**

Aconitase, AKO, ATP synthase (four protons for one ATP), BC10 new, Citrate synthase, CYOO HP, Enolase, FADOX, Fructose-bisphosphate aldolase, Fructose-bisphosphatase, NAD(P)H-flavin oxidoreductase, FRD5, FRDO, Fumarase, Glyceraldehyde-3-phosphate dehydrogenase, Isocitrate dehydrogenase (NADP), LDH D1, Malate synthase, MDH4, NADPHQR, NDH 1, 2-oxoglutarate synthase (rev), PDH2, Glucose-6-phosphate isomerase, Phosphoglycerate kinase, Phosphoglycerate mutase, Phosphoenolpyruvate synthase, Triose-phosphate isomerase.

Reactions which were in all minimal subnetworks in ***H. pylori* 26695:**

1-deoxy-D-xylulose 5-phosphate synthase, 1-deoxy-D-xylulose reductoisomerase, 1-hydroxy-2-methyl-2-(E)-butenyl 4-diphosphate reductase (dmpp), 1-hydroxy-2-methyl-2-(E)-butenyl 4-diphosphate reductase (ipdp), 2,3-diketo-5-methylthio-1-phosphopentane degradation reaction, 2-C-methyl-D-erythritol 2,4-cyclodiphosphate synthase, 2-C-methyl-D-erythritol 4-phosphate cytidyltransferase, 2-dehydropantoate 2-reductase, 2-oxoglutarate decarboxylase, 2-oxoglutarate synthase (rev), 2-succinyl-6-hydroxy-2,4-cyclohexadiene 1-carboxylate synthase, 3,4-Dihydroxy-2-butanone-4-phosphate synthase, 3-dehydroquinone dehydratase, irreversible, 3-dehydroquinone synthase, 3-deoxy -D-manno-octulosonic -acid 8-phosphate synthase, 3-deoxy-D-arabino-heptulosonate 7-phosphate synthetase, 3-deoxy-manno-octulosonate-8-phosphatase, 3-methyl-2-oxobutanoate hydroxymethyltransferase, 3-phosphoshikimate 1-carboxyvinyltransferase, 4,5-dihydroxy-2,3-pentanedione cyclization (spontaneous), 4-(cytidine 5-diphospho)-2-C-methyl-D-erythritol kinase, 4-aminobenzoate synthase, 5,10-methylenetetrahydrofolate reductase (NADH), 5-Methylthio-5-deoxy-D-ribulose 1-phosphate dehydratase, 5-amino-6-(5-phosphoribosylamino)uracil reductase, 5-methylthioribose kinase, 5-methylthioribose-1-phosphate isomerase, 6-carboxyhexanoate-CoA ligase, 8-amino-7-oxononanoate synthase, ADP-D-glycero-D-mannoheptose epimerase, AKO, AMMQT6, ASPO2, ATP synthase (four protons for one ATP), Acetate exchange, Acetate kinase, Acetate re-

reversible transport via proton symport, Acetyl-CoA carboxylase, Aconitase, Adenine phosphoribosyltransferase, Adenosylmethionine decarboxylase, Adenosylmethionine-8-amino-7-oxononanoate transaminase, Adenylate kinase, Adenylosuccinate lyase, Adenylosuccinate synthase, Adenylsuccinate lyase, Agmatinase, Alanine racemase, Ammonia exchange, Ammonia reversible transport, Anthranilate phosphoribosyltransferase, Arabinose-5-phosphate isomerase, Arginase, Arginine decarboxylase, Asparagine synthetase, Aspartate 1-decarboxylase, Aspartate carbamoyltransferase, Aspartate kinase, Aspartate transaminase, Aspartate-semialdehyde dehydrogenase, BC10 new, BTS2, Beta-ketoacyl-ACP synthase (2), BiomassHP published, C140SN, C160SN, C180SN, C181SN, C190cSN, CLPNS HP, CO₂ exchange, CO₂ transporter via diffusion, CTP synthase NH3, CYOO HP, Carbamoyl-phosphate synthase (glutamine-hydrolysing), Chorismate mutase, Chorismate synthase, Citrate synthase, Cysteine synthase, Cytidylate kinase (CMP), D-alanine-D-alanine ligase (reversible), D-glycero-D-manno-hepouse 1-phosphate adenylyltransferase, D-glycero-D-manno-heptose 1,7-bisphosphate phosphatase, D-glycero-D-manno-heptose 7-phosphate kinase, DASYN HP, DHNAOT2, DHNPA, DHORD3, DHPS, DTMP kinase, Dephospho-CoA kinase, Dethiobiotin synthase, Diamino-hydroxyphosphoribosylaminopyrimidine deaminase, Diaminopimelate decarboxylase, Diaminopimelate epimerase, Dihydrodipicolinate reductase (NADPH), Dihydrodipicolinate synthase, Dihydrofolate reductase, Dihydrofolate synthase, Dihydroneopterin monophosphate dephosphorylase, Dihydroneopterin triphosphate pyrophosphatase, Dihydroorotase, Dimethylallyltranstransferase, EX pime LPAREN e RPAREN, Enolase, FADOX, FMN adenylyltransferase, FRD5, FRDO, Formate exchange, Formate transport via diffusion, Fructose-bisphosphatase, Fructose-bisphosphate aldolase, Fumarase, GDP-D-mannose dehydratase, GFUCS, GLYCTO1, GMP synthase, GTP cyclohydrolase I, GTP cyclohydrolase II, Geranyltranstransferase, Glucosamine-1-phosphate N-acetyltransferase, Glucose-6-phosphate isomerase, Glutamate dehydrogenase (NADP), Glutamate racemase, Glutamine phosphoribosyldiphosphate amidotransferase, Glutamine synthetase, Glutamine-fructose-6-phosphate transaminase, Glyceraldehyde-3-phosphate dehydrogenase, Glycerol kinase, Glycine hydroxymethyltransferase, reversible, Glycolaldehyde dehydrogenase, Guanylate kinase (GMP:ATP), H⁺ exchange, HCO₃ equilibration reaction, Heme transport via ABC system, Homoserine O trans acetylase, Homoserine dehydrogenase (NADPH), Homoserine kinase, IMP cyclohydrolase, IMP dehydrogenase, Indole-3-glycerol-phosphate synthase, Inorganic diphosphatase, Isochorismate synthase, Isocitrate dehydrogenase (NADP), KAS HP, KAS HP2, KDOCT, L arganine reversible transport via proton symport, L lactate reversible transport via proton symport, L-Arginine exchange, L-Histidine exchange, L-Isoleucine exchange, L-Lactate exchange, L-Leucine exchange, L-Methionine exchange, L-Valine exchange,

