

FREIE UNIVERSITÄT BERLIN
FACHBEREICH MATHEMATIK UND INFORMATIK

PHD THESIS

**Metabolic Networks, Thermodynamic
Constraints, and Matroid Theory**

Arne Cornelis REIMERS

June 3, 2014

1. Gutachter:

Prof. Dr. Alexander BOCKMAYR

2. Gutachter:

Prof. Dr. Leen STOUGIE

Tag der Disputation:

14. August 2014

This thesis is dedicated to my dear friend
and future wife Alexandra, who fills my
life with joy and happiness every day.

Abstract

Background

Biological experiments are time-consuming and expensive. Hence, computer-aided experimental design is used more and more often to select those experiments that are likely to be successful. Constraint-based analysis of metabolic networks is such a method. It is used in bio-engineering for the production of bio-fuel and other valuable compounds and has been applied in medical research ranging from the virtual liver project to cancer-research.

Constraint-based methods do not predict a single outcome of experiments, but just constrain the space of possible outcomes. This way it is sufficient to only include confirmed information, which is a crucial aspect in systems biology, since many parameters are still unknown today.

The most famous constraint-based method is *flux balance analysis* (FBA). It is based on the steady-state assumption, which enforces that the reaction rates (flux vector) $v \in \mathbb{R}^{\mathcal{R}}$ must not over or under produce metabolites. Optimality criteria and additional constraints are used to exclude other physiologically unrealistic behaviors. For example, thermodynamic constraints incorporate energetic aspects.

Even with all these constraints, the flux space of metabolic networks is usually highly underconstrained. Hence, methods like *flux variability analysis* (FVA), *flux coupling analysis* (FCA) and *elementary flux modes* (EFM) have been developed. While it is possible to efficiently perform FVA and FCA on genome-scale networks, EFM enumeration is only suitable for small networks due to combinatorial explosion. On the other hand, FVA and FCA capture only specific aspects of the flux spaces, while the set of EFMs gives a comprehensive characterization.

Tightly connected to the problem of EFM enumeration is the problem of enumerating vertices of polyhedra. While it has been shown that there exists no total polynomial time algorithm (unless $\mathbf{P} = \mathbf{NP}$) to enumerate the vertices of (unbounded) polyhedra (which corresponds to enumeration of EFMs that satisfy an optimality criterion), the complexity for enumerating vertices of bounded polyhedra (which corresponds to the enumeration of all EFMs for flux spaces that are polyhedral cones) is still an open question.

Results

In this thesis, I focus on different applications of thermodynamic constraints, like the inference of metabolite concentrations, flux coupling analysis and the computation of flux modules. Since thermodynamic constraints are also not computationally trivial, I also investigate the computational complexity of various forms of decision problems related to thermodynamic constraints which lead me to theoretical obstacles for sampling methods. In many cases matroid theory turned out to be a powerful tool.

Computational complexity and sampling

I start with extending existing results from my master thesis on the characterization of thermodynamically feasible flux spaces and on the computational complexity of flux optimization problems. It turns out that many flavors of thermodynamically constrained flux optimization problems are NP-hard. I conclude that it is not possible to sample thermodynamically constrained flux spaces without losing important features. In an empirical study, I even show that without thermodynamic constraints flux through some reactions is never sampled, although non-zero flux is possible.

However, flux optimization with thermodynamic constraints can usually be solved efficiently with methods from constraint mixed integer programming. Using structural properties, like flux coupling analysis, I refine methods from my master thesis for flux optimization.

Optimization on metabolite concentrations

Thermodynamic constraints can not only be used to constrain the flux space, but they can also be used to infer information on possible metabolite concentrations. Hence, I also investigate optimization problems on metabolite concentrations/potentials. In particular, I present an optimization method for mixed integer linear programs with strict inequalities, since the space of feasible metabolite concentrations can only be described properly with strict inequalities.

Flux modules

Flux modules are one of the main applications of matroid theory in this thesis. They have originally been introduced by Kelk et al. as a structural property that can be found in the yield-optimal flux spaces of many genome-scale metabolic networks and used to compress the set of optimal yield EFMs considerably. However, their computation method requires the initial computation of all optimal yield EFMs. Although the number of optimal-yield EFMs is much smaller than the total number of EFMs it still is a major bottleneck of their method.

By introducing a mathematical definition of flux module I derive a method for finding flux modules that does not anymore require the computation of optimal-yield EFMs, but that instead requires the application of thermodynamically constrained FVA. Further research on the topic revealed that flux modules are closely linked to separators from matroid theory. Using this connection, I develop an algorithm using methods from matroid theory that is orders of magnitude faster than the previous methods.

These results can be extended to generalizations of flux modules, so called k -modules. The corresponding concept in matroid theory are k -separators. Closely connected to k -separators is the concept of branch-width in matroid theory, which turns out to be a valuable structural property for metabolic networks, too. In particular, I show in this thesis, how the EFMs can be enumerated in total polynomial time if the branch-width of the underlying matroid is bounded by a constant and the flux space is a polyhedral cone.

Sublinear growth of *Chlamydomonas reinhardtii*

When *Chlamydomonas reinhardtii* is grown in a photo bioreactor, its total growth rate decreases with increasing cell density. Cyanobacteria on the other hand, do not show this behavior. I show that this effect can be modeled in FBA models by introducing flux forcing reactions, i.e., reactions with positive lower flux bound or negative upper flux bound. I show that candidate reactions (sets) can be found by solving bilevel optimization problems.

Here again, thermodynamic constraints have to be used to investigate the flux forcing behavior of reactions contained in internal cycles. To solve these special bilevel programs (due to the thermodynamic constraints the inner problem is non-convex), I develop a global optimization method based on parametrized mixed integer linear programming.

Flux coupling analysis with thermodynamic constraints

Thermodynamic constraints can also be added to FCA to detect more couplings. Together with Yaron Goldstein, I developed a framework for running FCA on any kind of qualitative model, in particular FCA with thermodynamic constraints, efficiently. Using traditional FCA as a pre-processing step, we can now solve thermodynamically constrained FCA also in a few seconds.

Conclusion

With flux modules and algorithmic approaches to also include complex constraints I present methods in this thesis that simplify and speed-up the analysis of metabolic networks. This allows us to gain biological insights faster and develop better methods for the production of biofuels in bio-engineering and cancer therapies in medicine.

Zusammenfassung

Hintergrund

Biologische Experimente sind zeitraubend und teuer. Deshalb werden Computer immer häufiger benutzt um solche Experimente zu bestimmen, die am ehesten erfolgreich sind und zu neuen Erkenntnissen führen. Constraint-basierte Analyse von metabolischen Netzwerken ist eine solche Methode. Sie wird für die Produktion von Biotreibstoffen und wertvollen Nebenprodukten benutzt, aber auch in der medizinischen Forschung um die Funktionsweise der Leber und Krebs zu verstehen.

Constraint-basierte Methoden sagen nicht nur ein bestimmtes Ergebnis eines Experiments voraus, sondern beschränken lediglich den Lösungsraum. Dabei wird biologisches Wissen benutzt um unmögliche Ergebnisse auszuschließen. Somit sind diese Methoden auch dann anwendbar, wenn nur wenige Daten vorhanden sind. Dies ist ein wesentlicher Aspekt der Systembiologie, da heutzutage noch viele Parameter unbekannt sind.

In der *Fluss Balance Analyse* (FBA), wird z.B. angenommen, dass jeder Stoff genauso schnell produziert wie konsumiert wird (steady-state). Optimalitätskriterien und zusätzliche Nebenbedingungen werden benutzt um weitere physiologisch unrealistische Verhaltensweisen auszuschließen. Ein Beispiel dafür sind thermodynamische Nebenbedingungen um energetische Aspekte zu integrieren.

Aber auch mit all diesen Nebenbedingungen bleibt der Lösungsraum in der Regel hochgradig unterbestimmt. Deswegen wurden Flussvariabilitätsanalyse (FVA), Flusskoppelungsanalyse (FCA) und elementare Flussmoden (EFM) entwickelt. Während es möglich ist FVA und FCA auch auf großen, genomweiten, Netzwerken auszuführen, funktioniert EFM-Enumeration nur auf kleinen Netzwerken aufgrund von kombinatorischer Explosion. Im Gegensatz liefern FVA und FCA nur begrenzte Informationen, wohingegen die EFMs eine vollständige Charakterisierung des Lösungsraumes ermöglichen.

Eng verwandt zur EFM-Enumeration ist das Problem der Eckenenumeration von Polyedern. Es wurde gezeigt, dass die Aufzählung der Ecken von unbeschränkten Polyedern (welche Flussräumen mit Optimalitätskriterium entsprechen) nicht in polynomieller Zeit (in Eingabe und Ausgabe) möglich ist, außer wenn $\mathbf{P} = \mathbf{NP}$ gilt. Für beschränkte Polyeder ist diese Frage allerdings noch offen.

Ergebnisse

In dieser Arbeit betrachte ich unterschiedliche Anwendungen von thermodynamischen Nebenbedingungen. Diese reichen von der Vorhersage von Metabolitkonzentrationen über Flusskopplungsanalyse zur Berechnung von Flussmodulen. Zudem zeige ich, dass die Integration von thermodynamischen Nebenbedingungen aus Komplexitätstheoretischen Gesichtspunkten nicht trivial ist, indem ich unterschiedlichste Formulierungen und Szenarien analysiere. Dies wiederum lässt mich auf Hindernisse für Samplingmethoden schließen. Dabei erweist sich Matroidtheorie oft als hochnützlich Werkzeug.

Komplexität und Sampling

Ich beginne mit der Erweiterung der Ergebnisse aus meiner Masterarbeit zu der Charakterisierung des thermodynamisch gültigen Flussraumes und den Komplexitätstheoretischen Ergebnissen zu Flussoptimierungsproblemen. Ich zeige, dass viele Formen des thermodynamisch beschränkten Flussoptimierungsproblems NP-schwer sind. Angewendet auf Samplingmethoden heißt dies, dass es unmöglich ist den thermodynamisch gültigen Flussraum zu sampeln ohne wichtige Eigenschaften zu verlieren.

In einer empirischen Studie zeige ich sogar, dass selbst ohne thermodynamischen Nebenbedingungen kein Fluss durch manche Reaktionen berechnet wird, obwohl die deterministische FVA das Gegenteil beweist.

Optimierung und Metabolitkonzentrationen

Thermodynamische Nebenbedingungen beschränken nicht nur den Flussraum, sondern sie können auch benutzt werden um Informationen über mögliche Konzentrationen von Stoffen in der Zelle zu erfahren. Daher betrachte ich auch das Optimierungsproblem über Metabolitkonzentrationen bzw. -potentialen. Insbesondere zeige ich, wie gemischt-ganzahlige lineare Programme mit strikten Ungleichungen gelöst werden können, da der Raum der gültigen Metabolitkonzentrationen nicht abgeschlossen ist.

Flussmodule

Eines der zentralen Anwendungsgebiete von Matroidtheorie in dieser Arbeit sind Flussmodule. Diese wurden von Kelk et al. als eine strukturelle Eigenschaft der optimalen Flüsse erkannt, mit der der Flussraum wesentlich vereinfacht werden kann. Sie beobachteten, dass damit das Problem der EFM Enumeration wesentlich effizienter gelöst werden kann. Allerdings benötigt deren Methode zuerst die Berechnung gerade jener EFMs.

Mittels meiner mathematischen Definition von Flussmodulen leite ich eine Methode her, die die Flussmodule ohne EFM Enumeration berechnen kann. Allerdings benötigt diese

Methode die Anwendung von thermodynamisch beschränkter FVA. Weitere Forschung ergab, dass Flussmodule eng verwandt zu Separatoren aus der Matroidtheorie sind. Mittels diese Erkenntnis präsentiere ich nun einen Algorithmus mit dem ich Flussmodule von großen Netzwerken in wenigen Sekunden, also Größenordnungen schneller als die vorherigen Methoden, berechnen kann.

Diese Ergebnisse verallgemeinere ich dann für sogenannte k -module. Ich zeige, dass k -module wiederum eng verwandt mit k -Separatoren aus der Matroidtheorie sind und führe die branch-weite des zugrunde liegenden Matroids als ein Komplexitätsmaß für metabolische Netzwerke ein. Insbesondere zeige ich, dass EFMs in total polynomieller Zeit enumeriert werden können, wenn die branch-weite des zugrunde liegenden Matroids durch eine Konstante beschränkt ist und der Flussraum ein Kegel ist.

Sublineares Wachstum von *Chlamydomonas reinhardtii*

Wenn die eukariotische Grünalge *Chlamydomonas reinhardtii* in einem Photo-Bioreaktor wächst, dann schrumpft die Gesamtwachstumsrate mit wachsender Zelldichte. Cyanobakterien dagegen zeigen dieses Verhalten nicht. Ich zeige, dass dieses Verhalten mittels FBA und flusserzwingender Reaktionen modelliert werden kann. Dazu berechne ich potentielle Reaktionskandidaten mittels bi-level Optimierung.

Auch hier benötige ich thermodynamische Nebenbedingungen um auch Reaktionskandidaten zu untersuchen, die in inneren Kreisen des metabolischen Netzwerkes enthalten sind. Um diese bi-level Optimierungsprobleme zu lösen, präsentiere ich eine Methode basierend auf parametrisierter gemischt-ganzzahliger Programmierung, die auch mit nicht-konvexen inneren Problemen umgehen kann.

Flusskopplungsanalyse mit thermodynamischen Nebenbedingungen

Thermodynamische Nebenbedingungen können auch in der FCA angewendet werden um mehr Abhängigkeiten zu finden. Zusammen mit Yaron Goldstein habe ich ein Framework entwickelt mit dem wir FCA auf beliebigen qualitativen Modellen effizient ausführen können. Damit können wir nun auch FCA mit thermodynamischen Nebenbedingungen in wenigen Sekunden berechnen.

Fazit

Mittels Flussmodulen und algorithmischen Ansätzen um auch komplizierte Nebenbedingungen zu integrieren, zeige ich in dieser Arbeit Methoden auf, die die Analyse metabolischer Netzwerke vereinfachen und beschleunigen. Dadurch können biologische Erkenntnisse schneller gewonnen werden und bessere Methoden in der Biotechnologie zur Herstellung von Biotreibstoffen und in der Medizin für Krebstherapien entwickelt werden.

Acknowledgments

First of all I want to thank my supervisor Alexander Bockmayr for giving me the opportunity to work in the highly fascinating intersection of mathematics, computer science and biology. With thermodynamic constraints, he found a topic that led me to rich mathematical tools and thus allowed me to do highly interesting mathematical research. Also, I want to thank him for his great help in the research-auxiliary areas, like the presentation of research results in publications and the selection of suitable conferences.

Another important role in my research played Leen Stougie, whom I met the first time on the ENUMEX summer school in 2012 on enumeration methods and exact algorithms in computational biology. There, he presented the flux modules problem, which then became the most fascinating part of my research. He invited me twice to Amsterdam, with many fruitful discussions and results that can now be found in this thesis.

All this research would also have not been possible without the academic environment and the financial support of the Berlin Mathematical School, who provided me with a PhD stipend, and the International Max Planck Research School on Computational Biology and Scientific Computing (IMPRS-CBSC). Moreover, they did not give me only money, but there I also met many interesting people, some of which I now call my friends.

Similarly important for my research environment were my colleagues in the research groups *Mathematics in Life Sciences* and *Discrete Biomathematics*. Here, I want to mention in particular Yaron Goldstein with whom I had many interesting and productive discussions. But, I also want to thank Shahrad, Hannes, Aljoscha, Ling, Heike, Adam, Kirsten, Laszlo, Alexandra, Firdevs, Annika and Marco who all contributed to a friendly research environment. Many thanks go to Isabel Beckenbach, who pointed me to the concept of branch width, which turned out to be highly relevant.

I also want to thank my parents and my brother Eike for supporting me in my path and for many interesting mathematical discussions. In particular I want to thank Eike for his help on Lemma 6.2.6. Without his algebraic knowledge I probably wouldn't have found the connection of modules to matroid theory so quickly.

Last but not least I want to thank my wonderful Alexandra for rigorously proof-reading this thesis.

Berlin, June 3, 2014
Arne Reimers

Contents

| | |
|--|-----------|
| Abstract | i |
| Zusammenfassung | v |
| Acknowledgments | ix |
| 1 Introduction | 1 |
| 1.1 Metabolic Networks | 2 |
| 1.2 Steady-State Assumption | 4 |
| 1.3 Thermodynamic Constraints | 5 |
| 1.4 Constraint-Based Analysis | 6 |
| 1.4.1 Does the organism grow under certain conditions? | 6 |
| 1.4.2 How does the organism grow? | 7 |
| 1.4.3 How can a growth behavior be enforced? | 8 |
| 1.5 Mathematical Context | 8 |
| 1.6 Structure of this Thesis | 9 |
| 2 Feasible Pathways | 11 |
| 2.1 Basic Concepts and Notation | 11 |
| 2.1.1 Directed Reactions and Equivalence | 13 |
| 2.1.2 Flux Coupling | 15 |
| 2.2 Steady-State Assumption | 16 |
| 2.2.1 Average Fluxes are Steady-State | 16 |
| 2.2.2 Concentrations for Average Fluxes | 17 |
| 2.3 Polyhedral Flux Spaces | 19 |

| | | |
|----------|---|-----------|
| 2.4 | Elementary Modes | 19 |
| 2.5 | Matroid Theory | 21 |
| 2.5.1 | Notation and Definition | 21 |
| 2.5.2 | Matroids from Metabolic Networks | 23 |
| 2.5.3 | Matroid Connectivity | 25 |
| 2.6 | Thermodynamic Constraints | 26 |
| 2.6.1 | Chemical Potentials | 28 |
| 2.6.2 | Without Bounds on Chemical Potentials | 28 |
| 2.6.3 | With Bounds on Chemical Potentials | 29 |
| 3 | Computational Complexity | 37 |
| 3.1 | Introduction | 37 |
| 3.2 | Results | 39 |
| 3.3 | Reductions | 39 |
| 3.3.1 | Trivial Reductions | 39 |
| 3.3.2 | $1x2y3a4a$ is Equivalent to $1x2y3b4a$ for $x \in \{a, b\}, y \in \{b, c\}$. . . | 40 |
| 3.4 | Problems in P | 42 |
| 3.4.1 | $1a2c3a4a$ is in P | 42 |
| 3.4.2 | $1b2x3a4a$ is in P | 42 |
| 3.4.3 | $1b2b3b4a$ is in P | 42 |
| 3.4.4 | $1x2y3c4a$ is in P | 43 |
| 3.4.5 | $1x2c3b4a$ is in P | 43 |
| 3.5 | NP-hard problems | 43 |
| 3.5.1 | $1a2a3a4x$ is NP-hard | 43 |
| 3.5.2 | $1x2a3b4a$ and $1x2y3b4b$ are NP-hard | 44 |
| 3.5.3 | $1x2y3c4b$ and $1x2y3a4b$ are NP-hard | 48 |
| 3.6 | Problems with Unknown Complexity | 49 |
| 4 | Flux Optimization | 51 |
| 4.1 | Flux Balance Analysis | 51 |
| 4.2 | Thermodynamic Constraints | 52 |
| 4.3 | Mixed Integer Linear Programming | 53 |

| | | |
|----------|---|------------|
| 4.4 | Implicit Representation of the Potential Space & Constraint Programming | 55 |
| 4.4.1 | Practical Implementation | 56 |
| 4.4.2 | Coupled Reactions and Generalized Infeasible Sets | 59 |
| 4.4.3 | Partitioning the Flux Space | 63 |
| 4.5 | Heuristics | 63 |
| 4.5.1 | Cycle Subtraction | 64 |
| 4.5.2 | First Directions, then Fluxes | 65 |
| 4.6 | Conclusion | 68 |
| 5 | Potential Optimization | 69 |
| 5.1 | Strict Inequalities | 69 |
| 5.1.1 | Strict Inequalities in Linear Programming | 70 |
| 5.1.2 | Strict Inequalities in Mixed Integer Linear Programming | 71 |
| 5.1.3 | Integration into MILP-solver | 73 |
| 5.1.4 | Application to Thermodynamic Constraints in Metabolic Networks | 75 |
| 5.2 | Improving Bounds on Potentials | 77 |
| 5.2.1 | Introduction: the Graphic Case | 78 |
| 5.2.2 | The Flow Condition | 82 |
| 5.2.3 | Updating Bounds | 87 |
| 5.2.4 | Application on a Genome-Scale Network of <i>E. coli</i> | 97 |
| 5.2.5 | Dependence of Uncertainties in Equilibrium Constants | 100 |
| 6 | Modules in Metabolic Networks | 105 |
| 6.1 | Definition | 106 |
| 6.2 | Properties of (Linear) k -Modules | 106 |
| 6.2.1 | Restriction to Linear Vector Spaces | 107 |
| 6.2.2 | Matroid Theory for k -Modules | 112 |
| 6.2.3 | Finding k -Modules | 114 |
| 6.3 | Flux modules (0-modules) | 115 |
| 6.3.1 | Uniqueness of the Decomposition | 119 |
| 6.3.2 | Decomposition Theorem for EFMs | 121 |
| 6.3.3 | Finding Flux Modules | 125 |

| | | |
|----------|---|------------|
| 6.3.4 | Comparison of the Methods | 133 |
| 6.3.5 | Computed Modules in Genome-Scale Metabolic Networks | 136 |
| 6.3.6 | Visualizing the Interplay of Flux Modules | 139 |
| 6.3.7 | Conclusion | 144 |
| 6.4 | Decomposition Theorem for Linear 1-Modules | 145 |
| 6.4.1 | Computation of Linear 1-Modules | 149 |
| 6.4.2 | Linear 1-Modules in Practice | 150 |
| 6.5 | Decomposition Theorem for k -Modules | 153 |
| 6.5.1 | Elementary Flux Mode Enumeration and Vertex Enumeration | 154 |
| 6.5.2 | Branch-Width | 155 |
| 6.5.3 | Decomposition Theorem for Vertex Enumeration | 157 |
| 6.6 | Conclusion | 163 |
| 7 | Sublinear Growth & Flux Forcing Reactions | 165 |
| 7.1 | Motivation | 166 |
| 7.2 | Flux-Forcing Reactions can Explain the Effect | 170 |
| 7.3 | Sets of Diverting Reactions | 174 |
| 7.3.1 | Diverting Set of Type 1 | 174 |
| 7.3.2 | Diverting Set of Type 2 | 175 |
| 7.3.3 | Finding Flux Forcing Coefficients for Diverting Sets of Type 2 | 178 |
| 7.3.4 | Thermodynamically Constrained P | 180 |
| 7.4 | Multi-Parametric Thermodynamically Constrained Flux Balance Analysis (mpTFBA) | 180 |
| 7.4.1 | The Algorithm by Dua and Pistikopoulos | 181 |
| 7.4.2 | Adaptation to Thermodynamically Constrained FBA | 182 |
| 7.4.3 | Finding Cut Sets for Blocking Internal Cycles | 186 |
| 7.5 | Tight Integration of Bilevel Programming into Parametric Programming | 189 |
| 7.5.1 | Min-Max Problems | 189 |
| 7.6 | Implementation | 189 |
| 7.7 | Results | 190 |
| 7.8 | Discussion | 192 |

| | | |
|----------|--|------------|
| 8 | Flux Coupling Analysis with Thermodynamic Constraints | 197 |
| 8.1 | Introduction | 198 |
| 8.2 | Lattice Theory | 200 |
| 8.2.1 | FCA for arbitrary P | 200 |
| 8.2.2 | Algorithm for FCA in P | 202 |
| 8.3 | FCA in T | 202 |
| 8.4 | Implementation | 204 |
| 8.5 | Minimal Representation of Coupling Data | 205 |
| 8.5.1 | Minimal Extensions | 206 |
| 8.5.2 | Computation of Minimal Extensions | 206 |
| 8.6 | Discussion | 213 |
| 8.6.1 | Theoretical Differences | 213 |
| 8.6.2 | Practical Differences | 213 |
| 8.7 | Conclusion | 216 |
| 9 | Sampling the Thermodynamically Constrained Flux Space | 217 |
| 9.1 | Background | 217 |
| 9.2 | Theoretical Obstructions to Sampling | 218 |
| 9.3 | Practical Obstructions to Sampling | 222 |
| 9.3.1 | Method | 222 |
| 9.3.2 | Results | 224 |
| 9.4 | Discussion | 225 |
| 9.5 | Conclusion | 226 |
| A | Computational Results on Flux Modules | 229 |
| A.1 | Flux Modules | 229 |
| A.1.1 | Summary on <i>E. coli</i> iJR904 grown on L-tryptophan | 229 |
| A.1.2 | Summary on <i>E. coli</i> iJR904 grown on L-Threonine | 230 |
| A.1.3 | Summary on <i>E. coli</i> iJR904 grown on glucose | 231 |
| A.1.4 | Summary on <i>E. coli</i> iJR904 grown on L-Arginine | 232 |
| A.1.5 | Summary on <i>E. coli</i> iJR904 grown on citrate | 233 |
| A.1.6 | Summary on <i>E. coli</i> iJR904 grown on fumarate | 234 |

| | | |
|----------|---|------------|
| A.1.7 | Summary on <i>E. coli</i> iJR904 grown on L-glutamine | 235 |
| A.1.8 | Summary on <i>E. coli</i> iJR904 grown on Lactose | 236 |
| A.1.9 | Summary on <i>E. coli</i> iJR904 grown on L-malate | 237 |
| A.1.10 | Summary on <i>E. coli</i> iAF1260 grown on glucose, aerobic | 238 |
| A.1.11 | Summary on <i>E. coli</i> iAF1260 grown on glucose, anaerobic | 239 |
| A.1.12 | Summary on <i>E. coli</i> iAF1260 grown on L-Threonine | 240 |
| A.1.13 | Summary on <i>H. pylori</i> iIT341 | 241 |
| A.1.14 | Summary on <i>M. barkeri</i> iAF692 grown on methanol | 242 |
| A.1.15 | Summary on <i>M. tuberculosis</i> iNJ661 | 243 |
| A.1.16 | Summary on <i>S. aureus</i> iSB619 | 245 |
| A.1.17 | Summary on <i>S. cerevisiae</i> iND750 grown on D-glucose | 246 |
| A.2 | Optimal-Yield Elementary Flux Modes | 248 |
| A.2.1 | Elementary Flux Modes of <i>E. coli</i> iJR904 grown on <i>L-Arginine</i> . . | 248 |
| A.2.2 | Elementary Flux Modes of <i>E. coli</i> iJR904 grown on <i>L-Threonine</i> . | 250 |
| B | Notation | 251 |
| | Bibliography | 268 |
| | Index | 274 |
| | Kurzzusammenfassung | 275 |
| | Curriculum Vitae | 277 |
| | Declaration | 281 |

Chapter 1

Introduction

Large parts of this chapter were taken from my master thesis. It informally introduces metabolic networks and the questions that I worked on in this thesis. Mathematical definitions for essential concepts for this thesis are introduced in Chapter 2.

Living organisms are very complex systems. In the past biologists tried to deal with the problem by looking at single enzymes or very special subsystems only. These methods were ignoring the interactions between the subsystems and were never able to give a whole-cell system-wide view. With the increasing amount of sequenced genomes, experimental high-throughput analyses and computational technology, the field of systems biology emerged. Systems biologists do not try to understand single enzymes, but the functionality of the whole cell.

Many systems biologists work on the reconstruction of metabolic networks. Metabolic networks help to understand physiological processes, which in turn help to bring enzymes in a whole cell context and annotate them with functional properties [78]. But not only deeper understanding of the functionality of the cell can be obtained [151, 82], there are also many applications in medicine [11, 146, 8] and bioengineering [149, 49, 146, 153].

With the advent of whole-genome sequencing and annotated genome-databases, the reconstruction of genome-scale metabolic networks becomes easier and easier [39, 51, 110, 160]. Genome-scale models give us the possibility to analyze the behavior of the cell as a whole on the computer. This is why biologists also talk in terms of *in silico* experiments in contrast to *in vitro* and *in vivo* experiments that are carried out in the wet lab or on living organisms respectively. Since many genes can be associated to enzymes and the enzymes to the biochemical reactions they catalyze, the effects of gene knock-outs can be analyzed without even performing mutations on real cells [20, 144]. Also symbiotic and parasite metabolism can be analyzed, which is very difficult *in vitro* [159]. But, genome-scale models are huge, for example the *iAF1260* model of the bacterium *Escherichia coli* contains 2077 reactions and 1668 metabolites [43]. Hence, mathematical and computational methods are necessary to analyze those networks [156].

1.1 Metabolic Networks

Before we can talk about methods for metabolic models, we need to understand what a metabolic model is. A metabolic network consists of a set of metabolites \mathcal{M} (chemical species) and a set of reactions \mathcal{R} . But this is already everything all metabolic models have in common. The amount of information on the reactions and metabolites may vary depending on the purpose of the network. Information you may encounter in metabolic networks are

- **Stoichiometries (stoichiometric matrix $S \in \mathbb{R}^{\mathcal{M} \times \mathcal{R}}$):** Each column $r \in \mathcal{R}$ of the stoichiometric matrix stores how many metabolites the respective reaction is consuming ($S_{mr} < 0$) or producing ($S_{mr} > 0$). Every metabolic model contains the stoichiometries of the reactions, but the quality of the stoichiometries may differ. In some cases for example mass conservation is not guaranteed by the reaction. This can be caused when similar metabolites were lumped together, or the production/consumption of H_2O or electrons were ignored [55, 46]. The stoichiometric matrix stores the stoichiometric coefficients. See Figure 1.1 for a small real-world example and 1.2 for an even smaller made-up example.

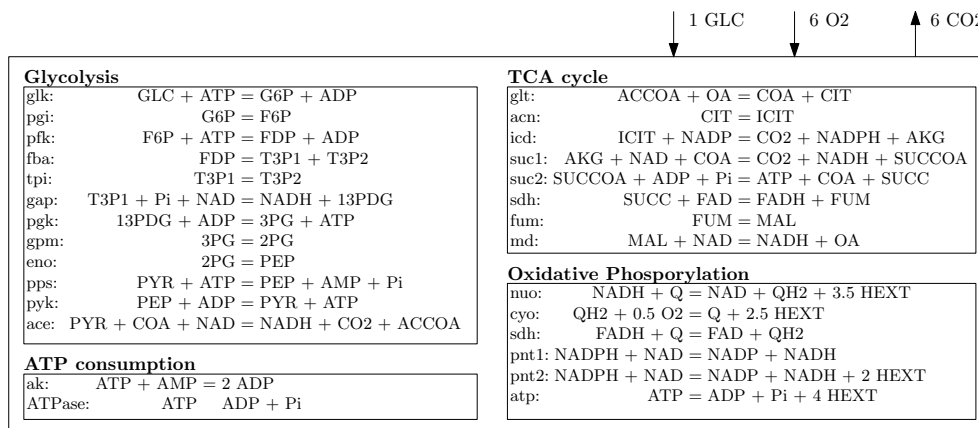
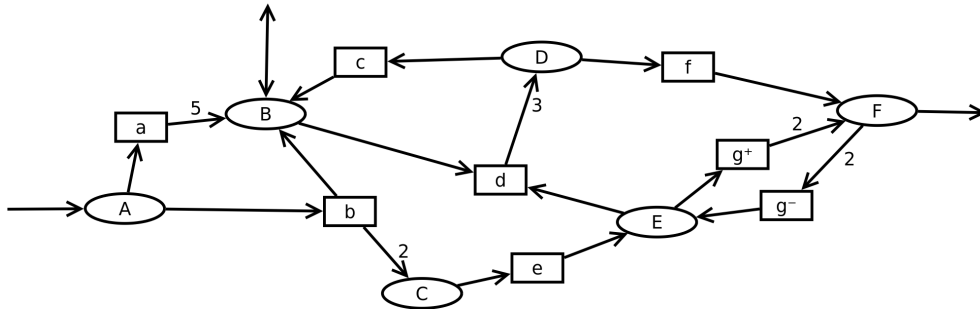


Figure 1.1: Network of *E. coli* core energy metabolism. Figure reproduced after [9].

- **Exchange reactions $\mathcal{E} \subset \mathcal{R}$:** Exchange reactions model the flow of nutrients and products in and out of the network. Hence, these reactions do not satisfy mass balance on purpose (else there could not be any inflow of nutrients). The reactions that are not exchange reactions are called *internal reactions* $\mathcal{I} := \mathcal{R} \setminus \mathcal{E}$.

One very important exchange reaction is the *biomass reaction*. It is possible to measure the types and amounts of amino-acids, cofactors, lipids, etc. the cell is built out of. If the cell grows, all those compounds need to be produced in exactly that ratio, else necessary amino acids etc. may be missing and essential proteins cannot be built. The biomass reaction consumes the compounds in exactly the necessary ratio. Hence, the biomass reaction measures how fast the cell grows [44].



$$S = \begin{matrix} & a & b & c & d & e & f & g & x_1 & x_2 & x_3 \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{pmatrix} -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 5 & 1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 2 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 3 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & -1 \end{pmatrix} \end{matrix}$$

Figure 1.2: Example of a metabolic network with the corresponding stoichiometric matrix S . The exchange reactions are denoted by x_1, x_2, x_3 and are simply indicated by arrows in the Petri-net type of drawing. Metabolites are drawn by ellipses and reactions by rectangles. Numbers on the arrows are the stoichiometric coefficients of the metabolite in the reaction connected by the arrow. If there is no number on an arrow, the stoichiometric coefficient is 1. Reaction g is split into two reactions to indicate that it is reversible.

- **Metabolite concentrations** $c \in \mathbb{R}^{\mathcal{M}}$: It is very hard to measure (or estimate) the concentrations of all metabolites, but the area of metabolomics shows great advances and more and more metabolite concentrations can be measured [35, 38, 59, 45]. This is valuable information from which we can estimate reaction directions and sometimes even reaction rates.
- **Flux** $v \in \mathbb{R}^{\mathcal{R}}$: The reaction rates are often called flux. The notion is analogous to flow in graphs. As with metabolite concentrations, the flux rates are often unknown and will often also depend on the environment. Some flux rates can be measured using isotope-labeling experiments [166, 133]. The variable J is also often found to denote flux in the literature (and also I have been using it in my master thesis).
- **Kinetics** $f : \mathbb{R}^{\mathcal{M}} \rightarrow \mathbb{R}^{\mathcal{R}}$: To compute flux of reactions from metabolite concentrations, information on the kinetics of the reaction is needed. The kinetics are usually functions that basically take metabolite concentrations as input, and compute the reaction rate. Good kinetic information is the rarest of all. Usually there

are a few default types of kinetic functions that are adapted using constants that specify enzyme concentration, temperature, pH etc [104]. Because kinetic data is so hard to obtain, methods were developed that also work with reduced kinetic information [152].

- **Regulatory Information:** Not every reaction that can happen, happens. For example, it may be the case that regulatory control of the cell inhibits that the needed enzyme is produced. Available transcriptomics data of gene expression is used to integrate these behaviors in the model [27, 162]. If an enzyme that catalyzes a reaction is not produced, this means that the reaction cannot carry any flux. In this case, this kind of information can simply be integrated into the model by deleting the corresponding reaction [122].

If kinetic information is available, a so-called kinetic model can be built. Since the reactions modify metabolite concentrations, the first derivative of the concentrations can be obtained from the flux:

$$\frac{d}{dt}c = Sv = Sf(c) \quad (1.1)$$

Hence we obtain a *system of ordinary differential equations* (ODE) describing the evolution and growth of the cell.

As already mentioned, good kinetic parameters are very hard to obtain. Hence constraint-based methods were developed that do not need kinetic parameters.

1.2 Steady-State Assumption

Constraint-based methods require the additional steady-state assumption, which is usually motivated as follows [122, 156]: If the environment does not change, the dynamical system (1.1) will reach a fixed point. In cases where the environment changes only slowly (compared to flux speeds), the error will not be very large. But the effect of this additional assumption is enormous, since Equation 1.1 simplifies to

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

which is also called the steady-state assumption or flux conservation. It can be considered as the metabolic equivalent to the flow conservation constraint in graphical flow problems. You simply need to replace the stoichiometric matrix by the incidence matrix of a digraph. See Figure 1.3 for an example.

The main difference between constraint-based methods and dynamical systems is, that we are not interested in the evolution of one flux vector but on the set of all feasible fluxes under the given constraints (e.g., at steady-state) [122, 137]. In this context additional information on the biologically feasible fluxes becomes vital. Some reactions for example can only proceed in one direction; hence, sign constraints on v are added. Sometimes it

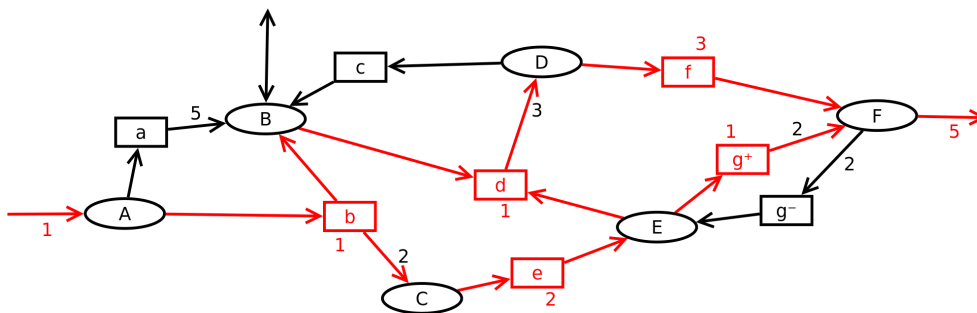


Figure 1.3: An example of a steady-state flux in a metabolic network. The reactions that carry flux are indicated in red. The amount of flux through the reactions can be read off from the red numbers next to the reactions. All stoichiometric coefficients are 1 except denoted otherwise (using black numbers).

is also possible to measure how fast one reaction can proceed; this leads to upper (and lower) bounds on the feasible fluxes.

1.3 Thermodynamic Constraints

One additional constraint that will be in the focus of this thesis is thermodynamic feasibility, which was first discussed by Oster in 1971 [115]. To understand thermodynamically feasible fluxes, we will have to include concentrations in a certain sense back into our formulation. This is no restriction on applicability, since it does not mean that we need to measure concentrations. As with flux, we can treat metabolite concentrations as unknown variables. A flux is called thermodynamically feasible if it does not violate the second law of thermodynamics. The second law of thermodynamics states that a reaction carries flux if and only if it reduces Gibbs free energy [10, 5, 124]. It is the same law that prohibits electrical current to go around a cycle if no energy source is attached [125]. We will see that this cycle-correspondence also holds for metabolic networks; for example the steady-state flux shown in Figure 1.4 is not thermodynamically feasible because it contains an internal cycle.

The reduction of Gibbs free energy is formulated using potential differences. Every metabolite $i \in \mathcal{M}$ in the network has a biochemical potential μ_i . This biochemical potential can usually be computed from its concentration c_i by

$$\mu_i = \mu_i^0 + RT \ln(c_i), \quad (1.3)$$

where R is the gas constant, T is temperature, and μ_i^0 is the equilibrium potential, which is different for each type of metabolite [124, 115]. The second law of thermodynamics can then be formulated using the chemical potentials as

$$\Delta\mu_r v_r < 0 \text{ or } \Delta\mu_r = 0 = v_r \quad \text{for every internal reaction } r \in \mathcal{I} \quad (1.4)$$

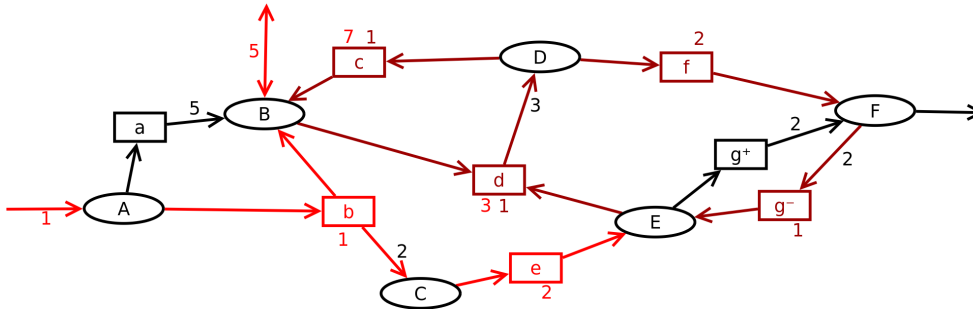


Figure 1.4: The steady-state flux shown in this metabolic network (again in red) is not thermodynamically feasible, because it contains an internal cycle (dark red)

where $\Delta\mu_r := \mu S_r$ is the potential difference induced by reaction $r \in \mathcal{I}$ (S_r denotes the r -th column of S). This formulation simply states that a reaction carries positive (resp. negative) flux, if and only if it has negative (resp. positive) potential difference. Note, that lumped metabolites (because they are not interesting by-products etc.) may significantly perturb potential differences. Thus, excellent stoichiometric information is necessary for these constraints to be applicable. But this is already everything that is needed to add these additional constraints.