L-glutamate 5-semialdehyde dehydratase (spontaneous), L-histidine transport via ABC system, L-isoleucine transport via ABC system, L-leucine transport via ABC system, L-methionine transport via ABC system, L-valine transport via ABC system, LDH D1, LPADSS HP, LPSSYN HP, LacR, MDH4, MECDPDH, MOAT HP, Malate synthase, Malic enzyme (NAD), Malonyl-CoA-ACP transacylase, Mannose-1-phosphate guanylyltransferase (GDP), Mannose-6-phosphate isomerase, Metb1, Methenyltetrahydrofolate cyclohydrolase, Methionine adenosyltransferase, Methylenetetrahydrofolate dehydrogenase (NADP), Methylthioadenosine nucleosidase, NAD kinase, NAD synthase (nh3), NAD(P)H-flavin oxidoreductase, NADPHQR, NDH 1, Naphthoate synthase, Nicotinate-nucleotide adenylyltransferase, Nicotinate-nucleotide diphosphorylase (carboxylating), Nucleoside-diphosphate kinase (ATP:CDP), Nucleoside-diphosphate kinase (ATP:GDP), Nucleoside-diphosphate kinase (ATP:UDP), Nucleoside-diphosphate kinase (ATP:dADP), Nucleoside-diphosphate kinase (ATP:dCDP), Nucleoside-diphosphate kinase (ATP:dGDP), Nucleoside-diphosphate kinase (ATP:dTDP), O-succinylbenzoate-CoA ligase, O-succinylbenzoate-CoA synthase, O-succinylhomoserine lyase (L-cysteine), O2 exchange, O2 transport diffusion, Octaprenyl pyrophosphate synthase, Ornithine transaminase, Orotate phosphoribosyltransferase, Orotidine-5-phosphate decarboxylase, PABB, PASYN HP, PDH2, PGPP HP, PGSA HP, PIMEtr, PPTGS, PSD HP, PSSA HP, Pantetheine-phosphate adenylyltransferase, Pantothenate kinase, Pantothenate synthase, Phenylalanine transaminase, Phosphate exchange, Phosphate reversible transport via symport, Phospho-N-acetylmuramoyl-pentapeptide-transferase (meso-2,6-diaminopimelate), Phosphoenolpyruvate synthase, Phosphoglucomutase, Phosphoglucosamine mutase, Phosphoglycerate dehydrogenase, Phosphoglycerate kinase, Phosphoglycerate mutase, Phosphomannomutase, Phosphopantothenate-cysteine ligase, Phosphopantothenoylcysteine decarboxylase, Phosphoribosylaminoimidazole carboxylase, Phosphoribosylaminoimidazole carboxylase (mutase rxn), Phosphoribosylaminoimidazole synthase, Phosphoribosylaminoimidazolecarboxamide formyltransferase, Phosphoribosylaminoimidazolesuccinocarboxamide synthase, Phosphoribosylanthranilate isomerase (irreversible), Phosphoribosylformylglycinamide synthase, Phosphoribosylglycinamide formyltransferase, Phosphoribosylglycinamide synthase, Phosphoribosylpyrophosphate synthetase, Phosphoserine phosphatase (L-serine), Phosphoserine transaminase, Phosphotransacylase, Prephenate dehydratase, Prephenate dehydrogenase, Protoheme exchange, Pyrimidine phosphatase, Pyrroline-5-carboxylate reductase, Quinolinate synthase, RFA HP, RFAC HP, Riboflavin kinase, Riboflavin synthase, Ribonucleoside-diphosphate reductase (ADP), Ribonucleoside-diphosphate reductase (CDP), Ribonucleoside-diphosphate reductase (GDP), Ribonucleoside-diphosphate reductase (UDP), Ribose-5-phosphate isomerase, Ribulose 5-phosphate 3-epimerase,

S-adenosylhomocysteine nucleosidase, S-ribosylhomocysteine cleavage enzyme, SHSL2r, Sedoheptulose 7-phosphate isomerase, Serine O-acetyltransferase, Shikimate dehydrogenase, Shikimate kinase, Sink ahcys(c), Sink needed to allow 4-hydroxy-5-methyl-3(2H)-furanone to leave system, Sink needed to allow S-Adenosyl-4-methylthio-2-oxobutanoate to leave system, Spermidine synthase, Succinate exchange, Succinate transport via proton symport, Succinyl-diaminopimelate desuccinylase, Succinyl-diaminopimelate transaminase, Superoxide dismutase, TMDSf, Tetrahydrodipicolinate succinylase, Thiamin exchange, Thiamine transport via ABC system, Thioredoxin reductase (NADPH), Threonine synthase, Transaldolase, Transketolase, Triose-phosphate isomerase, Tryptophan synthase (indoleglycerol phosphate), Tyrosine transaminase, U23GAAT HP, U2GAAT, U2GAAT2, UAGAAT HP, UAGDP2, UDP-N-acetylenolpyruvoylglucosamine reductase, UDP-N-acetylglucosamine 1-carboxyvinyltransferase, UDP-N-acetylglucosamine diphosphorylase, UDP-N-acetylglucosamine-N-acetylmuramyl-(pentapeptide)pyrophosphoryl-undecaprenol N-acetylglucosamine t..., UDP-N-acetylmuramoyl-L-alanine synthetase, UDP-N-acetylmuramoyl-L-alanyl-D-glutamate synthetase, UDP-N-acetylmuramoyl-L-alanyl-D-glutamyl-meso-2,6-diaminopimelate synthetase, UDP-N-acetylmuramoyl-L-alanyl-D-glutamyl-meso-2,6-diaminopimeloyl-D-alanyl-D-alanine synthetase, UDPglucose 4-epimerase, UHGADA HP, UMP kinase, UNK2, USHD HP, UTP-glucose-1-phosphate uridylyltransferase (irreversible), Undecaprenyl-diphosphatase, Urea exchange, Urea transport via facilitate diffusion, Uridylate kinase (dUMP), Valine transaminase.

M. barkeri str. Fusaro: The maximal biomass rate with ammonia uptake is 0.0268 h^{-1} . Without ammonia, the rate is 0.0095 h^{-1} . Per default the lower bound of ammonia is $-9999999 \text{ mmol/gDW/h}$ and of glucose 0. The upper bounds of both are $9999999 \text{ mmol/gDW/h}$. The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 484 non-blocked reactions. All the requirements can be fulfilled by using 364 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 289.

S. aureus N315: The maximal biomass rate with glucose uptake is 8.0759 h^{-1} . Without glucose, the rate is 4.8154 h^{-1} . The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 465 non-blocked reactions. Per default the lower bounds of oxygen and glucose are -1000 mmol/gDW/h . The upper bounds of both are 1000 mmol/gDW/h . All the requirements can be fulfilled by using 122 reactions. The number of essential reactions for

having at least 20% of the biomass reaction is 71.