This is the strength of thermodynamic constraints. Without much more additional information (except good quality stoichiometries), we can obtain more realistic results. In particular, we do not need to know information on reaction kinetics.

Since these thermodynamic constraints do not require much more additional information, they can easily be added to genome-scale metabolic networks [134, 43, 46, 123].

1.4 Constraint-Based Analysis

In this thesis I will present results for various kinds or aspects of analysis methods. To put these results into a broader application-oriented context, I will outline here the main approaches and questions in constraint-based analysis of metabolic networks.

1.4.1 Does the organism grow under certain conditions?

The simplest application of metabolic networks is to simulate an experiment on the computer with specified growth conditions, e.g., gene knock-outs and growth medium. *Flux balance analysis* (FBA) [163, 114] is then usually used to test if the organism has the metabolic capability to grow at all. If FBA predicts that no growth is possible, then it is a sign that it is useless to perform the experiment, because the organism will die. There are of course exceptions, e.g. when the correctness of the metabolic network should be verified.

1.4.2 How does the organism grow?

FBA does also predict some quantitative growth properties, like a maximal yield rate and a pathway that achieves this yield [40]. However, solutions of FBA are usually not unique [77]. Additionally, the objective criterion is often not followed completely *in vivo* such that in practice suboptimal yield is often observed.

Hence, a wide range of methods focuses on analyzing how the organism can grow [95, 63, 20, 58]. In particular the exclusion of growth behaviors can lead to model verification experiments and thus, model curation.

1.4.2.1 Yield maximization vs. growth maximization

In a very common FBA setup the uptake of nutrients is limited, while the flux through the remaining reactions is unlimited (except for irreversibility constraints). In these cases, FBA does actually not predict how fast the organism can grow, but how efficiently it can turn the nutrient into growth.

According to evolutionary theory (at least for bacteria) however, a high yield rate with low growth rate is not an evolutionary advantage if the growth medium is shared. Fast-growing bacteria would simply leave no nutrients for the more efficiently, but slower growing individuals. This is a serious problem if evolutionarily stable cultures with high yield rate are industrially desired [157].

Predicting how fast an organism can grow on the other hand is much more complicated since it involves bounding the flux rates of internal reactions [146, 3]. This requires the knowledge of kinetic parameters and, if done properly, metabolite concentrations have to be taken into account also. This leads to huge non-convex non-linear problems, which are currently practically not solvable [103, 167].

1.4.2.2 Essential reactions and flux variability analysis

A wide class of methods focuses on the non-uniqueness of possible solutions. The most famous analysis method for exploring the range of possible solutions is *flux variability analysis* (FVA) [22, 95]. For each reaction the minimal and maximal flux rate is computed. If for a reaction $r \in \mathcal{R}$ a flux rate of 0 is in this computed range, we can conclude that the reaction is not essential and that it is likely that the organism can also live without it [95].

Similar approaches also exist for bounding metabolite concentrations if thermodynamic constraints are added [11, 82, 171, 25, 106]. Due to the non-convexity of thermodynamic constraints this is however always comes at the expense with additional computational challenges.

1.4.2.3 Flux coupling analysis and elementary flux modes

Many reactions depend on each other. A normal variability analysis cannot uncover these dependencies, which is why alternative methods have been developed. In *elementary flux mode* (EFM) analysis [141] or related concepts [138, 89] minimal pathways through the network are enumerated. This way all dependencies are precisely captured (and indeed the EFMs generate the whole flux space). However, this comes with the cost of combinatorial explosion, which makes it unpractical for genome-scale metabolic networks.

Flux coupling analysis (FCA) [21, 31, 90, 30] and *elementary flux patterns* [73] are a compromise, where only dependencies between pairs of reactions or in small subsystems are computed. Applications range from the identification of co-regulated reactions [109] to lethality analysis [21].

1.4.3 How can a growth behavior be enforced?

In many cases, we do not only want to understand how living organisms work, but we also want to apply our knowledge. In medicine, we want to identify reactions for potential treatment sites; in bio-engineering, we want to identify a modification of the metabolic network that forces the organism to produce desired by-products, for example for the production of bio-fuel. Very popular are gene knock-outs, because they can easily be realized in the lab and can be modeled in metabolic networks by deleting the catalyzed reactions (reaction knock-out).

MOMA [144] and ROOM [147] assume that the regulation of the organism only adjusts minimally to the performed gene knock-outs and hence search for a flux distribution in the modified metabolic network that is as close as possible to a reference distribution of the original strain. Other methods, like OptKnock [20] and cut-sets [79, 87] assume that the organism will evolve to use optimally the reduced metabolic capabilities.

Mixed integer programming and bilevel optimization are frequently encountered techniques in this area and many of the network analysis methods are used for pre-processing or as sub-steps.

1.5 Mathematical Context

Analyzing metabolic networks is not only interesting from a biological perspective. First of all metabolic networks are nothing else than directed, weighted hypergraphs. Hence, we can use Petri-nets [126, 164], Lattices [58], and oriented matroids as a generalization of directed graphs [15, 9] to describe these networks.

The steady-state assumption and bounds on flux-rates give rise to a polyhedral flux space. Hence, many questions can be formulated using computational discrete geometry

and combinatorial optimization. Frequent examples are linear programming to optimize on flux spaces and vertex enumeration to enumerate EFMs. Complicated, non-convex constraints lead frequently to *mixed integer linear programming* (MILP) formulations, which despite theoretical hardness can often be solved efficiently due to intense research in the last decades.

Metabolic networks also give rise to new questions in polytope theory and related fields. Flux modules (Chapter 6) are a wonderful example for this. Originally motivated from metabolic networks, they are a structural property (formulated using matroid theory [118]) of polyhedra in general which I then used to develop a new method for vertex enumeration of polytopes.

1.6 Structure of this Thesis

In this thesis I present several rather independent results of mine for the analysis of metabolic networks. All works include thermodynamic constraints in some way, but often applications can also be found without thermodynamic constraints.

In Chapter 2 I mathematically introduce notation and discuss many fundamental properties of thermodynamic constraints. This way, Ch. 2 forms the basis for many of the results in this thesis.

In Chapter 3 I continue the analysis of thermodynamic constraints from a complexity theoretical point of view. In particular, I show that many flavors of thermodynamic constraints lead to NP-hard decision problems. This is the motivation for the computational methods developed in Ch. 4 and Ch. 5, where I investigate computational methods for solving the usually NP-hard computational problems in practice. The complexity theoretic results have also consequences on sampling methods for the thermodynamically constrained flux space. Hence, I present in Ch. 9 theoretical and practical obstacles for sampling methods.

In Chapter 4 I refine the optimization methods developed in my master thesis by introducing a new perspective to the branching procedure and by incorporating structural information like flux coupling data.

Chapter 5 deals with the problem of optimizing over metabolite potentials. Here, I address topological issues and show how MILPs with strict inequalities can be solved computationally. Since optimization with bounds on metabolite potentials is much harder than without, I present a new method to tighten and propagate metabolite bounds.

In Chapter 6 I present my results on flux modules and their generalization: k -modules. I stumbled over this field more by accident than on purpose, but it turned out to be very rich in mathematical theory. On the other hand it also exhibits the weakest links to thermodynamic constraints. Here, I introduce flux modules and k -modules as structural properties of metabolic networks and polyhedra in general. I show how k -modules are connected to matroid theory, which leads to efficient computation algorithms. Further-

more, I present a new method that utilizes k -modules to enumerate EFMs of metabolic networks resp. vertices of polytopes.

In Chapter 7 I present results obtained in a collaboration with G. Cogne on an unintuitive growth behavior of *Chlamydomonas reinhardtii*. I show how the problem can be modeled using metabolic networks and formalized as a bilevel optimization problem. Based on results from Ch. 4, I develop a method for solving these bilevel optimization problems.

In Chapter 8 I present work done together with Y. Goldstein, where we show how *flux coupling analysis* (FCA) can be generalized to arbitrary qualitative models. As a show case, we investigate the incorporation of thermodynamic constraints into FCA.

Chapter 2

Feasible Pathways

Abstract In this chapter we introduce basic concepts for the analysis of metabolic networks. Usually, every feasible pathway will have to satisfy the steady-state assumption. In addition, thermodynamic constraints restrict the signs that flux vectors can take.

Thermodynamic constraints come in many different forms throughout the literature. Not all of them are mathematically equivalent. Generally, thermodynamic constraints state that a reaction can only carry (positive) flux if it has a negative potential difference. However, sometimes it is required that if a reaction carries no flux, also the potential difference has to be zero. In other cases, a reaction is always allowed to have no flux. Both cases have legitimate biological background, but the mathematical properties change significantly.

Additional side constraints, like reaction irreversibilities, or knowledge of metabolite concentration modify the structure of the problem.

In this chapter, we mathematically introduce the different concepts of thermodynamic constraints. Furthermore, we will compare the different structural properties. This chapter will also supply us with the theoretical background that is then used in the following chapters.

This chapter mostly summarizes existing results. Most of the aspects have also been already discussed in my master thesis and are repeated here for completeness' sake.

2.1 Basic Concepts and Notation

Before we give a mathematical definition of steady-state flux, we first fix some notation. Every metabolic network has to consist of metabolites, reactions and a stoichiometric matrix:

Definition 2.1.1 (Metabolic network) A *three-tuple* $\mathcal{N} = (\mathcal{M}, \mathcal{R}, S)$, $S \in \mathbb{R}^{\mathcal{M} \times \mathcal{R}}$ is

called a metabolic network. Its metabolites are denoted by \mathcal{M} , the reactions by \mathcal{R} , and the stoichiometric matrix by S . \square

Note that for computation-theoretic results, we will assume that all entries of the stoichiometric matrix are rational. This also holds for other kind of (formally real-valued) input-parameters, like flux vectors.

Often, we have additional information like a set $\text{Irrev} \subset \mathcal{R}$ of irreversible reactions, lower and upper bounds $\ell, u \in \mathbb{R}^{\mathcal{R}}$ on flux values, a set of internal reactions $\mathcal{I} \subseteq \mathcal{R}$, exchange reactions $\mathcal{E} := \mathcal{R} \setminus \mathcal{I}$, etc. Since this additional information varies from case to case, it is not included in the definition itself but will be given next to it, e.g. “Let $\mathcal{N} = (\mathcal{M}, \mathcal{R}, S)$ be a metabolic network with irreversible reactions $\text{Irrev} \subseteq \mathcal{R}$.”

The definition is similar to the one of a graph with the exception that we encode the incidence information via the stoichiometric matrix and not in the reactions themselves, see Figure 1.2. To work properly with the stoichiometric matrix, we will write S_{*r} to denote the column corresponding to reaction r and S_{m*} to denote the row corresponding to metabolite m respectively. S_{mr} then denotes the entry corresponding to reaction r and metabolite m . If it is clear from the context, we will omit the ‘*’ and simply write S_r to denote the r .th column or the r .th row of S respectively. We will also use sets of indices to denote submatrices; $S_{\mathcal{I}}$ for example will denote the stoichiometries of all internal reactions \mathcal{I} . We will also use similar notation on vectors, like the flux vector v ; there, v_r denotes the flux through reaction r .

We use pr_A to denote the projection on the components A , i.e., $\text{pr}_A(v) = v_A$ and $\text{pr}_A(P) = \{v_A : v \in P\}$. For spaces $P^A \subseteq \mathbb{R}^A, P^B \subseteq \mathbb{R}^B$ the product is defined as $P^A \times P^B = \{v \in \mathbb{R}^{A \times B} : v_A \in P^A, v_B \in P^B\}$. For products over a family of flux spaces, we also use \prod .

Often, I will use arguments based on the closure of sets. Therefore, I use for $P \subseteq \mathbb{R}^n$

- $\text{conv}(P) := \{\sum_{i=1}^m \lambda_i v^i : \sum_{i=1}^m \lambda_i = 1, v^i \in P, \lambda_i \geq 0 \text{ for all } i = 1, \dots, m \text{ and } m \in \mathbb{N}\}$,
- $\text{cone}(P) := \{\sum_{i=1}^m \lambda_i v^i : v^i \in P, \lambda_i \geq 0 \text{ for all } i = 1, \dots, m \text{ and } m \in \mathbb{N}\}$,
- $\text{aff}(P) := \{\sum_{i=1}^m \lambda_i v^i : \sum_{i=1}^m \lambda_i = 1, v^i \in P, \text{ for all } i = 1, \dots, m \text{ and } m \in \mathbb{N}\}$,
- $\text{span}(P) := \{\sum_{i=1}^m \lambda_i v^i : v^i \in P \text{ for all } i = 1, \dots, m \text{ and } m \in \mathbb{N}\}$.

for matrices $D \in \mathbb{R}^{n \times k}$ I use

- $\text{span}(D) := \{D\alpha : \alpha \in \mathbb{R}^k\}$,
- $\text{ker}(D) := \{v : Dv = 0\}$.

In the Appendix B you find a summary of all notation used in this thesis.

Through out the thesis we will work with metabolic networks that satisfy the following two assumptions:

Assumption 2.1.1 (*S* does not contain zero columns) *In the whole thesis, we will assume that the stoichiometric matrix S of any metabolic network does not contain any zero columns (i.e., columns that contain only zeros). This is not a restriction for real-world applications, since zero columns correspond to reactions that do not involve any metabolites.* \square

Assumption 2.1.2 (Finite Models) *We will also assume that \mathcal{R} and \mathcal{M} are finite. It may be strange to mention this explicitly, but there are works by Hatzimanikatis et al. [62] that implicitly define metabolites and reactions. In such a setting this assumption may not be true.* \square

2.1.1 Directed Reactions and Equivalence

I developed the concept of pseudo-reaction in particular to make our lives much easier in Ch. 5. This subsection was not part of my master thesis.

When dealing with thermodynamic constraints, the direction of a reaction is a core property. In many cases we can simplify notation by assuming that all reactions are used in forward direction. This can usually be done by simply reversing reactions. Therefore we define the set of pseudo-reactions that contains the (proper) reactions and their reversed counterparts.

Definition 2.1.2 (Pseudo-reaction) *The set of pseudo-reactions $\overline{\mathcal{R}}$ is defined as*

$$\overline{\mathcal{R}} := \{(r, +) : r \in \mathcal{R}\} \dot{\cup} \{(r, -) : r \in \mathcal{R}\}$$

We identify each $r \in \mathcal{R}$ with $(r, +) \in \overline{\mathcal{R}}$. \square

Often, the set of internal pseudo-reactions is of special importance. Therefore, we define

$$\overline{\mathcal{I}} := \{(r, s) \in \overline{\mathcal{R}} : r \in \mathcal{I}, s \in \{+, -\}\}.$$

The reverse operator turns the direction of a pseudo-reaction and it helps us to simplify notation. Instead of writing $(r, -)$ we can then write $-r$ instead.

Definition 2.1.3 (Reverse operator) *The reverse operator $\text{rev} : \overline{\mathcal{R}} \rightarrow \overline{\mathcal{R}}$ is defined as*

$$\begin{aligned} (r, +) &\mapsto (r, -) \\ (r, -) &\mapsto (r, +) \end{aligned}$$

For $s \in \overline{\mathcal{R}}$ we also write $-s$ instead of $\text{rev}(s)$. \square

We want to work with pseudo-reactions as seamlessly as we do with proper reactions. Hence, we also define indexing of the stoichiometric matrix with pseudo-reactions and similarly indexing of vectors.

Definition 2.1.4 (Stoichiometry of a pseudo-reaction) *The stoichiometry of each pseudo-reaction is defined by $S : \bar{\mathcal{R}} \rightarrow \mathbb{R}^{\mathcal{M}}$ with*

$$\begin{aligned}(r, +) &\mapsto S_r \\ (r, -) &\mapsto -S_r.\end{aligned}$$

We also use subscript notation S_s to denote $S(s)$. □

Observation 2.1.1 *The subscript notation for $r \in \mathcal{R}$ introduced in Def. 2.1.4 is consistent with the subscript notation for matrices.* □

Definition 2.1.5 (Indexing with pseudo-reactions) *Let $v \in \mathbb{R}^A$ with $A \subseteq \mathcal{R}$ and $r \in A$. We define*

$$v_{(r,-)} := -v_r.$$

Often, it also makes little sense to treat lower and upper flux bounds separately. With pseudo-reactions we can deal with them in a unified way, since the upper bounds work as the lower bounds for the reversed reaction:

Definition 2.1.6 (Indexing flux bounds) *Let $\ell, u \in \mathbb{R}^A$ be flux bounds with $A \subseteq \mathcal{R}$ and $r \in A$. We define $\bar{\ell} \in \mathbb{R}^{\bar{\mathcal{R}}}$ with*

$$\begin{aligned}\bar{\ell}_{(r,+)} &:= \ell_r \\ \bar{\ell}_{(r,-)} &:= -u_r\end{aligned}$$

In particular, we can now define what the set of irreversible reactions are given flux bounds ℓ, u without having to pay additional attention to reactions that can only work in the reverse. Note, however, that often it is sufficient if just the set of irreversible reactions is given and hence we will also use the set Irrev of irreversible reactions if no flux bounds are given.

Definition 2.1.7 (Irreversible reactions) *Let $(\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network.*

For given flux bounds $\ell, u \in \mathbb{R}^{\mathcal{R}}$ we define

$$\text{Irrev} := \{r \in \bar{\mathcal{R}} : \bar{\ell}_r \geq 0\}.$$

If a metabolic network contains multiple reactions that actually do the same thing, then we want to call these reactions equivalent. A typical case would be that a network contains proper reactions r, s , where s is just the reverse of r . Since r, s are both proper reactions, it follows that $r \neq -s$ and an identification is problematic since $v_r = -v_s$ does not need to hold in general (and usually does not).

Definition 2.1.8 We define the equivalence relation \equiv on $\overline{\mathcal{R}}$ by

$$r \equiv s :\Leftrightarrow \exists \alpha > 0 : S_r = \alpha S_s.$$

For proper reactions r, s where s is the reverse of r , we can now write $r \equiv -s$.

Sometimes, it is also important to have an easy notation to access the metabolites involved in a reaction. Hence, we define for a pseudo-reaction r

- $r^+ := \{m \in \mathcal{M} : S_{mr} > 0\}$ the set of metabolites produced by r ,
- $r^- := \{m \in \mathcal{M} : S_{mr} < 0\}$ the set of metabolites consumed by r ,
- $\underline{r} := \{m \in \mathcal{M} : S_{mr} \neq 0\}$ the set of metabolites involved in r .

2.1.2 Flux Coupling

In this thesis, I will work with two different kinds of flux coupling. Both are introduced here to clarify the differences. This subsection was not part of my master thesis.

In this thesis we will deal with two types of flux coupling. Directed flux coupling and undirected flux coupling.

Directed flux coupling does not really have a biological interpretation but it is a useful concept that can be used to speed up algorithms.

Definition 2.1.9 (Directed flux coupling) For $P \subseteq \mathbb{R}^{\mathcal{R}}$ and $r, s \in \overline{\mathcal{R}}$ we define the flux coupling relation \rightarrow_P by

$$r \rightarrow_P s :\Leftrightarrow \forall v \in P : v_r > 0 \rightarrow v_s > 0.$$

If the space P is clear, we also just write \rightarrow . □

In contrast to directed flux coupling, undirected flux coupling [21] has a biological interpretation. It models the effect of a reaction knock-out (which is an abstraction of a gene knock-out).

Definition 2.1.10 (Undirected flux coupling) For $P \subseteq \mathbb{R}^{\mathcal{R}}$ and $r, s \in \mathcal{R}$ we define the flux coupling relation $\xrightarrow{=0}_P$ by

$$r \xrightarrow{=0}_P s :\Leftrightarrow \forall v \in P : v_r = 0 \rightarrow v_s = 0.$$

We observe that directed flux coupling is stronger than undirected flux coupling:

Observation 2.1.2 For $r, s \in \mathcal{R}$ it holds that $r \xrightarrow{=0}_P s$ if the following conditions hold

- $s \rightarrow_P r$ or $s \rightarrow_P -r$
- $-s \rightarrow_P r$ or $-s \rightarrow_P -r$

□

2.2 Steady-State Assumption

Although the steady-state assumption is a very fundamental concept, I found that there are different ways to motivate the steady-state assumption. This has fundamental consequences for the biological plausibility of additional constraints. In particular this is highly relevant in the context of thermodynamic constraints, which is why I added this discussion here.

The first property we derived in the introduction was the steady-state assumption, which lead to the notion of steady-state fluxes [122, 156]. An example can be seen in Figure 1.3.

Definition 2.2.1 (Steady-state flux) Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R}, S)$, we call a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ a steady-state flux in \mathcal{N} if

$$Sv = 0$$

In the introduction, we motivated this property by assuming that the dynamical system (1.1) will reach a fixed point if the environment is not (or only slowly) changing. However, we also know that some organisms are exhibiting oscillations. In that case, we can still motivate the steady-state assumption as follows.

2.2.1 Average Fluxes are Steady-State

If an internal metabolite would accumulate or deplete over a long time, the metabolite would eventually be available in such high quantities that it jams all other reactions or in the other case would not be available at all and hence, also block the operation of the cell. It follows that every metabolite (on the long time scale) must be produced at the same rate as it is consumed.

Mathematically, we can derive this property by studying the dynamical system describing the metabolic states of the organism. In contrast to Eq. 1.1 we can also allow enzyme concentrations E to change over time due to regulatory control:

$$\dot{c}(t) := \frac{dc(t)}{dt} = Sv(t), \quad v(t) = f(E(t), c(t)) \quad (2.1)$$

We assume that enzyme and metabolite concentrations are compact, in particular at every time point the metabolite concentrations c have to satisfy $0 \leq c(t) \leq c^{\max}$ for a fixed $c^{\max} \in \mathbb{R}^{\mathcal{M}}$ and similar for the enzyme concentrations E . Furthermore, assume that E and f are continuous. Hence, $v(t)$ is also bounded. Assume that the average concentration \bar{c} and flux \bar{v} are well defined (in general \bar{v} and \bar{c} may not exist and we would have to consider the set of accumulation points):

$$\bar{c} := \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T c(t) dt$$

$$\bar{v} := \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T v(t) dt$$

In particular the average flux \bar{v} is the phenotype that we would expect to see, if we observe the organism for a long enough time period.

We observe that also \bar{v} satisfies the steady-state assumption:

$$\begin{aligned} \int_0^T \dot{c}(t) dt &= c(T) - c(0) \\ \Rightarrow \int_0^T \dot{c}(t) dt &\leq c^{\max} \\ \Rightarrow \frac{1}{T} \int_0^T \dot{c}(t) dt &\leq \frac{c^{\max}}{T} \xrightarrow{T \rightarrow \infty} 0 \\ \Rightarrow \frac{1}{T} \int_0^T S f(E(t), c(t)) dt &\xrightarrow{T \rightarrow \infty} 0 \\ \Rightarrow S \frac{1}{T} \int_0^T f(E(t), c(t)) dt &\xrightarrow{T \rightarrow \infty} 0 \\ \Rightarrow S \frac{1}{T} \int_0^T v(t) dt &\xrightarrow{T \rightarrow \infty} 0 \\ &\Rightarrow S \bar{v} = 0 \end{aligned}$$

2.2.2 Concentrations for Average Fluxes

If the dynamical system converges to a steady-state, then it clearly follows that $\bar{v} = f(\bar{c})$. Feinberg and Horn [42] identified structural properties from which convergence to a steady-state can be concluded. However, in practice these properties are sometimes not satisfied. Feinberg and Horn even give a part of the glycolysis pathway as an example. Also, the oscillating predator-prey model of Lotka and Volterra can be modeled as a metabolic network with mass action kinetics:



Here, we assume that the concentration of B is kept constant and C is immediately transported out of the network. With simple mass-action kinetics (see [80] for more details on kinetics) (c_1 is the concentration of A_1 and c_2 is the concentration of A_2) we get:

$$\begin{aligned}v_1 &= c_1 \\v_2 &= c_1 c_2 \\v_3 &= c_2\end{aligned}$$

We derive the following ODE system:

$$\begin{aligned}\dot{c}_1 &= c_1 - c_1 c_2 \\ \dot{c}_2 &= c_1 c_2 - c_2\end{aligned}\tag{2.3}$$

Since the kinetic rate law f is not linear in general, we cannot use the same technique as in the previous section to show that $\bar{v} = f(\bar{c})$. Therefore, I would expect that a lot of ugly things can happen if the network is exhibiting oscillations, even if we only consider mass-action kinetics (and keep the enzyme concentrations constant).

Conjecture 2.2.1 *There exists a metabolic network with mass-action kinetics and initial concentrations $c_0 \in \mathbb{R}_{>0}^M$ such that there exists no $\bar{c} \in \mathbb{R}_{\geq 0}^M$ with $\bar{v} = f(\bar{c})$. \square*

It should be remarked that it is not so easy to come up with an example to prove the conjecture as it may seem. For example, for the network with Lotka-Volterra dynamics (Eq. 2.3), one can even show that the long-term behavior equals the steady-state of the system, since several flux rates are linear in the metabolite concentrations and hence, we can pull f out of the integral.

However, if indeed the conjecture is true, it follows that one must be careful when applying thermodynamic constraints or other concepts related to the existence of metabolite concentrations, like the growth maximization results by Stefan Müller et al. [103] and Wortel et al. [167]. It could for example be that a reaction may not be able to carry flux in a steady-state of the dynamical system, but carry flux in an average flux distribution \bar{v} . By a similar reason, it can also be that the growth rate is maximized by an oscillation and hence, maximal growth is not obtained by an EFM.

Nevertheless we conclude that the notion of average flux explains the success of the steady-state assumption in methods like *flux balance analysis* (FBA) to predict growth-rate [40], since growing cells are subject to oscillations, like the cell cycle and hence, clearly not in a physical steady-state.

2.3 Polyhedral Flux Spaces

Next to the steady-state condition, usually lower and upper bounds on the flux rates $\ell, u \in \mathbb{R}^{\mathcal{R}}, \mathbb{R}_{\infty} := \mathbb{R} \cup \{-\infty \cup \infty\}$ are given. Here, in abuse of notation $v_r \leq \infty$ ($v_r \geq -\infty$) means that v_r is unbounded from above (below). The space of feasible fluxes then forms a polyhedron, which we also call the *flux polyhedron*:

$$P := \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, \ell \leq v \leq u\}.$$

If the flux bounds only encode irreversibilities, i.e., $u_r = \infty$ for all $r \in \mathcal{R}$ and $\ell \in \{-\infty, 0\}^{\mathcal{R}}$, then P is a cone, the so called *flux cone*. Note that we do not need to specify all the bounds but it is sufficient to give a set of irreversible reactions $\text{Irrev} \subseteq \mathcal{R}$. The flux cone is then defined as the set

$$C := \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, v_{\text{Irrev}} \geq 0\}. \quad (2.4)$$

2.4 Elementary Modes

Elementary flux modes (or *elementary modes*, EFM for short) and *generating vectors* are one of the most commonly used tools to analyze metabolic networks, infer potential regulatory sites and find knock-out targets [141, 140, 143, 63, 36, 112, 7]. Basically, generating vectors, extreme rays, minimal metabolic behaviors [89] and elementary modes are reformulations of the flux cone that can give deeper insights into the functions of metabolic networks [120]. They all have their foundation in Minkowski-Weyl's theorem for polyhedral cones. See [60, 18] for an introduction into convex polytope theory. Cones can be described in two different ways: By an *outer description* (as defined in Eq.2.4) or an *inner description* (as generating vectors, etc.). A detailed discussion in the context of metabolic networks can be found in [88].

In the following we will recapitulate a few properties of elementary modes. A comprehensive study of elementary modes can be found in the PhD-Thesis of Terzer [155].

Definition 2.4.1 (Elementary flux mode) *Let $(\mathcal{M}, \mathcal{R}, S)$ be a metabolic network with irreversible reactions $\text{Irrev} \subseteq \mathcal{R}$.*

A steady-state flux vector $v \in \mathbb{R}^{\mathcal{R}} \setminus \{0\}, v_{\text{Irrev}} \geq 0$ is an elementary flux mode if and only if there exists no steady-state flux vector $w \in \mathbb{R}^{\mathcal{R}} \setminus \{0\}, w_{\text{Irrev}} \geq 0$ with

$$\text{supp}(w) \subset \text{supp}(v),$$

where $\text{supp}(x) := \{i \in \mathcal{R} : x_i \neq 0\}$ denotes the support of $x \in \mathbb{R}^{\mathcal{R}}$. □

The set of elementary modes is an inner description of the flux cone (as defined in Eq.2.4), i.e., every $v \in C$ is a conical combination of elementary flux modes [141]. The power of

elementary flux modes stems from the fact that a single elementary flux mode is simply a flux vector which can be much more directly interpreted than a set of inequalities. However, the set of elementary flux modes generally grows exponentially with the size of the metabolic network. Hence, enumeration of EFMs is up to now not possible on genome-scale metabolic networks.

Sometimes, we do not work on a cone, but we still want to have a notion that characterizes the flux modes with minimal support. Hence we define for $P \subseteq \mathbb{R}^{\mathcal{R}}$:

Definition 2.4.2 (Elementary flux mode w.r.t. P) *A steady-state flux vector $v \in P \setminus \{0\}$ is an elementary flux mode w.r.t. P if and only if there exists no flux vector $w \in P \setminus \{0\}$ with*

$$\text{supp}(w) \subset \text{supp}(v),$$

where supp denotes the support. The set of elementary modes w.r.t. P is denoted by $\text{EFM}(P)$. □

We observe that both definitions coincide in the case that $P = \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, v_{\text{Irrev}} \geq 0\}$.

Often, it is also very helpful to instead characterize elementary flux modes not using the support, but by the signed support, which will be discussed in more detail in Sec. 2.5.

Proposition 2.4.1 (EFM by signed support) *Let $P = F \cap Q$ with $F \subseteq \mathbb{R}^{\mathcal{R}}$ convex and $Q \subseteq \mathbb{R}^{\mathcal{R}}$ satisfying*

$$x \in Q \Rightarrow y \in Q \text{ for } y_{\mathcal{R} \setminus A} = x_{\mathcal{R} \setminus A}, y_A = 0, A \subseteq \mathcal{R}.$$

A steady-state flux vector $v \in P \setminus \{0\}$ is an elementary flux mode if and only if there exists no flux vector $w \in P \setminus \{0\}$ with $\text{supp}(w) \subset \text{supp}(v)$ and

$$w_i > 0 \rightarrow v_i > 0 \quad \forall i \in \mathcal{R} \quad (2.5)$$

$$w_i < 0 \rightarrow v_i < 0 \quad \forall i \in \mathcal{R} \quad (2.6)$$

PROOF We show both directions separately.

\Rightarrow : Since v is an EFM w.r.t. P it follows that there exists no w with smaller support and hence, the condition is satisfied.

\Leftarrow : Assume there exists a $v \in \mathbb{R}^{\mathcal{R}}$ that is not an EFM. Let $w \in P$ with $\text{supp}(w) \subset \text{supp}(v)$. Since one of (2.5), (2.6) is not satisfied, there exists an $r \in \mathcal{R}$ where v_r and w_r have different sign. Define $x : [0, 1] \rightarrow \mathbb{R}^{\mathcal{R}}$ by

$$\alpha \mapsto \alpha v + (1 - \alpha)w.$$

Since $\text{supp}(w) \subset \text{supp}(v)$ we observe that $x(\alpha) \neq 0$ for all $\alpha \in [0, 1]$. We observe that x is a continuous function where $x_r(\alpha)$ changes its sign from $\alpha = 0$ to $\alpha = 1$. Hence $\bar{\alpha} = \min_{\alpha \in [0,1]} : \text{supp}(x(\alpha)) \subset \text{supp}(v)$ is well defined and $x(\bar{\alpha}) \in Q$. By convexity we have $x(\bar{\alpha}) \in F$ and by continuity it follows that $x(\bar{\alpha})$ satisfies (2.5) and (2.6). ■

Note that Prop. 2.4.1 applies to all convex spaces (by choosing $Q = \mathbb{R}^{\mathcal{R}}$). We will later also see that thermodynamic constraints can be formulated in the form of Q and hence, this result will also apply to thermodynamically constrained flux spaces.

2.5 Matroid Theory

We can also describe metabolic networks with matroid theory[118]. We will see that steady-state flux modes correspond to *cycles* (also called *vectors*) in oriented matroids[15] and elementary flux modes correspond to *circuits*.

Matroids and oriented matroids describe the metabolic networks by the reactions \mathcal{R} , which are called elements, and cycles, which are sets of reactions. In the case of oriented matroids, the cycles are signed, i.e., for each element of a cycle we store whether it is a positive element or a negative element. For example assume that the flux vector $v = (-2, 3, 4, -5, 1)$ is a steady-state flux vector of some metabolic network. In matroid theory it follows that $\underline{C} = \{r_1, r_2, r_3, r_4, r_5\} = \text{supp}(v)$ is a cycle. In oriented matroids the cycle is signed and it is represented by $C^+ = \{r_2, r_3, r_5\}$ and $C^- = \{r_1, r_4\}$. We also write the cycle as the signed vector $(-, +, +, -, +)$.

In (oriented) matroid theory the notion of metabolite is lost. The book by Ziegler has a very nice introduction to oriented matroids in chapter 6 [172].

2.5.1 Notation and Definition

Let E be a set. Based on the notation used in [15], we call $C = (C^+, C^-)$ with $C^+ \dot{\cup} C^- \subseteq E$ a signed subset of E . Alternatively, we also use the incidence vector notation for C , i.e., $C \in \{-, 0, +\}^E$, where $C_i = - \Leftrightarrow i \in C^-$ and $C_i = + \Leftrightarrow i \in C^+$ for all $i \in E$. These two forms of notation will be used interchangeably. A signed set C can also be represented by the set of pseudo-reactions

$$\{(r, +) : r \in C^+\} \dot{\cup} \{(r, -) : r \in C^-\}.$$

We use this for indexing with signed sets and to consider elements of signed sets. Note however that not every set of pseudo-reactions can be represented by a signed set.

Since oriented matroids are a generalization of matroids, we sometimes are not interested in the signs and hence, $\underline{C} := C^+ \dot{\cup} C^-$ will denote the support of C .

We will now give an abstract definition of oriented matroids and state some of the main theorems. See the book by Björner et al. [15] for a comprehensive survey on oriented matroids.

Definition 2.5.1 (Oriented Matroid) *A tuple $\mathcal{M} = (E, \mathcal{C})$ with elements E and circuits $\mathcal{C} \subseteq \{-, 0, +\}^E$ is called an oriented matroid if the following circuit axioms are satisfied:*

C0 $\emptyset \notin \mathcal{C}$, where \emptyset , respectively 0 , are shorthand notations for (\emptyset, \emptyset) in the context of signed sets.

C1 $\mathcal{C} = -\mathcal{C}$, where $-\mathcal{C} := \{-C : C \in \mathcal{C}\}$

C2 for all $X, Y \in \mathcal{C}$ if $\underline{X} \subseteq \underline{Y}$, then $X = Y$ or $X = -Y$.

C3 for all $X, Y \in \mathcal{C}$ with $X \neq -Y$ and $e \in X^+ \cap Y^-$, there is a $Z \in \mathcal{C}$ such that

$$\begin{aligned} Z^+ &\subseteq (X^+ \cup Y^+) \setminus \{e\} \\ Z^- &\subseteq (X^- \cup Y^-) \setminus \{e\}. \end{aligned}$$

It can be shown that also a stronger form of axiom C3 can be derived.

Theorem 2.5.1 (Strong elimination, Theorem 3.2.5 in [15]) *Let \mathcal{C} be a collection of signed subsets of a set E satisfying (C0), (C1), (C2). Then (C3) is equivalent to*

C3' *for all $X, Y \in \mathcal{C}$ with $X \neq -Y$ and $e \in X^+ \cap Y^-$ and $f \in (X^+ \setminus Y^-) \cup (Y^- \setminus X^+)$, there is a $Z \in \mathcal{C}$ such that*

$$\begin{aligned} Z^+ &\subseteq (X^+ \cup Y^+) \setminus \{e\}, \\ Z^- &\subseteq (X^- \cup Y^-) \setminus \{e\}, \text{ and} \\ f &\in Z. \end{aligned}$$

Note that if we remove the orientation property, these axioms are the circuit axioms of ordinary matroids, i.e., $\underline{\mathcal{M}} := (E, \underline{\mathcal{C}})$ is a matroid. A more common definition of matroid is done via *independent* sets. This definition is inspired from linear algebra and generalizes the concept of *linearly independent* sets. We will denote the family of independent sets by \mathcal{J} , because we use the more commonly used letter \mathcal{I} for the set of internal reactions. To avoid confusion later on in the thesis, we will often just write “ A is independent” instead of $A \in \mathcal{J}$.

Definition 2.5.2 (Matroid) *A tuple $\mathcal{M} = (E, \mathcal{J})$ with elements E and independent sets $\mathcal{J} \subseteq 2^E$ is called a matroid if the following axioms are satisfied:*

I1 $\emptyset \in \mathcal{J}$.

I2 If $Y \in \mathcal{J}$ and $X \subseteq Y$, then $X \in \mathcal{J}$.

I3 If $I_1, I_2 \in \mathcal{J}$ and $|I_1| < |I_2|$, then there exists an $e \in I_2 \setminus I_1$ such that $I_1 \cup \{e\} \in \mathcal{J}$. \square

A set $A \subseteq \mathcal{R}$ is called dependent, if it is not independent ($A \notin \mathcal{J}$). The minimal dependent sets (w.r.t. set inclusion) are called the circuits of the matroid. For the proof of the equivalence of the two definitions, we refer the reader to the book by Oxley [118]. For us, the most important class of matroids will be *linear matroids*, where the independent sets are simply linearly independent sets:

Definition 2.5.3 (Linear Matroid) Let E be a finite set and let $A \in \mathbb{R}^{m \times E}$ be a matrix. The linear matroid represented by A is a matroid (E, \mathcal{J}) with

$$\mathcal{J} := \{X \subseteq E : \text{the columns of } A_X \text{ are linearly independent}\}.$$

2.5.2 Matroids from Metabolic Networks

We show now how metabolic networks can be interpreted as oriented matroids [111, 9] (the connection to matroid theory follows immediately). Since we are only working with signs in oriented matroid theory, we have to translate the real flux vectors into sign vectors:

Definition 2.5.4 (Sign function (signed support)) For $x \in \mathbb{R}^n$, define $\text{sign}(x) := s \in \{-, 0, +\}^n$ with:

$$s_i := \begin{cases} - & \text{if } x_i < 0 \\ 0 & \text{if } x_i = 0 \\ + & \text{if } x_i > 0 \end{cases} \text{ for all } i \in 1, \dots, n$$

It turns out that the sign function $\text{sign}(\cdot)$ is a very useful extension of the support function $\text{supp}(\cdot)$ and we will use it even in cases where we do not do purely oriented matroid theoretic arguments. For example, we have $x_{\text{sign}(x)} > 0$ ($\text{sign}(x)$ represented as a set of pseudo-reactions), which allows us to just work with positive vectors.

We have seen that circuits in matroid theory can be defined as dependent sets with minimal support. For the minimality condition we used set inclusion (\subset). Using the subset relation induced by the representation as a set of pseudo-reactions, we obtain the following set-inclusion for signed sets (which is the \subset -relation used in oriented matroid theory).

$$X \subseteq Y \Leftrightarrow X^+ \subseteq Y^+ \wedge X^- \subseteq Y^-.$$

Let \mathcal{V} be a collection of signed subsets of E . We can define the set of minimal signed subsets:

$$\text{Min}(\mathcal{V}) := \{v \in \mathcal{V} : \nexists w \in \mathcal{V} \setminus \{v\} \text{ s.t. } w \subseteq v\}$$

In the interpretation of metabolic networks as oriented matroids, the collection \mathcal{V} is the set of flux directions of steady-state fluxes (called *vectors* or *cycles*) and $\mathcal{C} = \text{Min}(\mathcal{V} \setminus \emptyset)$ are the circuits:

Proposition 2.5.1 (Metabolic networks are oriented matroids) *Let S be a stoichiometric matrix, then $(\mathcal{R}, \mathcal{C})$ defines an oriented matroid, where $\mathcal{C} = \text{Min}(\mathcal{V})$ and $\mathcal{V} = \{\text{sign}(v) : Sv = 0, v \neq 0\}$.* \square

Such a construction of an (oriented) matroid is very common, which is why (oriented) matroids that are generated by a stoichiometric matrix S are also called *realizable* (oriented) matroids. For matroids we can also use the definition via independent sets. Then, we define the independent sets of the matroid as sets $A \subseteq \mathcal{R}$ where S_A only contains linearly independent columns. It can be seen from basic linear algebra that the linearly independent sets satisfy the independence axioms of the matroid. In my master thesis I also repeated the proof that shows that the construction via the circuits define a matroid. Observe that the set of cycles \mathcal{V} can be obtained by taking unions of circuits (in the case of matroid theory, where we are only interested in the support); in the case of oriented matroid theory, we obtain the set \mathcal{V} by *composition* which is a generalization to union for signed sets.

Proposition 2.5.2 (Composition, Thm. 3.7.5 in [15]) *For $a = (a^+, a^-), b = (b^+, b^-) \in \mathcal{V}$ it holds that the composition*

$$(a^+, a^-) \circ (b^+, b^-) := (a^+ \cup (b^+ \setminus a^-), a^- \cup (b^- \setminus a^+))$$

is also a vector of the oriented matroid. \square

In linear algebra we observe that for v^a, v^b with $\text{sign}(v^a) = a, \text{sign}(v^b) = b$ it follows that for small enough $\varepsilon > 0$ it holds that

$$\text{sign}(v^a + \varepsilon v^b) = a \circ b.$$

The following two examples of oriented matroids will be important for the analysis of metabolic networks:

Definition 2.5.5 (Flux Mode Matroid) *Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R}, S)$, let $(\mathcal{R}, \mathcal{C})$ denote the oriented matroid obtained from S by Proposition 2.5.1. We will call $(\mathcal{R}, \mathcal{C})$ the flux mode matroid.* \square

Note, that the circuits of the flux mode matroid are the minimal sign vectors of steady-state fluxes. Hence, a circuit $C \in \mathcal{C}$ is the signed support of an elementary mode if and only if it obeys the sign constraints of the irreversibilities ($C^- \cap \text{Irrev} = \emptyset$).

Definition 2.5.6 (Internal Cycle Matroid) *Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R}, S)$ with internal reactions $\mathcal{I} \subseteq \mathcal{R}$, let $(\mathcal{I}, \mathcal{C})$ denote the oriented matroid obtained from $S_{\mathcal{I}}$ by Proposition 2.5.1. We will call $(\mathcal{I}, \mathcal{C})$ the internal cycle matroid.* \square

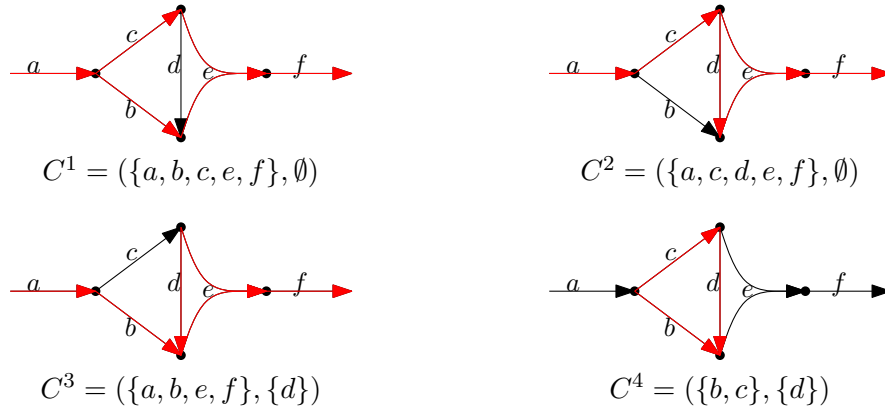


Figure 2.1: $C^1, C^2, C^3, C^4, -C^1, -C^2, -C^3, -C^4$ are the circuits of the flux mode matroid. $C^4, -C^4$ are the only circuits of the internal cycle matroid.

The internal cycle matroid consists of all the circuits that we do not want to have in thermodynamically feasible solutions. Observe that the circuits of the internal cycle matroid (in the following also called *internal circuits*) are a subset of the circuits of the flux mode matroid.

In my master thesis I showed that the internal cycle matroid is intrinsically linked to the space of potential differences by oriented matroid duality (cf. Thm. 2.6.1). In Figure 2.1 we can see an example of circuits of the two matroids. For an introduction to matroid duality see Ch. 3.4 in [15] and Ch. 2 in [118]. Here, we just mention that circuits in the dual matroid are called cocircuits, vectors (resp. cycles) in the dual matroid are called covectors (resp. cocycles).

The internal cycle matroid and the flux mode are also closely related in matroid theoretic terms. It can easily be seen that the internal cycle matroid is the restriction of the flux mode matroid to the internal reactions.

Proposition 2.5.3 (Restriction, Prop. 3.3.1 in [15]) *Let $M = (E, \mathcal{C})$ be an oriented matroid on elements E with circuits \mathcal{C} . For $F \subseteq E$ it holds that $M|_F := (F, \mathcal{C}')$ is an oriented matroid with*

$$\mathcal{C}' := \{C \in \mathcal{C} : \underline{C} \subseteq F\}.$$

We call $M|_F$ the restriction of M to F . □

Similarly, given a matroid M on elements E we call $M \setminus F := M|_{E \setminus F}$ the *deletion* of F from M .

2.5.3 Matroid Connectivity

Similar to connectivity in graphs, also a notion of connectivity can be defined for matroids. This will be of crucial importance in Ch. 6, where we will see that matroid

connectivity is closely linked to flux modules. However, matroid connectivity is not only important in the context of flux modules, but we can also use it for the analysis of thermodynamic constraints.

Proposition 2.5.4 (Prop. 4.1.2 of [118]) *Let M be a matroid on a set of elements E . The relation \sim defined by*

$$a \sim b :\Leftrightarrow \text{there exists a circuit containing } a \text{ and } b$$

is an equivalence relation on E . □

The equivalence classes of \sim are a generalization of 2-connected components from graph theory and hence, simply called *connected components*. Note that in the case of the flux mode matroid the connected components do not yet have an established biological interpretation, which will be discussed in Ch. 6.

2.6 Thermodynamic Constraints

In the following, we will partition the set of reactions \mathcal{R} into *internal reactions* \mathcal{I} and *exchange reactions* \mathcal{E} . The internal reactions will be subject to thermodynamic constraints, but the exchange reactions will not be. For metabolic networks, where we have split the reactions into internal and exchange reactions, we will often also write $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$.

We can now define thermodynamically feasible fluxes with respect to given metabolite potentials (which depend on the metabolite concentrations) by adding the condition that a reaction carries flux if and only if it has negative potential difference [10, 5, 124]:

Definition 2.6.1 (Strongly thermo. feasible flux w.r.t. metabolite potentials)

Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ and metabolite potentials $\mu \in \mathbb{R}^{\mathcal{M}}$, we call a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ strongly thermodynamically feasible in \mathcal{N} with respect to μ if

- v is a steady-state flux in \mathcal{N} .
- $\mu S_{*r} v_r < 0$ or $\mu S_{*r} = 0 = v_r$ for all $r \in \mathcal{I}$. □

See Figure 2.2 for an example of metabolite potentials that allow a strongly thermodynamically feasible flux. For practical reasons, it is useful to relax the second condition slightly such that reactions do not *have to* carry non-zero flux if they have non-zero potential difference, but *can* carry non-zero flux (see Figure 2.3).

For example if a reaction is catalyzed by an enzyme and the enzyme is not expressed, the reaction will carry nearly no flux although there may be non-zero potential difference. Theoretically, a reaction with negative potential difference always also happens

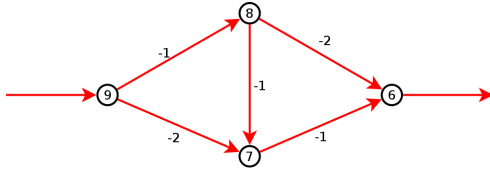


Figure 2.2: According to the potentials on the nodes and the induced potential differences on the reactions, every reaction must carry flux to be strongly thermodynamically feasible. This is possible, since for every metabolite that is produced, there also exists a reaction that consumes it.

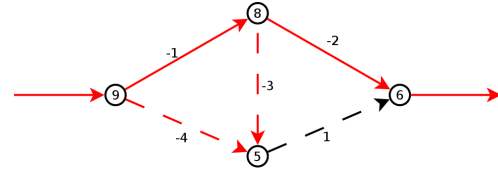


Figure 2.3: In this example, there exists no strongly thermodynamically feasible flux with these potentials, since the metabolite with potential 5 would have to be a sink, which is not allowed. But, there exists a weakly thermodynamically feasible flux, which carries only flux on the continuous arcs and not on the dashed ones (strong thermodynamics would force flux on the red, dashed arcs).

spontaneously. However, the flux rate is usually so low that it can be considered zero. Indeed, many kinetic rate laws are linear in the enzyme concentration [91]. Hence, these kinetic rate laws would also predict a zero flux even for negative potential difference if the enzyme concentration is zero.

Hence, we will call the thermodynamically feasible fluxes w.r.t. Definition 2.6.1 *strongly thermodynamically feasible* in contrast to *weakly thermodynamically feasible* fluxes that only have to satisfy the relaxed condition. For the definition of weakly thermodynamically feasible fluxes, we will follow the definition proposed by Beard et al. [9]. Since this will actually be the definition we will work with most of the time in this thesis, we will often simply omit the term “weakly”:

Definition 2.6.2 ((Weakly) thermo. feasible flux w.r.t. metabolite potentials)

Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S)$ and metabolite potentials $\mu \in \mathbb{R}^{\mathcal{M}}$, we call a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ thermodynamically feasible with respect to μ if

- v is a steady-state flux in \mathcal{N}
- $\mu_{S_{*r}} v_r < 0$ or $v_r = 0$ for all $r \in \mathcal{I}$. □

Sometimes, we want to apply an even weaker formulation of thermodynamic constraints. This is in particular the case when we have to deal with metabolite potentials.

Definition 2.6.3 (Relaxed thermo. feasible flux w.r.t. metabolite potentials)

Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S)$ and metabolite potentials $\mu \in \mathbb{R}^{\mathcal{M}}$, we call a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ relaxed thermodynamically feasible with respect to μ if

- v is a steady-state flux in \mathcal{N}
- $\mu S_{*r} v_r \leq 0$ for all $r \in \mathcal{I}$. □

2.6.1 Chemical Potentials

In many applications not all metabolite potentials are known. The potentials may actually also change when regulatory mechanisms modify metabolite concentrations. Hence, we usually have to treat chemical potentials as unknown variables.

Let $Q \subseteq \mathbb{R}^{\mathcal{M}}$ be the space of feasible metabolite potentials. Often we are not really interested in the metabolite potentials that make a flux vector thermodynamically feasible, but are happy with the existence of such.

Definition 2.6.4 (Thermodynamically feasible flux w.r.t. potential space) *Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ and a space of feasible metabolite potentials $Q \subseteq \mathbb{R}^{\mathcal{M}}$, we call a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ thermodynamically feasible with respect to Q if there exists $\mu \in Q$ such that v is thermodynamically feasible in \mathcal{N} w.r.t. μ . □*

Even if $Q = \mathbb{R}^{\mathcal{M}}$ not every steady-state flux vector is thermodynamically feasible. However, for $Q = \mathbb{R}^{\mathcal{M}}$ we have a nice alternative characterization of thermodynamic feasibility, which only generalizes in a limited extent to $Q \subset \mathbb{R}^{\mathcal{M}}$.

2.6.2 Without Bounds on Chemical Potentials

In the following we will call a flux $v \in \mathbb{R}^{\mathcal{R}}$ thermodynamically feasible if it is thermodynamically feasible w.r.t. $Q = \mathbb{R}^{\mathcal{M}}$.

Beard et al. [9] observed that a flux vector is thermodynamically feasible if it does not contain any internal cycles. More formally, they claimed the following result:

Theorem 2.6.1 (Equivalence of Thermodynamic Feasibility) *Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ is thermodynamically feasible if and only if it holds for all $w \in \mathbb{R}^{\mathcal{I}}$ with $\text{sign}(w) \subseteq \text{sign}(v_{\mathcal{I}})$ and $S_{\mathcal{I}} w = 0$ that $w = 0$. □*

Beard et al. [9] gave an unfortunately incomplete proof using oriented matroid theory. In my master thesis I then gave two different mathematically rigorous proofs. In the first proof, I extended the proof by Beard et al. and used matroid theory to fix the hole. The second proof used LP duality and was much shorter. A similar proof was later published by Noor et al. [108].

Thm. 2.6.1 has the nice implication that thermodynamically feasible flux through internal reactions is essentially bounded by flux through exchange reactions, i.e., if all exchange reactions (like nutrient uptake) are bounded, then the flux through every reaction in the network is also bounded. It follows that results obtained by flux variability analysis become more realistic [134, 101].

2.6.2.1 Elementary Modes, Circuits & Matroids

We can reformulate Thm. 2.6.1 using matroid theory as follows:

Corollary 2.6.1 (Equivalence of Thermodynamic Feasibility (Matroid version))

Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ is thermodynamically feasible if and only if there exists no internal circuit C with $C \subseteq \text{sign}(v)$. \square

This is the reason, why in the literature thermodynamic constraints with $Q = \mathbb{R}^{\mathcal{M}}$ are also referred to as *loop-law constraints*. The name originates from electric circuit theory, where internal cycles are called loops. Since a *loop* is something different in matroid theory, I will use the term “loop” only to emphasize that I mean thermodynamic feasibility w.r.t. an unconstrained metabolite space.

We already observed that the internal circuits (cf. Def. 2.5.6) are a subset of the circuits of the flux mode matroid (cf. Def. 2.5.5). By the minimality property of circuits, the following corollary follows immediately:

Corollary 2.6.2 *Let $(\mathcal{M}, \mathcal{R}, S)$ be a metabolic network and \mathcal{C} the circuits of the corresponding flux mode matroid. It holds that every circuit $C \in \mathcal{C}$ that is not an internal circuit is thermodynamically feasible.*

PROOF Since C is a circuit, there exists no circuit $D \subset C$, in particular no internal circuit. The corollary follows by Thm. 2.6.1. \blacksquare

We can formulate a similar result for strongly thermodynamically feasible fluxes:

Proposition 2.6.1 (Equivalence of Strongly Thermodynamic Feasibility)

Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ is strongly thermodynamically feasible if and only if there exists a covector c of the internal cycle matroid (Def. 2.5.6) with $\text{sign}(v_{\mathcal{I}}) = c$.

PROOF By oriented matroid duality, it follows that for every covector c of the internal cycle matroid there exists a vector $\mu \in \mathbb{R}^{\mathcal{M}}$ such that $c = \mu S_{\mathcal{I}}$. Choosing μ as the metabolite potentials yields the result. \blacksquare

2.6.3 With Bounds on Chemical Potentials

Compared to my master thesis, this section now includes a discussion on the causes of uncertainty on data of chemical potentials. We observe that we may not always want to assume box shaped potential spaces. Therefore, I generalized Thm. 2.6.3 accordingly.

With increasing advances in metabolomics, more and more systems biologists want to include metabolite concentration information into metabolic pathway analysis [11, 171, 123, 82, 83, 43, 46, 25].

As already explained in the introduction, we theoretically can compute the chemical potential μ_i of a metabolite $i \in \mathcal{M}$ from its concentration value c_i by the following formula:

$$\mu_i = \mu_i^0 + RT \ln(c_i) \quad (1.3)$$

In practice R, T can be considered as known constants, but this is usually not the case for the equilibrium constants μ_i^0 and the concentration c_i .

2.6.3.1 Uncertainties of Concentration

Although it is possible to measure concentrations using mass spectrometry and other methods [35, 38, 59, 45, 12], such data is always only measured and hence imprecise. Additionally, often only relative concentrations are measured, but for the computation of chemical potentials, we need absolute concentrations [12]. Furthermore, compartmentalization and other factors may distort the measurement. Hence, if we want to understand the capabilities of a metabolic network we should not rely too strongly on the measurement data, but always allow a range of tolerance to account for measurement errors.

This is the case, if we have data on measured concentrations. In many cases however, we do not have any such data at all. Here, we have to allow a wide range of concentrations. Tepper et al. [154] choose 10nM as the minimal concentration (this corresponds to about 1 molecule in a cell of the size of *E. coli*) and 100mM as maximal concentration (maximal total metabolite concentration in *E. coli* has been measured to be 300mM [12]). However, Bennet et al. also measured glutamate concentrations of about 150mM in *E. coli* for growth under glycerol. Also, channeling effects may cause local concentrations to be much higher than 100mM.

Nevertheless, I think that 10nM – 100mM are a good range of default concentration bounds that however may need to be corrected for certain metabolites if further biological knowledge is available.

2.6.3.2 Uncertainties of Equilibrium Constants

Uncertainties in the equilibrium constants of formation (μ_i^0 values) are more complicated than uncertainty of concentrations, because μ_i^0 is an energetic potential. Energetic potentials do not exist as such but are always obtained from potential differences by a normalization method. Hence, there exists no direct way to measure chemical potentials, although we can estimate the μ_i^0 values for nearly all metabolites using the group

contribution method developed by Mavrouniotis [96]. This method was further improved by Jankowski et al. [70] and Noor et al. [107] and is surprisingly precise. However, due to the nature of chemical potentials, estimation errors are dependent on each other.

To understand the dependence of the estimation errors, we briefly explain how these *equilibrium constants of formation* are obtained. Instead of the equilibrium constants of formation for metabolites μ^0 , the *equilibrium constants of reactions* $\Delta\mu^0$ can be measured. One of the main works that list such measurements is the book by Alberty [5] that lists equilibrium constants for many reactions in the central carbon metabolism. By defining certain metabolites to have an equilibrium constant of formation of 0, we can compute the remaining equilibrium constants of formation by using the following relationship:

$$\Delta\mu^0 = \mu^0 S,$$

where S is the stoichiometric matrix. Note, that if we define the equilibrium constant of formation for too many metabolites, we may obtain an inconsistent system. If we define not enough equilibrium constants of formation, then the system becomes under-constrained.

We further observe that we can rewrite the formula for potential differences as follows:

$$\begin{aligned} \Delta\mu &= \mu S \\ &= (\mu^0 + RT \ln(c_i)) S \\ &= \mu^0 S + RT \ln(c_i) S \\ &= \Delta\mu^0 + RT \ln(c_i) S \end{aligned}$$

Consider the example network shown in Fig. 2.4. Assume that we define $\mu_B^0 = 0$. It follows by the reaction $B \rightarrow C$ that $\mu_C^0 \in [1, 2]$. Hence, using reaction $A \rightarrow B + C$ we obtain $\mu_A^0 \in [-1, 1]$. However, it will never be possible that $\mu_C^0 = 2$ and $\mu_A^0 = -1$. It can easily be seen that the effect can accumulate in large reaction networks.

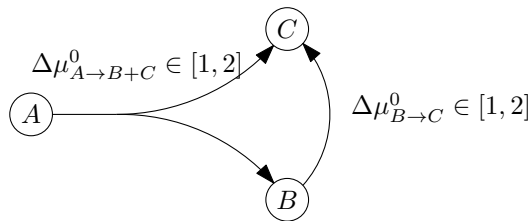


Figure 2.4: A small example network showing that we lose precision if we ignore dependencies on equilibrium constants of formation.

Hence, Noor et al. [107] developed the so-called *component contribution* method to obtain more precise estimates of $\Delta\mu^0$ and thus, more precise estimates on $\Delta\mu$.

We conclude that if we ignore the dependencies between the errors in the estimates for equilibrium constants of formation, we can simply add the error tolerances of the concentrations and equilibrium constants of formation and thus obtain lower and upper bounds on the chemical potentials. On the other hand, we can obtain much smaller error tolerances, if we include the dependencies. In this case however, we are not able to model simply with lower and upper bounds of chemical potentials, but the space Q of feasible potentials will be more complicated. However, Q will still be polyhedral, since we can formulate it as follows:

$$Q = \left\{ \mu \in \mathbb{R}^{\mathcal{M}} : \begin{array}{l} \Delta\mu^{0,\min} \leq S\mu^0 \leq \Delta\mu^{0,\max} \\ c^{\min} \leq c \leq c^{\max} \end{array}, \exists c \in \mathbb{R}^{\mathcal{M}} \right\},$$

where $\Delta\mu^{0,\min}$ and $\Delta\mu^{0,\max}$ denote the tolerance bounds of the equilibrium constants for each reaction, and c^{\min}, c^{\max} denote the logarithm of the minimal resp. maximal allowed concentration for each metabolite.

2.6.3.3 Infeasible Reaction Subsets

A similar theorem to Theorem 2.6.1 can also be stated for cases where $Q \subseteq \mathbb{R}^{\mathcal{M}}$ is a polyhedron. A similar result for box-shaped Q (only constrained by lower and upper bounds) was already discovered by Mavrovouniotis in 1996 [97]. Although Mavrovouniotis was a bit sloppy regarding strict and weak inequalities and the statement of the theorem as well as his proof are not completely correct, he had the correct idea anyway and I presented a new proof in my masters thesis. Here, I will repeat the preliminary steps that lead to the proof and then show the generalized theorem for polyhedral Q .

Essential is the notion of *minimally infeasible sets* (MIS) called bottlenecks by Mavrovouniotis, which in this case can be represented as $s \in \{-, 0, +\}^{\mathcal{I}}$ like the circuits of the internal cycle matroid. Indeed, if $Q = \mathbb{R}^{\mathcal{M}}$, the minimally infeasible sets will be precisely the circuits of the internal cycle matroid. Similar to the circuits, we also use the signed set notation $s = (s^+, s^-)$ with $s^+ = \{i : s_i = +\}$ and $s^- = \{i : s_i = -\}$. In Figure 2.5, you can see a network with an infeasible set that is not an internal circuit.

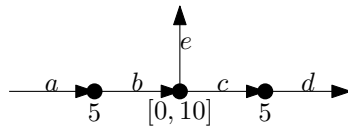


Figure 2.5: Because the potential difference between the left and right metabolite is $5 - 5 = 0$, no flux is possible through reactions b, c at the same time. Hence, $(\{b, c\}, \emptyset)$ is an infeasible set and the only feasible pathways are $(\{a, b, e\}, \emptyset)$ and $(\{c, d\}, \{e\})$.

To capture the notion of infeasible sets more precisely, we introduce the notion of a *subnetwork*. Each vector $s \in \{-, 0, +\}^{\mathcal{I}}$ represents a subnetwork of the original network

in the following way:

Definition 2.6.5 (Subnetwork) Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network.

Given a vector $s \in \{-, 0, +\}^{\mathcal{R}}$ we call the network $\mathcal{N}^s = (\mathcal{M}, \mathcal{R}^s := \underline{s}, S^s)$ the subnetwork of \mathcal{N} w.r.t. s . The stoichiometric matrix S^s of the subnetwork contains the same stoichiometries of the forward reactions as S , i.e., $S_{*i}^s = S_{*i}$ for all $i \in s^+$. For the backward reactions the signs of the stoichiometric coefficients are flipped, i.e., $S_{*i}^s = -S_{*i}$ for all $i \in s^-$.

The internal reactions \mathcal{I}^s are defined by $\mathcal{I}^s := \underline{s} \cap \mathcal{I}$. \square

If we are given a flux v and are interested in properties of the reactions that carry flux, we can simplify things by looking at the subnetwork of flux carrying reactions. This subnetwork is exactly the network \mathcal{N}^s , where $s = \text{sign}(v)$.

Proposition 2.6.2 (Proposition 7 in [99]) Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network. Let v be a flux in \mathcal{N} .

Let $s = \text{sign}(v)$. Then v is a steady-state flux, if and only if $v^s \in \mathbb{R}^s$ with $v_r^s = s_r v_r$ is a steady-state flux in \mathcal{N}^s . Additionally $v^s > 0$. \square

Thus, it suffices to check whether it is thermodynamically feasible if all reactions in \mathcal{N} proceed in forward direction. If it is not feasible, we have found an infeasible direction configuration that also cannot occur in the original network. This infeasible direction configuration will be captured by the notion of *infeasible set* (IS):

Definition 2.6.6 (infeasible set) Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network with metabolite potential space $Q \subseteq \mathbb{R}^{\mathcal{M}}$. A vector $s \in \{-, 0, +\}^{\mathcal{I}}$ is called an infeasible set, if there exists no $\mu \in Q$ such that $\mu S_{*i}^s < 0$ for all $i \in \mathcal{I}^s$ of the subnetwork \mathcal{N}^s . \square

These infeasible sets behave as one naturally would expect infeasible sets to behave. I.e., if you make the set larger, it still stays infeasible:

Proposition 2.6.3 (Proposition 8 in [99]) Let $v \in \{-, 0, +\}^{\mathcal{I}}$ be an infeasible set in a metabolic network \mathcal{N} . Let $v \subseteq w \in \{-, 0, +\}^{\mathcal{I}}$. Then w is also an infeasible set. \square

Assume now, we are given a flux vector v . If v is thermodynamically feasible w.r.t. Q , all internal reactions where v has positive flux must have negative potential differences. Translated to the theory of infeasible sets, $\text{sign}(v_{\mathcal{I}})$ must not be an infeasible set. As a consequence of Proposition 2.6.3 we now only have to check for each (inclusion) minimal infeasible set s whether $s \subseteq \text{sign}(v_{\mathcal{I}})$.

Theorem 2.6.2 (Theorem 3 in [99]) Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network with metabolite potential space $Q \subseteq \mathbb{R}^{\mathcal{M}}$. Then there exists a collection of minimally infeasible sets $\mathcal{C} \subseteq \{-, 0, +\}^{\mathcal{I}}$ such that a flux vector v is thermodynamically feasible w.r.t. to Q if and only if $s \not\subseteq \text{sign}(v_{\mathcal{I}})$ for every $s \in \mathcal{C}$. \square

In the case, where $Q = \mathbb{R}^M$, we know that the minimal infeasible sets are the circuits of the internal cycle matroid. Since oriented matroids are well studied, we know how to find those circuits and how to check whether $\text{sign}(v_{\mathcal{I}})$ of a flux v contains such a circuit. The following Theorem 2.6.3 answers the generalized question. It is a generalization of Theorem 4 in [99], which was originally stated by Mavrovouniotis [97] in a similar fashion in 1996. There, we only analyzed the case where the potential space was given by $Q = \{\mu \in \mathbb{R}^M : \ell \leq \mu \leq u\}$ with $\ell, u \in \mathbb{R}^M$.

Since we also want to model dependencies between uncertainties, we generalize the result to polyhedral potential spaces $Q \subseteq \mathbb{R}^M$. Note that most of the new complexity is due to the fact that Q may be unbounded, which was not allowed in [99]. We will apply the following theorem only on subnetworks \mathcal{N}^s for flux vectors w with $s = \text{sign}(w_{\mathcal{I}})$. Hence, the set of reactions \mathcal{R} consists only of internal reaction and hence it makes sense to consider potential differences for them.

Theorem 2.6.3 *Let $\mathcal{N} = (\mathcal{M}, \mathcal{R}, S)$ be a metabolic network with polyhedral potential space $Q \neq \emptyset$. Then the following are equivalent:*

1. *There exists a $\mu \in Q$ such that $\mu S_i < 0 \forall i \in \mathcal{R}$.*
2. *For all $v \in \mathbb{R}^{\mathcal{R}}$ with $v \geq 0, \sum_{i \in \mathcal{R}} v_i = 1$ it holds that there exists a $\mu \in Q$ such that $\mu S v < 0$.*

PROOF Since Q is a polyhedron, we can assume that there exists a matrix A and vector b s.t. $Q = \{\mu \in \mathbb{R}^M : \mu A \leq b\}$.

The first statement holds if and only if the following minimization problem has a solution less than 0:

$$\min\{\alpha : \alpha - \mu S_i \geq 0 \forall i \in \mathcal{R}, \mu A \leq b, \alpha \geq -1\} \quad (2.7)$$

Since $Q \neq \emptyset$, it follows that (2.7) is feasible, since we can choose α arbitrary large. Clearly, (2.7) is also bounded.

Hence, it follows from the duality theorem that the solution value is equal to

$$\begin{aligned} & \max\{bx - z : -Sv + Ax = 0, \mathbf{1}v + z = 1, v \geq 0, x \leq 0, z \geq 0\} \\ & = \max\{\max\{bx : Ax = Sv, x \leq 0\} - z : \mathbf{1}v + z = 1, v \geq 0, z \geq 0\} \end{aligned} \quad (2.8)$$

Claim 2.6.1 *It holds that*

$$\begin{aligned} & \max\{\max\{bx : Ax = Sv, x \leq 0\} - z : \mathbf{1}v + z = 1, v \geq 0, z \geq 0\} \\ & = \max\{\min\{\mu S v : \mu A \leq b\} - z, \mathbf{1}v + z = 1, v \geq 0, z \geq 0\} \end{aligned} \quad (2.10)$$

PROOF The dual LP of $\max\{bx : Ax = Sv, x \leq 0\}$ is $\min\{\mu S v : \mu A \leq b\}$. Let $(\tilde{v}, \tilde{z}, \tilde{x})$ be an optimal solution of (2.9). It follows that $\max\{bx : Ax = S\tilde{v}, x \leq 0\}$ is feasible and bounded. We now show both directions of the equality separately.

\leq : By the duality theorem, we obtain

$$\begin{aligned} & \max\{\max\{bx : Ax = Sv, x \leq 0\} - z, \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \\ &= \max\{bx : Ax = S\tilde{v}, x \leq 0\} - \tilde{z} \\ &= \min\{\mu S\tilde{v} : \mu A \leq b\} - \tilde{z} \\ &\leq \max\{\min\{\mu Sv : \mu A \leq b\} - z, \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \end{aligned}$$

\geq : We observe that by duality $\min\{\mu S\tilde{v} : \mu A \leq b\}$ is bounded and feasible. Clearly, $\inf\{\mu Sv : \mu A \leq b\}$ is also always feasible and hence,

$$f(v, z) := \max(\min\{\mu S\tilde{v} : \mu A \leq b\} - \tilde{z}, \inf\{\mu Sv : \mu A \leq b\} - z)$$

is a continuous function defined on all $(v, z) \in \mathbb{R}^{\mathcal{R}} \times \mathbb{R}$ and $\{(v, z) \in \mathbb{R}^{\mathcal{R}} \times \mathbb{R} : \mathbb{1}v + z = 1, v \geq 0, z \geq 0\}$ is a compact set. Hence, the optimum of $\max\{f(v) : \mathbb{1}v + z = 1, v \geq 0, z \geq 0\}$ exists. Let (\bar{v}, \bar{z}) be an optimal solution. Clearly, $f(\tilde{v}, \tilde{z}) \leq f(\bar{v}, \bar{z})$ hence, $f(\bar{v}, \bar{z}) = \min\{\mu S\tilde{v} : \mu A \leq b\} - \bar{z}$. By the duality theorem, it follows that

$$\begin{aligned} & \max\{\min\{\mu Sv : \mu A \leq b\} - z, \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \\ &\leq \max\{f(v) : \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \\ &= f(\bar{v}, \bar{z}) \\ &= \min\{\mu S\bar{v} : \mu A \leq b\} - \bar{z} \\ &= \max\{bx : Ax = S\bar{v}, x \leq 0\} - \bar{z} \\ &\leq \max\{\max\{bx : Ax = Sv, x \leq 0\} - z, \mathbb{1}v + z = 1, v \geq 0, z \geq 0\}. \end{aligned}$$

Claim 2.6.2 *It holds that*

$$\begin{aligned} 0 &> \max\{\min\{\mu Sv : \mu A \leq b\} - z, \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \\ &\Leftrightarrow 0 > \max\{\min\{\mu Sv : \mu A \leq b\}, \mathbb{1}v = 1, v \geq 0\} \end{aligned}$$

PROOF We show both directions separately:

\Rightarrow : By choosing $z = 0$ we reduce the feasible domain and thus we obtain

$$\begin{aligned} 0 &> \max\{\min\{\mu Sv : \mu A \leq b\} - z, \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \\ &\Rightarrow 0 > \max\{\min\{\mu Sv : \mu A \leq b\}, \mathbb{1}v = 1, v \geq 0\} \end{aligned}$$

\Leftarrow : If the optimum of (2.10) is attained for $z = 0$, the claim follows immediately. Hence, assume the optimum is attained with $z > 0$. We observe that $z \leq 1$ and thus it follows that

$$\begin{aligned} 0 &> \max\{\min\{\mu Sv : \mu A \leq b\}, \mathbb{1}v = 1, v \geq 0\} \\ &\Rightarrow 0 \geq \max\{\min\{\mu Sv : \mu A \leq b\}, \mathbb{1}v + z = 1, v \geq 0\} && \text{(scaling)} \\ &\Rightarrow 0 > \max\{\min\{\mu Sv : \mu A \leq b\} - z, \mathbb{1}v + z = 1, v \geq 0\} \end{aligned}$$

Hence, the claim follows. ■

We summarize:

$$\begin{aligned}
 & \text{There exists a } \mu \in Q \text{ such that } \mu S_i < 0 \forall i \in \mathcal{R} \\
 \Leftrightarrow & 0 > \min\{\alpha : \alpha - \mu S_i \geq 0 \forall i \in \mathcal{R}, \mu A \leq b, \alpha \geq -1\} \\
 \Leftrightarrow & 0 > \max\{\max\{bx : Ax = Sv, x \leq 0\} - z : \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \quad (\text{LP duality}) \\
 \Leftrightarrow & 0 > \max\{\min\{\mu Sv : \mu A \leq b\} - z, \mathbb{1}v + z = 1, v \geq 0, z \geq 0\} \quad (\text{Claim 2.6.1}) \\
 \Leftrightarrow & 0 > \max\{\min\{\mu Sv : \mu A \leq b\}, \mathbb{1}v = 1, v \geq 0\} \quad (\text{Claim 2.6.2})
 \end{aligned}$$

which concludes the proof. ■

In the theory of infeasible sets, we can restate Theorem 2.6.3 as follows:

Corollary 2.6.3 *Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network with polyhedral metabolite potential space Q .*

A steady-state flux $v \in \mathbb{R}^{\mathcal{R}}$ is thermodynamically feasible in \mathcal{N} w.r.t. Q if and only if for all $w \in \mathbb{R}^{\mathcal{I}}$ with $\text{sign}(w) \subseteq \text{sign}(v_{\mathcal{I}})$ and $w \neq 0$ exists a $\mu \in Q$ such that $\mu S_{\mathcal{I}} w < 0$. \square

So the main result of Corollary 2.6.3 is, that if v is not thermodynamically feasible, we can find a witness that proves infeasibility. In the cases of circuits these are the internal circulations w , which satisfy $S_{\mathcal{I}} w = 0$. Now it is easy to see, that all internal circulations are also infeasible sets.

Another special case is used by Nolan, Lee, Xu, Boghihian et al. [105, 170, 16]. They are only given potentials of boundary metabolites. In this case, all minimal infeasible sets are elementary modes where the exchange reactions have been removed.

For general metabolite concentration bounds we observe:

Observation 2.6.1 *The proof of Theorem 2.6.3 also gives us an LP formulation that either proves thermodynamic feasibility of a flux vector v or returns a witness that proves infeasibility. We only need to solve the LP (2.8) on the subnetwork $\mathcal{N}^{\text{sign}(v_{\mathcal{I}})}$. If the solution value is less than zero we obtain feasibility. If the solution value is larger or equal to zero, we obtain a witness for infeasibility (an infeasible set). \square*

Chapter 3

Computational Complexity

Abstract The computational complexity of the question whether there exists a thermodynamically feasible flux through a given reaction depends highly on how the decision problem is formulated. In this chapter I analyze the complexity of this decision problem depending on

- the formalism used for thermodynamic constraints,
- whether the target reaction is internal or external,
- what kind of reactions can be irreversible,
- and if bounds on metabolite concentrations are given.

The chapter extends work from my master thesis, where I showed that finding a positive thermodynamically feasible flux through a given internal reaction is NP-hard w.r.t. to weak thermodynamic constraints if the network contains irreversible internal reactions.

3.1 Introduction

In the analysis of metabolic networks an essential question is whether a given reaction can carry positive flux or not. Reactions that cannot carry positive flux are usually indicators of modeling errors. If we only impose the steady-state condition (i.e., the flux space is a polyhedron), we can answer the question whether a given reaction can carry positive flux by solving a linear program.

When we add thermodynamic constraints, the computational complexity of this decision problem becomes unclear. Hence, we now consider the complexity of the THERMOFLUX problem (Prob. 3.1.1). Depending on what kinds of values we allow for the input parameters and what formalism of thermodynamic constraints we use, the complexity of the

problem changes. An introduction to computational complexity theory can be found in [119].

Problem 3.1.1 (Thermoflux)

Given:

- metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S \in \mathbb{Q}^{M \times R})$
- irreversible reactions $\text{Irrev} \subseteq \mathcal{R}$
- upper and lower potential bounds μ^{\min}, μ^{\max}
- objective reaction $r \in R$

Question: Does there exist a thermodynamically feasible w.r.t. potential bounds μ^{\min}, μ^{\max} flux v with $v_{\text{Irrev}} \geq 0$ and $v_r > 0$? □

We consider the following 4 independent restrictions for the input / formalisms of thermodynamic constraints (the numbering will be used to reference the corresponding combination of restrictions):

1. Choice of r
 - (a) r is element of the internal reactions ($r \in \mathcal{I}$)
 - (b) r is element of the exchange reactions ($r \in \mathcal{E}$)
2. Irreversibilities
 - (a) all reactions can be irreversible ($\text{Irrev} \subseteq \mathcal{R}$)
 - (b) all internal reactions are reversible ($\text{Irrev} \subseteq \mathcal{E}$)
 - (c) all reactions are reversible ($\text{Irrev} = \emptyset$)
3. Thermodynamics formalism
 - (a) weak thermodynamic constraints (Def. 2.6.2)
 - (b) strong thermodynamic constraints (Def. 2.6.1)
 - (c) relaxed thermodynamic constraints (Def. 2.6.3)
4. Bounds
 - (a) without bounds on metabolite potentials ($\mu^{\min} = -\infty, \mu^{\max} = \infty$)
 - (b) with bounds on metabolite potentials ($\mu^{\min}, \mu^{\max} \in \mathbb{R}_{\infty}^{\mathcal{M}}, \mu^{\min} \leq \mu^{\max}$)

3.2 Results

The complexity results are summarized in Tab. 3.1. First of all we remark that all variants of the THERMOFLUX problem are in NP, because a YES-answer can always be proven by giving a flux vector and corresponding metabolite potentials.

We observe that adding any kind of a-priori thermodynamic information, either in form of bounds on metabolite potentials, or irreversibilities of internal reactions makes the THERMOFLUX problem hard. However, if only limited thermodynamic information is added the problem seems to become easier.

We also note that it is unclear at the moment if we can incorporate irreversibility constraints for exchange reactions and still solve the THERMOFLUX problem in polynomial time.

Table 3.1: Summary of the complexity results for the THERMOFLUX problem.

| | | Irrev $\subseteq \mathcal{R}$ | | Irrev $\subseteq \mathcal{E}$ | | Irrev = \emptyset | | | |
|---------|------------|-------------------------------|---------------------|-------------------------------|---------------------|---------------------|---------------------|----|----|
| | | $r \in \mathcal{I}$ | $r \in \mathcal{E}$ | $r \in \mathcal{I}$ | $r \in \mathcal{E}$ | $r \in \mathcal{I}$ | $r \in \mathcal{E}$ | | |
| weak | wo. bounds | NP-hard | P | ? | P | P | P | 4a | 3a |
| | w. bounds | NP-hard | NP-hard | NP-hard | NP-hard | NP-hard | NP-hard | 4b | |
| strong | wo. bounds | NP-hard | NP-hard | ? | P | P | P | 4a | 3b |
| | w. bounds | NP-hard | NP-hard | NP-hard | NP-hard | NP-hard | NP-hard | 4b | |
| relaxed | wo. bounds | P | P | P | P | P | P | 4a | 3c |
| | w. bounds | NP-hard | NP-hard | NP-hard | NP-hard | NP-hard | NP-hard | 4b | |
| | | 1a | 1b | 1a | 1b | 1a | 1b | | |
| | | 2a | | 2b | | 2c | | | |

3.3 Reductions

To show the complexity results of Tab. 3.1, we do not show the complexity for each case separately, but employ the fact that we can show some reduction between the problems.