T. maritima MSB8: The maximal biomass rate with H₂O uptake is 0.2284 h⁻¹. Without H₂O, the rate is 0.1043 h⁻¹. Per default the lower bound of H₂O is -1000 mmol/gDW/h and of glucose 0. The upper bounds of both are 1000 mmol/gDW/h. The requirements are that at least 99.9 % of both rates can be realized by the subnetwork. The original network consists of 385 non-blocked reactions. All the requirements can be fulfilled by using 282 reactions. The number of essential reactions for having at least 20% of the biomass reaction is 267.

10.2 Appendix for Computing iMCSs using the dual approach

We provide here some computation results. We used the method provided by [Tobalina et al., 2016].

10.2.1 Time: Computations using Tobalina's tool

Following we show results we obtained computing a given number of MCSs using the script provided by [Tobalina et al., 2016]. We computed iMCSs using the original and the projected flux cone for all networks from the BiGG Models Database [King et al., 2016] which include a biomass reaction (which was the target reaction). In each table the results for several networks can be found.

network id: The id of the network on the BiGG Models Database. **Nr iMCSs:** computed number of iMCSs. **time original cone:** time (in seconds) needed to compute the given number of iMCSs in the original cone. **time projected cone:** time (in seconds) needed to compute the given number of iMCSs in the projected cone. **relative time:** relative time needed to compute the given number of iMCSs in the original cone compared to the time needed using the projected cone.

| network id | Nr. iMCSs | time original cone | time pro- jected cone | relative time |
|------------|-----------|--------------------------|--------------------------------|------------------|
| iNF517 | 10 | 3.72 | 1.89 | 1.9 |
| iNF517 | 100 | 90.85 | 25.76 | 3.5 |
| iNF517 | 300 | 220.63 | 307.87 | 0.7 |
| iHN637 | 10 | 3.42 | 1.94 | 1.7 |
| iHN637 | 100 | 75.19 | 18.77 | 4 |
| iHN637 | 300 | 105.65 | 147.45 | 0.7 |
| iJB785 | 10 | 4.6 | 2.34 | 1.9 |
| iJB785 | 100 | 129.78 | 23.68 | 5.4 |
| iJB785 | 300 | 171.15 | 79.89 | 2.1 |
| iJN678 | 10 | 5.83 | 3.56 | 1.6 |
| iJN678 | 100 | 108.87 | 62.63 | 1.7 |
| iJN678 | 300 | 116.25 | 139.61 | 0.8 |
| iNJ661 | 10 | 6.93 | 2.45 | 2.8 |
| iNJ661 | 100 | 151.87 | 27.85 | 5.4 |
| iNJ661 | 300 | 199.37 | 176.55 | 1.1 |
| iJN746 | 10 | 4.36 | 1.94 | 2.2 |
| iJN746 | 100 | 110.47 | 27.18 | 4.0 |
| iJN746 | 300 | 220.34 | 127.33 | 1.7 |
| iJR904 | 10 | 5.42 | 2.13 | 2.5 |
| iJR904 | 100 | 163.67 | 44.59 | 3.6 |
| iJR904 | 300 | 413.21 | 3470.9 | 0.1 |
| iYO844 | 10 | 6.29 | 2.04 | 3 |
| iYO844 | 100 | 140.5 | 23.82 | 5.8 |
| iYO844 | 300 | 341.45 | 831.65 | 0.4 |
| iND750 | 10 | 4.39 | 1.98 | 2.2 |
| iND750 | 100 | 119.11 | 26.78 | 4.4 |
| iND750 | 300 | 333.39 | 422.98 | 0.7 |
| iAF987 | 10 | 7.72 | 3.75 | 2 |
| iAF987 | 100 | 185.17 | 65.18 | 2.8 |
| iAF987 | 300 | 232.78 | 158.11 | 1.4 |
| iMM904 | 10 | 7.96 | 3.41 | 2.3 |
| iMM904 | 100 | 202 | 58.47 | 3.4 |
| iMM904 | 300 | 904.74 | 7451.6 | 0.1 |
| iPC815 | 10 | 10.81 | 3.51 | 3 |
| iPC815 | 100 | 239.87 | 35.79 | 6.7 |
| iPC815 | 300 | 561.4 | 799.45 | 0.7 |
| iYL1228 | 10 | 8.09 | 4.07 | 1.9 |
| iYL1228 | 100 | 304.93 | 45 | 6.7 |
| iYL1228 | 300 | 721.87 | 671.75 | 1 |
| iAF1260 | 10 | 15.7 | 14.96 | 1 |
| iAF1260 | 100 | 267.36 | 213.12 | 1.2 |
| iAF1260 | 300 | 858.29 | 698.43 | 1.2 |

Table 10.1

| network id | Nr. iMCSs | time original cone | time pro- jected cone | relative time |
|----------------|-----------|--------------------------|--------------------------------|------------------|
| iAF1260b | 10 | 14.37 | 14.12 | 1 |
| iAF1260b | 100 | 299.49 | 210.84 | 1.4 |
| iAF1260b | 300 | 858.73 | 714.9 | 1.2 |
| iSDY_1059 | 10 | 10.03 | 12.41 | 0.8 |
| iSDY_1059 | 100 | 275.76 | 114.79 | 2.4 |
| iSDY_1059 | 300 | 956.48 | 844.14 | 1.1 |
| STM_v1_0 | 10 | 10.61 | 15.35 | 0.6 |
| STM_v1_0 | 100 | 259.59 | 166.67 | 1.5 |
| STM_v1_0 | 300 | 522.03 | 651.11 | 0.8 |
| iJO1366 | 10 | 11.96 | 15.52 | 0.7 |
| iJO1366 | 100 | 298.89 | 266.76 | 1.1 |
| iJO1366 | 300 | 639.19 | 737.32 | 0.8 |
| iSbBS512_1146 | 10 | 11.71 | 13.36 | 0.8 |
| iSbBS512_1146 | 100 | 259.87 | 125.93 | 2 |
| iSbBS512_1146 | 300 | 931.68 | 802.77 | 1.1 |
| iSBO_1134 | 10 | 17.92 | 13.8 | 1.2 |
| iSBO_1134 | 100 | 438.95 | 169.67 | 2.5 |
| iSBO_1134 | 300 | 793.02 | 901.53 | 0.8 |
| iS_1188 | 10 | 17.04 | 11.41 | 1.4 |
| iS_1188 | 100 | 396.48 | 105.59 | 3.7 |
| iS_1188 | 300 | 898.43 | 612.89 | 1.4 |
| iSFV_1184 | 10 | 13.74 | 11.12 | 1.2 |
| iSFV_1184 | 100 | 267.1 | 97.04 | 2.7 |
| iSFV_1184 | 300 | 813.46 | 543.42 | 1.4 |
| iSF_1195 | 10 | 17.16 | 10.72 | 1.6 |
| iSF_1195 | 100 | 327.34 | 95.33 | 3.4 |
| iSF_1195 | 300 | 900.21 | 621.7 | 1.4 |
| iSFxv_1172 | 10 | 15.59 | 12.97 | 1.2 |
| iSFxv_1172 | 100 | 295.51 | 105.29 | 2.8 |
| iSFxv_1172 | 300 | 912.71 | 615.1 | 1.4 |
| iSSON_1240 | 10 | 11.37 | 11.82 | 0.9 |
| iSSON_1240 | 100 | 286.05 | 123.57 | 2.3 |
| iSSON_1240 | 300 | 945.38 | 741.46 | 1.2 |
| iECH74115_1262 | 10 | 11.96 | 13.44 | 0.8 |
| iECH74115_1262 | 100 | 286.11 | 122.8 | 2.3 |
| iECH74115_1262 | 300 | 975.04 | 709.68 | 1.3 |
| iE2348C_1286 | 10 | 15.51 | 15.11 | 1 |
| iE2348C_1286 | 100 | 350.26 | 121.83 | 2.8 |
| iE2348C_1286 | 300 | 943.23 | 686.27 | 1.3 |
| iG2583_1286 | 10 | 11.29 | 14.3 | 0.7 |
| iG2583_1286 | 100 | 285.99 | 117.54 | 2.4 |
| iG2583_1286 | 300 | 970.75 | 625.94 | 1.5 |