3.3.1 Trivial Reductions

The following 3 reductions are trivial. We use placeholders x, y, z to denote any choice of formulation and $a \preceq b$ means that if b can be solved in polynomial time, then a can be solved in polynomial time (a is easier or equal to b).

- $1b2x3y4z \preceq 1a2x3y4z$ by declaring the exchange reaction an internal reaction and adding a new fully coupled exchange reaction.
- $1x2c3y4z \preceq 1x2b3y4z \preceq 1x2a3y4z$ ($2c$ is a special case of $2b$ which is a special case of $2a$)
- $1x2y3z4a \preceq 1x2y3z4b$ ($4a$ is a specialization of $4b$)

3.3.2 $1x2y3a4a$ is Equivalent to $1x2y3b4a$ for $x \in \{a, b\}$, $y \in \{b, c\}$

We observe that if a reaction $r \in \mathcal{R}$ can carry positive strongly thermodynamically feasible steady-state flux (that satisfies the flux bounds), then r can also carry positive weakly thermodynamically feasible steady-state flux (that satisfies the flux bounds), because the constraints are weaker in the latter case.

Hence, we only have to show the other direction. We consider the flux mode matroid \mathcal{O} represented by the metabolic network \mathcal{N} . We observe that $\mathcal{T} := \mathcal{O} \setminus \mathcal{E}$ is the internal cycle matroid. Assume that r can carry weakly thermodynamically feasible steady-state flux (that satisfies the flux bounds). Let v be a minimal sign-vector of the flux vector realizing this (i.e., v is a circuit of \mathcal{O}). Define

$$A := \bigcup_{\substack{C \text{ circuit of } \mathcal{T} \\ v \cap C \neq \emptyset}} \underline{C}$$

$$B := \{s \in \underline{v} : s \notin \underline{C} \forall C \text{ circuit of } \mathcal{T}\}$$

Clearly, $\underline{v} \subseteq A \cup B \cup \mathcal{E}$. We observe the following

Lemma 3.3.1 *Let \mathcal{N}' be the metabolic network that only contains the reactions in $A \cup B \cup \mathcal{E}$, where A, B are defined as above with exchange reactions $\mathcal{E}' := \mathcal{E} \cup B$. Then it holds that if $v' \in \mathbb{R}^{A \cup B \cup \mathcal{E}}$ is strongly thermodynamically feasible in \mathcal{N}' , then $w \in \mathbb{R}^{\mathcal{R}}$ with $w_{A \cup B \cup \mathcal{E}} = v'$ and $w_s = 0$ for all $s \notin A \cup B \cup \mathcal{E}$ is strongly thermodynamically feasible in \mathcal{N} .*

PROOF Let \mathcal{O}' be the oriented matroid represented by \mathcal{N}' . It follows that $\mathcal{O}' = \mathcal{O}|_{A \cup B \cup \mathcal{E}}$. Let $\mathcal{T}' = \mathcal{O}'|_A$ be the internal cycle matroid of \mathcal{N}' . Since v' is strongly thermodynamically feasible in \mathcal{N}' , there exists a covector $c' \in \{-, 0, +\}^A$ of \mathcal{T}' with $\text{sign}(v'_A) = c'$.

By Prop. 2.5.4 it follows that A is the union of connected components of \mathcal{T} . Hence, by Prop. 4.2.1 of [118], it follows that A is a separator in \mathcal{T} . Thus, c' is also a covector of \mathcal{T} .

For each $r \in B$, we observe that $\{r\}$ is a cocircuit of \mathcal{T} (also called coloop). By composition (Prop. 2.5.2) it follows that $\text{sign}(v_B)$ is a covector in \mathcal{T} .

Again, by composition it follows that $c = c' \circ \text{sign}(v_B)$ is a covector of \mathcal{T} . We observe that $c_A := c'$, $c_B := \text{sign}(v_B)$ and $c_{\mathcal{T} \setminus (A \cup B)} = 0$. Hence, $\text{sign}(w_{\mathcal{T}}) = c$ and since w is clearly a steady-state flux, w is strongly thermodynamically feasible. ■

It follows that we can assume w.l.o.g. that for every $s \in \mathcal{I} \setminus \underline{v}$ there exists a circuit c that contains s and at least one element of v . With Alg. 1 we now construct from v a vector v' of \mathcal{O} with $\underline{v}' \supseteq \mathcal{I}$ that contains no internal cycles. It follows that v' is strongly thermodynamically feasible. Since v' is a vector of \mathcal{O} it also follows that there exists a steady-state flux vector with exactly the signs of v' .

Algorithm 1 Algorithm to compute a strongly thermodynamically feasible flux vector from a weakly thermodynamically feasible flux vector.

Input: vector v of \mathcal{O} for which no circuit w of \mathcal{T} with $w \subseteq v$ exists.

while $\exists s \in \mathcal{I} \setminus v$ **do**

Let c be a circuit of \mathcal{T} with $\underline{c} \cap v \neq \emptyset \neq \underline{c} \setminus v$ and $\underline{c} \setminus v$ minimal.

if \exists circuit w of $\mathcal{T} : w \subseteq v \circ c$ **then**

$c := -c$

end if

Set $v := v \circ c$.

end while

Return: v

Proposition 3.3.1 *Let be given a signed vector v for which there exists no signed vector w of \mathcal{T} with $w \subseteq v$. Then, Algorithm 1 computes a signed vector v' with $v' \supseteq v$ and $v' \supseteq \mathcal{I}$. Furthermore, v' satisfies that there exists no signed vector w of \mathcal{T} with $w \subseteq v'$.*

PROOF We show the following loop invariant of the while loop:

Loop invariant: v is a vector of \mathcal{O} for which no circuit w of \mathcal{T} with $w \subseteq v$ exists.

We have to show that the loop invariant also holds at the end of each iteration. We observe that by construction of A , there always exists a circuit c of \mathcal{T} with $\underline{c} \cap v \neq \emptyset$ and $\underline{c} \setminus v \neq \emptyset$ if there exists an $s \in \mathcal{I} \setminus v$.

Hence, the existence of such a circuit with the minimality condition follows immediately, since the number of circuits is finite.

We note that it is unclear whether the composition with c or $-c$ will yield a vector that does not contain internal cycles. However, the following claim shows that at least one of the two options will not produce internal cycles:

Claim 3.3.1 *If c is a circuit of \mathcal{T} with $\underline{c} \cap v \neq \emptyset$, $\underline{c} \setminus v \neq \emptyset$ and $\underline{c} \setminus v$ minimal, then either $v \circ c$ or $v \circ -c$ does not contain an internal cycle.*

PROOF Let $s \in \underline{c} \setminus v$. W.l.o.g. assume that $s \in c^+$. Assume there exist internal circuits $w \subseteq v \circ c$ and $w' \subseteq v \circ -c$. By minimality of $\underline{c} \setminus v$ it follows that $w \setminus v = w' \setminus v = \underline{c} \setminus v$. Hence, it follows that $s \in w^+ \subseteq (v \circ c)^+$ and $s \in w'^- \subseteq (v \circ -c)^-$. By circuit axiom (C3) there exists a circuit \tilde{w} with

$$\begin{aligned} \tilde{w}^+ &\subseteq (w^+ \cup w'^+) \setminus \{s\} \subseteq ((v \circ c)^+ \cup (v \circ -c)^+) \setminus \{s\} \\ \tilde{w}^- &\subseteq (w^- \cup w'^-) \setminus \{s\} \subseteq ((v \circ c)^- \cup (v \circ -c)^-) \setminus \{s\} \end{aligned}$$

It follows that $\tilde{w} \setminus v \subseteq \underline{c} \setminus v$. By minimality of c it follows that $\tilde{w} \setminus v = \emptyset$. By construction (composition), it follows that $\tilde{w} \subseteq v$. Hence, v contained an internal circuit, which is a contradiction. ■

It follows, that at the end of the loop iteration, v is a vector of \mathcal{O} for which no circuit w of \mathcal{T} with $w \subseteq v$ exists.

Finally, we observe that the support of v is growing with each iteration. Hence, the algorithm will terminate. ■

Since Alg. 1 is correct, it follows that for every weakly thermodynamically feasible steady-state flux that satisfies the irreversibilities in the exchange reactions, we can construct a strongly thermodynamically feasible steady-state flux that satisfies the irreversibilities in the exchange reactions. A close look at the construction reveals that all compositions are performed on internal reactions. Hence, we can even conclude that there exists a strongly thermodynamically feasible flux vector with exactly the same exchange fluxes as the weakly thermodynamically feasible flux vector.

3.4 Problems in P

3.4.1 1a2c3a4a is in P

We observe that, since no reaction is irreversible, the supports of steady-state flux vectors are precisely the cycles of the flux mode matroid (Def. 2.5.5). By Cor. 2.6.1 it follows that we only have to check if there exists a circuit that contains the target reaction r and an exchange reaction $s \in \mathcal{E}$. We recall Prop. 2.5.4 that states that there exists a circuit C containing s and r if and only if s and r are contained in the same connected component. This is a sufficient condition, since all reactions are reversible and $C, -C$ are both circuits. Since the connected components can be computed in polynomial time using the algorithm by Krogdahl [28, 81], the result follows. For more details see Sec. 6.3.3.3, where the same method is used to compute flux modules.

This result relates to the fact that the 2-vertex disjoint paths problem on undirected graphs is solvable in polynomial time [145, 131]. We remark that the complexity of the problem remains unknown if the set of irreversible reactions is small, or if it only contains exchange reactions.

3.4.2 1b2x3a4a is in P

For Problem 1b2a3a4a we can use the cycle elimination method, see also Thm. 4.5.1. Hence, 1b2b3a4a and 1b2c3a4a are also in P.

3.4.3 1b2b3b4a is in P

Fleming et al. showed in [48] that a non-linear reformulation of the strict thermodynamic flow problem can be reformulated as a convex optimization problem. Given a steady-state flux $v \in \mathbb{R}^{\mathcal{R}}$ they compute a thermodynamically feasible flux (w.r.t. to their

non-linear reformulation) \tilde{v} with $\tilde{v}_\varepsilon = v_\varepsilon$. In my masters thesis I showed that the non-linear reformulation by Fleming et al. [48] is equivalent to the strong formulation of thermodynamic constraints (Sec. 5.3 in [99]). It follows that to decide if strongly thermodynamically feasible flux is possible through an exchange reaction, we only have to solve the FBA problem.

However, it should be noted that the addition of sign-constraints to internal reactions will break the method. The hardness of $1b2a3b4a$ shows this formally.

Furthermore, we observe that while the primal of the non-linear reformulation does not contain variables for potentials, the dual does. However, if we would add additional constraints to the dual to restrict the range of the potentials, we break the steady-state assumption of the primal. Hence, the method cannot be extended to finding a strongly thermodynamically feasible flux vector that also satisfies metabolite bounds. Indeed, there exist networks where no feasible pathway exists (due to the strong thermodynamic constraints, also the zero flux can be infeasible).

3.4.4 $1x2y3c4a$ is in P

We can set all metabolite potential to 0. It follows that every potential difference is 0. Hence, flux in any direction is feasible for the relaxed thermodynamic constraint. It follows that FBA with relaxed thermodynamic constraints is just FBA. Since FBA can be solved with linear programming in polynomial time, also this problem can be solved in polynomial time.

3.4.5 $1x2c3b4a$ is in P

Since $1a2c3b4a$ and $1a2c3a4a$ are equivalent (Sec. 3.3.2) and $1a2c3a4a$ is in P, it follows that also $1a2c3b4a$ is in P. By declaring the exchange reaction an internal reaction and coupling it to a new exchange reaction, we observe that also $1b2c3b4a$ is in P.

3.5 NP-hard problems

3.5.1 $1a2a3a4x$ is NP-hard

In my master thesis ([99], Thm. 5; [101], Thm. 1) I showed that $1a2a3a4a$ is NP-hard by reduction to the 2-vertex disjoint path problem on directed graphs [50]. By one of the trivial reductions it follows that $1a2a3a4b$ is also NP-hard.

3.5.2 $1x2a3b4a$ and $1x2y3b4b$ are NP-hard

We show hardness of $1b2a3b4a$ and $1b2c3b4a$ using similar constructions in parallel. The hardness of the other variants follows using the trivial reductions mentioned above.

We do a reduction from 3SAT. Let f be a 3SAT formula with variables x_1, \dots, x_n and clauses C_1, \dots, C_m . For each variable $x_i, i = 1, \dots, n$ we construct a subnetwork as shown in Fig. 3.1. It consists of metabolites $M_i^+ := \{m_{ij}^+ : x_i \in C_j\}$ and metabolites $M_i^- := \{m_{ij}^- : \bar{x}_i \in C_j\}$ and one reversible exchange reaction r_i with stoichiometric coefficient 1 for each $m \in M_i^+$ and stoichiometric coefficient -1 for each $m \in M_i^-$.

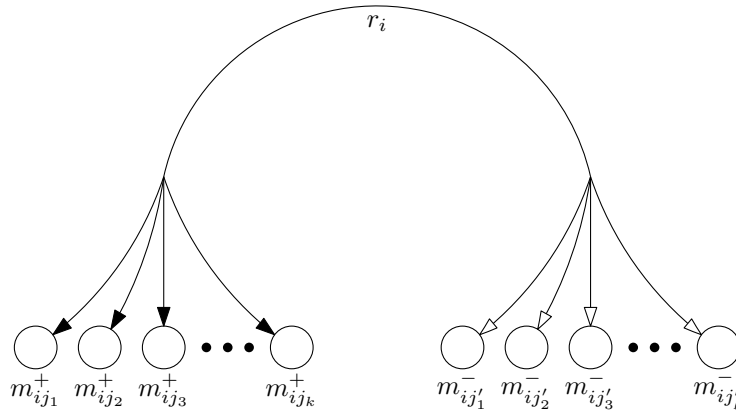


Figure 3.1: The subnetwork for variable x_i , where x_i is used in clauses $C_{j_1}, C_{j_2}, \dots, C_{j_k}$ and \bar{x}_i is used in clauses $C_{j'_1}, C_{j'_2}, \dots, C_{j'_l}$. It is represented by the reversible exchange reaction r_i that transforms metabolites $M_i^+ = \{m_{ij_1}^+, m_{ij_2}^+, \dots, m_{ij_k}^+\}$ into metabolites $M_i^- = \{m_{ij'_1}^-, m_{ij'_2}^-, \dots, m_{ij'_l}^-\}$ and vice versa. The reverse direction is marked with white arrows.

Before we construct the subnetworks for the clauses, we define the flux-decoupling motif shown in Fig. 3.2 for $1b2a3b4a$ and Fig. 3.3 for $1b2c3b4b$.

We observe that this motif decouples the input flux rate from the output flux rate by still keeping the direction of the flux. Furthermore, the motif will be connected to the rest of the network using exchange reactions. Hence, it is sufficient to analyze the thermodynamically feasible fluxes inside the motif and only care about the steady-state condition on a system wide perspective.

Lemma 3.5.1 *Let T be the space of strictly thermodynamically feasible fluxes through the network shown in Fig. 3.2 (Fig. 3.3). Then it holds that $r_i \rightarrow_T r_o$ and $r_o \rightarrow_T r_i$ (according to Def. 2.1.9), i.e., $\text{sign}(v_{r_i}) = \text{sign}(v_{r_o})$.*

PROOF We first observe that metabolites o and o' always have the same potential. In the case of Fig. 3.3 this is trivial. In the case of Fig. 3.2 this is clearly induced by the two irreversible reactions (drawn just with one arrow).

It follows by the strong thermodynamic constraint that if o' has outflow through r'_o , then also o has to have outflow through r_o and vice versa. The same holds for inflow. We conclude that i has inflow/outflow through r_i if and only if o has outflow/inflow through r_o . ■

Lemma 3.5.2 *For all $a, b \in \mathbb{R}$ with $\text{sign}(a) = \text{sign}(b) = \text{sign}(a-b)$ there exists a strongly thermodynamically feasible flux through the networks shown in Fig. 3.2 and Fig. 3.3 with $v_{r_i} = a$ and $v_{r_o} = b$.*

PROOF Define $\mu_o = \mu_{o'} = 0$, $v_{h_2} = v_{r_o} = b$, $v_{h_1} = v_{r_{o'}} = a - b$ and $v_{r_i} = a$. For the irreversible reactions (if they exist), we define a flux of 0. It is easy to verify that this is a steady-state flux distribution.

We now only have to find feasible potentials:

Case $a > 0$: Define $\mu_i = 1$. It follows that $\mu_o - \mu_i = \mu_{o'} - \mu_i = -\mu_i < 0$ and hence, v is strongly thermodynamically feasible.

Case $a < 0$: Define $\mu_i = -1$. It follows that $\mu_o - \mu_i = \mu_{o'} - \mu_i = -\mu_i > 0$ and hence, v is strongly thermodynamically feasible.

Case $a = 0$: Define $\mu_i = 0$. It follows that $\mu_o - \mu_i = \mu_{o'} - \mu_i = -\mu_i = 0$ and hence, v is strongly thermodynamically feasible, since also $b = a - b = 0$. ■

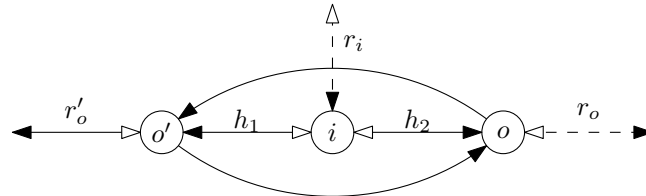


Figure 3.2: Decoupling motif with irreversible internal reactions. The reactions r_o , r'_o and r_i are exchange reactions. The decoupling motif has two metabolites i, o which will be used as its interface when used in the construction. This motif will be connected to the rest of the network using exchange reactions, marked here as dashed arrows. The reverse direction is marked with a white arrow.

For each clause $C_j = (l_a \vee l_b \vee l_c)$, where $l_a, l_b, l_c \in \{x_1, \dots, x_n\} \cup \{\bar{x}_1, \dots, \bar{x}_n\}$ we construct a sub-network as shown in Fig. 3.4.

We observe the following two properties:

Lemma 3.5.3 *Let $a_1, a_2, a_3, b \in \mathbb{R}$ be arbitrary but fixed. If there exists a $k \in \{1, 2, 3\}$ with $a_k > b > 0$, then there exists a strongly thermodynamically feasible flux vector v with $v_{r_{\text{ex}}} = b$ and $v_{s_l r_{i_l}} = a_l$ for $l = 1, 2, 3$.*

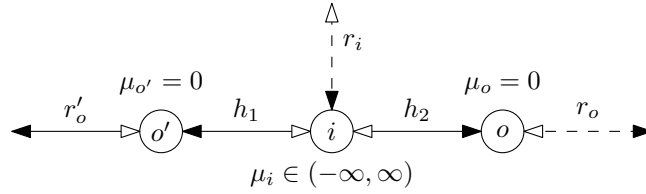


Figure 3.3: Decoupling motif with metabolite potentials. The reactions r_o , r'_o and r_i are exchange reactions. The decoupling motif has two metabolites i, o which will be used as its interface when used in the construction. This motif will be connected to the rest of the network using exchange reactions, marked here as dashed arrows. The reverse direction is marked with a white arrow.

PROOF Assume w.l.o.g. that $k = 1$. Define $\varepsilon := \min \left\{ \frac{b}{4}, \frac{a_1 - b}{4} \right\}$. Define

$$\begin{aligned} w_2 &:= \max \left\{ \min \left\{ \frac{a_2}{2}, \varepsilon \right\}, -\varepsilon \right\} \\ w_3 &:= \max \left\{ \min \left\{ \frac{a_3}{2}, \varepsilon \right\}, -\varepsilon \right\} \\ w_1 &:= b - w_2 - w_3 \end{aligned}$$

We observe that $\frac{b}{2} = b - 2\frac{b}{4} \leq w_1 \leq b + 2\frac{a_1 - b}{4} = \frac{a_1 + b}{2} < a_1$. Hence, by Lemma 3.5.2 there exists a thermodynamically feasible flux v with

$$\begin{aligned} v_{g_{i_1, j}} &= w_1 \\ v_{g_{i_2, j}} &= w_2 \\ v_{g_{i_3, j}} &= w_3 \end{aligned}$$

By the steady-state condition it follows that $v_{r_{\text{ex}}} = w_1 + w_2 + w_3 = b$. ■

Lemma 3.5.4 *There exists no strongly thermodynamically feasible flux vector v with $v_{r_{\text{ex}}} > 0$ and $v_{s_k r_{i_k}} \leq 0$ for $k = 1, 2, 3$.*

PROOF Assume there exists a strongly thermodynamically feasible flux vector v with $v_{r_{\text{ex}}} > 0$ and $v_{s_k r_{i_k}} \leq 0$ for $k = 1, 2, 3$. By Lemma 3.5.1 it follows that $v_{g_{i_k, j}} \leq 0$. The result follows from the steady-state condition on c_j . ■

Finally, we add one exchange reaction r_{ex} that consumes each of the metabolites $c_j, j = 1, \dots, m$ at the same rate.

Theorem 3.5.1 *There exists a strongly thermodynamically feasible flux vector v with $v_{r_{\text{ex}}} > 0$ if and only if f is satisfiable.*

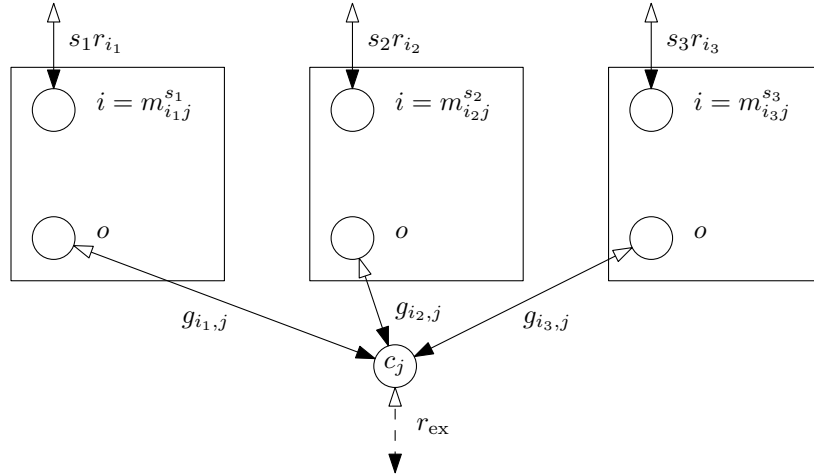


Figure 3.4: The subnetwork for clause C_j . The boxes denote instances of the decoupling motif, where only the nodes i and o are marked. In the upper row, the input metabolites (i) correspond to the literals of the clause. For example if $C_j = (x_{i_1} \vee \bar{x}_{i_2} \vee x_{i_3})$ then $m_{i_1 j}^{s_1} = m_{i_1 j}^+$, $m_{i_2 j}^{s_2} = m_{i_2 j}^-$, $m_{i_3 j}^{s_3} = m_{i_3 j}^+$. The reactions connecting the output-metabolite (o) of the decoupling motif are connected to the metabolite c_j with exchange reactions. The reaction r_{ex} symbolizes the exchange reaction that will be added in the last step to take up flux from c_j .

PROOF \Rightarrow : Let v be a strongly thermodynamically feasible flux vector v with $v_{r_{\text{ex}}} > 0$. By Lemma 3.5.4 it follows for each clause C_j , $j = 1, \dots, m$ that at least one of the input reactions $s_1 r_{i_1}$, $s_2 r_{i_2}$, $s_3 r_{i_3}$ must carry positive flux. It follows that

$$x_i \mapsto \begin{cases} \text{true} & v_{r_i} > 0 \\ \text{false} & v_{r_i} \leq 0 \end{cases} \quad \forall i = 1, \dots, n$$

is a satisfying assignment of f .

\Leftarrow : Let \mathcal{A} be a satisfying assignment of f . For each $i = 1, \dots, n$ define

$$\begin{aligned} v_i &:= 2 && \text{if } \mathcal{A}(x_i) = \text{true} \\ v_i &:= -2 && \text{if } \mathcal{A}(x_i) = \text{false} \end{aligned}$$

Hence, at least one literal is satisfied for every clause and by Lemma 3.5.3 it follows that we can extend v to a strongly thermodynamically feasible flux through each clause with $v_{r_{\text{ex}}} = 1$. It is easy to see that we can combine all these strongly thermodynamically feasible fluxes to a strongly thermodynamically feasible flux through the whole network with $v_{r_{\text{ex}}} = 1$. \blacksquare

Since the metabolic network described here can be generated in polynomial time for a given 3SAT-formula f it follows by Thm. 3.5.1 that Problem 1b2a3b4a and Problem 1b2c3b4b are NP-hard.

3.5.3 $1x2y3c4b$ and $1x2y3a4b$ are NP-hard

We show hardness of $1b2c3c4b$ and $1b2c3a4b$. The remaining problems are more general and hence, also NP-hard.

The reduction for both kinds of problems ($1b2c3c4b$ and $1b2c3a4b$) is the same. Hence, we do the proofs together.

Again, we do a reduction from 3SAT. Assume we are given a 3SAT formula f with variables x_1, \dots, x_k and clauses c_1, \dots, c_n .

For each variable x_i , we construct a network as shown in Fig, 3.5. We observe that for weak thermodynamic constraints and relaxed thermodynamic constraints it holds that either m_{x_i} or $m_{\bar{x}_i}$ can be produced by this subnetwork but not both at the same time.

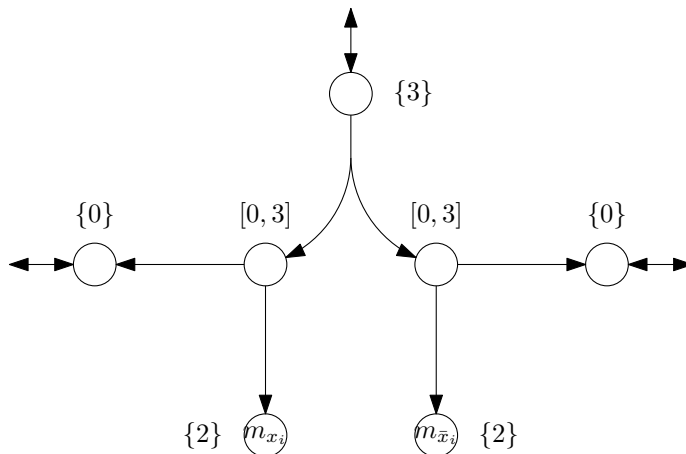


Figure 3.5: Subnetwork for each variable x_i . All reactions are reversible, the exchange reactions are drawn using arrows in both directions. For internal reactions the main direction is indicated. It is easy to see that by the given metabolite potentials this subnetwork can not produce m_{x_i} and $m_{\bar{x}_i}$ at the same time.

For each clause $c_j = (\ell_1 \vee \ell_2 \vee \ell_3)$, where ℓ_1, ℓ_2, ℓ_3 are literals, we introduce a metabolite m_{c_j} that has arcs from $m_{\ell_1}, m_{\ell_2}, m_{\ell_3}$ (i.e., the metabolites representing positive / negative variables). The network is shown in Fig. 3.6. The metabolite m_{c_j} is assigned a potential of 1 while the metabolites $m_{\ell_1}, m_{\ell_2}, m_{\ell_3}$ have a fixed potential of 2. It follows that no flow from m_{c_j} to $m_{\ell_1}, m_{\ell_2}, m_{\ell_3}$ is possible. It follows that a metabolite m_ℓ , where ℓ is a literal can only have inflow from the subnetwork that was constructed for the variable. Hence, either m_{x_i} or $m_{\bar{x}_i}$ can have outflow to any clause.

Finally, we require that every clause has some outflow by adding the subnetwork shown in Fig. 3.7 with exchange reaction r . It can easily be seen that f is satisfiable if and only if r can have flux.

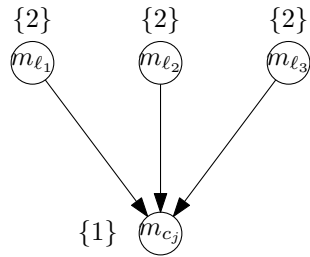


Figure 3.6: Subnetwork for clause $c_j = (\ell_1 \vee \ell_2 \vee \ell_3)$.

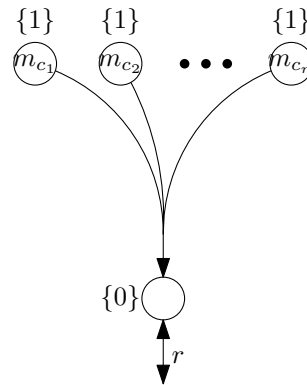


Figure 3.7: Subnetwork to enforce outflow of each clause subnetwork. Only if this outflow is possible, the target exchange reaction r can carry flux.

3.6 Problems with Unknown Complexity

We do not know whether $1a2b3a4a$ and $1a2b3b4a$ are solvable in polynomial time.

We only know by Sec. 3.3.2 that the problems $1a2b3a4a$ and $1a2b3b4a$ have the same complexity.

Chapter 4

Flux Optimization

Abstract The optimization of fluxes through a metabolic network is a frequent approach to reduce the space of possible solutions. While the optimization problem can be efficiently solved if only the steady-state assumption and flux bounds are given, the incorporation of thermodynamic constraints makes the problem NP-hard (Ch. 3). To still be able to solve the flux optimization problem efficiently in practice, I present here several strategies from the area of mixed integer linear programming to approach the problem. The presented methods are implemented in the `metaopt-toolbox` <http://sourceforge.net/projects/metaopt/>.

This section extends the work from my master thesis on the flux optimization problem, which has also been published in [99]. I briefly summarize the results from my master thesis and then present improvements for the formalism that I determined there as best.

4.1 Flux Balance Analysis

Flux Balance Analysis [163, 114] is a very popular analysis method in the field of metabolic network analysis. It is a very powerful tool to check if a gene-knockout will remove a certain metabolic function and therefore turn out to be lethal without having to actually run the biological experiment.

It comes in many variants, but most of them have the following common features:

- The flux space is constrained by the steady-state condition (Def. 2.2.1).
- Reactions have lower and upper flux bounds $\ell, u \in \mathbb{R}^{\mathcal{R}}$.
- An objective function (usually flux through a biomass reaction) is maximized.

We observe that these constraints define a polyhedral flux space

$$P = \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, \ell \leq v \leq u\}.$$

In the following we will further assume that c is a linear objective function, which leads us to the following linear program:

$$\begin{aligned} \max \quad & cv \\ \text{subject to} \quad & Sv = 0 \\ & \ell \leq v \leq u \end{aligned}$$

This formulation is very strong, since on the one hand it has been shown to give good results on some cases [40], does not require a lot of biological information and such linear programs, even with thousands of variables as in the case of genome-scale metabolic networks can typically be solved in less than a second by current solvers. In particular the formalism is easily extendable. If additional information is available it can simply be added in the form of additional constraints.

4.2 Thermodynamic Constraints

As we know from Chapter 2.6, thermodynamic constraints prohibit flux through internal cycles. This important property, in practice, comes with drawbacks on the mathematical properties. While the space of steady-state fluxes P is polyhedral and hence closed and convex, these properties are lost if we add strong thermodynamic constraints (Def. 2.6.1).

Proposition 4.2.1 (Proposition 9 in [99]) *There exist metabolic networks, where the space*

$$T_{\text{strong}} := \{v \in P : \exists \mu \text{ s.t. } \mu S_i v_i < 0 \text{ or } \mu S_i = 0 = v_i \forall i \in \mathcal{I}\}$$

of strongly thermodynamically feasible fluxes is neither closed nor convex. □

By relaxation to weakly thermodynamically constrained fluxes (Def. 2.6.2) we obtain a closed flux space. Hence optimization problems on the weakly thermodynamically constrained flux space have either an optimum or are unbounded.

Proposition 4.2.2 (Lemma 2 in [99]) *Let $Q \subseteq \mathbb{R}^{\mathcal{M}}$. The space*

$$T := \{v \in P : \exists \mu \in Q \text{ s.t. } \mu S_{*i} v_i < 0 \text{ or } v_i = 0 \forall i \in \mathcal{I}\}$$

of (weakly) thermodynamically feasible fluxes is closed. □

But in general the set of weakly thermodynamically feasible fluxes is not the closure of the strongly thermodynamically feasible fluxes. In particular, the space of strongly thermodynamically feasible fluxes can be empty while the space of weakly thermodynamically feasible fluxes is not.

It can also be easily seen that the space of weakly thermodynamically feasible fluxes is not convex. This comes as no surprise, since we already observed that the THERMOFLUX problem (1a2a3a4a) is NP-hard (cf. Sec. 3.2).

In the literature different approaches ranging from Monte Carlo methods [33, 32] over continuous non-linear programming [66, 47] to mixed integer programming [9, 134, 25, 69, 65] have been considered to solve thermodynamically constrained optimization problems. A comparison of the different methods can be found in [99]. Here, we will only consider MILP-based approaches.

4.3 Mixed Integer Linear Programming

To obtain a mixed integer programming formulation for thermodynamic constraints, we have to rewrite the quadratic constraint that links the fluxes to the potential differences:

$$\mu S_i v_i < 0 \text{ or } v_i = 0.$$

Since this constraint is only a sign constraint, the quadratic part can easily be reformulated as follows:

$$v_i > 0 \Rightarrow \Delta\mu_i < 0 \quad \forall i \in \mathcal{I} \quad (4.1)$$

$$v_i < 0 \Rightarrow \Delta\mu_i > 0 \quad \forall i \in \mathcal{I} \quad (4.2)$$

Note, that if $v_i = 0$ then $\Delta\mu_i$ can be anything. Hence, if and only if a steady-state flux v satisfies (4.1) and (4.2) then v is weakly thermodynamically feasible.

For an MILP formulation we have to capture the sign of the variables, we can either use SOS-constraints or big-M formulations. In the case of big-M formulations we would introduce boolean variables $a, b \in \{0, 1\}^{\mathcal{I}}$ with

- $v_i \leq 0$ if $a_i = 0$
- $v_i \geq 0$ if $b_i = 0$
- $a_i = 1$ if $\mu S_i < 0$
- $b_i = 1$ if $\mu S_i > 0$
- $a_i + b_i \leq 1$

for each $i \in \mathcal{I}$.

This leads to the following MILP, where ε is a small constant and M is a large constant:

$$\begin{aligned}
 \alpha_{\text{MILP}} &:= \max cv \\
 \text{s.t. } & Sv = 0 \\
 & \Delta\mu = \mu S \\
 & \ell^v \leq v \leq u^v \\
 & -Mb_i \leq v_i \leq Ma_i \quad \forall i \in \mathcal{I} \\
 & -Ma_i + \varepsilon b_i \leq \Delta\mu_i \leq Mb_i - \varepsilon a_i \quad \forall i \in \mathcal{I} \\
 & a_i + b_i \leq 1 \quad \forall i \in \mathcal{R} \\
 & a, b \in \{0, 1\}^{\mathcal{I}}
 \end{aligned}$$

As already discussed in my master thesis [99], we observe the following weaknesses:

- The LP relaxation is very weak. Usually, M has to be very large to not be restrictive. Therefore, even a small fractional a_i or b_i will imply that flux through reaction i is unconstrained. It follows that the optimum of the LP relaxation will equal the optimum of traditional FBA. This property is not surprising, since the ThermoFlux problem 1a2a3a4a (see Sec. 3.2) is NP-hard and hence, the optimization problem is not approximable (APX-hard).
- Cutting planes barely have any effect. Any cutting plane on the decision variables a, b will still allow fractional values of a, b and hence, due to the big-M constraints this will have barely any effect on the continuous variables v, μ .

Hence, we draw the conclusion that clever branching strategies will help us more than cutting plane methods. In my masters thesis I devised the branching method described in Alg. 2 that uses the reformulation of thermodynamic constraints using infeasible sets (Thm. 2.6.1, Thm. 2.6.2). Essentially, what we do is the following: We solve FBA (without any thermodynamic constraints). This gives us a flux vector v . For v to be thermodynamically feasible, there must exist a $\mu \in Q$ that satisfies

$$\begin{aligned}
 \mu S_i &< 0 & \forall i \in P \\
 \mu S_i &> 0 & \forall i \in N,
 \end{aligned} \tag{4.3}$$

where $P := \{i \in \mathcal{I} : v_i > 0\}$ and $N := \{i \in \mathcal{I} : v_i < 0\}$, i.e., $(P, N) = \text{sign}(v)$.

If the computed solution is infeasible, it must contain an infeasible direction set $C \subseteq (P, N)$. Such a minimal infeasible set can be found in polynomial time as we observed in Obs. 2.6.1. Since C must not be contained in a feasible solution, we block one of the reaction directions of C after another and repeat.

Algorithm 2 Constraint handler that enforces thermodynamic feasibility

Input: A flux v
Find violated minimal infeasible set $C = (C^+, C^-) \subset \text{sign}(v)$ (see Theorem 2.6.2).
if we found no infeasible set C **then**
 Return feasible
else
 for $i \in C^+$ **do**
 if $\ell_i^J \leq 0$ **then**
 Create child node with $u_i^J = 0$.
 end if
 end for
 for $i \in C^-$ **do**
 if $u_i^J \geq 0$ **then**
 Create child node with $\ell_i^J = 0$.
 end if
 end for
 Add created child nodes to branch and bound tree and continue solving nodes.
end if

4.4 Implicit Representation of the Potential Space & Constraint Programming

If the number of minimal infeasible reaction sets is very small, which is the case for several genome-scale networks without bounds on metabolite potentials, then the branching method described in the previous section quickly tests all possibilities and computes the optimal solution. However this method becomes quickly inefficient for larger numbers of infeasible sets, and a closer analysis of the method reveals that we do many redundant branchings.

W.l.o.g. let $C = (C^+, \emptyset)$ be an infeasible set. By Thm. 2.6.3 there exists a $v \in \mathbb{R}^{\mathcal{R}}$ with $\text{sign}(v) = C$ and $\mu S v \geq 0$ for all $\mu \in Q$. It follows that for each $\mu \in Q$ there exists at least one $i \in C^+$ with $\mu S_i \geq 0$. Hence, instead of blocking forward flux through reaction i , we can also restrict the space of feasible potentials to $Q' = \{\mu \in Q : \mu S_i \geq 0\}$. The direct effect on the feasible flux directions is the same.

Assume now that we blocked forward flux of reaction i by adding the implicit constraint $\mu S_i \geq 0$ and we now find a new flux vector w that obeys the added constraint (i.e., $w_i \leq 0$) but contains an infeasible set $C' = (C'^+, C'^-)$ w.r.t. Q with $i \in C'^-$. If we would just branch as described in the previous section, we would also have to consider the case where we additionally enforce $v_i \geq 0$ and hence block reaction i entirely. However, we will see that if we work on Q' we can find a smaller infeasible set that does not contain i and hence we can skip one branching possibility.

Application of Thm. 2.6.3 yields that there exists a $w' \in \mathbb{R}^{\mathcal{R}}$ with $\text{sign}(w') = C'$ and

$\mu S w' \geq 0$ for all $\mu \in Q$. We observe that

$$\begin{aligned} \mu S w' &= \mu S_{\underline{C}'} w'_{\underline{C}'} = \mu S_{\underline{C}' \setminus \{i\}} w'_{\underline{C}' \setminus \{i\}} + \mu S_i w'_i \geq 0 & \forall \mu \in Q \\ &\Rightarrow \mu S_{\underline{C}' \setminus \{i\}} w'_{\underline{C}' \setminus \{i\}} \geq 0 & \forall \mu \in Q', \end{aligned}$$

since $w'_i < 0$ and $\mu S_i \geq 0$ for all $\mu \in Q'$. We conclude that we can find a smaller infeasible set (that does not contain i) w.r.t. Q' than w.r.t. Q . Under Q we do not know that $\mu S_i \geq 0$ and hence, we cannot perform the simplification. We can also not read this off from the irreversibility constraint, since the irreversibility may originate from kinetic arguments (e.g. reverse flux is always so slow that it can be considered zero) and not from thermodynamic considerations.

4.4.1 Practical Implementation

Recall Obs. 2.6.1. Let w be a given flux vector. For simplicity of notation we again just consider the subnetwork $\mathcal{N}^{\text{sign}(w_I)}$ as in Thm. 2.6.3. Hence we can assume w.l.o.g. that $w > 0$ and that \mathcal{R} contains only internal reactions.

We observed that to compute an infeasible set for a polyhedral potential space $Q = \{\mu : \mu A \leq b\}$ we only have to solve an LP (Eq. 2.8):

$$\text{opt} = \max\{bx - z : -Sv + Ax = 0, \mathbf{1}v + z = 1, v \geq 0, x \leq 0, z \geq 0\}$$

We recall that w was feasible if $\text{opt} < 0$. If an infeasible set exists, it can be read of from the solution variables v . We observe from this LP that if we constrain Q further to $Q' = \{\mu : \mu A \leq b, \mu B \leq 0\}$ we simply add variables for each column in B to the LP:

$$\max\{bx - z : -Sv + Ax + By = 0, \mathbf{1}v + z = 1, v \geq 0, x, y \leq 0, z \geq 0\} \quad (4.4)$$

When we add constraints for the branching conditions, these constraints are exactly of the form $\mu B \leq 0$. We further observe that the columns of B are columns of S corresponding to the reactions where we blocked a direction. It follows that instead of setting $v_i > 0$ in a feasible solution of (4.4) we can also set $y_j < 0$ if $S_i = B_j$ (e.g., if we branched on reaction i before). This way we can get a smaller infeasible set and save time in the branching process. However, the objective function will not enforce the computation of a v with minimal support.

Hence, we will now reformulate the LP to obtain a formulation, where we can minimize the support of v . First of all, we observe that we actually do not have to solve the optimization problem (4.4), but it is sufficient to test whether the following system has a feasible solution.

$$\begin{aligned} bx - z &\geq 0, \\ -Sv + Ax + By &= 0, \\ \mathbf{1}v + z &= 1, \\ v \geq 0, x \leq 0, y \leq 0, z &\geq 0 \end{aligned} \quad (4.5)$$

If it is inconsistent, it follows that w is feasible. If we find a feasible solution, then the maximum of LP (4.4) is at least 0 and we know that w is infeasible and we can also use the v from the solution to compute an infeasible set.

A naive approach would now try to minimize $\mathbb{1}v$ over all feasible solutions. This however will not yield any improvement if there exists no solution with $bx > 0$.

Proposition 4.4.1 *If there exists no solution of (4.5) with $bx > 0$ then every consistent solution is also an optimal solution under $\min \mathbb{1}v$.*

PROOF We observe that minimizing $\mathbb{1}v$ is equivalent to maximizing z . To obtain $z > 0$ it is necessary that there exist solutions with $bx > 0$. Since this is not possible, we know that every consistent solution satisfies $z = 0$ and hence, $\mathbb{1}v = 1$. It follows that also every optimal solution satisfies $\mathbb{1}v = 1$ and hence optimality is no restriction. ■

If there exists a solution with $bx > 0$, it is hard to show a minimality result in general, but for special matrices $B = SX$ where X contains positive entries we can show the following result. Observe that such constraints are added when incorporating branching decisions into the potential space Q .

Proposition 4.4.2 *Assume $Q = \{\mu : \mu A \leq b, \mu SX \leq 0\} \neq \emptyset$ with $X \in \mathbb{R}_+^{\mathcal{R} \times k}$. If there exists a solution of (4.5) with $bx > 0$ then each optimal solution of*

$$\min\{\mathbb{1}v : bx - z \geq 0, -Sv + Ax + SXy = 0, \mathbb{1}v + z = 1, v \geq 0, x \leq 0, y \leq 0, z \geq 0\}$$

*satisfies for each $j \in \{1, \dots, k\}$ that there exists an $i \in \text{supp}(X_{*j})$ with $v_i = 0$.*

PROOF Let (v, x, y, z) be an optimal solution. We observe that $bx > 0$.

Assume there exists a $j \in \{1, \dots, k\}$ with $v_i > 0$ for all $i \in \text{supp}(X_{*j})$.

Let $\lambda = \min_{i \in \text{supp}(X_{*j})} \frac{v_i}{X_{ij}}$. Clearly, $\lambda > 0$. Define $v' := v - \lambda X_{*j}$, and $y' \in \mathbb{R}^k$ with

$$y'_k := \begin{cases} y_k & \text{if } k \neq j \\ y_j - \lambda & \text{if } k = j \end{cases}.$$

We observe that by construction, $v' \geq 0$.

It follows that

$$-Sv' + Ax + SXy' = -Sv + \lambda SX_{*j} + Ax + SXy - \lambda SX_{*j} = -Sv + Ax + SXy = 0.$$

However, it holds that $\mathbb{1}v' + z < 1$. We need equality for a feasible solution. This we can obtain by scaling (v', x, y') to a solution (v'', x'', y'') and choosing z'' appropriately. To obtain a contradiction we have to show that after the scaling we obtain $\mathbb{1}v'' < \mathbb{1}v$ (i.e., $z'' > z$).

Therefore we need to control z .

Claim 4.4.1 $bx = z$

PROOF Assume $bx > z$. Hence, $\lambda := \frac{1}{bx+1-z} < 1$ is well defined. Define $x' = \lambda x$, $y' = \lambda y$, $v' = \lambda v$ and $z' = 1 - \lambda(1 - z)$. It follows that

$$\begin{aligned} bx' - z' &= \lambda bx - (1 - \lambda(1 - z)) = \frac{bx}{bx+1-z} - \frac{bx+1-z - (1-z)}{bx+1-z} = 0, \\ -Sv' + Ax' + By' &= \lambda(-Sv + Ax + By) = 0, \\ \mathbb{1}v' + z' &= \lambda\mathbb{1}v + (1 - \lambda(1 - z)) = \frac{1-z}{bx+1-z} + \frac{bx+1-z - (1-z)}{bx+1-z} = 1, \\ v' &\geq 0, x' \leq 0, y' \leq 0, z' \geq 0 \end{aligned}$$

and hence, (v', x', y', z') is a feasible solution with $\mathbb{1}v' = \lambda\mathbb{1}v < \mathbb{1}v$. This is a contradiction to the optimality of v . ■

It follows that when we choose $v''(\lambda) = \lambda v'$, $x''(\lambda) = \lambda x$, $y''(\lambda) = \lambda y'$ we can chose $z''(\lambda) = bx'' = \lambda bx = \lambda z$ and satisfy for all $\lambda \geq 0$

$$\begin{aligned} bx''(\lambda) - z''(\lambda) &\geq 0 \\ -Sv''(\lambda) + Ax''(\lambda) + SY''(\lambda) &= 0 \\ v''(\lambda) &\geq 0, x''(\lambda) \leq 0, y''(\lambda) \leq 0, z''(\lambda) \geq 0. \end{aligned}$$

For $\lambda = 1$ we satisfy by construction $\mathbb{1}v''(\lambda) + z''(\lambda) < 1$. Since scaling is continuous and for sufficiently large λ we obtain $\mathbb{1}v''(\lambda) + z''(\lambda) > 1$ it follows that there exists a $\lambda > 1$ that satisfies $\mathbb{1}v''(\lambda) + z''(\lambda) = 1$. It follows that $z''(\lambda) > z$ and hence, $\mathbb{1}v''(\lambda) < \mathbb{1}v$. This is a contradiction to the optimality of v . ■

4.4.1.1 Without Bounds on Metabolite Potentials

While Prop. 4.4.2 gives us some hope of a general method to obtain smaller infeasible sets, it is restricted to the case where we can find a solution with $bx > 0$. In particular, if we do not have bounds on metabolite concentrations then x is 0-dimensional (no constraint on metabolite potentials) and hence, $bx = 0$ always.

Let us look again at LP 4.5.

We observe that if $Q = \{\mu : \mu A \leq b, \mu B \leq 0\} \neq \emptyset$ (the potential space is consistent), then every feasible solution of LP (4.5) has $v \neq 0$. In particular for such a given v , there exists a reaction r with $v_r > 0$. Due to scaling, we can hence find a solution of:

$$\begin{aligned} bx &\geq 0, \\ -Sv + Ax + By &= 0, \\ v_r &= 1 \\ v &\geq 0, x \leq 0, y \leq 0 \end{aligned} \tag{4.6}$$

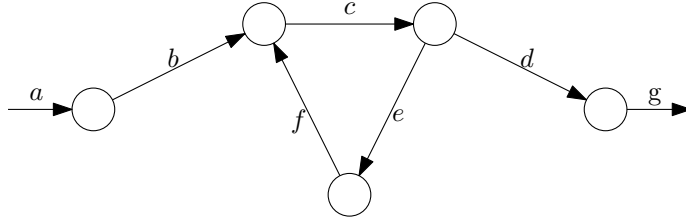


Figure 4.1: Only reactions (arcs) that carry flux are shown in this example. Since $\{c, e, f\}$ forms an internal cycle, the depicted flow is thermodynamically infeasible.

Assuming that we know such a reaction r already, we can then simply minimize $\mathbb{1}v$ under the constraints in (4.6). In the special case, where Q is unconstrained (A, B have no columns) and the infeasible sets are cycles, we observe that this approach will search for a circuit that contains r and hence compute a minimal infeasible set.

4.4.2 Coupled Reactions and Generalized Infeasible Sets

Let us now consider the example network shown in Fig. 4.1. Assume that all reactions carry flux. We observe that $\{c, e, f\}$ is an infeasible set, since its reactions form an internal cycle. We observe further that reactions e, f are coupled, i.e., if e carries positive flux that f has to carry positive flux and vice versa. We observe that blocking either e or f has the same effect and we do not have to consider both options. However, neither $\{c, e\}$ nor $\{c, f\}$ forms an infeasible set and by the current theory we would have to consider both cases.

We will now generalize the notion of infeasible set so that we do not have to consider both cases.

Definition 4.4.1 $\mathcal{A} \in \mathbb{R}^{\mathcal{R} \times k}$ with $\text{supp}(\mathcal{A}_i), \text{supp}(\mathcal{A}_j)$ disjoint for all distinct $i, j \in \{1, \dots, k\}$ is called an infeasible set for a potential space Q if

$$\begin{aligned} \mu &\in Q \\ \mu S \mathcal{A}_j &< 0 & \forall j = 1, \dots, k \end{aligned}$$

is inconsistent. □

We observe that this definition is indeed a generalization of an infeasible set, since we can model every classical infeasible set as an infeasible set according to Def. 4.4.1:

Proposition 4.4.3 $A \in \{-, 0, +\}^{\mathcal{R}}$ is an infeasible set, i.e., (4.3) is inconsistent with $(P, N) = A$ if and only if $\mathcal{A} \in \mathbb{R}^{\mathcal{R} \times \mathcal{A}}$ with

$$\mathcal{A}_{ij} = \begin{cases} 1 & i \in A^+ \wedge i = j \\ -1 & i \in A^- \wedge i = j \\ 0 & \text{otherwise} \end{cases}$$

is an infeasible set.

PROOF Clearly $\text{sign}(\mathcal{A}_j) = (\{j\}, \emptyset)$ for $j \in A^+$ and $\text{sign}(\mathcal{A}_j) = (\emptyset, \{j\})$ for $j \in A^-$. It follows that

$$\begin{aligned} \mu &\in Q, \\ \mu S \mathcal{A}_j &= \mu S_j < 0 && \forall j \in A^+ \\ \mu S \mathcal{A}_j &= -\mu S_j < 0 && \forall j \in A^- \end{aligned}$$

is inconsistent (i.e., \mathcal{A} is an infeasible set) if and only if A is an infeasible set. \blacksquare

We can now use this generalization to branch on *groups of reactions*. For example, in the infeasible cycle $\{c, e, f\}$, we can branch on the conditions $\mu S_c \geq 0$, $\mu(S_e + S_f) \geq 0$. In this case the first branching decision translates to the well known “no positive flux through reaction c .” The second branching decision however translates to: “through either reaction e or f no positive flux is possible.” In general this is not very helpful, since such a condition can not necessarily be formulated using linear constraints. If, however, e, f are coupled (as in the example), we can use this to get rid of extra branching decisions. In the example of reactions e, f the condition would simply translate to: “no positive flux is possible through e and f ”. A linear constraint that enforces the condition for the flux space in this special case can easily be found.

We observe that we can do the same simplification if e, f were only directionally coupled ($e \rightarrow f$), i.e., if $v_e > 0$ implies $v_f > 0$ (Def. 2.1.9). Consequently $v_f \leq 0$ implies $v_e \leq 0$. Hence, we can simplify ($v_f \leq 0$ or $v_e \leq 0$) to $v_e \leq 0$.

4.4.2.1 Computing Generalized Infeasible Sets

We have observed that we can reduce the number of branching decision of infeasible sets (Def. 2.6.6) using generalized infeasible sets (Def. 4.4.1) and flux coupling information. We now discuss how we can compute such generalized infeasible sets.

Theorem 4.4.1 *Let $\mathcal{A} \in \mathbb{R}^{\mathcal{R} \times k}$ be an infeasible set. Then there exists $\lambda_1, \lambda_2 \geq 0$ such that $\mathcal{B} \in \mathbb{R}^{\mathcal{R} \times (k-1)}$ is an infeasible set with $\mathcal{B}_i = \mathcal{A}_i, i = 1 \dots, k-2$ and $\mathcal{B}_{k-1} = \lambda_1 \mathcal{A}_{k-1} + \lambda_2 \mathcal{A}_k$.*

PROOF By definition of infeasible set with $Q = \{\mu : \mu A \leq b\} \neq \emptyset$, we have that the system

$$\begin{aligned} \mu A &\leq b \\ \mu S A &< 0 \end{aligned} \tag{4.7}$$

is inconsistent. We observe that we can define a modified metabolic network with stoichiometric matrix $S' := SA$. By Thm. 2.6.2 and the same arguments as above, it follows

that (4.7) is inconsistent if and only if

$$\begin{aligned} bx - z &\geq 0 \\ -SAv + Ax &= 0 \\ \mathbb{1}v + z &= 1 \\ v \geq 0, x \leq 0, z &\geq 0 \end{aligned}$$

is consistent.

If $v_{k-1} = 0$, we observe that we can drop the $(k-1)$ -st column of \mathcal{A} to obtain \mathcal{B} , i.e., by choosing $\lambda_1 = 0$ and $\lambda_2 = 1$. The theorem then holds trivially. Hence for the following we assume that $v_{k-1} > 0$.

We now choose $v' \in \mathbb{R}_+^{k-1}$ as $v'_i := v_i$ for $i = 1, \dots, k-1$ and $\lambda_1 := 1, \lambda_2 := \frac{v_k}{v_{k-1}}$.

For $\mathcal{B} \in \mathbb{R}^{\mathcal{R} \times (k-1)}$ with $\mathcal{B}_i = \mathcal{A}_i, i = 1 \dots, k-2$ and $\mathcal{B}_{k-1} = \lambda_1 \mathcal{A}_{k-1} + \lambda_2 \mathcal{A}_k$ it follows that

$$\begin{aligned} S\mathcal{B}v' &= \sum_{i=1}^{k-2} S\mathcal{A}_i v_i + S\mathcal{B}_{k-1} v'_{k-1} \\ &= \sum_{i=1}^{k-2} S\mathcal{A}_i v_i + \lambda_1 S\mathcal{A}_{k-1} v_{k-1} + \lambda_2 S\mathcal{A}_k v_{k-1} \\ &= \sum_{i=1}^{k-2} S\mathcal{A}_i v_i + S\mathcal{A}_{k-1} v_{k-1} + S\mathcal{A}_k v_k \\ &= SA v \end{aligned}$$

We observe that $\lambda_2 \geq 0$ and hence, v' also satisfies the sign constraints. However, $\mathbb{1}v' + z = 1$ does not have to hold anymore. We observe that $\mathbb{1}v \geq \mathbb{1}v' > 0$ since $v'_{k-1} = v_{k-1} > 0$. It follows that for all $\mu \geq 1$ with $x'' = \mu x, v'' = \mu v', z'' = \mu z$ it holds that

$$\begin{aligned} bx'' - z'' &= \mu(bx - z) \geq 0, \\ -S\mathcal{B}v'' + Ax'' &= \mu(-S\mathcal{B}v' + Ax) = 0, \\ v'' \geq 0, x'' \leq 0, z'' &\geq 0 \end{aligned}$$

Clearly, there exists a $\mu \geq 1$ that also satisfies $\mathbb{1}v'' + z'' = 1$. Hence, \mathcal{B} is an infeasible set and the theorem follows. ■

As a side product of Thm. 4.4.1 we observe that if we know a witness v for an infeasible set, then we can directly compute (recursive application of Thm. 4.4.1) a generalized infeasible set with arbitrary reactions grouped together.

In practice we can now run Alg. 3 to minimize the number of branchings, while still being able to realize all branchings also as linear constraints in the flux space. Let v be

a witness of an infeasible set Y . As observed in Sec. 2.6.3.3 we will use the equivalent interpretation of Y as a set of pseudo-reactions ($Y \subseteq \bar{\mathcal{R}}$). Core step of the algorithm is to compute a minimal set $X \subseteq Y$ that represents the branching decisions of Y . The following proposition states that we can compute X efficiently:

Proposition 4.4.4 *A minimal set $X \subseteq Y$ of pseudo-reactions such that each pseudo-reaction $r \in Y$ is coupled from at least one pseudo-reaction $s \in X$ ($s \rightarrow r$) can be computed by analyzing the coupling graph $G = (Y, E)$ with*

$$E = \{(a, b) \in Y^2 : a \rightarrow b\}$$

as follows:

- Start with $X = \emptyset$.
- Find all strongly connected components.
- For each strongly connected component C test if there exist $r \in C, s \in Y \setminus C$ with $s \rightarrow r$. If this is not the case, add an arbitrary reaction from C to X .

PROOF First of all we observe that by transitivity of the coupling relation \rightarrow , we can collapse each strongly connected component C to a single representing node $c \in C$ and assume w.l.o.g. that G is acyclic.

By transitivity and reflexivity, it follows now that for every node r there exists a node s with in-degree 0 such that $s \rightarrow r$. Hence, the algorithm computes a set X such that for each $r \in Y$ there exists a $s \in X$ with $s \rightarrow r$.

The minimality of the computed X is also easy to observe, since every node with in-degree 0 must be an element of X . ■

We observe that by Thm. 4.4.1 \mathcal{A} will be a generalized infeasible set with branchings that can be executed by blocking the corresponding directed reaction in X .

Algorithm 3 Algorithm that computes a generalized infeasible set with a smaller amount of branching decisions.

Input: A witness v of an infeasible set Y
 Find a minimal set of pseudo-reactions $X \subseteq Y$ such that for each pseudo-reaction $r \in Y$ there exists a pseudo-reaction $s \in X$ such that $s \rightarrow r$.
 Set \mathcal{A} as in Prop. 4.4.3.
for $r \in Y \setminus X$ **do**
 Find a reaction $s \in X$ with $s \rightarrow r$.
 $\mathcal{A}_s = \mathcal{A}_s + \frac{v_r}{v_s} \mathcal{A}_r$.
 Delete the column corresponding to r from \mathcal{A} .
end for

4.4.3 Partitioning the Flux Space

A big problem with the proposed branching methods is that we actually do not partition the solution space in general. The subproblems may actually overlap. This of course can lead to massive double-computation.

The way out is paved by the idea of *interdiction branching* [92]. Interdiction branching is a method developed for ILPs, where conventional branching on fractional variables leads to one very small subproblem and one huge subproblem. This is precisely the case in flux optimization problems with thermodynamic constraints if we would simply branch on single reactions. Blocking flux (i.e., fixing the flux rate to 0) will induce a small subproblem, while setting flux to be bigger than 0 only reduces the problem very little (if it does not make it even more complicated). Hence, we have chosen to branch on infeasible sets. The same idea is followed by interdiction branching.

From the theory of interdiction branching, we will here only use a very simple idea on how to partition the solution space from an infeasible set.

Assume that we computed an infeasible set consisting of reactions r, s, t (in forward direction). In our classical branching scheme, we branch on the following 3 cases:

1. $\Delta\mu_r \geq 0$
2. $\Delta\mu_s \geq 0$
3. $\Delta\mu_t \geq 0$

The key insight is now that if our solution is not contained in the first case, it has to satisfy $\Delta\mu_r < 0$. Hence, we can strengthen our three branching cases to

1. $\Delta\mu_r \geq 0$,
2. $\Delta\mu_r \leq 0 \wedge \Delta\mu_s \geq 0$,
3. $\Delta\mu_r \leq 0 \wedge \Delta\mu_s \leq 0 \wedge \Delta\mu_t \geq 0$.

If we branch into the third condition, this implies that we block r, s in backward direction and block t in forward direction. Hence, these branching conditions can also be directly enforced on the flux space.

4.5 Heuristics

Another key aspect of constraint programming are heuristics. In my master thesis, I introduced a heuristic that turns thermodynamically infeasible solutions into thermodynamically feasible (without bounds on concentrations) solutions by subtracting internal

cycles. Here, we will quickly recall this heuristic, since it continues to be the main workhorse for solving thermodynamically constrained flux optimization problems.

While the cycle subtraction heuristic works very well for unconstrained potential spaces, the effectiveness decreases dramatically as soon as we work with bounds on metabolite potentials. This is simply caused by the fact that the infeasible sets are not cycles anymore. Hence, an infeasible set that is preventing a solution from being feasible cannot be resolved by subtracting the infeasible set, since this would destroy the steady-state condition. Hence, we will discuss a new approach for designing heuristics for the general case.

4.5.1 Cycle Subtraction

A comprehensive discussion of this method can be found in my master thesis [99] and in my publication on fast thermodynamically constrained flux variability analysis [101].

For theoretical results on this heuristic, we had to observe that internal flux forcing reactions and objective reactions in internal cycles can cause problems. To define the set $\bar{\mathcal{C}}$ of reactions in internal cycles, we recall that we can uniformly deal with lower and upper bounds using pseudo-reactions (Def. 2.1.6) and define the set of irreversible reactions Irrev given these bounds (Def. 2.1.7).

Definition 4.5.1 (Reactions in Internal Cycles) *Let $(\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network with irreversible reactions $\text{Irrev} \subseteq \bar{\mathcal{R}}$. The set of reactions in internal cycles is given by*

$$\bar{\mathcal{C}} := \{r \in \bar{\mathcal{I}} : S_{\mathcal{I}}v = 0, v_{\text{Irrev}} \geq 0, v_r > 0 \exists v \in \mathbb{R}^{\mathcal{I}}\}$$

Definition 4.5.2 *Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network.*

Given lower and upper flux bounds $\ell, u \in \mathbb{R}^{\mathcal{R}}$, a pseudo-reaction r is called flux-forcing if $\bar{\ell}_r > 0$.

For a linear objective function $c \in \mathbb{R}^{\mathcal{R}}$, a pseudo-reaction r is called objective if $c_r > 0$. \square

Theorem 4.5.1 *Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network with lower and upper flux bounds $\ell, u \in \mathbb{R}^{\mathcal{R}}$ and a linear objective function $c \in \mathbb{R}^{\mathcal{R}}$. Let $F \subseteq \bar{\mathcal{R}}$ be the set of reactions that are flux-forcing, and let $O \subseteq \bar{\mathcal{R}}$ be the set of objective reactions.*

Given a steady-state flux v with $\ell \leq v \leq u$, a thermodynamically feasible flux v^ with $cv \leq cv^*$ and $\ell \leq v^* \leq u$ can be computed in polynomial time w.r.t. the size of the network \mathcal{N} if $\bar{\mathcal{C}} \cap (F \cup O) = \emptyset$. Furthermore, it holds that $v_{\mathcal{E}}^* = v_{\mathcal{E}}$ and there exists an internal circulation $w \in \mathbb{R}^{\mathcal{I}}$ with $S_{\mathcal{I}}w = 0$, $\text{sign}(w) \subseteq \text{sign}(v)$ and $v_{\mathcal{I}}^* + w = v_{\mathcal{I}}$.*

PROOF This theorem was already stated in a similar fashion as Thm. 9 in [99] and Thm. 2 in [101]. However, due to its more refined formulation with pseudo-reactions, it is not identical. However, the proof goes along exactly the same lines.

The key to the proof of Thm. 4.5.1 is Alg. 4, which gives the wanted polynomial-time algorithm.

Therefore, we observe that by construction of Alg. 4 we only subtract internal cycles with smaller sign-support. Thus, we get that $\text{sign}(v^*) \subseteq \text{sign}(v)$, $|v_r^*| \leq |v_r|$ for all $r \in \mathcal{R}$, and that there exists a $w \in \mathbb{R}^{\mathcal{I}}$ with $S_{\mathcal{I}}w = 0$, $\text{sign}(w) \subseteq \text{sign}(v)$ and $v_{\mathcal{I}}^* + w = v_{\mathcal{I}}$. Clearly, the flux bounds for all non-flux forcing reactions stay satisfied.

For every flux forcing reaction $r \in F$ we know that $r \notin \bar{\mathcal{C}}$ and hence, $w_r \leq 0$. However, since r is flux-forcing, we also know that $v_r \geq 0$ and thus we conclude that $w_r = 0$. Therefore, also the flux bounds for the flux-forcing reactions stay satisfied and v^* is a steady-state flux that satisfies the flux bounds.

Since cycles are subtracted until no internal cycle can be found, the thermodynamic feasibility of v^* follows (Prop. 5 in [99]).

We observe that $c_{\mathcal{I}}w \leq 0$ since there exists no pseudo-reaction $r \in \bar{\mathcal{R}}$ with $c_r > 0$ and $w_r > 0$ since $O \cap \bar{\mathcal{C}} = \emptyset$. Hence, it follows that $cv \leq cv^*$.

Similarly it follows that $v_{\mathcal{E}} = v_{\mathcal{E}}^*$ (Obs. 1 in [99]). ■

Note that if v is chosen to maximize cv , then clearly $cv = cv^*$ follows.

Algorithm 4 This algorithm computes a thermodynamically feasible flux out of a possibly thermodynamically infeasible flux, if the conditions of Thm. 4.5.1 are satisfied. It runs in polynomial time. ($\mathbf{1}$ denotes a vector where all entries are 1.)

Input: A steady-state flux v

repeat

$$I^+ := \{i \in \mathcal{I} : v_i \geq 0\}$$

$$I^- := \{i \in \mathcal{I} : v_i \leq 0\}$$

$$L := \arg \max \left\{ \begin{array}{l} \mathbf{1}L_{I^+} - \mathbf{1}L_{I^-} : S_{\mathcal{I}}L = 0, \\ v_{I^-} \leq L_{I^-} \leq 0, \\ v_{I^+} \geq L_{I^+} \geq 0 \end{array} \right\}$$

$$v_{\mathcal{I}} := v_{\mathcal{I}} - L$$

until $\mathbf{1}L_{I^+} - \mathbf{1}L_{I^-} = 0$

return v

4.5.2 First Directions, then Fluxes

In the cycle deletion heuristic, we took a thermodynamically infeasible solution that satisfied the steady-state condition and modified it to also satisfy the thermodynamic constraints. Since this approach seems to be rather difficult for general potential spaces,

we will now go the other way round: We will first assign directions to all reactions to make them thermodynamically consistent and then compute a steady-state flux vector through this system that obeys the previously assigned directions.

Since computing an optimal steady-state flux vector that satisfies the assigned reaction directions can easily be solved using linear programming, we now discuss how the potentials can be chosen from which the reaction directions are inferred.

The following observations and conclusions are made on *E. coli* iAF1260. For this model information on equilibrium constants was published in the supplementary material of [43]. Since the default model already contains a lot of irreversibility information that makes the potential bounds redundant, I removed all the irreversibility constraints for internal reactions (for more details see Sec. 5.2.4).

4.5.2.1 Approach 1: Mean Values from Lower and Upper Bounds

Let us assume that the potential space is only constrained by lower and upper bounds $\ell, u \in \mathbb{R}^M$ on the metabolite potentials.

Since the lower and upper bounds originate from uncertainties around a physiologically likely potential, it makes sense to simply use this potential as our guess. Hence, we simply use the mean potential

$$\mu_{\text{mean}} := \frac{u + \ell}{2}$$

and set reaction r to be in forward direction if $\mu_{\text{mean}} S_r < 0$ and set it to backwards if $\mu_{\text{mean}} S_r > 0$.

In practice we will also have to deal with metabolites on which no potential bounds could be estimated. These metabolites have bounds of $-\infty$ and ∞ , where the average is not defined. However, even if I assume all reactions incident to metabolites with infinity bounds to be reversible, the resulting metabolic network did not allow a feasible steady-state flux vector. Note that the 0-flux vector is not feasible in *E. coli* iAF1260, since it always requires some positive flux through an artificial *ATP maintenance* reaction.

The problem of this approach lies in the fact that the metabolite concentrations are unknown and the lower and upper potential bounds are hence also not equally distributed around the actual concentration.

4.5.2.2 Approach 2: Orientation on Steady-State Solution

Since the heuristic is usually called during the branch and bound framework of the constraint integer programming solver, we can assume that there is a current steady-state solution v available (which is however thermodynamically infeasible).

If we cannot find a potential vector $\mu \in Q$ that proves thermodynamic feasibility of v

(i.e., $\mu S_r v_r < 0$ for all $r \in \text{supp}(v)$), we can at least try to maximize the number of reactions with $\mu S_r v_r < 0$.

To do this, we solve

$$\begin{aligned}
 & \min \alpha s && (4.8) \\
 \text{s.t.} & \mu S_r - s_r \leq 0 && \forall r \in \mathcal{R} : \alpha_r > 0 \\
 & \mu S_r + s_r \geq 0 && \forall r \in \mathcal{R} : \alpha_r < 0 \\
 & \mu \in Q \\
 & s \geq 0
 \end{aligned}$$

with $\alpha_r = 1$ for all $v_r > 0$, $\alpha_r = -1$ for all $v_r < 0$ and 0 otherwise.

Since we minimize a L_1 -norm, it is likely that we compute a solution with small support in s . It follows that for many reactions in $\text{supp}(v)$, we obtain $\mu S_r v_r \leq 0$. Although, we actually require strict inequality, this discrepancy did not seem to be too relevant in practice (i.e., all solutions computed this way were actually thermodynamically feasible also with strict inequalities).

The directions obtained from this approach at least allowed flux through *ATP maintenance* and the resulting LP of the flux-optimization step was feasible. However, no flow through the biomass reaction was possible.

4.5.2.3 Approach 3: Reiteration using Dual Variables

We observe that the choice of α in approach 2 was rather arbitrary. Instead, we could have chosen $\alpha = v$ to make reactions with a lot of flux more important. Even more important, α does not need to have the same sign as the reference flux distribution. We now want to exploit that the dual solution of an LP tells us which inequalities are constraining the solution. In our case, the inequalities of the FBA problem are (with a few exceptions) the irreversibility constraints induced from the computed μ . Hence, the dual solution of the flux optimization step tells us through which reactions we should send flux in the opposite direction than in the current solution to obtain a better flux.

Changing α_r according to the reduced costs and resolving (4.8) will lead to a new solution that is more likely to also produce some biomass. This can be reiterated until a good enough solution is found, or the quality of the solutions (for example measured in the objective value) decreases.

In practice, this heuristic indeed managed to find thermodynamically feasible fluxes that also had flux through the biomass reaction. However, this required many iterations of resolving.

4.6 Conclusion

In this chapter we have seen that we can reduce the number of branching decisions by branching on the potential space instead of just the flux space and by making sure that the branching decisions actually form a partition. Furthermore, we saw that we can easily incorporate additional structural information like flux coupling data to reduce the number of branchings even more. However, an empirical verification of these improvements turned out to be difficult, because the run time depends highly on the order of the branchings, which easily changes when the branching method is modified. In particular, these improvements were not sufficient to enable thermodynamic FVA on *Human recon.* 1.

For the case that also bounds on metabolite potentials should be included, it should be noted that for well-curated models, like *E. coli* iAF1260 most of the thermodynamic information is already encoded in the reaction reversibilities. Hence, the bounds on metabolite potentials do not pose a significant restriction of the flux space and the standard algorithm works very well. On the other hand, if reversibility information is removed, the optimization problem becomes very hard. It should be noted that the problem seems to be particularly hard for weak thermodynamic constraints (Def. 2.6.2), as investigated in this section.

It is interesting that if the problem is formulated with strong thermodynamic constraints (Def. 2.6.1) or relaxed thermodynamic constraints (Def. 2.6.3), then MILP approaches, for example with FASIMU [68, 69], can solve the flux optimization problem efficiently in a few minutes (oral communication from Andreas Hoppe). However, with weak thermodynamic constraints also FASIMU is not able to solve the flux optimization problem (takes more than a day).

Chapter 5

Potential Optimization

Abstract Thermodynamic constraints are frequently not only used as constraints for the flux space, but also to infer metabolite concentrations or at least concentration ranges [82, 25, 106].

In the first part of this chapter, we will discuss how we can deal with the strict inequalities in the definitions of weak and strong thermodynamic constraints (Def. 2.6.2, 2.6.1) in linear and mixed integer linear programs. We will observe that this will lead to similar techniques as applied already for flux optimization.

In the second part of this chapter, we will discuss a method to tighten the ranges of feasible metabolite potentials. This does not only have applications for potential optimization, but can also be used to tighten constraints and detect blocked reactions for flux optimization.

5.1 Strict Inequalities

We recall that in the case of the strong (Def. 2.6.1) and weak (Def. 2.6.2) formulation of thermodynamic constraints, we have a constraint of the form

$$\mu S_r v_r < 0 \text{ or } v_r = 0 \text{ (and } \mu S_r = 0) \quad \forall r \in \mathcal{I}.$$

While we can deal with the “or” using standard integer programming techniques, the strict inequality makes problems. In the case of flux optimization for the weak formulation we were able to circumvent the problem by using a formulation based on infeasible sets and combinatorial Benders’ cuts (see Sec. 4). This only worked since the projection to the flux variables is topologically closed.

Let us now consider the very simple example network shown in Fig. 5.1 and assume that there is a fixed flux of $\mathbb{1}$ through the network. It follows that $\mu_A > \mu_B$ must hold. It follows that $\mu_A = 0$ is not feasible, because $\mu_B < 0$ is not allowed. However $\mu_A = \varepsilon$ for every $\varepsilon > 0$ is possible, because we can choose $\mu_B = 0$. Hence, the potential space is not

closed. We observe that it is irrelevant whether we use the weak or the strong definition of thermodynamic constraints, since we assumed a flux of 1 through all reactions.

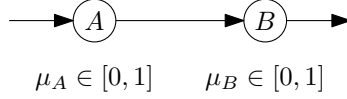


Figure 5.1: A very simple example network. The bounds on potentials are specified as closed intervals under the metabolites. It is easy to see (under the assumption that each reaction carries flux) that metabolite A cannot attain its minimal potential and B cannot attain its maximal potential. However, all other potentials can be attained.

We conclude that when we optimize on metabolite potentials, it may be that the optimization problem is bounded but an optimum does not exist. For many practical applications it may be sufficient to compute a near-optimal solution. Often found solutions (e.g. [69, 25, 137]) simply tighten each strict inequality $ax < b$ by an $\varepsilon > 0$ to $ax \leq b - \varepsilon$.

This approach can lead to numerical instabilities (ε usually has to be orders of magnitude smaller than the other parameters) and will always have the flavor of artificial arbitrariness. Hence, we rather would want to compute the supremum or infimum instead.

5.1.1 Strict Inequalities in Linear Programming

Let us consider the following feasible domain of an LP with strict inequalities:

$$F := \{x \in \mathbb{R}^m : Ax \leq e, Cx < f\}.$$

For sets $A \subseteq \mathbb{R}^m$ let \bar{A} denote the (topological) closure of A .

The question of how to deal with strict inequalities does not seem to be widely discussed in the literature. In the case of linear programs the reason is also rather evident as observed by Goberna et al. [56] (in the original formulation, they even allow an infinite set of constraints):

Proposition 5.1.1 (Prop. 1.1 in [56]) *If $F \neq \emptyset$, then it holds that $\bar{F} = F'$ for*

$$F' := \{x \in \mathbb{R}^m : Ax \leq e, Cx \leq f\}.$$

It follows that in the case of linear programming, we only have to make sure that the stated problem is feasible and then we can forget about the strict inequalities and solve using conventional methods. We observe that in the case of linear programming the shrinking of the constraints by ε is indeed unnecessary and only poses the risk of numerical instability.

5.1.2 Strict Inequalities in Mixed Integer Linear Programming

Let us here consider the following feasible domain of an MILP with strict inequalities:

$$F := \{(x, y) \in \mathbb{R}^m \times \mathbb{Z}^n : Ax + By \leq e, Cx + Dy < f\}.$$

In the case of MILP, the existence of a feasible solution in F is not sufficient to guarantee equivalence between the closure \bar{F} of F and its relaxation

$$F' = \{(x, y) \in \mathbb{R}^m \times \mathbb{Z}^n : Ax + By \leq e, Cx + Dy \leq f\}.$$

as the following example shows: The system $\{(x, y) \in \mathbb{R} \times \{0, 1\} : x > 0, x - y < 0\}$ has a feasible solution $(0.5, 1)$ but $(0, 0)$ is not in the closure, since there exists no feasible solution with $y = 0$.

However, it is sufficient to add cuts to exclude infeasible assignments.

For the following results define

$$\begin{aligned} F(y) &:= \{x \in \mathbb{R}^m : (x, y) \in F\} \\ F'(y) &:= \{x \in \mathbb{R}^m : (x, y) \in F'\} \\ Y^* &:= \{y \in \mathbb{Z}^n : F(y) \neq \emptyset\}. \end{aligned}$$

We observe that Y^* satisfies a certain convexity property, i.e., if $y' \in \mathbb{Z}^n$ can be written as a convex combination of points in Y^* , then $y' \in Y^*$. Hence, infeasible points can be cut off:

Lemma 5.1.1 *For each $y' \in \mathbb{Z}^n \setminus Y^*$ there exists a $g \in \mathbb{R}^n, h \in \mathbb{R}$ such that $gy \leq h$ for all $y \in Y^*$ and $gy' > h$.*

PROOF Assume that there exists no $g \in \mathbb{R}^n, h \in \mathbb{R}$ such that $gy \leq h$ for all $y \in Y^*$ and $gy > h$. This can be equivalently reformulated using the following LP:

$$\begin{aligned} 0 &\geq \max gy' - h \\ \text{s.t } &gy - h \leq 0 && \forall y \in Y^* \\ &g \in \mathbb{R}^n, h \in \mathbb{R} \end{aligned}$$

It can be easily seen that the LP is feasible and it is bounded by assumption. It follows by LP-duality that the dual LP is feasible and satisfies

$$\begin{aligned} 0 &\geq \min 0\alpha && (5.1) \\ &\sum_{y \in Y^*} \alpha y = y' \\ &\sum_{y \in Y^*} \alpha = 1 \\ &\alpha \geq 0. \end{aligned}$$

It follows that y' can be written as a convex combination of points in Y^* . By definition of Y^* there exists for each $y \in Y^*$ a $x(y) \in F(y)$ and hence $(x(y), y) \in F$. Let α be a feasible solution of LP (5.1). It follows that

$$y' = \sum_{y \in Y^*} \alpha y \quad \text{and define}$$

$$x' := \sum_{y \in Y^*} \alpha x(y).$$

By convexity it follows that $Ax' + By' \leq e$ and $Cx' + Dy' < f$. Since $y' \in \mathbb{Z}^n$ it follows that $(x', y') \in F$ and hence, $F(y') \neq \emptyset$ and $y' \in Y^*$. ■

Theorem 5.1.1 *There exists a (possibly infinite) set of linear inequalities $g_i y \leq h_i, i \in I$ such that*

$$\bar{F} = \{(x, y) \in F' : g_i y \leq h_i \forall i \in I\}.$$

PROOF By definition of Y^* it follows that

$$F = \bigcup_{y \in \mathbb{Z}^n} F(y) = \bigcup_{y \in Y^*} F(y). \quad (5.2)$$

By Lemma 5.1.1 there exists for each $y' \in I := \mathbb{Z}^n \setminus Y^*$ a $g_{y'} \in \mathbb{R}^n, h_{y'} \in \mathbb{R}$ with

$$\begin{aligned} g_{y'} y &\leq h_{y'} & \forall y \in Y^* \\ g_{y'} y' &> h_{y'}. \end{aligned}$$

It follows that

$$g_i y \leq h_i \forall i \in I$$

is satisfied for $y \in \mathbb{Z}^n$ if and only if $y \in Y^*$. By Prop. 5.1.1 we have $\overline{F(y)} = F'(y)$ for all $y \in Y^*$ and hence, it follows by (5.2) that

$$\begin{aligned} \bar{F} &\supseteq \bigcup_{y \in Y^*} \overline{F(y)} \\ &= \bigcup_{y \in \mathbb{Z}^n} \{x \in \mathbb{R}^m : Ax + By \leq e, Cx + Dy \leq f, g_i y \leq h_i \forall i \in I\} \\ &= \{(x, y) \in \mathbb{R}^m \times \mathbb{Z}^n : Ax + By \leq e, Cx + Dy \leq f, g_i y \leq h_i \forall i \in I\} \\ &= \{(x, y) \in F' : g_i y \leq h_i \forall i \in I\}. \end{aligned}$$

Assume now that $\bar{F} \supset \{(x, y) \in F' : g_i y \leq h_i \forall i \in I\}$. Hence there exists a sequence $(x_n, y_n)_{n \in \mathbb{N}} \in F$ with $(x_n, y_n) \rightarrow (x, y)$ and $(x, y) \notin F'$ or $g_i y > h_i$ for one $i \in I$.