Table 10.2

| network id | Nr. iMCSs | time original cone | time pro- jected cone | relative time |
|---------------|-----------|--------------------------|--------------------------------|------------------|
| iECED1_1282 | 10 | 12.65 | 14.53 | 0.8 |
| iECED1_1282 | 100 | 302.32 | 125.97 | 2.3 |
| iECED1_1282 | 300 | 949.33 | 706.32 | 1.3 |
| iECSP_1301 | 10 | 11.58 | 13.83 | 0.8 |
| iECSP_1301 | 100 | 299.57 | 111.53 | 2.6 |
| iECSP_1301 | 300 | 967.46 | 673.95 | 1.4 |
| iML1515 | 10 | 12.46 | 17.67 | 0.7 |
| iML1515 | 100 | 306.07 | 249.36 | 1.2 |
| iML1515 | 300 | 1106.3 | 1499.3 | 0.7 |
| iEC042_1314 | 10 | 12.14 | 14.89 | 0.8 |
| iEC042_1314 | 100 | 313.36 | 126.75 | 2.4 |
| iEC042_1314 | 300 | 929.27 | 720.44 | 1.2 |
| iECNA114_1301 | 10 | 12.39 | 14.8 | 0.8 |
| iECNA114_1301 | 100 | 253.7 | 130.09 | 1.9 |
| iECNA114_1301 | 300 | 1050.3 | 785.74 | 1.3 |
| iECs_1301 | 10 | 14.12 | 14.9 | 0.9 |
| iECs_1301 | 100 | 227.55 | 131.57 | 1.7 |
| iECs_1301 | 300 | 1006.4 | 752.61 | 1.3 |
| iECIAI39_1322 | 10 | 11.29 | 10.57 | 1 |
| iECIAI39_1322 | 100 | 255.92 | 90.32 | 2.8 |
| iECIAI39_1322 | 300 | 998.69 | 637.59 | 1.5 |
| iZ_1308 | 10 | 13.05 | 14.88 | 0.8 |
| iZ_1308 | 100 | 342.24 | 131.84 | 2.5 |
| iZ_1308 | 300 | 1025.8 | 752.7 | 1.3 |
| iUTI89_1310 | 10 | 11.58 | 15.7 | 0.7 |
| iUTI89_1310 | 100 | 346.11 | 131.61 | 2.6 |
| iUTI89_1310 | 300 | 921.2 | 697.96 | 1.3 |
| ic_1306 | 10 | 13.18 | 15.47 | 0.8 |
| ic_1306 | 100 | 313.83 | 205.57 | 1.5 |
| ic_1306 | 300 | 1006.9 | 885.66 | 1.1 |
| iLF82_1304 | 10 | 11.44 | 14.63 | 0.7 |
| iLF82_1304 | 100 | 281.53 | 130.9 | 2.1 |
| iLF82_1304 | 300 | 1002.2 | 771.52 | 1.2 |
| iECOK1_1307 | 10 | 14.19 | 14.74 | 0.9 |
| iECOK1_1307 | 100 | 330.72 | 157.33 | 2.1 |
| iECOK1_1307 | 300 | 1033.7 | 733.25 | 1.4 |
| iECS88_1305 | 10 | 11.85 | 14.6 | 0.8 |
| iECS88_1305 | 100 | 260.83 | 146.71 | 1.7 |
| iECS88_1305 | 300 | 988.1 | 683.37 | 1.4 |

Table 10.3

| network id | Nr. iMCSs | time original cone | time pro- jected cone | relative time |
|---------------|-----------|--------------------------|--------------------------------|------------------|
| iECABU_c1320 | 10 | 12.79 | 14.91 | 0.8 |
| iECABU_c1320 | 100 | 307.85 | 137.38 | 2.2 |
| iECABU_c1320 | 300 | 1017.9 | 692.52 | 1.4 |
| iAPECO1_1312 | 10 | 11.5 | 14.72 | 0.7 |
| iAPECO1_1312 | 100 | 284.07 | 138.23 | 2 |
| iAPECO1_1312 | 300 | 1003.5 | 756.72 | 1.3 |
| iNRG857_1313 | 10 | 12.7 | 15.82 | 0.8 |
| iNRG857_1313 | 100 | 296.37 | 151.53 | 1.9 |
| iNRG857_1313 | 300 | 1118.8 | 774.48 | 1.4 |
| iUMN146_1321 | 10 | 11.66 | 15.69 | 0.7 |
| iUMN146_1321 | 100 | 336.86 | 129.51 | 2.6 |
| iUMN146_1321 | 300 | 1066.7 | 745.19 | 1.4 |
| iECP_1309 | 10 | 11.46 | 14.93 | 0.7 |
| iECP_1309 | 100 | 234.68 | 130.25 | 1.8 |
| iECP_1309 | 300 | 964.16 | 701.24 | 1.3 |
| iECUMN_1333 | 10 | 15.08 | 14.09 | 1 |
| iECUMN_1333 | 100 | 223.53 | 111.38 | 2 |
| iECUMN_1333 | 300 | 852.58 | 668.38 | 1.2 |
| iB21_1397 | 10 | 14.96 | 13.06 | 1.1 |
| iB21_1397 | 100 | 309.4 | 115.8 | 2.6 |
| iB21_1397 | 300 | 1095.2 | 750.11 | 1.4 |
| iBWG_1329 | 10 | 15.93 | 14.54 | 1 |
| iBWG_1329 | 100 | 356.67 | 213.48 | 1.6 |
| iBWG_1329 | 300 | 815.63 | 693.38 | 1.1 |
| iECD_1391 | 10 | 13.53 | 13.06 | 1 |
| iECD_1391 | 100 | 384.91 | 115.81 | 3.3 |
| iECD_1391 | 300 | 1021.9 | 750.14 | 1.3 |
| iECDH10B_1368 | 10 | 14.89 | 16.13 | 0.9 |
| iECDH10B_1368 | 100 | 368.16 | 234.13 | 1.5 |
| iECDH10B_1368 | 300 | 772.36 | 689.5 | 1.1 |
| iECSF_1327 | 10 | 21.86 | 13.3 | 1.6 |
| iECSF_1327 | 100 | 368.12 | 218.4 | 1.6 |
| iECSF_1327 | 300 | 803.02 | 678.55 | 1.1 |
| iEcSMS35_1347 | 10 | 15.47 | 15.05 | 1 |
| iEcSMS35_1347 | 100 | 329.95 | 133.37 | 2.4 |
| iEcSMS35_1347 | 300 | 1036.5 | 739.45 | 1.4 |
| iECB_1328 | 10 | 13.78 | 13.97 | 0.9 |
| iECB_1328 | 100 | 284.36 | 120.41 | 2.3 |
| iECB_1328 | 300 | 967.93 | 687.05 | 1.4 |