Since $\bar{F} \subseteq F'$, it follows that there exists an $i \in I$ with $g_i y > h_i$. By definition of convergence, there exists a $(x', y') \in F$ with $\|x - x'\|_2^2 + \|y - y'\|_2^2 \leq \varepsilon$ for an $\varepsilon < 1$. Hence, it follows that $\|y - y'\|_2^2 < 1$ and thus, $y = y'$.

We conclude that $F(y) \neq \emptyset$ and hence $g_i y \leq h_i$ for all $i \in I$, a contradiction. ■

5.1.3 Integration into MILP-solver

By Prop. 5.1.1 we know a simple condition to check if an LP is consistent with the strict inequalities. We can apply this condition for each LP-relaxation that is solved during the solving of the MILP.

Since the integer variables y are treated as continuous variables in the LP relaxation, we also do not distinguish between them in this section. Instead we use y to denote those integer variables that have already been fixed and x denotes the continuous and unfixed variables. This applies to tentative solutions and nodes in the branch-and-bound tree.

5.1.3.1 Feasibility Check at Nodes

To check if a LP with strict inequalities is feasible, we simply have to solve:

$$\begin{aligned} v &= \max z \\ \text{s.t. } Ax + By &\leq e \\ Cx + Dy + \mathbf{1}z &\leq f \\ x &\in \mathbb{R}^m \\ z &\in \mathbb{R} \end{aligned}$$

This we can do for tentative solutions and for nodes of the branch-and-bound tree.

Additionally to the usual solving procedures, we obey the following three conditions:

- In the case of the branch-and-bound tree, $v \leq 0$ implies that the node is infeasible and can be cut off.
- In the case of a tentative solution, $v \leq 0$ implies the solution is infeasible and must be discarded.
- If we find a solution with $v > 0$, we know that there exists a solution satisfying the strict inequalities and any solution satisfying the relaxation with weak inequalities lies in the closure. Hence, we can continue as usual.

Since we check every solution for strict feasibility and only cut off nodes that cannot contain a solution satisfying all strict inequalities, it follows that the modified method will compute precisely an optimal solution in the closure of the strict feasible domain.

5.1.3.2 Adding Combinatorial Benders' Cuts for Boolean Decision Variables

Of course, when we find an infeasible solution, we want to learn from that infeasibility and cut off as many other similar infeasible solutions as possible. This can be done using a similar technique to combinatorial Benders' cuts [24]. Here we will only consider the case where the fixed decision variables y are Boolean.

Theorem 5.1.2 *Assume that $F'(y) \neq \emptyset$. Then $F(y) = \emptyset$ if and only if the following system is consistent:*

$$\begin{aligned} \alpha(e - By) + \beta(f - Dy) + \gamma &\leq 0 \\ \alpha A + \beta C &= 0 \\ \gamma + \beta \mathbf{1} &= 1 \\ \alpha, \beta, \gamma &\geq 0 \end{aligned} \tag{5.3}$$

PROOF Since $F'(y) \neq \emptyset$ it follows that the following system (for fixed y) is feasible and bounded:

$$\begin{aligned} z^* &= \max z \\ Ax + By &\leq e \\ \mathbf{1}z + Cx + Dy &\leq f \\ z &\leq 1 \end{aligned}$$

Hence, z^* exists and $z^* = 0$ if and only if $F(y) = \emptyset$ (since $F'(y) \neq \emptyset$, there exists a feasible solution with $z = 0$). Using the duality theorem, we conclude that $F(y) = \emptyset$ if and only if

$$\begin{aligned} 0 &= \min \alpha(e - By) + \beta(f - Dy) + \gamma \\ \alpha A + \beta C &= 0 \\ \gamma + \beta \mathbf{1} &= 1 \\ \alpha, \beta, \gamma &\geq 0 \end{aligned}$$

which is equivalent to consistency of

$$\begin{aligned} \alpha(e - By) + \beta(f - Dy) + \gamma &\leq 0 \\ \alpha A + \beta C &= 0 \\ \gamma + \beta \mathbf{1} &= 1 \\ \alpha, \beta, \gamma &\geq 0. \end{aligned}$$

■

Now we can use Theorem 5.1.2 to derive combinatorial Benders' cuts for a given solution (x^*, y^*) . To obtain strong cuts, we want to have as many non-zero coefficients in the cut as possible. Assume we have α, β, γ satisfying (5.3).

Recall that we assumed for this subsection that all decision variables y are Boolean. For this case we observe the following:

- If $y_i^* = 1$ and $\alpha B_i + \beta D_i \leq 0$, then α, β, γ will also prove infeasibility if $y_i = 0$ (and the rest of $y_j = y_j^*$ for all $j \neq i$). This is the case because the first inequality of (5.3) will stay satisfied and the rest of the inequalities do not involve y .

- Similarly, if $y_i^* = 0$ and $\alpha B_i + \beta D_i \geq 0$, then α, β, γ will also prove infeasibility if $y_i = 1$.

We can find such a solution by solving (with sufficient large constant M):

$$\begin{aligned}
 & \min \mathbf{1}v \\
 & \alpha B_i + \beta D_i \leq Mv_i && \text{for all } i \text{ with } y_i^* = 1 \\
 & \alpha B_i + \beta D_i \geq -Mv_i && \text{for all } i \text{ with } y_i^* = 0 \\
 & \alpha(e - By^*) + \beta(f - Dy^*) + \gamma = 0 \\
 & \alpha A + \beta C = 0 \\
 & \gamma + \beta \mathbf{1} = 1 \\
 & \alpha, \beta, \gamma \geq 0 \\
 & v \in \{0, 1\}^n
 \end{aligned}$$

Observe that, even if we solve the LP relaxation, we are likely to get a v with small support.

For each feasible solution v of this MILP and each feasible solution $(x, y) \in F$ of the original problem we know that y must be distinct from y^* in at least one index i , where $v_i > 0$, i.e.,

$$\sum_{i \in \text{supp}(v): y_i^* = 1} (1 - y_i) + \sum_{i \in \text{supp}(v): y_i^* = 0} y_i \geq 1.$$

We remark that although we know that even for general integer variable y there exists a valid inequality that cuts of the current solution (Lemma 5.1.1), it is unclear how such a cut can be computed efficiently.

5.1.4 Application to Thermodynamic Constraints in Metabolic Networks

Let us now come back to the motivation for which we looked at MILPs with strict inequalities: Thermodynamic constraints. Let us recall Prop. 5.1.1: If all integer decision variables are fixed, we only need to check if there exists a feasible solution and then work with the weak inequalities.

In the case of weak thermodynamic constraints w.r.t. a metabolite space $Q = \{\mu : \mu A \leq b\}$ the decision variables are the directions of the reactions which are implied by a given flux distribution. It follows that the feasibility test is precisely the same test that we already used and developed to test thermodynamic feasibility of a flux distribution. Furthermore, the cutting planes and branching strategies developed for flux optimization can also be applied to enforce the strict inequalities.

In the following we will see that the cuts that we derive for dealing with strict inequalities are the same inequalities that we have already been investigating for flux optimization.

We recall that we can formulate thermodynamic constraints as follows using Boolean decision variables (with sufficiently large M):

$$\begin{aligned}
 Sv &= 0 && \text{(steady-state assumption)} \\
 \ell &\leq v \leq u && \text{(flux bounds)} \\
 \mu A &\leq b && \text{(physiological potentials)} \\
 \Delta\mu - \mu S &= 0 && \text{(potential differences)} \\
 v &\leq Ma && \text{(indicator for possible fwd. flux)} \\
 v &\geq -Mb && \text{(indicator for possible bwd. flux)} \\
 \Delta\mu &< M(\mathbf{1} - a) && \text{(if fwd. flux, potential difference must be negative)} \\
 \Delta\mu &> -M(\mathbf{1} - b) && \text{(if bwd. flux, potential difference must be positive)} \\
 a + b &\leq \mathbf{1} && \text{(only fwd. or bwd. flux is possible)} \\
 a, b &\in \{0, 1\}^{\mathcal{I}}
 \end{aligned} \tag{5.4}$$

Since we now want to also optimize over the chemical potentials, we cannot project out μ as we did for flux optimization. As discussed in Sec. 5.1.3 we will deal with the strict inequalities by replacing them with weak inequalities and making sure that the subproblems stay feasible with respect to the strict inequalities.

As we previously observed, the link between the fluxes v and the potential differences $\Delta\mu$ is very weak due to the large (M) coefficients on the Boolean variables. Hence, in the following we will only consider restrictions in the form of fixed Boolean decision variables.

The fixed decision variables take effect by defining subsets N, P with $N \cup P \subset \mathcal{I}$. N is the set of reactions that must have negative potential difference. P is the set of reactions that must have positive potential difference. For the remaining reactions the flux direction has not yet been decided and hence, they do not pose any direct constraint on the sign of the potential difference. Thus, we consider the direction unconstrained.

We analyze the following system:

$$\begin{aligned}
 \mu S_i &< 0 && \forall i \in N \\
 \mu S_i &> 0 && \forall i \in P \\
 \mu A &\leq b
 \end{aligned}$$

Additionally, we will assume that the potential space is non-empty, i.e., $Q \neq \emptyset$.

Clearly, the system is consistent if the following optimization problem is feasible and has

an optimal solution greater than zero:

$$\begin{aligned}
 & \max z \\
 \text{s.t. } & \mu S_i + z \leq 0 && \forall i \in N \\
 & \mu S_i - z \geq 0 && \forall i \in P \\
 & \mu A \leq b \\
 & z \leq 1
 \end{aligned}$$

We observe that this optimization problem is just a reformulation of Eq. (2.7). Hence, we see that we do not have to develop a new theory for optimizing metabolite potentials but we can just reuse the results from flux optimization and apply them on MILP (5.4) with weak inequalities.

5.2 Improving Bounds on Potentials

Let us consider a metabolic network with n alternative pathways of length k from a metabolite A to a metabolite F , as shown in Fig. 5.2. It can easily be seen that no thermodynamically feasible flow is possible through this network due to the metabolite potential bounds. However, if we apply the approach of minimal infeasible sets this will not be determined immediately but only after checking k^n possibilities.

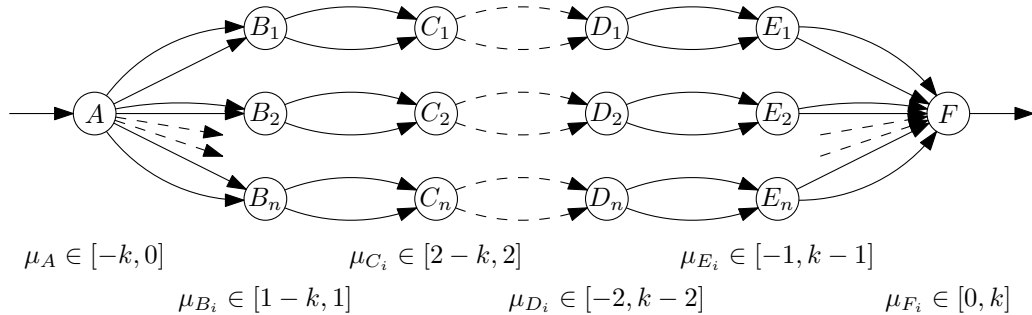


Figure 5.2: We observe that there is no thermodynamically feasible flow possible from metabolite A to metabolite F , since the highest potential at A ($\mu_A \leq 0$) is not bigger than the lowest potential at F ($\mu_F \geq 0$).

Since no two reactions are coupled, it follows that for each chain we get an infeasible set consisting of k pairs of parallel reactions. This induces (at least) k branching decisions. Since none of these branching decisions will influence the other chains, we have to do the branching decisions for the other chains also. It follows that we get k^n branching decisions. We observe that we are even rather lucky in this case, because with the extended branching scheme discussed in Sec. 4.4 it follows that with each branching decision on one reaction, we automatically also fix the potential difference and the parallel reac-

tion gets blocked automatically, too. Hence, for each pathway k branching decisions are sufficient to determine that the pathway cannot operate.

5.2.1 Introduction: the Graphic Case

The central idea of the approach is to abstract from steady-state fluxes and to just work in the metabolite space. Since the potential of a metabolite is unconstrained (except by the given potential bounds) if the metabolite is neither produced nor consumed, we study the range of feasible metabolite potentials under the assumption that the metabolite is produced and consumed. This condition can easily be translated to a condition in the potential space.

We will first analyze the graphic case, because it is easier. Let $G = (V, A)$ be a digraph (that represents the metabolic network). Let $W^- \subseteq V$ be the nodes with in-degree 0. We can understand these nodes as metabolites that can be supplied from the environment. Let $W^+ \subseteq V$ be the nodes with out-degree 0. We can understand these nodes as metabolites that can be secreted to the environment.

Let $\ell, u \in \mathbb{R}^V$ be lower and upper bounds on the potentials, i.e., we assume that the potential space has the form $Q = \{\mu \in \mathbb{R}^V : \ell \leq \mu \leq u\}$.

To simplify notation, we will allow $-\infty, \infty$ as feasible potential values to deal with metabolites that are essentially not producible, hence we will work with potentials $\mu \in \mathbb{R}_\infty^V$.

We will now develop a feasibility concept for the assumption that *every* metabolite is produced and consumed. Although this assumption will not be satisfied in practice, it will lead to a set of definitions that we can later use for less restricted cases.

In the graph-world this assumption means that for every node $v \in V \setminus W^-$ there must exist an arc $(w, v) \in A$ from a node w with higher potential (enforced by *upper-bound feasible*, see Def. 5.2.1) and for every $v \in V \setminus W^+$ there must exist an arc $(v, w) \in A$ to a node w with lower potential (enforced by *lower-bound feasible*, see Def. 5.2.2). Additionally the pre-defined upper bounds (for upper-bound feasible) and lower bounds (for lower-bound feasible) have to be satisfied. In general, it will be impossible to find a potential $\mu \in \mathbb{R}_\infty^V$ that is both upper-bound feasible and lower-bound feasible at the same time. However it is easy to see that it is always possible to find a potential $\mu \in \mathbb{R}_\infty^V$ that is only upper-bound feasible, or respectively lower-bound feasible.

Definition 5.2.1 (upper-bound feasible) $\mu \in \mathbb{R}_\infty^V$ is called *upper-bound feasible* if for all $v \in V$

- $\mu_v \leq \min(u_v, \max_{(w,v) \in A} \mu_w)$ or
- $\mu_v \leq u_v$ and $v \in W^-$. □

Definition 5.2.2 (lower-bound feasible) $\mu \in \mathbb{R}_\infty^V$ is called lower-bound feasible if for all $v \in V$

- $\mu_v \geq \max(\ell_v, \min_{(v,w) \in A} \mu_w)$ or
- $\mu_v \geq \ell_v$ and $v \in W^+$. □

While upper-bound feasible and lower-bound feasible define a local feasibility property, we can define global feasibility using paths.

Definition 5.2.3 (Path to $v \in V$) We call $P = (p_1, \dots, p_k)$ a path to v if

- $p_1 \in W^-$
- $p_k = v$
- $(p_i, p_{i+1}) \in A$ for all $1 \leq i < k$ □

Definition 5.2.4 (Path from $v \in V$) We call $P = (p_1, \dots, p_k)$ a path from v if

- $p_k \in W^+$
- $p_1 = v$
- $(p_i, p_{i+1}) \in A$ for all $1 \leq i < k$ □

We can now use the paths to define potentials $\mu^{\max} \in \mathbb{R}_\infty^V$ and $\mu^{\min} \in \mathbb{R}_\infty^V$ with:

$$\mu_v^{\max} := \max_{P \text{ path to } v} \left(\min_{w \in P} u_w \right)$$

$$\mu_v^{\min} := \min_{P \text{ path from } v} \left(\max_{w \in P} \ell_w \right)$$

We will now show that μ^{\max} and μ^{\min} are upper-bound feasible, resp. lower-bound feasible (Prop. 5.2.1, Cor. 5.2.2) and that the following theorem holds:

Theorem 5.2.1 Let $(u, w) \in A$. If $\mu_u^{\max} \leq \mu_w^{\min}$, then there exists no thermodynamically feasible flow through (u, w) . □

While in the case of graphs, we actually can compute μ^{\max} and μ^{\min} in polynomial time, this will not be the case for metabolic networks in general. There we will have to use the maximal upper-bound feasible potential and minimal lower-bound feasible potential to approximate μ^{\max} and μ^{\min} .

Corollary 5.2.1 *Let $\bar{\mu}^{\max}, \bar{\mu}^{\min} \in \mathbb{R}_{\infty}^V$ such that*

$$\begin{aligned} \mu &\leq \bar{\mu}^{\max} \text{ for all upper bound feasible } \mu \\ \mu &\geq \bar{\mu}^{\min} \text{ for all lower bound feasible } \mu \end{aligned}$$

Then it holds for $(u, w) \in A$ with $\bar{\mu}_u^{\max} \leq \bar{\mu}_w^{\min}$ that there exists no thermodynamically feasible flow through (u, w) .

PROOF By Prop. 5.2.1, Cor. 5.2.2 it follows that μ^{\max}, μ^{\min} are upper-bound feasible, resp. lower-bound feasible. Hence,

$$\mu_u^{\max} \leq \bar{\mu}_u^{\max} \leq \bar{\mu}_w^{\min} \leq \mu_w^{\min}$$

and the result follows by Thm. 5.2.1. ■

We now give the deferred proofs for the promised results.

Proposition 5.2.1 *μ^{\max} is upper-bound feasible.*

PROOF W.l.o.g. it suffices to take the maximum over only simple paths, because if a path contains a cycle, we could simply remove the cycle and would then only take the minimum over a subset of the original values.

Since there is only a finite number of simple paths in a graph, μ^{\max} is well defined.

We will show that the upper-bound property is satisfied for all $v \in V$.

If $v \in W^-$, there exists only one path to v , namely (v) . By definition $\mu_v^{\max} = u_v$, hence the property is satisfied for v .

Otherwise, $v \notin W^-$ and every maximizing path P to v has at least length 2. Hence, we can write $P = (p_1, \dots, p_k, v)$, where possibly $k = 1$. Define $Q := (p_1, \dots, p_k)$. Since Q is a path to p_k , it follows by definition of μ^{\max} that

$$\begin{aligned} \mu_{p_k}^{\max} &\geq \min_{w \in Q} u_w \\ \mu_v^{\max} &= \min_{w \in P} u_w = \min \left(u_v, \min_{w \in Q} u_w \right) \\ \Rightarrow \mu_v^{\max} &\leq \min \left(u_v, \mu_{p_k}^{\max} \right) \\ &\leq \min \left(u_v, \max_{(w,v) \in A} \mu_w^{\max} \right). \end{aligned}$$

Hence, μ^{\max} is upper-bound feasible. ■

Corollary 5.2.2 *μ^{\min} is lower-bound feasible.*

PROOF Let G' be the graph G with reversed arcs. Consequently, every path P from v in G bijectively corresponds to a path P' to v in G' (by reversing the path). Define additionally $\ell' = -u$ and $u' = -\ell$.

It follows that μ is lower-bound feasible in G w.r.t. ℓ, u if and only if $-\mu$ is upper-bound feasible in G' w.r.t. ℓ', u' . Also, the definition of μ^{\min} in G with ℓ, u is equivalent to the definition of μ^{\max} in G' with ℓ', u' .

Hence, by Prop. 5.2.1, the corollary follows. \blacksquare

PROOF (THM. 5.2.1) Assume there exists a thermodynamically feasible flow through (u, w) . It follows that there exists a path $P = (p_1, \dots, p_k = u, w = p_{k+1}, \dots, p_n)$ and potentials μ satisfying

$$\begin{aligned} \mu_{p_i} &> \mu_{p_{i+1}} && \forall i = 1, \dots, p_{n-1} \\ \ell_v &\leq \mu_v \leq u_v && \forall v \in P \end{aligned}$$

By definition of μ^{\max} and μ^{\min} , we have $\mu_u \leq \mu_u^{\max}$ and $\mu_w \geq \mu_w^{\min}$. It follows that

$$\mu_u \leq \mu_u^{\max} \leq \mu_w^{\min} \leq \mu_w,$$

a contradiction to $\mu_u = \mu_{p_k} > \mu_{p_{k+1}} = \mu_w$. \blacksquare

As concluding remark, we observe that the approximation of μ^{\max}, μ^{\min} by $\bar{\mu}^{\max}, \bar{\mu}^{\min}$ in Corollary 5.2.1 will be perfect in acyclic parts but can be arbitrarily bad if cycles are part of the network:

Proposition 5.2.2 *It holds that*

$$\begin{aligned} \mu_v^{\max} &= \min \left(u_v, \max_{(w,v) \in A} \mu_w^{\max} \right) \\ \mu_v^{\min} &= \max \left(\ell_v, \min_{(w,v) \in A} \mu_w^{\min} \right). \end{aligned}$$

PROOF We only show the result for μ^{\max} , since the result for μ^{\min} can then be obtained using the same construction as in Cor. 5.2.2.

Assume $v \in V$ fixed but arbitrary. If $\mu_v^{\max} = u_v$, the claim follows immediately. Hence, we will only consider the case where $\mu_v^{\max} < u_v$. Let $w \in V$ such that $(w, v) \in A$ and μ_w^{\max} is maximized. By definition of μ^{\max} , there exists a path $P = (p_1, \dots, p_k)$ to w with $\min_{x \in P} u_x = \mu_w^{\max}$. By defining $Q := (p_1, \dots, p_k, v)$ it follows that Q is a path to v and

$$\mu_v^{\max} \geq \min_{x \in Q} u_x = \min \left(u_v, \min_{x \in P} u_x \right) = \min(u_v, \mu_w^{\max}).$$

Since $\mu_v^{\max} < u_v$, it follows that $\mu_v^{\max} \geq \mu_w^{\max}$ and together with Prop. 5.2.1, we obtain equality. \blacksquare

Observation 5.2.1 *It is, however, not the case that μ^{\max} is the maximal upper-bound feasible μ . Consider a cycle, where all vertices have a maximal potential of 10 and all arcs to vertices of the cycle start at vertices with a maximal potential of 1. It follows that $\mu_v^{\max} = 1$ for all cycle vertices, although $\mu_v = 10$ would also lead to a feasible solution. \square*

We also observe that also with μ^{\max} and μ^{\min} we do not get an “if and only if” condition in Thm. 5.2.1. Let us consider the graph shown in Fig. 5.3. We observe that the arc from G to H can only be operated by a thermodynamically infeasible cycle. Hence, it cannot carry flow. However, Thm. 5.2.1 does not predict this, since $-1 = \mu_H^{\min} < \mu_G^{\max} = 1$.

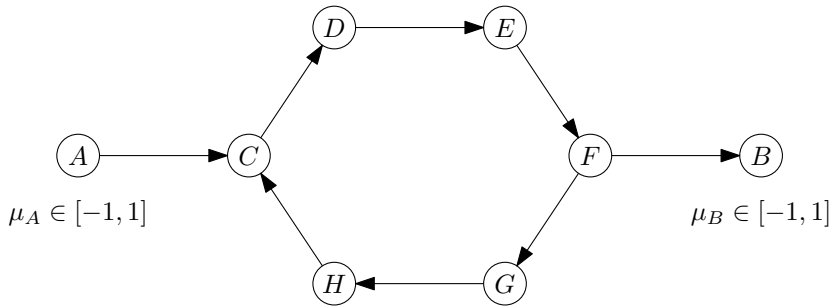


Figure 5.3: For the nodes A, B bounds on the potentials are given. The remaining nodes are unconstrained (or have irrelevant big bounds). Since every node is reachable from A , it follows that $\mu_v^{\max} = 1$ for each node v . Similarly since we can reach B from every node, it follows that $\mu_v^{\min} = -1$ for each node v .

5.2.1.1 Computation of μ^{\max} (μ^{\min} analogous)

In the case of graphs, we can compute μ^{\max} exactly in polynomial time using Alg. 5. Analogously, we can compute μ^{\min} .

We observe that in Alg. 5 every node is only updated once, since the update value decreases monotonically. It follows that we will only look at each edge only once (if implemented via adjacency lists). Hence, we obtain a running time of $O(|V| \log |V| + |A|)$ for Alg. 5.

5.2.2 The Flow Condition

Now we want to apply the ideas from the previous subsection to metabolic networks. For simplicity we assume for the rest of the theoretical work in this chapter that all reactions in the metabolic network are irreversible. This can be obtained by splitting all reversible reactions into a forward and a backward reaction. This simplification comes with the cost that circulation using only the forward and backward reaction become possible. This will require some additional consideration, which is discussed in Sec. 5.2.3.1.

Algorithm 5 This algorithm computes μ^{\max} in polynomial time

Input: A graph $G = (V, A)$
 $\mu_v^{\max} := -\infty$ for all $v \in V$.
 $\mu_v^{\max} = u_v$ for every $v \in W^-$.
 $Q := W^+$
while $Q \neq \emptyset$ **do**
 $v := \arg \max\{\mu_v^{\max} : v \in Q\}$
 for $w \in V$ with $(v, w) \in A$ **do**
 if $\mu_w^{\max} < \max\{u_w, \mu_v^{\max}\}$ **then**
 $\mu_w^{\max} := \max\{u_w, \mu_v^{\max}\}$
 $Q := Q \cup \{w\}$
 end if
 end for
end while
return μ^{\max}

The following results can also be formulated using pseudo-reactions (Def. 2.1.2) without splitting reversible reactions. This however leads to some notational inconveniences (for example for set intersection with the internal reactions), which is why it is not done here.

The problem with metabolic networks is that we do not have the notion of “path“ to a vertex anymore. However, we can characterize paths through the network.

Definition 5.2.5 (Path) We call $F \subseteq \mathcal{R}$ a path in a metabolic network $(\mathcal{M}, \mathcal{R}, S)$ if for every $m \in \mathcal{M}$ holds

$$(\exists r \in F : m \in r^+) \Leftrightarrow (\exists r \in F : m \in r^-)$$

We call $\mathcal{P} := \{F \subseteq \mathcal{R} : F \text{ is path}\}$ the set of all paths. □

Definition 5.2.6 (Steady-State path) We call $F \subseteq \mathcal{R}$ a steady-state path in a metabolic network $(\mathcal{M}, \mathcal{R}, S)$ if $F = \text{supp}(v)$ for $v \in \mathbb{R}^{\mathcal{R}}$ with $Sv = 0$ and $v \geq 0$. We call $\mathcal{P}_s := \{F \subseteq \mathcal{R} : F \text{ is steady-state path}\}$ the set of steady-state paths. □

Observation 5.2.2 $\mathcal{P}_s \subseteq \mathcal{P}$. □

Again, we assume that for every metabolite m , we are given an upper potential bound $u_m \in \mathbb{R}_\infty$ and a lower potential bound $\ell_m \in \mathbb{R}_\infty$.

Similar to Thm. 5.2.1, we want to find bounds μ^{\max}, μ^{\min} on potential vectors μ under the condition that the metabolite is produced / consumed. This leads us to the following definitions:

Definition 5.2.7 (Metabolite in a path) For $F \subseteq \mathcal{R}$ the set of produced / consumed metabolites is

$$\mathcal{M}(F) := \bigcup_{r \in F} \underline{r}.$$

Definition 5.2.8 (μ -bound) $\mu^{\max}, \mu^{\min} : \mathcal{R} \rightarrow \mathbb{R}_{\infty}^{\mathcal{M}}$ are called a μ -bound w.r.t. $\mathcal{Q} \subseteq \mathcal{P}$ of a metabolic network $(\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, \mathcal{S})$ if it holds for every path $F \in \mathcal{Q}$ and every $\mu \in \mathbb{R}^{\mathcal{M}}$ with

$$\begin{aligned}\mu &\leq u \\ \mu &\geq \ell \\ \mu S_{F \cap \mathcal{I}} &< 0\end{aligned}$$

that $\mu^{\min}(F)_m \leq \mu_m \leq \mu^{\max}(F)_m$ for all $m \in \mathcal{M}(F)$, where

$$\begin{aligned}\mu^{\min}(F)_m &:= \max_{r \in F} \mu^{\min}(r)_m \\ \mu^{\max}(F)_m &:= \min_{r \in F} \mu^{\max}(r)_m\end{aligned}$$

and $\ell \leq \mu^{\min}(r), \mu^{\max}(r) \leq u$ for all $r \in \mathcal{R}$. □

Note that μ -bounds additionally link a source reaction to each metabolite potential. In Sec. 5.2.3.1 we will see that this is an important feature to deal with reversible reactions. By trivial thermodynamic considerations, we know that for every path F for which there exists a $\mu \in \mathbb{R}^{\mathcal{M}}$ with

$$\mu S_{F \cap \mathcal{I}} < 0$$

it holds that it is impossible that $r, s \in F$ with $-r \equiv s$. Hence, if we want to check if flux through a reaction r is possible, we can exclude all paths that contain a reaction equivalent to $-r$.

Furthermore, we know that if a metabolite m is consumed/produced by F it must also be produced/consumed by reactions in the network. This allows us to further restrict the range of feasible values for μ_m (see Fig. 5.4). We therefore define $\mu^{\min*}, \mu^{\max*}$ for $F \subseteq \overline{\mathcal{R}}$ and $r \in \overline{\mathcal{R}}$ where $-F := \{s \in \mathcal{R} : r \in F, s \equiv -r\}$ with:

$$\begin{aligned}\mu^{\min*}(F)_m &:= \max \left\{ \min_{\substack{r \in \mathcal{R} \setminus -F \\ m \in r^-}} \mu^{\min}(r)_m, \mu^{\min}(F)_m \right\} \\ \mu^{\max*}(F)_m &:= \min \left\{ \max_{\substack{r \in \mathcal{R} \setminus -F \\ m \in r^+}} \mu^{\max}(r)_m, \mu^{\max}(F)_m \right\} \\ \mu^{\min*}(r) &:= \mu^{\min*}(\{r\}) \\ \mu^{\max*}(r) &:= \mu^{\max*}(\{r\})\end{aligned}$$

We observe the following:

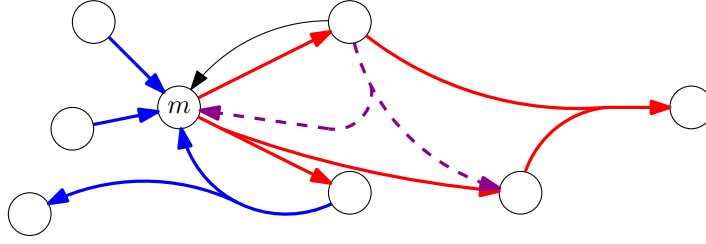


Figure 5.4: Consider the network without the dashed reaction. The reactions in F are marked in red. The reactions marked in blue additionally contribute to $\mu^{\max^*}(F)_m$. The black reaction does not contribute to $\mu^{\max^*}(F)_m$, because its reverse is already contained in F and hence, we would get a thermodynamically infeasible cycle if we would use it to produce m . Note that the reactions in F do not provide an upper bound on μ_m at all, i.e., $\mu^{\max}(F)_m = \infty$. If the purple reaction however, would be added to F , then it would also provide an upper bound on μ_m , since it has to be able to produce m .

Proposition 5.2.3 *It holds for $A \subseteq B \subseteq \mathcal{R}$ that*

$$\begin{aligned}\mu^{\min^*}(A) &\leq \mu^{\min^*}(B) \\ \mu^{\max^*}(A) &\geq \mu^{\max^*}(B)\end{aligned}$$

PROOF We only show the case for μ^{\min^*} , the case for μ^{\max^*} is analogous.

Let $m \in \mathcal{M}$ be arbitrary but fixed. Since $\{r \in \mathcal{R} \setminus -A : m \in r^-\} \supseteq \{r \in \mathcal{R} \setminus -B : m \in r^-\}$, it follows that

$$\min_{\substack{r \in \mathcal{R} \setminus -A \\ m \in r^-}} \mu^{\min}(r)_m \leq \min_{\substack{r \in \mathcal{R} \setminus -B \\ m \in r^-}} \mu^{\min}(r)_m.$$

Since $A \subseteq B$, it also follows that

$$\mu^{\min}(A)_m = \max_{r \in A} \mu^{\min}(r)_m \leq \max_{r \in B} \mu^{\min}(r)_m = \mu^{\max}(B)_m.$$

Hence, the proposition follows. ■

Proposition 5.2.4 *Let $F \subseteq \mathcal{R}$ with $-F \cap F = \emptyset$. Then it holds for all $m \in M := \{m \in \mathcal{M} : \exists r, s \in F : m \in r^+, m \in s^-\}$ that*

$$\begin{aligned}\mu^{\max^*}(F)_m &= \mu^{\max}(F)_m \\ \mu^{\min^*}(F)_m &= \mu^{\min}(F)_m\end{aligned}$$

PROOF Again, we only show the case for $\mu^{\min*}$, the case for $\mu^{\max*}$ is analogous.

Let $m \in M$ be arbitrary but fixed. Since $-F \cap F = \emptyset$, we observe that

$$\{r \in \mathcal{R} \setminus -F : m \in r^-\} \supseteq \{r \in F : m \in r^-\}.$$

Since $m \in M$ it follows that $\{r \in F : m \in r^-\} \neq \emptyset$ and hence,

$$\min_{\substack{r \in \mathcal{R} \setminus -F \\ m \in r^-}} \mu^{\min}(r)_m \leq \min_{r \in F : m \in r^-} \mu^{\min}(r)_m \leq \max_{r \in F} \mu^{\min}(r)_m = \mu^{\min}(F)_m.$$

By definition of $\mu^{\min*}$, it follows now that $\mu^{\min*}(F)_m = \mu^{\min}(F)_m$. ■

Note that M in Prop. 5.2.4 satisfies $M = \mathcal{M}(F)$ if $F \in \mathcal{P}$.

The usefulness of μ -bounds can be seen in the following theorem:

Theorem 5.2.2 *If μ^{\max}, μ^{\min} is a μ -bound w.r.t. $\mathcal{Q} \subseteq \mathcal{P}$ and $r \in \mathcal{I}$ with*

$$\mu^{\min*}(r)_{r^+} S_{r^+,r} + \mu^{\max*}(r)_{r^-} S_{r^-,r} \geq 0,$$

then there exists no thermodynamically feasible path $F \in \mathcal{Q}$ with $r \in F$.

PROOF Assume there exists a thermodynamically feasible path $F \in \mathcal{Q}$ with $r \in F$. Let μ denote the potentials that prove thermodynamic feasibility of F .

Since F is thermodynamically feasible it holds that

$$\begin{aligned} \mu &\leq u \\ \mu &\geq \ell \\ \mu S_{F \cap \mathcal{I}} &< 0 \end{aligned}$$

It follows that $\mu^{\min}(F)_m \leq \mu_m \leq \mu^{\max}(F)_m$ for all $m \in \mathcal{M}(F)$ by definition of μ -bound w.r.t. \mathcal{Q} . It is easy to see that no reaction in F is equivalent to $-r$ and hence, it follows for all $m \in \mathcal{M}(F)$ by Prop. 5.2.3 and Prop. 5.2.4 that

$$\mu^{\min*}(r)_m \leq \mu^{\min*}(F)_m = \mu^{\min}(F)_m \leq \mu_m \leq \mu^{\max}(F)_m = \mu^{\max*}(F)_m \leq \mu^{\max*}(r)_m.$$

Since $r \in F$, we also have $\underline{r} \subseteq \mathcal{M}(F)$ and thus,

$$\begin{aligned} 0 &\leq \mu^{\min*}(r)_{r^+} S_{r^+,r} + \mu^{\max*}(r)_{r^-} S_{r^-,r} \\ &\leq \mu_{r^+} S_{r^+,r} + \mu_{r^-} S_{r^-,r} && \text{(since } S_{r^+,r} \geq 0 \text{ and } S_{r^-,r} \leq 0) \\ &= \mu S_r \\ &< 0 && \text{(by thermodynamic feasibility)} \end{aligned}$$

This is a contradiction and hence, there cannot exist a path $F \in \mathcal{Q}$ with $r \in F$. ■

5.2.3 Updating Bounds

Let μ^{\max}, μ^{\min} be a μ -bound w.r.t. $\mathcal{Q} \subseteq \mathcal{P}$ of a metabolic network $(\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$. Let $\mu \in \mathbb{R}^{\mathcal{R}}$ with

$$\begin{aligned} \mu &\leq u \\ \mu &\geq \ell \\ \mu S_{F \cap I} &< 0 \end{aligned}$$

for a path $F \in \mathcal{Q}$.

We observe that it holds for every metabolite $m \in \mathcal{M}(F)$ and reaction $r \in F$ with $m \in r^+$ that

$$\begin{aligned} \mu S_r &< 0 \\ \Leftrightarrow \mu_m S_{mr} &< \sum_{x \in r^-} |S_{xr}| \mu_x - \sum_{x \in r^+ \setminus \{m\}} |S_{xr}| \mu_x \\ \Rightarrow \mu_m S_{mr} &< \sum_{x \in r^-} |S_{xr}| \mu^{\max}(F)_x - \sum_{x \in r^+ \setminus \{m\}} |S_{xr}| \mu^{\min}(F)_x \\ \Rightarrow \mu_m S_{mr} &< \sum_{x \in r^-} |S_{xr}| \mu^{\max^*}(F)_x - \sum_{x \in r^+ \setminus \{m\}} |S_{xr}| \mu^{\min^*}(F)_x \quad (\text{by Prop. 5.2.4}) \\ \Rightarrow \mu_m S_{mr} &< \sum_{x \in r^-} |S_{xr}| \mu^{\max^*}(r)_x - \sum_{x \in r^+ \setminus \{m\}} |S_{xr}| \mu^{\min^*}(r)_x \quad (\text{by Prop. 5.2.3}). \end{aligned}$$

We observe that the last equation holds irrespective of the path F and hence, if

$$\mu^{\max}(r)_m S_{mr} > \sum_{x \in r^-} |S_{xr}| \mu^{\max^*}(r)_x - \sum_{x \in r^+ \setminus \{m\}} |S_{xr}| \mu^{\min^*}(r)_x,$$

we can improve the bound of $\mu^{\max}(r)_m$ to

$$\mu^{\max}(r)_m := \frac{1}{S_{mr}} \left(\sum_{x \in r^-} |S_{xr}| \mu^{\max^*}(r)_x - \sum_{x \in r^+ \setminus \{m\}} |S_{xr}| \mu^{\min^*}(r)_x \right)$$

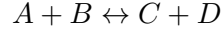
Similarly, if $m \in r^-$, we can update

$$\mu^{\min}(r)_m := \max \left\{ \mu^{\min}(r)_m, \frac{1}{S_{mr}} \left(\sum_{x \in r^+} |S_{xr}| \mu^{\min^*}(r)_x - \sum_{x \in r^- \setminus \{m\}} |S_{xr}| \mu^{\max^*}(r)_x \right) \right\}$$

5.2.3.1 Reversible Reactions

We observe that adding reactions to a network will never allow us to tighten a μ -bound, since we can always assume the additional reactions to be unused.

Hence, to observe that we have to pay special attention to reversible reactions, it is sufficient to consider a metabolic network that consists solely of the internal reversible reaction r :



Originally we assumed that all reactions are irreversible to simplify notation, which can be obtained by splitting reactions. Hence, when we talk about a reversible reaction, we mean that there exists a reaction $s \in \mathcal{R}$ with $s \equiv -r$. By abuse of notation, we also write $-r$ instead of s .

Let us further assume that $\ell_m = 20, u_m = 100$ for all $m = A, B, C, D$. It can easily be seen that many other values also produce the following undesirable effect.