Table 10.4

| network id | Nr. iMCSs | time original cone | time pro- jected cone | relative time |
|-------------------|-----------|--------------------------|--------------------------------|------------------|
| iECBD_1354 | 10 | 15.49 | 14.62 | 1 |
| iECBD_1354 | 100 | 280.94 | 112.19 | 2.5 |
| iECBD_1354 | 300 | 1047.5 | 740.2 | 1.4 |
| iEcDH1_1363 | 10 | 16.27 | 14.21 | 1.1 |
| iEcDH1_1363 | 100 | 231.95 | 124.05 | 1.8 |
| iEcDH1_1363 | 300 | 974.78 | 725.05 | 1.3 |
| iEcHS_1320 | 10 | 11.52 | 15.17 | 0.7 |
| iEcHS_1320 | 100 | 191.56 | 118.66 | 1.6 |
| iEcHS_1320 | 300 | 923.79 | 695.49 | 1.3 |
| iECDH1ME8569_1439 | 10 | 15.38 | 14.62 | 1 |
| iECDH1ME8569_1439 | 100 | 258.29 | 117.93 | 2.1 |
| iECDH1ME8569_1439 | 300 | 961.29 | 695.8 | 1.3 |
| iEC55989_1330 | 10 | 13.36 | 14.14 | 0.9 |
| iEC55989_1330 | 100 | 285.86 | 130.3 | 2.1 |
| iEC55989_1330 | 300 | 983.52 | 704.04 | 1.3 |
| iETEC_1333 | 10 | 12.14 | 14.92 | 0.8 |
| iETEC_1333 | 100 | 281.72 | 210.17 | 1.3 |
| iETEC_1333 | 300 | 920.7 | 922.43 | 0.9 |
| iECO103_1326 | 10 | 14.32 | 13.97 | 1 |
| iECO103_1326 | 100 | 313.92 | 132.72 | 2.3 |
| iECO103_1326 | 300 | 1014.7 | 733.64 | 1.3 |
| iY75_1357 | 10 | 15.39 | 14.3 | 1 |
| iY75_1357 | 100 | 308.99 | 116.25 | 2.6 |
| iY75_1357 | 300 | 970.79 | 694.79 | 1.3 |
| iECO111_1330 | 10 | 16.34 | 15.48 | 1 |
| iECO111_1330 | 100 | 339.57 | 133.69 | 2.5 |
| iECO111_1330 | 300 | 1031.2 | 671.57 | 1.5 |
| iEcE24377_1341 | 10 | 14.06 | 15.6 | 0.9 |
| iEcE24377_1341 | 100 | 315.44 | 137.55 | 2.2 |
| iEcE24377_1341 | 300 | 1003.5 | 735.81 | 1.3 |
| iECIAI1_1343 | 10 | 11.22 | 16.41 | 0.6 |
| iECIAI1_1343 | 100 | 257.7 | 117.42 | 2.1 |
| iECIAI1_1343 | 300 | 646.43 | 625.98 | 1 |
| iEcolC_1368 | 10 | 11.62 | 14.25 | 0.8 |
| iEcolC_1368 | 100 | 269.26 | 118.57 | 2.2 |
| iEcolC_1368 | 300 | 880.3 | 706.04 | 1.2 |
| iECSE_1348 | 10 | 11.54 | 13.34 | 0.8 |
| iECSE_1348 | 100 | 276.51 | 126.37 | 2.1 |
| iECSE_1348 | 300 | 992.89 | 716.74 | 1.3 |

Table 10.5

| network id | Nr. iMCSs | time original cone | time pro- jected cone | relative time |
|---------------|-----------|--------------------------|--------------------------------|------------------|
| iUMNK88_1353 | 10 | 15.06 | 15.56 | 0.9 |
| iUMNK88_1353 | 100 | 238.71 | 125.53 | 1.9 |
| iUMNK88_1353 | 300 | 1027.6 | 700.89 | 1.4 |
| iEKO11_1354 | 10 | 16.9 | 15.28 | 1.1 |
| iEKO11_1354 | 100 | 224.72 | 125.88 | 1.7 |
| iEKO11_1354 | 300 | 1036.8 | 735.07 | 1.4 |
| iECO26_1355 | 10 | 13.07 | 14.05 | 0.9 |
| iECO26_1355 | 100 | 270.47 | 122.56 | 2.2 |
| iECO26_1355 | 300 | 991.09 | 720.59 | 1.3 |
| iECW_1372 | 10 | 11.87 | 12.3 | 0.9 |
| iECW_1372 | 100 | 305.59 | 119.5 | 2.5 |
| iECW_1372 | 300 | 971.51 | 707.72 | 1.3 |
| iWFL_1372 | 10 | 13.16 | 12.28 | 1 |
| iWFL_1372 | 100 | 289.42 | 119.56 | 2.4 |
| iWFL_1372 | 300 | 983.33 | 706.67 | 1.3 |
| iMM1415 | 10 | 38.01 | 21.59 | 1.7 |
| iMM1415 | 100 | 981.9 | 359.02 | 2.7 |
| iMM1415 | 300 | 11005 | 54895 | 0.2 |
| iLB1027_lipid | 10 | 79.44 | 49.92 | 1.5 |
| iLB1027_lipid | 100 | 1706.1 | 906.78 | 1.8 |
| iLB1027_lipid | 300 | 4068.9 | 4266.8 | 0.9 |
| iCHOv1 | 10 | 96.76 | 30.58 | 3.1 |
| iCHOv1 | 100 | 2337.7 | 467.6 | 4.9 |
| iCHOv1 | 300 | 26197 | 22744 | 1.1 |

10.2.2 Cardinality: Computations using Tobalina's tool

Following we show results we obtained computing a given number of MCSs using the script provided by [Tobalina et al., 2016]. For computing MCSs we used the original flux cone and for computing iMCSs we used the projected flux cone for all networks from the **BiGG Models Database** [King et al., 2016] which include a biomass reaction (which was the target reaction). In each table the results for several networks can be found.

network id: The id of the network on the **BiGG Models Database**. **Card:** cardinality of the MCSs. **Nr MCSs:** number of MCSs of the corresponding cardinality computed. For the first 4 networks we computed 1000 MCSs for the last two we computed 500. **Nr iMCSs:** Number of MCSs of the corresponding cardinality computed. **Difference:** The difference between MCSs and iMCSs of a certain cardinality. For the first network we computed 1000 MCSs. If we computed more, we probably computed also MCSs of cardinality 6 or higher.

| network id | Card | Nr MCSs | Nr iMCSs | Difference |
|------------|------|---------|----------|------------|
| iHN637 | 1 | 716 | 540 | 176 |
| | 2 | 100 | 78 | 22 |
| | 3 | 53 | 12 | 41 |
| | 4 | 41 | 14 | 27 |
| | 5 | 0 | 0 | 0 |
| | 6 | 0 | 26 | -26 |
| | 7 | 0 | 4 | -4 |
| | 8 | 0 | 98 | -98 |
| | 9 | 0 | 49 | -49 |
| | 10 | 0 | 87 | -87 |
| iJB785 | 1 | 864 | 749 | 115 |
| | 2 | 46 | 70 | -24 |
| | 3 | 0 | 21 | -21 |
| | 4 | 0 | 38 | -38 |
| | 5 | 0 | 32 | -32 |
| iJN678 | 1 | 861 | 749 | 112 |
| | 2 | 49 | 43 | 6 |
| | 3 | 0 | 4 | -4 |
| | 4 | 0 | 14 | -14 |
| | 5 | 0 | 29 | -29 |
| | 6 | 0 | 71 | -71 |
| iNJ661 | 1 | 724 | 590 | 134 |
| | 2 | 83 | 42 | 41 |
| | 3 | 103 | 14 | 89 |
| | 4 | 0 | 8 | -8 |
| | 5 | 0 | 46 | -46 |
| | 6 | 0 | 96 | -96 |
| | 7 | 0 | 76 | -76 |
| | 8 | 0 | 38 | -38 |
| iJN746 | 1 | 584 | 434 | 150 |
| | 2 | 194 | 54 | 140 |
| | 3 | 132 | 46 | 86 |
| | 4 | 0 | 222 | -222 |
| | 5 | 0 | 153 | -153 |
| iJR904 | 1 | 510 | 246 | 264 |
| | 2 | 128 | 21 | 107 |
| | 3 | 272 | 28 | 244 |
| | 4 | 0 | 27 | -27 |
| | 5 | 0 | 44 | -44 |
| | 6 | 0 | 44 | -44 |
| iYO844 | 1 | 318 | 231 | 87 |
| | 2 | 92 | 50 | 42 |
| | 3 | 0 | 18 | -18 |
| | 4 | 0 | 40 | -40 |
| | 5 | 0 | 71 | -71 |
| iND750 | 1 | 289 | 215 | 74 |
| | 2 | 121 | 48 | 73 |
| | 3 | 0 | 40 | -40 |
| | 4 | 0 | 103 | -103 |
| | 5 | 0 | 4 | -4 |

| network id | Card | Nr MCSs | Nr iMCSs | Difference |
|---------------|------|---------|----------|------------|
| iAF987 | 1 | 410 | 402 | 8 |
| | 2 | 0 | 8 | -8 |
| iMM904 | 1 | 278 | 219 | 59 |
| | 2 | 132 | 48 | 84 |
| | 3 | 0 | 29 | -29 |
| | 4 | 0 | 55 | -55 |
| | 5 | 0 | 59 | -59 |
| iPC815 | 1 | 317 | 247 | 70 |
| | 2 | 93 | 63 | 30 |
| | 3 | 0 | 100 | -100 |
| iYL1228 | 1 | 313 | 246 | 67 |
| | 2 | 97 | 63 | 34 |
| | 3 | 0 | 75 | -75 |
| | 4 | 0 | 26 | -26 |
| iAF1260 | 1 | 389 | 303 | 86 |
| | 2 | 21 | 77 | -56 |
| | 3 | 0 | 30 | -30 |
| iAF1260b | 1 | 389 | 303 | 86 |
| | 2 | 21 | 74 | -53 |
| | 3 | 0 | 33 | -33 |
| iSDY_1059 | 1 | 368 | 288 | 80 |
| | 2 | 42 | 37 | 5 |
| | 3 | 0 | 85 | -85 |
| STM_v1.0 | 1 | 395 | 324 | 71 |
| | 2 | 15 | 79 | -64 |
| | 3 | 0 | 7 | -7 |
| iJO1366 | 1 | 410 | 358 | 52 |
| | 2 | 0 | 51 | -51 |
| iSbBS512_1146 | 1 | 377 | 292 | 85 |
| | 2 | 33 | 45 | -12 |
| | 3 | 0 | 73 | -73 |
| iSBO_1134 | 1 | 379 | 292 | 87 |
| | 2 | 31 | 68 | -37 |
| | 3 | 0 | 49 | -49 |
| iS_1188 | 1 | 371 | 288 | 83 |
| | 2 | 39 | 61 | -22 |
| | 3 | 0 | 61 | -61 |
| iSFV_1184 | 1 | 382 | 296 | 86 |
| | 2 | 28 | 62 | -34 |
| | 3 | 0 | 52 | -52 |
| iSF_1195 | 1 | 369 | 284 | 85 |
| | 2 | 41 | 62 | -21 |
| | 3 | 0 | 63 | -63 |
| iSFxv_1172 | 1 | 373 | 287 | 86 |
| | 2 | 37 | 60 | -23 |
| | 3 | 0 | 63 | -63 |
| iSSON_1240 | 1 | 370 | 285 | 85 |
| | 2 | 40 | 64 | -24 |
| | 3 | 0 | 61 | -61 |