Since r is the only reaction in the network, it follows that r can impossibly carry flow in any direction. However, if we want to update the bounds on r (with using $\mu^{\max^*}(\emptyset)$ instead of $\mu^{\max^*}(r)$) we obtain:

$$\begin{aligned} 100 &= \mu^{\max}(r)_C < \mu^{\max^*}(\emptyset)_A + \mu^{\max^*}(\emptyset)_B - \mu^{\min^*}(\emptyset)_D = 100 + 100 - 20 = 180 \\ 20 &= \mu^{\min}(r)_C > \mu^{\min^*}(\emptyset)_A + \mu^{\min^*}(\emptyset)_B - \mu^{\max^*}(\emptyset)_D = 20 + 20 - 100 = -60 \end{aligned}$$

Hence, the potential bounds of C cannot be tightened. By symmetry, also none of the other potential bounds can be tightened.

However, if we use the original definition, we observe that $\mu^{\max^*}(r)_A = -\inf = \mu^{\max^*}(r)_B$, since $-r$ is excluded as a producer of A and B and no other reaction exists that can produce A or B . In this case, we immediately deduce that r must be blocked.

Although we could have easily deduced in this example that r must be blocked by flux variability analysis, we observe that without the source-aspect in the definition of μ -bound we only would have very limited abilities to tighten bounds on metabolite potentials for potentials used in reversible reactions.

5.2.3.2 Using Flux Coupling

Let us now consider the two reactions shown in Fig. 5.5. Let us assume that $\mu_2, \mu_3 \in [-5, 5]$, $\mu_1 \in [-2, 1]$, and $\mu_4 \in [-1, 2]$. If we consider both reactions together, we clearly see that if one of the two reactions is used, it must hold that $\mu_1 \in [-1, 1]$ and $\mu_4 \in [-1, 1]$.

If we only analyze the reactions one after another, we cannot improve any bounds,

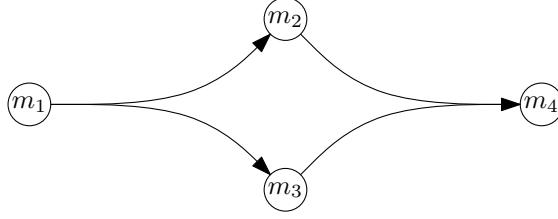


Figure 5.5: Using coupling information to propagate bounds can induce tighter bounds. Assume that metabolites m_1, m_4 are reused in the network, while m_2, m_3 only link the displayed reactions. Although we immediately see that $\mu_1 > \mu_4$ for all potential vectors μ that allow flow through any of the two reactions, the variability in the potentials for μ_2, μ_3 does not allow us to infer this property by propagating reactions one by one.

because

$$\begin{aligned}
 \mu_2^{\max} &= 5 \leq 1 + 5 = \mu_1^{\max} - \mu_3^{\min} \\
 \mu_3^{\max} &= 5 \leq 1 + 5 = \mu_1^{\max} - \mu_2^{\min} \\
 \mu_4^{\max} &= 2 \leq 5 + 5 = \mu_2^{\max} + \mu_3^{\max} \\
 \mu_2^{\min} &= -5 \geq -1 - 5 = \mu_4^{\min} - \mu_3^{\max} \\
 \mu_3^{\min} &= -5 \geq -1 - 5 = \mu_4^{\min} - \mu_2^{\max} \\
 \mu_1^{\min} &= -2 \geq -5 - 5 = \mu_2^{\min} + \mu_3^{\min}.
 \end{aligned}$$

Hence, we now want to include flux coupling data into the bound update mechanism. Therefore, we observe that we can also formulate the update step as a linear optimization problem (we only consider the case where $m \in r^+$, because $m \in r^-$ is analogous):

$$\begin{aligned}
 \mu^{\max}(r)_m &:= \sup \mu_m \\
 &\text{s.t. } \mu S_r < 0 \\
 &\mu \geq \mu^{\min*}(r) \\
 &\mu \leq \mu^{\max*}(r).
 \end{aligned}$$

Note that by Prop. 5.1.1, we can replace the sup by max and the strict inequalities by weak inequalities, which allows us to use ordinary LP solvers.

We can easily extend this formalism for cases where we not only want to compute an upper flux bound on m under the condition that reaction r carries flux but a set of reactions F carries flux:

$$\begin{aligned}
 \mu^{\max}(r)_m &:= \sup \mu_m \\
 \text{s.t. } \mu S_F &< 0 \\
 \mu &\geq \mu^{\min*}(F) \\
 \mu &\leq \mu^{\max*}(F).
 \end{aligned} \tag{5.5}$$

We encounter such a set F in directed flux coupling analysis: Let $F := \{s \in \overline{\mathcal{R}} : r \rightarrow s\}$ be the set of reactions that are directionally coupled from r , i.e., if r carries positive flux, then all the reactions in F also carry positive flux. Hence, if we compute the bound under the condition that r carries positive flux, we can also assume that all reactions in F carry positive flux.

In the case of the two coupled reactions in the example of Fig. 5.5, we obtain the constraints

$$\begin{aligned}
 \mu_2 + \mu_3 - \mu_1 &\leq 0 \\
 \mu_4 - \mu_2 - \mu_3 &\leq 0
 \end{aligned}$$

which imply that $\mu_4 \leq \mu_1$ and thus, we obtain the tightened bounds that we want.

5.2.3.3 Using EFMs through subnetworks

Let $A \subseteq \mathcal{R}$ be an arbitrary but fixed subset of reactions. By considering the set of elementary pathways through A , we can generalize the approach developed for flux coupling analysis.

Define

$$\begin{aligned}
 B^+ &:= \{m \in \mathcal{M} : m \in r^+, m \in s^- \exists r \in A, s \in \mathcal{R} \setminus A\} \\
 B^- &:= \{m \in \mathcal{M} : m \in r^-, m \in s^+ \exists r \in A, s \in \mathcal{R} \setminus A\}.
 \end{aligned}$$

We observe that B^+ is the set of metabolites that allow outflow of the subnetwork A , and B^- is the set of metabolites that allow inflow of A .

It is easy to see that the projection $v = w_A$ of every feasible flux vector $w \in \{w : Sw = 0, w \geq 0\}$ in the metabolic network satisfies

$$\begin{aligned}
 S_{\mathcal{M}(A) \setminus B^+, A} v &\leq 0 \\
 S_{\mathcal{M}(A) \setminus B^-, A} v &\geq 0 \\
 v &\geq 0.
 \end{aligned} \tag{5.6}$$

Therefore, we know that if there is flux through A , it can be obtained by a convex combination of the elementary modes E' of (5.6).

Since we are only interested in the pathways that contain our metabolite m of interest (for which we want to compute improved bounds), we can restrict the EFMs that we have to look at to

$$E := \{\text{supp}(e) : e \in E', m \in \underline{r} \exists r \in \text{supp}(e)\}.$$

Proposition 5.2.5 *Let $A \subseteq \mathcal{R}$ and $m \in \mathcal{M}(A) \setminus (B^+ \cap B^-)$. Let $F \in \mathcal{P}_s$. If $m \in \mathcal{M}(F)$, then there exists an $e \in E$ with $e \subseteq F$.*

PROOF If $m \notin B^+$, there exists no reaction outside of A that consumes m . It follows that if $m \in \mathcal{M}(F)$, there exists an reaction $r \in F \cap A$. Similarly we can conclude for $m \notin B^-$ that there exists an $r \in F \cap A$.

Since $F \in \mathcal{P}_s$, there exists a $w \in \mathbb{R}^{\mathcal{R}}$ with $Sw = 0, w \geq 0$. Therefore, w_A satisfied the equations in (5.6) and $w_A \in \text{cone}(E')$. Since $r \in F$ and hence, $w_r > 0$, it follows that there exists an elementary mode $e \in E'$ with $\text{supp}(e) \subseteq F$ and $e_r > 0$. By definition of E , it follows that $\text{supp}(e) \in E$ and the statement is proven. \blacksquare

Note, that we can reduce the space of feasible fluxes in (5.6) by using k -modules (see Ch. 6) to additionally restrict the space of feasible interface fluxes.

We observe now how we can use the set E to improve potential bounds on m :

Theorem 5.2.3 *Let $A \subseteq \mathcal{R}$ and E be defined as above. Let μ^{\min}, μ^{\max} be a μ -bound w.r.t $Q \subseteq \mathcal{P}_s$ and $F \in Q, \tilde{\mu} \in \mathbb{R}^{\mathcal{M}}$ with*

$$\begin{aligned} \ell &\leq \tilde{\mu} \leq u \\ \tilde{\mu} S_{F \cap \mathcal{I}} &< 0 \end{aligned}$$

Then it holds for every $m \in \mathcal{M}(A) \setminus (B^+ \cap B^-)$ with $m \in \mathcal{M}(F)$ that

$$\tilde{\mu}_m \leq \max_{e \in E} \sup \mu_m : \mu S_{e \cap \mathcal{I}} < 0, \mu^{\min^*}(e) \leq \mu \leq \mu^{\max^*}(e) \quad (5.7)$$

$$\tilde{\mu}_m \geq \min_{e \in E} \inf \mu_m : \mu S_{e \cap \mathcal{I}} < 0, \mu^{\min^*}(e) \leq \mu \leq \mu^{\max^*}(e) \quad (5.8)$$

PROOF Assume false, then w.l.o.g. there exists a $F \in \mathcal{P}_s, \tilde{\mu} \in \mathbb{R}^{\mathcal{M}}$ with $m \in \mathcal{M}(F), \ell \leq \tilde{\mu} \leq u$, and $\tilde{\mu} S_{F \cap \mathcal{I}} < 0$ with (min case is analogous)

$$\tilde{\mu}_m > \max_{e \in E} \sup \mu_m : \mu S_{e \cap \mathcal{I}} < 0, \mu^{\min^*}(e) \leq \mu \leq \mu^{\max^*}(e).$$

By Prop. 5.2.5 we know that there exists an $f \in E$ with $f \subseteq F$. By definition of μ -bound (Def. 5.2.8) and Prop. 5.2.4 it follows that

$$\tilde{\mu} S_{f \cap \mathcal{I}} < 0, \mu^{\min^*}(f) = \mu^{\min}(f) \leq \tilde{\mu} \leq \mu^{\max}(f) = \mu^{\max^*}(f).$$

Thus,

$$\begin{aligned}
 \tilde{\mu}_m &> \max_{e \in E} \sup \mu_m : \mu S_{e \cap \mathcal{I}} < 0, \mu^{\min^*}(e) \leq \mu \leq \mu^{\max^*}(e) \\
 &\geq \sup \mu_m : \mu S_{f \cap \mathcal{I}} < 0, \mu^{\min^*}(f) \leq \mu \leq \mu^{\max^*}(f) \\
 &\geq \tilde{\mu}_m,
 \end{aligned}$$

which is a contradiction. ■

In practice we may frequently save the effort of checking if a pathway is indeed able to satisfy the potential differences with strict inequalities:

Corollary 5.2.3 *Under the conditions of Thm. 5.2.3 it follows that*

$$\begin{aligned}
 \mu_m &\leq \max_{e \in E} \max \mu_m : \mu S_{e \cap \mathcal{I}} \leq 0, \mu^{\min^*}(e) \leq \mu \leq \mu^{\max^*}(e) \\
 \mu_m &\geq \min_{e \in E} \min \mu_m : \mu S_{e \cap \mathcal{I}} \leq 0, \mu^{\min^*}(e) \leq \mu \leq \mu^{\max^*}(e)
 \end{aligned}$$

PROOF By relaxing the strict inequality constraints, we only make the feasible solution space bigger. ■

However, with the relaxation we accept pathways that only operate with equality of potential differences. Without equality, some metabolites may not get any inflow and hence, their potential bound will be $-\infty$ or ∞ . In the strict formulation such effects cannot happen:

Corollary 5.2.4 *If we use $A = \mathcal{R}$ and $Q = \mathcal{P}_s$, then the bounds of Thm. 5.2.3 are tight.*

PROOF Let m be a fixed but arbitrary metabolite. Let $e \in E'$ with $\text{supp}(e) \in E$, $\mu \in \mathbb{R}^{\mathcal{M}}$ be the maximizer of (5.7) for μ_m . Since $A = \mathcal{R}$, it follows that $B^+ = B^- = \emptyset$ and w.l.o.g. $\mathcal{M}(A) = \mathcal{M}$ (we can ignore metabolites that are not contained in any reaction). It follows that we have

$$\begin{aligned}
 S_e &= 0 \\
 \mu S_r &< 0 && \forall r \in \text{supp}(e) \cap \mathcal{I} \\
 \ell &\leq \mu^{\min^*}(e) \leq \mu \leq \mu^{\max^*}(e) \leq u
 \end{aligned}$$

Hence, e is a feasible pathway in the whole network. Thus, there exists a flow through m and the bound value can be attained. ■

5.2.3.4 Using Branching

To be most effective, the set A should be chosen with care.

For the following, assume, we want to update the upper bound (lower bound is analogous). Let us also assume that all reactions are irreversible. Since we will compare different subsets of \mathcal{R} , let us denote $E(A)$ as the set of elementary modes E defined above for the set A .

In practice it usually is too much overhead if $\mu^{\max}(r), \mu^{\min}(r)$ also bound potentials of metabolites not produced/consumed by r , since the effect is too indirect. Hence, we assume that μ^{\max}, μ^{\min} only tighten bounds for metabolites that are produced/consumed by the respective reactions, i.e.,

$$\mu^{\max}(r)_n = u_n \quad \forall n \in \mathcal{M} \setminus r^+, r \in \mathcal{R} \quad (5.9)$$

$$\mu^{\min}(r)_n = \ell_n \quad \forall n \in \mathcal{M} \setminus r^-, r \in \mathcal{R} \quad (5.10)$$

Let m be the metabolite whose bound we want to update. Assume we want to update the bound coming from a reaction r (i.e., the bound under the assumption that r carries flux).

Define $C(r) := \{s : r \rightarrow s\}$ the set of reactions coupled to r .

We choose $A = \{r\} \cup C(r)$. Clearly, there exists only one EFM through A that contains r .

To strengthen the update, we now want to select a metabolite b that is either only consumed or only produced by A . For simplicity assume that b is consumed by A (production case is analogous). Let B be the set of reactions that produce b .

We now want to derive conditions for when we should try to extend the set A by B . First of all, since b is only consumed by A , we observe that

$$\mu^{\max^*}(A)_b = \max \left\{ \max_{s \in \mathcal{R} \setminus -A} \mu^{\max}(s)_b, \mu^{\max}(A)_b \right\} = \max_{s \in \mathcal{R} \setminus -A} \mu^{\max}(s)_b.$$

Hence, reactions s that achieve this optimum are particularly interesting and we derive the following necessary condition:

Proposition 5.2.6 *Assume no updates on μ^{\max}, μ^{\min} are possible using single-reaction updates, (5.9), (5.10) hold and b is a metabolite only consumed by A and B is the set of reactions producing b .*

Let

$$\begin{aligned} \bar{\mu}^D &:= \max_{\substack{a \in B \\ a \notin -A}} \max \left\{ \mu_m : \mu S_{A \cup \{a\}} \leq 0, \mu^{\min^*}(A \cup \{a\}) \leq \mu \leq \mu^{\max^*}(A \cup \{a\}) \right\} \\ \bar{\mu}^A &:= \max \left\{ \mu_m : \mu S_A \leq 0, \mu^{\min^*}(A) \leq \mu \leq \mu^{\max^*}(A) \right\}. \end{aligned}$$

If $\bar{\mu}^D < \bar{\mu}^A$ then every reaction $a \in B \setminus -A$ with

$$\mu^{\max}(a)_b = \mu^{\max^*}(A)_b \quad (5.11)$$

satisfies $|\underline{a} \cap \mathcal{M}(A)| \geq 2$.

PROOF Assume $\bar{\mu}^D < \bar{\mu}^A$ and there exists a reaction $a \in B \setminus -A$ satisfying (5.11) and $|\underline{a} \cap \mathcal{M}(A)| = 1$.

Define $A' = A \cup \{a\}$. We observe by (5.9), (5.10) that

$$\begin{aligned} \mu^{\max^*}(A')_{\underline{a} \setminus \mathcal{M}(A)} &= \mu^{\max^*}(\{a\})_{\underline{a} \setminus \mathcal{M}(A)} \\ \mu^{\min^*}(A')_{\underline{a} \setminus \mathcal{M}(A)} &= \mu^{\min^*}(\{a\})_{\underline{a} \setminus \mathcal{M}(A)} \\ \mu^{\max^*}(A')_{\mathcal{M}(A) \setminus \underline{a}} &= \mu^{\max^*}(A)_{\mathcal{M}(A) \setminus \underline{a}} \\ \mu^{\min^*}(A')_{\mathcal{M}(A) \setminus \underline{a}} &= \mu^{\min^*}(A)_{\mathcal{M}(A) \setminus \underline{a}} \end{aligned}$$

Furthermore, since

$$\begin{aligned} \mu^{\max^*}(A')_b &= \min \left\{ \max_{s \in \mathcal{R} \setminus -A'} \mu^{\max}(s)_b, \mu^{\max}(A')_b \right\} \\ &\geq \min \left\{ \mu^{\max}(a)_b, \mu^{\max}(A')_b \right\} && \text{(since } a \notin -A') \\ &= \min \left\{ \mu^{\max}(a)_b, \mu^{\max}(A)_b \right\} && \text{(by Def. of } \mu^{\max}) \\ &\geq \min \left\{ \mu^{\max}(a)_b, \mu^{\max^*}(A)_b \right\} && \text{(by Def. of } \mu^{\max^*}) \\ &= \mu^{\max^*}(A)_b && \text{(by (5.11))} \\ &\geq \mu^{\max^*}(A')_b && \text{(by Prop. 5.2.3)} \end{aligned}$$

it follows that $\mu^{\max^*}(A')_b = \mu^{\max^*}(A)_b$.

Similarly, we conclude from

$$\begin{aligned} \mu^{\min^*}(A')_b &= \mu^{\min}(A')_b && \text{(Prop. 5.2.4)} \\ &= \max \left\{ \mu^{\min}(a)_b, \mu^{\min}(A)_b \right\} && \text{(by Def. of } \mu^{\min}) \\ &= \mu^{\min}(A)_b && \text{(since } \mu^{\min}(a)_b = \ell_b) \\ &\leq \mu^{\min^*}(A)_b && \text{(by Def. of } \mu^{\min^*}) \\ &\leq \mu^{\min^*}(A')_b && \text{(by Prop. 5.2.3)} \end{aligned}$$

that $\mu^{\min^*}(A')_b = \mu^{\min^*}(A)_b$.

Since no updates on μ^{\max}, μ^{\min} are possible using single-reaction updates, there exists $\mu_{\underline{a}}^1$ with $\mu_b^1 = \mu^{\max}(a)_b, \mu_{\underline{a}}^{\min^*}(a)_{\underline{a}} \leq \mu_{\underline{a}}^1 \leq \mu^{\min^*}(a)_{\underline{a}}$ and $\mu S_a \leq 0$.

Let μ^2 be an optimizer of $\bar{\mu}^A$. Since $\mu^{\max}(a)_b = \mu^{\max^*}(A)_b$, it follows that $\mu_b^2 \leq \mu^{\max}(a)_b$. Define now $\mu^3 \in \mathbb{R}^{\mathcal{M}}$ by

$$\begin{aligned} \mu_{\mathcal{M}(A)}^3 &:= \mu_{\mathcal{M}(A)}^2 \\ \mu_{\mathcal{M} \setminus \mathcal{M}(A)}^3 &:= \mu_{\mathcal{M} \setminus \mathcal{M}(A)}^1. \end{aligned}$$

It follows by construction that $\mu^3 S_A \leq 0$ and since $\mu_b^3 \leq \mu_b^1$ that $\mu^3 S_a \leq 0$. From the previous observations, it is also easy to see that

$$\begin{aligned} \mu^{\min*}(A')_{\mathcal{M}(A)\setminus a} &= \mu^{\min*}(A)_{\mathcal{M}(A)\setminus a} \leq \mu^3_{\mathcal{M}(A)\setminus a} \leq \mu^{\max*}(A)_{\mathcal{M}(A)\setminus a} = \mu^{\max*}(A')_{\mathcal{M}(A)\setminus a} \\ \mu^{\min*}(A')_{a\setminus\mathcal{M}(A)} &= \mu^{\min*}(\{a\})_{a\setminus\mathcal{M}(A)} \leq \mu^3_{a\setminus\mathcal{M}(A)} \leq \mu^{\max*}(\{a\})_{a\setminus\mathcal{M}(A)} = \mu^{\max*}(A')_{a\setminus\mathcal{M}(A)} \\ \mu^{\min*}(A')_b &= \mu^{\min*}(A)_b \leq \mu_b^3 \leq \mu^{\max*}(A)_b = \mu^{\max*}(A')_b. \end{aligned}$$

Thus, μ^3 satisfies all constraints for a feasible solution of $\bar{\mu}^D$ and thus, $\bar{\mu}^D \geq \bar{\mu}^A$, which is a contradiction and concludes the proof. \blacksquare

We conclude that we should not branch on a metabolite b (i.e., investigate possibilities for extending the set A via this metabolite) if there exists a reaction a that only induces the active bound of the metabolite b , but does not have any interaction with the other reactions in A . Such kinds of updates are already covered by normal bound propagation.

Of course the applications of Prop. 5.2.6 is rather limited, because we cannot make such a statement as soon as there are additional reactions coupled to a that then indirectly interact with A in the end. Similar situations occur if we decide to continue branching after branching on b .

Since branching is subject to combinatorial explosion, it should be used very restrictively. In the most extreme case, we might decide to only branch if a sufficient condition for improvement is satisfied. For example, the following is a sufficient condition on whether we will see an improvement by branching on b :

Proposition 5.2.7 *Assume no updates on μ^{\max}, μ^{\min} are possible using single-reaction updates, (5.9), (5.10) hold and b is a metabolite only consumed by A and B is the set of reactions producing b .*

Consider the LP

$$\bar{\mu}^A := \max \{ \mu_m : \mu S_A \leq 0, \mu^{\min*}(A) \leq \mu \leq \mu^{\max*}(A) \} \quad (5.12)$$

with the dual

$$\min \mu^{\min*}(A)y + \mu^{\max*}(A)z : S_A x + y + z = 0, x \geq 0, y \leq 0, z \geq 0.$$

Assume the following conditions are satisfied for an optimizer (x, y, z) of the dual:

- $z_b > 0$
- For every reaction $a \in B \setminus -A$ with $\mu^{\max}(a)_b = \mu^{\max*}(A)_b$ there exists a metabolite c with
 - $z_c > 0$ and there exists a reaction $d \in C(a)$ with $\mu^{\max}(d)_c < \mu^{\max*}(A)_c$ or
 - $y_c < 0$ and there exists a reaction $d \in C(a)$ with $\mu^{\min}(d)_c > \mu^{\min*}(A)_c$.

Then it holds that

$$\bar{\mu}^A > \bar{\mu}^D := \max_{\substack{a \in B \\ a \notin -A}} \max \left\{ \mu_m : \begin{array}{l} \mu S_{A \cup C(a)} \leq 0, \\ \mu^{\min^*}(A \cup C(a)) \leq \mu \leq \mu^{\max^*}(A \cup C(a)) \end{array} \right\}. \quad (5.13)$$

PROOF First of all, we observe that the LP (5.13) only has additional constraints and tightened bounds compared to LP (5.12) by Prop. 5.2.3. Hence, every dual solution of (5.12) is also dual feasible for (5.13). In particular, (x, y, z) is a dual feasible solution of LP (5.13).

Since b is only consumed by A , we observe that for every $a \in B \setminus -A$ it holds that

$$\mu^{\max}(a)_b \leq \max_{s \in \mathcal{R} \setminus -A} \mu^{\max}(s)_b = \min \left\{ \max_{s \in \mathcal{R} \setminus -A} \mu^{\max}(s)_b, \mu^{\max}(A)_b \right\} = \mu^{\max^*}(A)_b.$$

Hence, if $\mu^{\max}(a)_b \neq \mu^{\max^*}(A)_b$, then $\mu^{\max}(a)_b < \mu^{\max^*}(A)_b$. In this case it follows that $\mu^{\max^*}(A \cup C(a))_b < \mu^{\max^*}(A)_b$ and since $z_b > 0$, we conclude (using Prop. 5.2.3) that

$$\begin{aligned} \bar{\mu}^A &= \mu^{\min^*}(A)y + \mu^{\max^*}(A)z \\ &> \mu^{\min^*}(A \cup C(a))y + \mu^{\max^*}(A \cup C(a))z \\ &\geq \max \left\{ \mu_m : \begin{array}{l} \mu S_{A \cup C(a)} \leq 0, \\ \mu^{\min^*}(A \cup C(a)) \leq \mu \leq \mu^{\max^*}(A \cup C(a)) \end{array} \right\}. \quad (\text{by LP duality}) \end{aligned}$$

Thus, we now only have to consider reactions $a \in B \setminus -A$ with $\mu^{\max}(a)_b = \mu^{\max^*}(A)_b$. Assume $\mu^{\max}(d)_c < \mu^{\max^*}(A)_c$ with $z_c > 0$. It follows by Prop. 5.2.3 that

$$\mu^{\max^*}(A \cup C(a))_c \leq \mu^{\max}(d)_c < \mu^{\max^*}(A)_c.$$

Therefore, $\mu^{\max}(A \cup C(a))_c z_c < \mu^{\max^*}(A)_c z_c$ and by Prop. 5.2.3 it follows that

$$\begin{aligned} &\mu^{\max^*}(A \cup C(a))z < \mu^{\max^*}(A)z \\ \Rightarrow &\mu^{\min^*}(A \cup C(a))y + \mu^{\max^*}(A \cup C(a))z < \mu^{\min^*}(A)y + \mu^{\max^*}(A)z. \end{aligned}$$

The case for $\mu^{\min}(d)_c > \mu^{\min^*}(A)_d$ with $y_z < 0$ is analogous and we also derive

$$\mu^{\min^*}(A \cup C(a))y + \mu^{\max^*}(A \cup C(a))z < \mu^{\min^*}(A)y + \mu^{\max^*}(A)z.$$

It follows that

$$\begin{aligned} \bar{\mu}^A &= \mu^{\min^*}(A)y + \mu^{\max^*}(A)z \\ &> \mu^{\min^*}(A \cup C(a))y + \mu^{\max^*}(A \cup C(a))z \\ &\geq \max \left\{ \mu_m : \begin{array}{l} \mu S_{A \cup C(a)} \leq 0, \\ \mu^{\min^*}(A \cup C(a)) \leq \mu \leq \mu^{\max^*}(A \cup C(a)) \end{array} \right\} \quad (\text{by LP duality}), \end{aligned}$$

which proves the proposition. ■

Hence, the dual solution of (5.12) gives important information about which metabolites are preferred branching targets. Common LP solvers provide this information usually for free.

5.2.3.5 Using 1-Modules

Chains of reactions that are linked by multiple metabolites in parallel are not likely to propagate bounds in full strength. However, if we can group reactions to a linear 1-module (Def. 6.1.2), we can replace the reactions with just one reaction representing the 1-module (see Sec. 6.4). In general, such a replacement will not always be useful, because the linear 1-module will not respect the potentials of metabolites hidden by the module. Hence, it may actually give worse bounds. Since we can choose, we can simply go for the smaller one in the course of the algorithm.

However, most of the linear 1-modules in metabolic networks are sets of fully coupled reactions, where already the methods using flux coupling data give the same results. Another frequently occurring 1-module are sets of parallel reactions. In this case, even the single reaction propagation method is sufficiently powerful.

This only leaves a few sets of 1-modules that are a combination of fully coupled reactions and parallel reactions or indeed non-decomposable 1-modules, see Sec. 6.4.2. Hence, I did not implement this approach.

5.2.4 Application on a Genome-Scale Network of *E. coli*

5.2.4.1 Method

I tested the effects of potential propagation (in terms of blocked reactions) on a modified *E. coli* iAF1260 model. The modification consisted in turning all internal reactions into reversible reactions. Potential bounds were obtained from the equilibrium constant estimates given in the supplementary material to the paper on the model [43]. According to the uncertainties δ in equilibrium constants μ^0 (both values obtained from [43]) and to respect metabolite concentrations between $c_{\min} = 1nM$ and $c_{\max} = 100mM$, the bounds for metabolite potentials were computed by (see also Sec. 2.6.3)

$$\begin{aligned}\ell &= \mu^0 - \delta + RT \ln(c_{\min}) = \mu^0 - \delta - 12.2 \\ u &= \mu^0 + \delta + RT \ln(c_{\max}) = \mu^0 + \delta - 1.4\end{aligned}$$

I chose $c_{\min} = 1nM$, because a concentration of $1nM$ corresponds to less than 1 molecule per cell [154]. As maximum concentration I chose $c_{\max} = 100mM$, because this is about the maximal metabolite concentration measured in *E. coli* [12]. The only exception is water, where the concentration is fixed to $1M$ by default.

I implemented the update algorithm using μ -bounds that are specific to the source reaction that feed the respective metabolites. However, I only identify forward and backward reaction directions for reactions that are marked as reversible. The *E. coli* iAF1260 network also contains duplicate reactions. I did not implement a method that detects these duplicate reactions. Hence, the problematic effect of reversible reactions will still occur for these cases. Since there is only a small number of such reactions, I decided that the implementation overhead is probably not worth the improvement.

For the integration of flux coupling data, I did not use information from a full FCA, but I just used trivial flux coupling information. This means, I only looked at metabolites that are only consumed / produced by one reaction. For example, if a metabolite is only consumed by one reaction r , then r is directionally coupled from all reactions that produce the metabolite. Such flux couplings are of course not comprehensive, but can be computed very efficiently.

To test the effects of potential propagation, I measured the number of reaction directions that were inferred to be blocked. Since potential propagation and flux variability analysis are two contrasting methods to infer blocked reactions, I applied them iteratively one after the other.

5.2.4.2 Results

The results using the different improvements can be seen in Fig. 5.6.

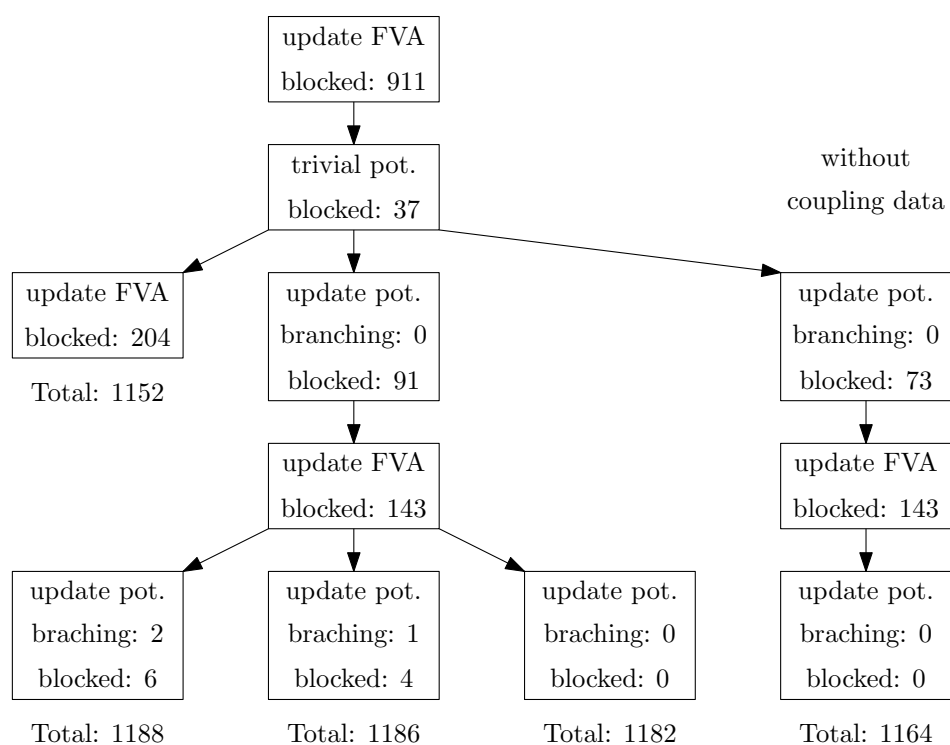


Figure 5.6: Results using the different improvements for propagating bounds.

We observe that although most of the blocked reaction directions originate from blocked (directions of) exchange reactions, about 270 reaction directions can be inferred using thermodynamic considerations. Important is the alternation between potential propagation and FVA. Already the 37 reaction directions that can be determined to be blocked by trivial thermodynamic considerations imply 204 additional blocked reaction directions

that can be determined by FVA.

If we would on the other hand only run potential propagation instead, we would just find 91 additional blocked reactions. It should be noted that a large part of these 91 additional blocked reactions are also determined to be blocked by FVA.

Furthermore, we observe that the integration of even trivial coupling data (metabolites that are only produced/consumed by one reaction) leads to about three times the number of additionally blocked reactions compared to the number of reactions that can be inferred as blocked from FVA and trivial thermodynamic considerations. Without flux coupling data, we find $1164 - 1152 = 12$ additional blocked reactions, while with flux coupling data we find $1182 - 1152 = 30$ additional blocked reactions.

We also note that the integration of branching strategies has only a mild impact. By performing 1 branching operation per update step, we obtain 4 additional blocked reactions and with 2 branching steps, we get 6 additional blocked reactions.

5.2.4.3 Conclusions

Overall, we conclude that, although non-trivial blocked reactions can be inferred, the impact of the method is rather disappointing. In contrast to the 2382 reactions of the *E. coli* iAF1260 network, the number of 36 additional reaction directions that could be identified as blocked is very small.

In comparison, the curated network contains 1807 irreversible reactions of which are 4 blocked. FVA on this network alone already identifies 2132 reactions that are blocked in backward direction and 1015 reactions that are blocked in forward direction.

What could be the reasons for this weak performance? One possibility could be that the method, even with the branching feature is not strong enough to unravel the complexities of the metabolic network. Also, the few reactions in the network that occur twice could be preventing us from deriving stronger results. However, since the improvements obtainable by branching on metabolites are very small, I do not think that this is the main cause.

Rather, I expect that the effect is due to the uncertainties in the potential bounds. For example, Tepper et al. [154] work with a minimal metabolite concentration of $10nM$ which would increase metabolite potential lower bounds by 1.4. Also, the current approach is neglecting that uncertainties on equilibrium constants are not independent [107]. However, it is unclear to me at the moment how improved models for equilibrium constant uncertainties, as discussed in Sec. 2.6.3.2 can be integrated best into bound propagation. One possibility would be to not propagate bounds on the potentials, but bounds on the logarithms of concentrations. As we will see in Sec. 5.2.5 this approach however, is not guaranteed to be better than the original approach.

5.2.5 Dependence of Uncertainties in Equilibrium Constants

Noor et al. [107] showed that information of equilibrium constants on internal reactions can be much more precise than data obtained simply by group contribution method.

To use data of equilibrium constants on reactions, let us assume that it is given in the following way:

- We are given an upper and lower bound ℓ_r^0, u_r^0 on the equilibrium constant $\Delta\mu_r^0$ for each internal reaction $r \in \mathcal{I}$.
- We are given a consistent vector of reference equilibrium constants $\overline{\Delta\mu^0}$, i.e., $\ell^0 \leq \overline{\Delta\mu^0} \leq u^0$ and $\overline{\Delta\mu^0} = \mu^0 S$ has a solution.
- We are given a lower and upper bound c_m^{\min}, c_m^{\max} on the concentration c_m for each metabolite $m \in \mathcal{M}$.
- We assume that the concentrations are already given in a logarithmic scale, i.e., the potential difference $\Delta\mu_r$ of a reaction $r \in \mathcal{I}$ is computed by $\Delta\mu_r = \Delta\mu_r^0 + cS_r$.

Instead of propagating bounds on potentials, we could now propagate bounds on the concentrations:

$$\begin{aligned}
 c^{\max}(v, r) &= \max c_v \\
 \text{s.t. } &K\Delta\mu^0 = 0 \\
 &\Delta\mu_r^0 + cS_r \leq 0 \\
 &\ell^0 \leq \Delta\mu^0 \leq u^0 \\
 &c^{\min} \leq c \leq c^{\max}
 \end{aligned} \tag{5.14}$$

Note that we could strengthen this formalism and exclude cycles formed by forward and backward reactions similar to $\mu^{\min*}, \mu^{\max*}$. For simplicity reasons, this is omitted here.

We want to investigate how the formulation (5.14) compares to the classical formulation (which uses bounds on potentials). We will do this by analyzing when reactions can be inferred as blocked. To be able to compare the two methods, we have to translate the bounds on equilibrium constants on the reactions $\Delta\mu^0$ to bounds on equilibrium constants on metabolites μ^0 .

Using $\overline{\Delta\mu^0}$, we can find an assignment of μ^0 by solving

$$\overline{\Delta\mu^0} = \mu^0 S_{\mathcal{I}}.$$

However, this system is usually highly under-determined. Clearly, we can define a matrix J and a vector k such that the system

$$\begin{aligned}
 \overline{\Delta\mu^0} &= \mu^0 S_{\mathcal{I}} \\
 k &= \mu^0 J
 \end{aligned} \tag{5.15}$$

has a unique solution. A typical choice is to define the equilibrium constants of some metabolites to be 0 [5]. This can be realized by choosing $k = 0$ and J as the corresponding sub-matrix of the identity matrix.

For fixed J and k , we can now compute lower and upper bounds on the metabolite equilibrium constants ℓ_m^μ, u_m^μ for each $m \in \mathcal{M}$:

$$\ell_m^\mu := \min \mu_m^0 : \Delta\mu^0 = \mu^0 S, k = \mu^0 J, \ell^0 \leq \Delta\mu^0 \leq u^0 \quad (5.16)$$

$$u_m^\mu := \max \mu_m^0 : \Delta\mu^0 = \mu^0 S, k = \mu^0 J, \ell^0 \leq \Delta\mu^0 \leq u^0 \quad (5.17)$$

5.2.5.1 Where Formulation 5.14 Fails

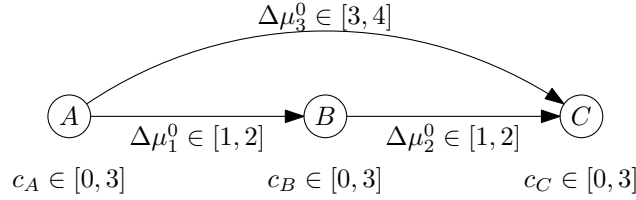


Figure 5.7: Example where concentration propagation does not detect an infeasible pathway.

In Figure 5.7 we see a network, where no flow from A to C is possible. This can be seen at reaction 3 which has an equilibrium constant of at least 3, hence this cannot be canceled by the concentrations of the metabolites (at most a potential difference of 0 is achievable).

If we use the classical formulation and choose J, k such that the equilibrium constant of metabolite A is fixed to $\mu_A^0 = 0$, it follows by reaction 3 that $\mu_C^0 \in [3, 4]$. We compute potential bounds $\mu_A \in [0, 3]$, $\mu_B \in [1, 5]$ and $\mu_C \in [3, 7]$. Propagation of potential bounds yield an upper potential bound for each metabolite of 3 and also a lower potential bound of 3. Hence, classical potential propagation infers that no flux is possible.

If, however, we use concentration propagation, we infer that $c_B \in [0, 2]$ and $c_C \in [0, 1]$ by propagating in forward direction and if we propagate in backward direction, we additionally get $c_B \in [1, 2]$ and $c_A \in [2, 3]$. It follows for reaction 1 that $\Delta\mu_1 \in [1, 2] + [1, 2] - [2, 3] = [-1, 2]$. Hence a negative potential difference is possible and hence, we cannot derive a blocked flux. The same applies for reaction 2. Reaction 3 is derived as blocked, but one path is sufficient to transport flow.

5.2.5.2 Where the Classical Formulation Fails

First of all, we observe that J, k can always be chosen so badly that the inferred bounds on μ^0 are so weak that no blocked reactions can be inferred. For example, we can simply

choose J to be a slightly perturbed submatrix of the stoichiometric matrix S . Consider the reaction $A \rightarrow B$ with $\Delta\mu^0 \in [1, 2]$. By adding the equality $(1 - \varepsilon)\mu_B^0 - \mu_A^0 = 0$, we obtain as bounds for μ_B that

$$\frac{1}{\varepsilon} \leq \mu_B \leq \frac{2}{\varepsilon}$$

which is clearly covering a much wider range than the interval $[1, 2]$ and thus it ceases to be possible to infer that $A \rightarrow B$ can only operate in backward direction. Hence, we assume in the following that J, k are chosen in the best possible way, resp. ask the question if there exists a choice of J, k that imply potential bounds on μ^0 from which we can infer that a given reaction is blocked. It should be noted though that in practice we cannot assume in general that J and k are chosen as best as possible.