Table 10.7

| network id | Card | Nr MCSs | Nr iMCSs | Difference |
|----------------|------|---------|----------|------------|
| iECH74115_1262 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 65 | -30 |
| | 3 | 0 | 50 | -50 |
| iE2348C_1286 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 63 | -28 |
| | 3 | 0 | 52 | -52 |
| iG2583_1286 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECED1_1282 | 1 | 378 | 295 | 83 |
| | 2 | 32 | 62 | -30 |
| | 3 | 0 | 53 | -53 |
| iECSP_1301 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iML1515 | 1 | 356 | 282 | 74 |
| | 2 | 54 | 93 | -39 |
| | 3 | 0 | 34 | -34 |
| iEC042_1314 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 60 | -25 |
| | 3 | 0 | 56 | -56 |
| iECNA114_1301 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECs_1301 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECIAI39_1322 | 1 | 358 | 277 | 81 |
| | 2 | 52 | 82 | -30 |
| | 3 | 0 | 51 | -51 |
| iZ_1308 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iUTI89_1310 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 54 | -54 |
| | 4 | 0 | 1 | -1 |
| ic_1306 | 1 | 370 | 290 | 80 |
| | 2 | 40 | 61 | -21 |
| | 3 | 0 | 59 | -59 |
| iLF82_1304 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 62 | -27 |
| | 3 | 0 | 54 | -54 |
| iECOK1_1307 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 62 | -27 |
| | 3 | 0 | 54 | -54 |
| iECS88_1305 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECABU_c1320 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 60 | -25 |
| | 3 | 0 | 56 | -56 |

Table 10.8

| network id | Card | Nr MCSs | Nr iMCSs | Difference |
|-------------------|------|---------|----------|------------|
| iAPECO1_1312 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iNRG857_1313 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iUMN146_1321 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 60 | -25 |
| | 3 | 0 | 56 | -56 |
| iECP_1309 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 60 | -25 |
| | 3 | 0 | 56 | -56 |
| iECUMN_1333 | 1 | 377 | 294 | 83 |
| | 2 | 33 | 62 | -29 |
| | 3 | 0 | 54 | -54 |
| iB21_1397 | 1 | 379 | 298 | 81 |
| | 2 | 31 | 41 | -10 |
| | 3 | 0 | 71 | -71 |
| iBWG_1329 | 1 | 410 | 353 | 57 |
| | 2 | 0 | 57 | -57 |
| | 3 | 0 | 71 | -71 |
| iECD_1391 | 1 | 379 | 298 | 81 |
| | 2 | 31 | 41 | -10 |
| | 3 | 0 | 71 | -71 |
| iECDH10B_1368 | 1 | 410 | 347 | 63 |
| | 2 | 0 | 62 | -62 |
| | 3 | 0 | 71 | -71 |
| iECSF_1327 | 1 | 410 | 353 | 57 |
| | 2 | 0 | 57 | -57 |
| | 3 | 0 | 71 | -71 |
| iEcSMS35_1347 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 60 | -25 |
| | 3 | 0 | 56 | -56 |
| iECB_1328 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 63 | -28 |
| | 3 | 0 | 53 | -53 |
| iECBD_1354 | 1 | 379 | 298 | 81 |
| | 2 | 31 | 41 | -10 |
| | 3 | 0 | 71 | -71 |
| iEcDH1_1363 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iEcHS_1320 | 1 | 375 | 293 | 82 |
| | 2 | 35 | 62 | -27 |
| | 3 | 0 | 55 | -55 |
| iECDH1ME8569_1439 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iEC55989_1330 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 54 | -54 |

Table 10.9

| network id | Card | Nr MCSs | Nr iMCSs | Difference |
|----------------|------|---------|----------|------------|
| iETEC_1333 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 62 | -27 |
| | 3 | 0 | 54 | -54 |
| iECO103_1326 | 1 | 376 | 295 | 81 |
| | 2 | 34 | 61 | -27 |
| | 3 | 0 | 54 | -54 |
| iY75_1357 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECO111_1330 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 54 | -54 |
| iEcE24377_1341 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 62 | -27 |
| | 3 | 0 | 54 | -54 |
| iECIAI1_1343 | 1 | 377 | 296 | 81 |
| | 2 | 33 | 64 | -31 |
| | 3 | 0 | 49 | -49 |
| iEcolC_1368 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECSE_1348 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 62 | -27 |
| | 3 | 0 | 53 | -53 |
| iUMNK88_1353 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 54 | -54 |
| iEKO11_1354 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECO26_1355 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 55 | -55 |
| iECW_1372 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 54 | -54 |
| iWFL_1372 | 1 | 375 | 294 | 81 |
| | 2 | 35 | 61 | -26 |
| | 3 | 0 | 54 | -54 |
| iMM1415 | 1 | 208 | 124 | 84 |
| | 2 | 157 | 89 | 68 |
| | 3 | 44 | 35 | 9 |
| | 4 | 1 | 87 | -86 |
| | 5 | 0 | 75 | -75 |
| iLB1027_lipid | 1 | 410 | 354 | 56 |
| | 2 | 0 | 56 | -56 |
| iCHOv1 | 1 | 130 | 76 | 54 |
| | 2 | 208 | 114 | 94 |
| | 3 | 69 | 70 | -1 |
| | 4 | 3 | 54 | -51 |
| | 5 | 0 | 94 | -94 |

Table 10.10

Zusammenfassung

Systembiologie ist ein interdisziplinäres Feld, welches Mathematik, Informatik und Ingenieurwissenschaften vereint, um biologische Prozesse zu analysieren. Aufgrund erfolgreicher Anwendungen in der Medizin und Biotechnologie wurde dieses Feld in den letzten zwei Jahrzehnten immer wichtiger. Ziel ist es, biologische Systeme mathematisch zu beschreiben und somit die benötigte Zeit und Kosten der Forschung in Laboren zu reduzieren. Um dies zu realisieren, werden Algorithmen und Techniken entworfen, die eine breite Anwendung finden. Durch moderne DNA-Sequenzierung können Daten generiert werden, die biologische Entitäten beschreiben. Hierdurch wird immer mehr Wissen gewonnen, welches zur Rekonstruktion von metabolischen Netzwerken beitragen kann. Nachteil dieser neuen Techniken ist, dass die so gesammelten Daten oft zu groß sind um sie von Hand zu analysieren. Daher braucht es automatisierte Methoden, die relevante Informationen extrahieren.

In dieser Arbeit stellen wir verschiedene Algorithmen vor, die gegebene metabolische Netzwerke sinnvoll reduzieren. Wir präsentieren einen Algorithmus, der von einem gegebenen Netzwerk minimale Teilnetzwerke berechnet, in denen weiterhin vordefinierte Prozesse stattfinden können. Zudem entwickeln wir eine Methode, um eine minimale Teilmenge der elementaren Flussmodi zu berechnen, die das Netzwerk vollständig erzeugen und beschreiben können. Die Größe dieser Teilmengen ist im Vergleich zu der Gesamtmenge der elementaren Flussmodi signifikant reduziert. Darüber hinaus stellen wir ein Verfahren vor, welches die Anzahl der Variablen eines gegebenen Problems reduziert, um somit (bereits existierende) Algorithmen zu beschleunigen. Desweiteren entwickeln wir eine neue Methode die *minimal cut sets* in einem projizierten Netzwerk berechnet, wodurch wir *minimal cut sets* von größerer Kardinalität als zuvor berechnen können. Die Projektion von metabolischen Netzwerken führt auch zu anderen Anwendungen, wie zum Beispiel die Berechnung von *minimal metabolic behaviours*.

Schwerpunkt diese Arbeit ist die Entwicklung mathematischer Methoden, deren Anwendungsbereich metabolische Netzwerke sind. Um diese Modelle zu erstellen und zu validieren verwenden wir lineare ganzzahlige Optimierung, polyedrische Kegel, lineare Algebra und orientierte Matroide.

Curriculum vitae

| | |
|--------------|---|
| Name: | Annika Röhl |
| Experiences: | Oct. 2013 - August 2018 Teaching assistant Freie Universität, Berlin. Teaching assistant and research in the field optimization and bioinformatics. Jan. 2011–Sept. 2013 E-Learning author and assistant Digital Spirit, Berlin. Realization, generation and revision of learning content and generation of E-Learning programs. May 2008–March 2010 Student assistant Georg-August-Universität, Göttingen. 2009 and 2010 Employee SourceTalk Tage, Göttingen. |
| Education | Oct. 2013 - August 2018 PhD student in Mathematics Freie Universität, Berlin. Supervisor: Prof. Dr. Alexander Bockmayr. Oct. 2004–July 2010 Diploma Georg-August-Universität Göttingen |
| Publications | EFM-Recorder - Faster Elementary Mode Enumeration via Reaction Coupling Order Proceedings on the Strasbourg Spring School on Advances in Systems and Synthetic Biology 2015, Strasbourg, France A mixed-integer linear programming approach to the reduction of genome-scale metabolic networks BMC Bioinformatics, 2017, 18:2 Reaction Splitting and Minimum Sets of Elementary Flux Modes Proceedings on the Lyon Spring School on Advances in Systems and Synthetic Biology 2017, Lyon, France |

| | |
|--------------------|---|
| Conference Talks | <p>EFM-Recorder - Faster Elementary Mode Enumeration via Reaction Coupling Order Strasbourg Spring School on Advances in Systems and Synthetic Biology 2015, Strasbourg, France</p> <p>Reaction Splitting and Minimum Sets of Elementary Flux Modes Proceedings on the Lyon Spring School on Advances in Systems and Synthetic Biology 2017, Lyon, France</p> |
| Invited Talks | <p>A mixed-integer linear programming approach to the reduction of genome-scale metabolic networks Group of Marie-France Sagot, Université Claude Bernard (Lyon 1), November 2016, Lyon, France</p> <p>Reduction of Networks: Optimization and Projection Group of Daniel Segrè, Boston University, December 2017, Boston, USA</p> <p>Reduction of Networks: Optimization and Projection, Group of Albert-László Barabási, Center for Complex Network Research, December 2017, Boston, USA</p> |
| Conference Posters | <p>Elementary Mode Enumeration via Flux Coupling ALGO 2014, Wrocław, Poland</p> <p>Fast Minimal Network Finder GCB 2016, Berlin, Germany</p> <p>Fast Minimal Network Finder ISGSB 2016, Jena, Germany</p> |

Selbständigkeitserklärung

Hiermit versichere ich, dass ich alle Hilfsmittel und Hilfen angegeben und auf dieser Grundlage die Arbeit selbständig verfasst habe. Die Arbeit ist nicht schon einmal in einem früheren Promotionsverfahren eingereicht worden.

Berlin, den 22. Juni, 2018

Annika Röhl

Acknowledgement

Der erste und größte Dank geht an meinen Betreuer Alexander Bockmayr, der mir diese Arbeit erst ermöglicht hat und mich all die Jahre unterstützt hat. Er hat mir alle Freiheiten gelassen die ich brauchte und mich mit seiner unendlichen Geduld in meinem Vorhaben unterstützt. Ohne Druck und mit sehr viel Expertise hat er mich bei dieser Reise begleitet, ganz unter dem Motto *der Weg ist das Ziel*.

Mein zweiter Dank geht an meinen Zweitgutachter, Jürgen Zanghellini. Jürgens Arbeiten sind für mich immer wieder eine Inspiration. Zweitgutachter zu sein erfordert viel Zeit und ich bin ihm sehr dankbar dafür, dass er sich diese nimmt.

Danken möchte ich auch Knut Reinert mit dem die Zusammenarbeit immer sehr entspannt war. Das Gleiche gilt für Heike Siebert, die immer ein offenes Ohr für mich hatte. Ein weiterer Dank geht an Marie-France Sagot. Der Austausch mit ihr hat mich sehr motiviert und angespornt.

Meine Arbeit wäre ohne meine ArbeitkollegInnen nicht möglich gewesen. Deswegen geht ein ganz herzlicher Dank an die Gruppen *Mathematics in Life Sciences* und *Discrete Biomathematics*. Vielen Dank an Adam, Aljoscha, Elisa, Firdevs, Hannes, Katinka, Kirsten, Lin, Ling, Manon, Markus, Melania, Neveen, Robert, Tanguy, Therese und natürlich Katja, das Herz unserer Gruppe! Ganz besonders möchte ich Yaron Goldstein danken, der mir meinen Start unglaublich erleichtert hat und auch danach immer für ein Gespräch zu haben war. Mit Alexandra und Arne Reimers habe ich wohl am meisten über meine Dissertation diskutiert. Ohne *die Reimers* wäre meine Arbeit nicht so lebhaft und mit weniger Lachen verbunden gewesen und uns verbindet mittlerweile eine wunderbare Freundschaft. Ein ganz besonderer Dank geht an Richard, der nicht nur mir so manche Mittagspause versüßt hat!

Außerdem möchte ich meinen Freunden und meiner Familie danken. Allen voran Oma Martha, ohne die ich nicht da wäre, wo ich jetzt bin und natürlich

meinem Bruder Benedikt, ohne den diese Arbeit genauso aussehen würde wie jetzt auch. Auch möchte ich meiner Mutter danken, dafür dass sie jeden Tag erfolgreich kämpft. Für die wunderbaren Ablenkungen möchte ich Ben, Ireen, Matze und Steffen danken.

Ganz besonders danke ich Arne. Er ist immer für mich da und hat mich mit seiner Geduld und unbeirrbarer Zuversicht durch so manches Tief manövriert. Seit über zehn Jahren ist er der tollste und wichtigste Mensch in meinem Leben.

Widmen möchte ich diese Arbeit aber einer anderen Person. Es hätte sich wahrscheinlich niemand so sehr darüber gefreut, dass ich ihr diese Dissertation widme, wie sie selbst. Es macht mich unendlich glücklich, dass ich Jenny kennenlernen durfte und umso trauriger, dass ihr Leben viel zu kurz war. Jenny, ich vermisse dich jeden Tag und deswegen widme ich dir diese Arbeit.