It is much harder to come up with an example, where there exists no choice of J, k (such that (5.15) has a unique solution) s.t. that the classical formulation detects no reaction as blocked. A potential example can be seen in Fig. 5.8.

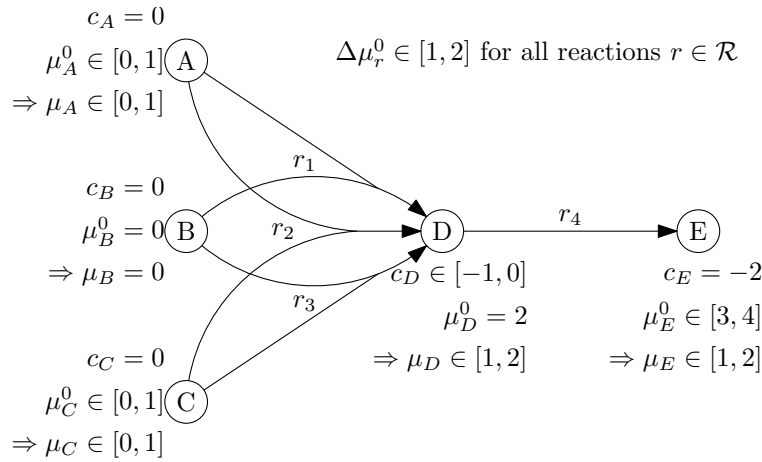


Figure 5.8: Example where classical potential propagation fails.

First of all, we observe that by Formulation 5.14 there cannot be any flow through the network. Let us consider the reaction $A+B \rightarrow D$ (the other two reactions that produce D are analogous). For thermodynamic feasibility, we have to satisfy $\Delta\mu_r^0 + c_D - c_A - c_B < 0$. Since $\Delta\mu_r^0 \in [1, 2]$, $c_D \in [-1, 0]$, $c_A = c_B = 0$, it follows that $\Delta\mu_r^0 + c_D - c_A - c_B \geq 0$. This is a contradiction, hence no thermodynamically feasible flux is possible. By propagation, also $D \rightarrow E$ can be deduced as blocked.

With the choice of J, k as shown in Fig. 5.8 (i.e., to fix $\mu_B^0 = 0, \mu_D^0 = 2$) it is easy to see where the classical potential propagation method is able to cheat. We use reaction $A + C \rightarrow D$. We exploit that $\mu_A = 1, \mu_B = 1$ is a feasible assignment. This way it is not possible to restrict μ_D to $[1, 1]$. It is easy to see that for all choices of J, k that fix μ^0 to a fixed value for 2 metabolites, the situation looks alike.

Conjecture 5.2.1 *Let $\overline{\Delta\mu^0}$ be a consistent vector of reference equilibrium constants for the network in Fig. 5.8 and let J, k such that (5.15) has a unique solution.*

Then there exists a flux vector $v \in \mathbb{R}^{\mathcal{R}}$ and a $\mu \in \mathbb{R}^{\mathcal{M}}$ with

$$\begin{aligned}v_1 + v_2 + v_3 &= v_4 = 1 \\ \mu_A + \mu_B - \mu_D &< 0 \text{ or } v_1 = 0 \\ \mu_A + \mu_C - \mu_D &< 0 \text{ or } v_2 = 0 \\ \mu_B + \mu_C - \mu_D &< 0 \text{ or } v_3 = 0 \\ \mu_D - \mu_E &< 0 \\ v_1, v_2, v_3 &\geq 0 \\ \ell_m^\mu &\leq \mu \leq u_m^\mu\end{aligned}$$

for potential bounds ℓ^μ, u^μ as defined in (5.16). □

Chapter 6

Modules in Metabolic Networks

Abstract The topic of modules (flux modules) was never planned to be part of my research work. It started as a by-product of my work on fast thermodynamically constrained flux variability analysis. In a summer school in Bertinoro, 2012, Leen Stougie presented the problem of finding modules in metabolic networks. Although I had known of the result that there exist such kind of things as modules in metabolic networks [75] from presentations at conferences, I was not aware of the (computational) problem that was preventing them to turn their results into a practical method.

In this summer school, Leen said that the problem are cycles in the network that allow unbounded flux through reactions. Although it turned out that the cycles are in the end not a problem, this gave me the idea that I can apply thermodynamic constraints to get rid of the problem. Thermodynamic constraints prevent internal cycles and if all nutrients / energy sources are bounded, it follows that this also eliminates all unbounded fluxes.

I quickly realized that to work mathematically with modules, I need a definition. I ended up defining modules by a property that was only slightly hinted at in the original publication [75] and was not used at all in the original characterization. From the new definition, the original characterization, an improved detection algorithm and a nice decomposition theorem for elementary modes follow [100].

Because of these results I came in contact with the original researchers, in particular Leen Stougie with whom I started a very fruitful collaboration. In this collaboration I found the connection to matroid theory and we were able to improve my first method significantly [102].

While the notion of module works very well for certain flux spaces, there is a large class of flux spaces, where it utterly fails, since no interesting modules exist. Hence, I started early on to also work on generalizations. Luckily the connection to matroid theory also works for the generalization. This finally led to a decomposition result not only for special cases of metabolic networks but for polyhedra in general [129].

During the course of my PhD work, I have been writing 3 publications on modules [100, 102, 129]. During my research, I continuously refined and generalized my results. To avoid repetition and to tell a coherent story, this chapter does not present the results in chronological order as they have been published.

6.1 Definition

Although, we usually only work on the steady-state flux space of a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S)$, or the thermodynamically constrained flux space, we can define the concept of flux module (or in short module) for an arbitrary flux space $P \subseteq \mathbb{R}^{\mathcal{R}}$.

Definition 6.1.1 (Flux Module) $A \subseteq \mathcal{R}$ is a P -module if there exists a $d \in \mathbb{R}^{\mathcal{M}}$ s.t. $S_A v_A = d$ for all $v \in P$. We call d the interface flux (constant interface) of the module. \square

Many results also hold for the following generalizations of flux module, called k -modules. In particular, we will discuss two variants: k -modules and linear k -modules. k -modules are a proper generalization of flux module, since the definition of flux module is equivalent to 0-module. The notion of linear k -module is a bit more restrictive than the affine counterpart, but we will see that linear k -modules will play a very important role later on.

Definition 6.1.2 (k -module) Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$. $A \subseteq \mathcal{R}$ is a P - k -module if there exists a $d \in \mathbb{R}^{\mathcal{M}}$ and a $D \in \mathbb{R}^{\mathcal{M} \times k}$ s.t.

$$\forall v \in P \exists \alpha \in \mathbb{R}^k : S_A v_A = d + D\alpha.$$

We call d the constant interface of the module and D the variable interface. If we can choose $d = 0$, then we call A a linear k -module. \square

Biologically, we can understand k -modules as follows: In addition to the fixed function (the interface flux d), k -modules also allow additional variable functions (spanning a k -dimensional space). Since biological subsystems often have several side functions, this increases the applicability significantly. In other application areas, we can understand a k -module as a subsystem that only has few (k) interactions to the rest of the system.

6.2 Properties of (Linear) k -Modules

The following observations follow directly from the definition and hopefully clarify the connections between the 3 definitions introduced above.

Observation 6.2.1 Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$.

- (i) $A \subseteq \mathcal{R}$ is a P -module iff A is a P -0-module;
- (ii) Every set A with k elements is a linear k -module; in particular every $r \in \mathcal{R}$ is a linear 1-module;
- (iii) A is a (linear) P - $(k-1)$ -module $\Rightarrow A$ is a (linear) P - k -module;
- (iv) Let $B \subseteq \mathcal{R}$ be a 0-module. It holds for all $A \subseteq \mathcal{R} \setminus B$ that A is a k -module if and only if $A \dot{\cup} B$ is a k -module;
- (v) \emptyset is a k -module. □

For the special case of flux-modules, Obs. 6.2.1(iv) is discussed in more detail in Lemma 6.3.1. In particular, we observe that the statement does not hold in general for non-disjoint union.

We also observe that for a given k -module A the variable interface D is not unique (unless $k = 0$). However, the linear space spanned by D , i.e., $\text{span}(D) := \{D\alpha : \alpha \in \mathbb{R}^k\}$ is unique, if k is chosen to be minimal.

Proposition 6.2.1 *Let $P \subseteq \mathbb{R}^{\mathcal{R}}$, $A \subseteq \mathcal{R}$ and let k be minimal s.t. A is a P - k -module. Let D, D' be two different variable interfaces of A . Then $\text{span}(D) = \text{span}(D')$.*

PROOF Assume $\text{span}(D) \neq \text{span}(D')$. Let $v^0 \in P$ be arbitrary but fixed. Define $d := S_A v_A^0$. By definition of variable interface, it follows that

$$\begin{aligned} \forall v \in P : S_A v_A &\in d + \text{span}(D), \\ \forall v \in P : S_A v_A &\in d + \text{span}(D') \\ \Rightarrow \forall v \in P : S_A v_A &\in d + (\text{span}(D) \cap \text{span}(D')). \end{aligned} \tag{6.1}$$

Since $\text{span}(D) \neq \text{span}(D')$, it follows that $\text{span}(D) \cap \text{span}(D') \subset \text{span}(D)$. Hence,

$$\dim(\text{span}(D) \cap \text{span}(D')) < \dim(\text{span}(D)).$$

It follows that there exists a $D'' \in \mathbb{R}^{\mathcal{M} \times \ell}$ with $\ell < k$ and $\text{span}(D'') = \text{span}(D) \cap \text{span}(D')$. By Eq. 6.1 it follows that D'' is a variable interface of A and hence, k was not minimal; a contradiction. ■

6.2.1 Restriction to Linear Vector Spaces

A core property of flux modules is that the precise form of the flux space is irrelevant. We only have to look at the smallest affine linear space in which P is embedded. The smallest affine linear space that contains P is characterized by those reactions V that do not show variability:

$$\begin{aligned}
 V &:= \{r \in \mathcal{R} : v_r^{\max} \neq v_r^{\min}\}, \text{ where} & (6.2) \\
 v_r^{\max} &:= \sup\{v_r : v \in P\} \\
 v_r^{\min} &:= \inf\{v_r : v \in P\}
 \end{aligned}$$

Theorem 6.2.1 *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}, b \in \mathbb{R}^{\mathcal{M}}$. Then it holds for all $A \subseteq \mathcal{R}$ that*

$$A \text{ is } P\text{-}k\text{-module} \Leftrightarrow A \cap V \text{ is a (linear) } \ker(S_V)\text{-}k\text{-module,}$$

where V is the set of variables that are not constant as defined in (6.2). \square

Theorem 6.2.2 *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}, b \in \mathbb{R}^{\mathcal{M}}$. Then it holds for all $A \subseteq V$, where V is defined as in (6.2) that*

$$A \text{ is a linear } P\text{-}k\text{-module} \Leftrightarrow A \text{ is a (linear) } \ker(S_V|d)\text{-}k\text{-module,}$$

where $d = b - S_{\mathcal{R} \setminus V} v_{\mathcal{R} \setminus V}$ for a $v \in P$ and $(S_V|d)$ denotes the matrix obtained by horizontal concatenation of S_V and d . \square

Thm. 6.2.1 and Thm. 6.2.2 are very similar. Therefore, we will derive the lemmas necessary for the proofs in parallel and explain similarities as well as differences on the way. First however, we observe one important case where linear k -modules and k -modules coincide:

Observation 6.2.2 *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$. If $0 \in P$, then it holds for all $A \subseteq \mathcal{R}$ that*

$$A \text{ is a linear } P\text{-}k\text{-module} \Leftrightarrow A \text{ is a } P\text{-}k\text{-module.}$$

PROOF \Rightarrow : By definition.

\Leftarrow : Since $0 \in P$ it follows that there exists a $\alpha \in \mathbb{R}^k$ with $0 = S_A 0_A = D\alpha + d$. Hence, $d = D(-\alpha)$, which completes the proof. \blacksquare

In particular, this case applies to linear vector spaces as considered in Theorems 6.2.1, 6.2.2.

The following Lemma shows why we do not need a convexity assumption on P :

Lemma 6.2.1 *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$. It holds for all $A \subseteq \mathcal{R}$ that*

$$A \text{ is a (linear) } P\text{-}k\text{-module} \Leftrightarrow A \text{ is a (linear) } \text{aff}(P)\text{-}k\text{-module.}$$

The variable and constant interfaces are the same.

PROOF We only show that if A is a P - k -module, then A is also an $\text{aff}(P)$ - k -module. The other direction is trivial since $P \subseteq \text{aff}(P)$.

Let D be the variable interface of A and d the constant interface of the P - k -module A . Let $x \in \text{aff}(P)$ be arbitrary but fixed. It follows that there exist $\lambda_1, \dots, \lambda_n \in \mathbb{R}$ and $x^1, \dots, x^n \in P$ s.t.

$$x = \sum_{i=1}^n \lambda_i x^i, \quad 1 = \sum_{i=1}^n \lambda_i.$$

Since $x^i \in P$, there exists an $\alpha^i \in \mathbb{R}^k$ with $S_A x_A^i = D\alpha^i + d$ for every $i = 1, \dots, n$. It follows that

$$S_A x_A = \sum_{i=1}^n \lambda_i S_A x_A^i = \sum_{i=1}^n \lambda_i (D\alpha^i + d) = D \sum_{i=1}^n \lambda_i \alpha^i + d \sum_{i=1}^n \lambda_i = D \sum_{i=1}^n \lambda_i \alpha^i + d.$$

This concludes the proof. ■

We can simplify the space that we have to analyze even further if the relative interior w.r.t. $\ker(S)$ is non-empty:

Lemma 6.2.2 *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$. If there exist $x \in P$ and $\epsilon > 0$ s.t. $x+w \in P$ for all $w \in \ker S$ with $\|w\|_\infty < \epsilon$ (i.e., the relative interior of P is non-empty), then for all $A \subseteq \mathcal{R}$*

$$A \text{ is a } P\text{-}k\text{-module} \Leftrightarrow A \text{ is a } \ker(S)\text{-}k\text{-module}.$$

PROOF \Leftarrow : Let $v^1 \in P$ be arbitrary but fixed. We define $d = S_A v^1$. Let $v^2 \in P$ be arbitrary. By definition of P , it follows that $v^1 - v^2 \in \ker S$. Since A is a $\ker(S)$ - k -module, it follows that there exists an $\alpha \in \mathbb{R}^k$ such that $S_A(v_A^2 - v_A^1) = D\alpha$, where D is the variable interface of the $\ker(S)$ - k -module A . Thus, $S_A v_A^2 = S_A v_A^1 + D\alpha = d + D\alpha$ and A is an affine P - k -module.

\Rightarrow : Assume A is a P - k -module. It follows that there exist $d \in \mathbb{R}^k, D \in \mathbb{R}^{\mathcal{M} \times k}$ s.t. for all $v \in P$ there exists an $\alpha \in \mathbb{R}^k$ s.t. $S_A v_A = d + D\alpha$. In particular, it follows that there exists $\alpha^x \in \mathbb{R}^k$ s.t. $S_A x_A = d + D\alpha^x$.

For a proof by contradiction we assume that A is not a $\ker(S)$ - k -module. It follows that there exist $w \in \ker(S)$ s.t. for all $\alpha \in \mathbb{R}^k$ it holds that $S_A w_A \neq D\alpha$. By definition of x , there exists an $\epsilon > 0$ s.t. $x + \epsilon w \in P$. We conclude that

$$\begin{aligned} S_A(x_A + \epsilon w_A) &= S_A x_A + \epsilon S_A w_A \neq d + D\alpha^x + D\alpha && \text{for all } \alpha \in \mathbb{R}^k. \\ \Rightarrow S_A(x_A + \epsilon w_A) &\neq d + D\alpha && \text{for all } \alpha \in \mathbb{R}^k. \end{aligned}$$

This is a contradiction. ■

Lemma 6.2.2 does not work for linear k -modules, since the right hand side b can introduce a non-zero component which would not be covered by non-affine k -modules. However, the result also applies to k -modules if $b = 0$. Then k -modules and linear k -modules are equivalent:

Observation 6.2.3 *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0\}$. If there exists $x \in P$ and $\epsilon > 0$ s.t. $x + w \in P$ for all $w \in \ker S$ with $\|w\|_{\infty} < \epsilon$ (i.e., the relative interior of P is non-empty), then for all $A \subseteq \mathcal{R}$*

$$A \text{ is a } P\text{-}k\text{-module} \Leftrightarrow A \text{ is a linear } P\text{-}k\text{-module.}$$

PROOF \Leftarrow : By definition.

\Rightarrow : By Lemma 6.2.2 it follows that A is a $\ker(S)$ - k -module. Since $P \subseteq \ker(S)$, the claim follows. \blacksquare

If we want to generalize to flux spaces with right hand side $b \neq 0$, we have to eliminate b first.

Lemma 6.2.3 *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$ and assume there exists a $x \in P$ and $\epsilon > 0$ s.t. $x + w \in P$ for all $w \in \ker S$ with $\|w\|_{\infty} < \epsilon$ (i.e., the relative interior of P is non-empty).*

Let $Q = \{(v, t) \in \mathbb{R}^{\mathcal{R}} \times \mathbb{R} : Sv - bt = 0\} = \ker(S| - b)$.

It holds for all $A \subseteq \mathcal{R}$

$$A \text{ is linear } P\text{-}k\text{-module} \Leftrightarrow A \text{ is } Q\text{-}k\text{-module.}$$

PROOF \Leftarrow : Since A is a Q - k -module, there exist by Obs. 6.2.2 a $D \in \mathbb{R}^{\mathcal{M} \times k}$ s.t. for all $(w, t) \in Q$ there exists an $\alpha \in \mathbb{R}^k$ s.t. $S_A w_A = D\alpha$. Let $v \in P$ be arbitrary but fixed. It follows that $(v, 1) \in Q$. Hence, there exists an $\alpha \in \mathbb{R}^k$ s.t. $S_A v_A = D\alpha$ and A is a linear P - k -module.

\Rightarrow : Since A is a linear P - k -module, there exists $D \in \mathbb{R}^{\mathcal{M} \times k}$ s.t. for all $v \in P$ there exists an $\alpha \in \mathbb{R}^k$ s.t. $S_A v_A = D\alpha$. In particular, there exists $\alpha^x \in \mathbb{R}^k$ s.t. $S_A x_A = D\alpha^x$. Assume there exists a $(w, t) \in Q$ s.t. $S_A w_A \neq D\alpha$ for all $\alpha \in \mathbb{R}^k$.

Case $t \neq 0$: Define $w' := \frac{w}{t}$. It follows that for all $\alpha \in \mathbb{R}^k$ we have $S_A w'_A \neq D\alpha$. Since $(w', 1) \in Q$, it follows that $S w' = b$. By definition of x , there exists an $\epsilon > 0$ s.t. $v := (1 - \epsilon)x + \epsilon w' \in P$. It follows that

$$\begin{aligned} S_A v_A &= (1 - \epsilon)S_A x_A + \epsilon S_A w'_A = (1 - \epsilon)D\alpha^x + \epsilon S_A w'_A \\ &\neq (1 - \epsilon)D\alpha^x + \epsilon D\alpha = D((1 - \epsilon)\alpha^x + \epsilon\alpha) && \forall \alpha \in \mathbb{R}^k \\ \Rightarrow S_A v_A &\neq D\alpha && \forall \alpha \in \mathbb{R}^k. \end{aligned}$$

This is a contradiction, since $v \in P$.

Case $t = 0$: Since $Sw = 0$, it follows by definition of x that there exists an $\varepsilon > 0$ s.t. $v := x + \varepsilon w' \in P$. It follows that

$$\begin{aligned} S_A v_A &= S_A x_A + \varepsilon S_A w_A = D\alpha^x + \varepsilon S_A w_A \\ &\neq D\alpha^x + \varepsilon D\alpha = D(\alpha^x + \varepsilon\alpha) && \forall \alpha \in \mathbb{R}^k \\ \Rightarrow S_A v_A &\neq D\alpha && \forall \alpha \in \mathbb{R}^k. \end{aligned}$$

This is a contradiction, since $v \in P$. Hence, A is a Q module. \blacksquare

To obtain a polyhedron with non-empty relative interior, we project to only those variables that are non-constant. The following simple lemma shows that the modules stay invariant under this operation.

Lemma 6.2.4 *It holds for all $A \subseteq V$, where V is defined as in (6.2):*

$$A \text{ is (linear) } P\text{-}k\text{-module} \Leftrightarrow A \text{ is (linear) } \text{pr}_V P\text{-}k\text{-module}.$$

PROOF By definition of projection, we have the following equivalence (for all $d \in \mathbb{R}^M$, $D \in \mathbb{R}^{M \times k}$):

$$\begin{aligned} S_A v_A &= d + D\alpha && \forall v \in P \exists \alpha \in \mathbb{R}^k \\ \Leftrightarrow S_{A \cap V} v_{A \cap V} &= d + D\alpha && \forall v \in \text{pr}_V P \exists \alpha \in \mathbb{R}^k \end{aligned}$$

since $A \subseteq V$. \blacksquare

If we allow non-zero constant interface, we can strengthen Lemma 6.2.4:

Lemma 6.2.5 *Let $P \subseteq \{v \in \mathbb{R}^R : Sv = b\}$ and V be defined as in (6.2). It holds for all $A \subseteq \mathcal{R}$ that*

$$A \text{ is } P\text{-}k\text{-module} \Leftrightarrow A \cap V \text{ is } \text{pr}_V P\text{-}k\text{-module}.$$

PROOF We first show that A is a P - k -module iff $A \cap V$ is a P - k -module.

\Leftarrow : By definition.

\Rightarrow : Let $x \in P$ be arbitrary but fixed. Define $f := S_{A \setminus V} x_{A \setminus V}$. It follows by definition of V that $f = S_{A \setminus V} v_{A \setminus V}$ for all $v \in P$. Since A is a P - k -module, there exists a $d \in \mathbb{R}^M$, $D \in \mathbb{R}^{M \times k}$ s.t. for all $v \in P$ there exists an $\alpha \in \mathbb{R}^k$ s.t.

$$\begin{aligned} d + D\alpha &= S_A v_A = S_{A \cap V} v_{A \cap V} + S_{A \setminus V} v_{A \setminus V} = S_{A \cap V} v_{A \cap V} + f \\ \Rightarrow d - f + D\alpha &= S_{A \cap V} v_{A \cap V}. \end{aligned}$$

Hence, also $V \cap A$ is a P - k -module.

By Lemma 6.2.4 it follows that $A \cap V$ is a P -module iff $A \cap V$ is a $\text{pr}_V P$ -module and hence, the lemma follows. ■

Note that the construction does not translate to linear k -modules since then the elimination of fixed variables can introduce a non-zero constant interface. The differences of the two theorems consists of the following points:

- For Thm. 6.2.1 we can use Lemma 6.2.5 and obtain results for $A \subseteq \mathcal{R}$, while for Thm. 6.2.2 we can only use Lemma 6.2.4 and only obtain results for $A \subseteq V$.
- For Thm. 6.2.1 we can use Lemma 6.2.2, while for Thm. 6.2.2 we have to use Lemma 6.2.3 and thus, have to extend the stoichiometric matrix by an extra column.

Thm. 6.2.1 and Thm. 6.2.2 now follow directly from the previous lemmas.

PROOF (THM. 6.2.1) Let v^{\max}, v^{\min} be defined as in (6.2). Define $d := b - S_{\mathcal{R} \setminus V} v_{\mathcal{R} \setminus V}^{\max}$. By definition of V , it follows that $\text{pr}_V P \subseteq \{v : S_V v = d\}$. By Lemma 6.2.1 it follows for $A \subseteq \mathcal{R}$ that $A \cap V$ is an $\text{pr}_V P$ - k -module iff $A \cap V$ is a $\text{aff}(\text{pr}_V P)$ - k -module. We observe that $\text{aff}(\text{pr}_V P) = \{v : S_V v = d\}$ and hence $\text{aff}(\text{pr}_V P)$ has non-empty relative interior. By Lemma 6.2.2 it follows that $A \cap V$ is a $\text{pr}_V P$ - k -module iff $A \cap V$ is a $\ker(S_V)$ - k -module. By Lemma 6.2.5 it follows that A is a P - k -module iff $A \cap V$ is a $\ker(S_V)$ - k -module. ■

PROOF (THM. 6.2.2) By definition of V , it follows that d is well defined. By definition of V , it follows that $\text{pr}_V P \subseteq \{v : S_V v = d\}$.

By Lemma 6.2.1 it follows that $A \subseteq V$ is a linear $\text{pr}_V P$ - k -module iff A is a linear $\text{aff}(\text{pr}_V P)$ - k -module. We observe that $\text{aff}(\text{pr}_V P) = \{v : S_V v = d\}$ and hence $\text{aff}(\text{pr}_V P)$ has non-empty relative interior. By Lemma 6.2.3 it follows that A is a linear $\text{pr}_V P$ - k -module iff A is a $\ker(S_V|d)$ - k -module. By Lemma 6.2.4 it follows that A is a linear P - k -module iff A is a $\ker(S_V|d)$ - k -module. ■

6.2.2 Matroid Theory for k -Modules

In the previous section we observed that we can restrict ourselves to the analysis of linear vector spaces of the form $P = \ker S$.

Matroid theory is a very powerful theory to analyze discrete properties (like modules) of linear vector spaces. It turns out that k -modules correspond to $(k + 1)$ -separators in matroid theory. Here, we only work with unoriented matroids hence, circuits etc. are just represented by sets of reactions and not by signed-sets.

Definition 6.2.1 (k -separator, [118]) *Let M be a matroid on the element set \mathcal{R} and r its rank function (see [118] for details). A set $A \subseteq \mathcal{R}$ is a k -separator if and only if*

$$r(A) + r(\mathcal{R} \setminus A) - r(\mathcal{R}) < k.$$

We recall that every matrix defines a matroid, the so called linear matroid (Def. 2.5.3). This matroid now provides us with the link between matroid theory and modules:

Theorem 6.2.3 *$A \subseteq \mathcal{R}$ is a $(\ker S)$ - k -module if and only if A is a $k + 1$ -separator in the linear matroid M represented by S . \square*

We will now prove Thm. 6.2.3, which connects k -modules to $(k + 1)$ -separators. First, we characterize the dimension of the interface space:

Lemma 6.2.6 *Let $A \subseteq \mathcal{R}$ and $S \in \mathbb{R}^{\mathcal{M} \times \mathcal{R}}$. Then it holds that*

$$\dim(S_{A \text{pr}_A} \ker(S)) = \dim(\ker(S)) - \dim(\ker(S) \cap X^\perp) - \dim(\ker(S) \cap X),$$

where $X = \{v \in \mathbb{R}^{\mathcal{R}} : v_r = 0 \forall r \notin A\}$ and $X^\perp = \{v \in \mathbb{R}^{\mathcal{R}} : v_r = 0 \forall r \in A\}$.

PROOF Define $L = \text{pr}_A(\ker(S))$ and consider the linear map

$$\begin{aligned} f : \ker(S) &\rightarrow L \\ v &\mapsto \text{pr}_A(v). \end{aligned}$$

Since $\ker(f) = \ker(S) \cap \ker(\text{pr}_A)$ it follows by the fundamental theorem on homomorphisms that

$$\dim(L) = \dim(\ker(S)) - \dim(\ker(S) \cap \ker(\text{pr}_A)).$$

Observe that $\ker(\text{pr}_A) = X^\perp$. Hence, we get

$$\dim(L) = \dim(\ker(S)) - \dim(\ker(S) \cap X^\perp).$$

We can identify $L \subseteq \mathbb{R}^A$ with $L \times 0^{\mathcal{R} \setminus A} \subseteq \mathbb{R}^{\mathcal{R}}$. Observe that

$$\dim(S(L)) = \dim(S_A(L)) = \dim(S_{A \text{pr}_A} \ker(S)).$$

It follows again by the fundamental theorem on homomorphisms that

$$\dim(S(L)) = \dim(L) - \dim(L \cap \ker(S)).$$

With the identification, we also observe that $L = X \cap \ker(S)$. We conclude

$$\begin{aligned} \dim(S_{A \text{pr}_A} \ker(S)) &= \dim(S(L)) = \dim(L) - \dim(L \cap \ker(S)) \\ &= \dim(\ker(S)) - \dim(\ker(S) \cap X^\perp) - \dim(L \cap \ker(S)) \\ &= \dim(\ker(S)) - \dim(\ker(S) \cap X^\perp) - \dim(X \cap \ker(S)), \end{aligned}$$

which concludes the proof. \blacksquare

Lemma 6.2.7 *$A \subseteq \mathcal{R}$ is a k -module if and only if $\dim(S_{A \text{pr}_A} \ker S) \leq k$.*

PROOF We have

$$\begin{aligned}
 & \dim(S_{\text{Apr}_A} \ker S) \leq k \\
 \Leftrightarrow \exists D \in \mathbb{R}^{\mathcal{M} \times k} : & \quad S_{\text{Apr}_A} \ker S \subseteq \{D\alpha, \alpha \in \mathbb{R}^k\} \\
 \Leftrightarrow \exists D \in \mathbb{R}^{\mathcal{M} \times k} : & \quad \{S_{Av_A} : v \in \ker S\} \subseteq \{D\alpha, \alpha \in \mathbb{R}^k\} \\
 \Leftrightarrow \exists D \in \mathbb{R}^{\mathcal{M} \times k} \forall v \in \ker S \exists \alpha \in \mathbb{R}^k : & \quad S_{Av_A} = D\alpha,
 \end{aligned}$$

which concludes the proof. ■

PROOF (THM. 6.2.3) It follows from the definition of rank that

$$\begin{aligned}
 \dim(\ker S) &= |\mathcal{R}| - r(\mathcal{R}), \\
 \dim(\ker S \cap X) &= |A| - r(A), \\
 \dim(\ker S \cap X^\perp) &= |\mathcal{R} \setminus A| - r(\mathcal{R} \setminus A).
 \end{aligned}$$

By Lemma 6.2.7 a set $A \subseteq \mathcal{R}$ is a k -module if and only if

$$\begin{aligned}
 & k \geq \dim(S_{\text{Apr}_A} \ker S) \\
 \Leftrightarrow & k \geq \dim(\ker S) - \dim(\ker S \cap X^\perp) - \dim(\ker S \cap X) \quad (\text{by Lemma 6.2.6}) \\
 \Leftrightarrow & k \geq (|\mathcal{R}| - r(\mathcal{R})) - (|A| - r(A)) - (|\mathcal{R} \setminus A| - r(\mathcal{R} \setminus A)) \\
 \Leftrightarrow & k \geq r(A) + r(\mathcal{R} \setminus A) - r(\mathcal{R})
 \end{aligned}$$

Hence, A is a $k + 1$ -separator if and only if A is a k -module. ■

6.2.3 Finding k -Modules

Summarizing the results from the previous two subsections, we conclude that a polyhedron contains a k -module if and only if the corresponding matroid contains a $k + 1$ separator. It follows that if we want to test whether a module contains a non-trivial k -module, i.e, a k -module containing more than k elements, we only have to check whether the corresponding matroid is $k + 1$ -connected. Algorithms for testing connectivity have been developed by Bixby and Cunningham [28, 81, 13, 14]. If k is assumed fixed, the connectivity test can be performed in polynomial time [14] using matroid intersection.

Algorithms that directly compute decompositions into k -modules have been studied in the context of branch-decompositions [116, 117].

For the case of 1-separators and 2-separators there are designated algorithms [81, 13], which are discussed later on in Sec. 6.3.3.3 and Sec. 6.4.1.

6.3 Flux modules (0-modules)

These results are taken from my first paper on flux modules [100]. The only exception is the generalization of Thm. 6.3.2.

Here, we consider for a flux space P the properties of flux modules (P -0-modules), simply called P -modules.

We observe that if and only if $P \neq \emptyset$ then the interface of each P -module is well defined. Hence, all of the following theorems will require $P \neq \emptyset$. If $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$, then \mathcal{R} is itself a P -module. We also observe that, given two disjoint P -modules A and B , their union is again a P -module:

Lemma 6.3.1 *Let $P \subseteq \mathbb{R}^{\mathcal{R}}$ be an arbitrary flux space and let A, B be P -modules with interfaces d^A, d^B respectively.*

If A and B are disjoint, then $A \dot{\cup} B$ is a P -module with interface $d^A + d^B$.

If $A \subseteq B$, then $B \setminus A$ is a P -module with interface $d^B - d^A$. In particular, if $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$, then $\mathcal{R} \setminus A$ is a P -module for every P -module A . \square

PROOF For every $v \in P$, we have $S_A v_A = d^A$ and $S_B v_B = d^B$. If A and B are disjoint, this implies $S_{A \dot{\cup} B} v_{A \dot{\cup} B} = S_A v_A + S_B v_B = d^A + d^B$. Hence, $A \dot{\cup} B$ is a P -module with interface $d^A + d^B$. Similarly, if $A \subseteq B$ are P -modules, we have $S_{B \setminus A} v_{B \setminus A} = S_B v_B - S_A v_A = d^B - d^A$, for all $v \in P$, so $B \setminus A$ is a P -module. \blacksquare

We remark that Lemma 6.3.1 is generally not true for non-disjoint modules. A counterexample can be found with

$$S = \begin{pmatrix} 1 & -1 & 1 & -1 \end{pmatrix}, \quad P = \left\{ \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} \right\}.$$

We now focus on flux spaces consisting of steady-state fluxes and the more restricted flux spaces, where the fluxes are also thermodynamically feasible. In this section we refer by thermodynamically feasible as in Sec. 2.6.2, i.e., the case without bounds on the concentrations.

Definition 6.3.1 (Steady-State and Thermodynamically Constrained Flux Space)

Given a metabolic network $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ with right-hand side b and flux bounds ℓ, u , the steady-state flux space F is defined as

$$F := \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b, \ell \leq v \leq u\}. \quad (6.3)$$

The thermodynamically constrained flux space T is defined as

$$T := \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b, \ell \leq v \leq u, v \text{ thermo. feasible}\}. \quad (6.4)$$

For F -modules (if $F \neq \emptyset$) and T -modules (if $T \neq \emptyset$) $A \subseteq \mathcal{R}$ with interface d we define

$$F^A := \{v \in \mathbb{R}^A : S_A v = d, \ell_A \leq v \leq u_A\}. \quad (6.5)$$

If $T \neq \emptyset$, we similarly define for T -modules $A \subseteq \mathcal{R}$ with interface d

$$T^A := \{v \in \mathbb{R}^A : S_A v = d, \ell_A \leq v \leq u_A, v \text{ thermo. feasible} \}, \quad (6.6)$$

and in addition

$$\tilde{T}^A := \text{pr}_A(T). \quad (6.7)$$

Observe that F^A is also well defined if A is both a F -module and a T -module (if $T \neq \emptyset$), because $T \subseteq F$ and thus the interfaces must coincide. We will later see (Application of Cor. 6.3.1 with $\mathcal{I} = \emptyset$) that for F -modules A we have $F^A = \text{pr}_A(F)$. While $\tilde{T}^A \subseteq T^A$ holds for any T -module A , we do not always have $\tilde{T}^A = T^A$ as the following example shows (see however Cor. 6.3.1):

Example 6.3.1 Consider the network shown in Fig. 6.1.

We observe that in this example reaction g is blocked by thermodynamic constraints, because it forms an internal cycle with e , which always has to be present. We further observe that if d is carrying flow, then f cannot carry any flow, since otherwise $\{d, e, f\}$ would form an internal cycle. Since $v_e = 1$, it follows that $v_b + v_d \geq 1$, since metabolite 1 has no other inflow.

We further observe that $v_b \leq 1$. If $v_d = 0$, it follows that $v_b = 1$ and thus $v_k = 0$ and $v_\ell = 1$. If $v_d > 0$, it follows as observed above that $v_f = 0$, hence $v_\ell \geq v_d + v_b \geq 1$. Since $v_\ell + v_k = 1$, it follows that $v_\ell = 1$ and $v_k = 0$. We conclude that, in any case, we have $v_k = 0$ and $v_\ell = 1$.

Since $v_k = 0$, it follows that $v_d + v_b = 1$ and hence, $v_f = 0$. It also follows that $v_n = 0$ and $v_m = 1$.

We observe that reactions g, k, l, f, m, n only carry fixed flows. Hence, they all form modules by themselves. It follows that $A = \{g, k, l, f, m, n\}$ is a module, as shown in Fig. 6.2. In this module, we also have a thermodynamically feasible flux vector $v \in T^A$ with $v_g = v_k = v_n = v_f = 1, v_\ell = v_m = 0$. Observe that this flux vector has flux through reactions that we actually derived as blocked in the whole network, i.e., $v \notin \tilde{T}^A = \text{pr}_A T$. We conclude that the restriction of $\bar{\ell}_c \leq 0$ is necessary for Cor. 6.3.1.

In what follows, we will use P to denote general flux spaces, while T and F always denote the flux spaces defined above.

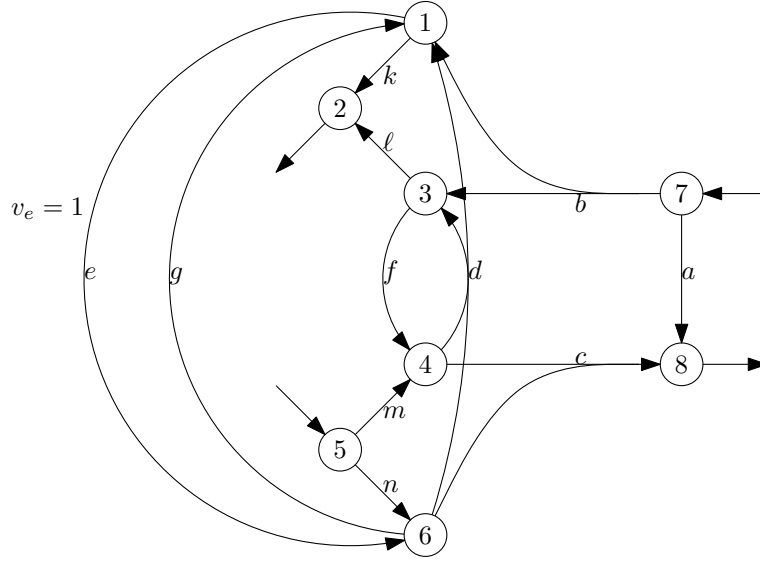


Figure 6.1: Artificial example network that shows that the restrictions on the flux bounds are really necessary. The exchange reactions are the reactions that only involve one metabolite, they all have a fixed flux of 1. All the other reactions are irreversible internal reactions. For all internal reactions, except reaction e , there exist no flux bounds. The flux through reaction e is fixed to 1. Furthermore, every metabolite has to be produced at the same rate as it is consumed.

Before we study the decomposition of the flux space T , we first investigate the decomposition of the flux space of a union of two T -modules. In contrast to the original version published in [100], we employ the slightly strengthened Thm. 4.5.1 to characterize the thermodynamically feasible fluxes. Therefore we use $\bar{\ell}$ as defined in Def. 2.1.6 to denote lower and upper bounds in a uniform way and recall Def. 4.5.1:

$$\bar{\mathcal{C}} := \{r \in \bar{\mathcal{I}} : S_{\mathcal{I}}v = 0, v_{\text{Irrev}} \geq 0, v_r > 0 \exists v \in \mathbb{R}^{\mathcal{I}}\}$$

Note that many of the following theorems can be strengthened in the case $\bar{\ell}_{\bar{\mathcal{C}}} \leq 0$.

Lemma 6.3.2 *Assume $T \neq \emptyset$. Then for any disjoint T -modules A and B we have*

$$\tilde{T}^{A \dot{\cup} B} \subseteq \tilde{T}^A \times T^B \subseteq T^A \times T^B \subseteq F^{A \dot{\cup} B}.$$

If $\bar{\ell}_{\bar{\mathcal{C}}} \leq 0$, then $\tilde{T}^{A \dot{\cup} B} = T^A \times T^B$. □

PROOF By the definition of module, we have $\tilde{T}^{A \dot{\cup} B} \subseteq \tilde{T}^A \times \tilde{T}^B$ and $\tilde{T}^A \subseteq T^A, \tilde{T}^B \subseteq T^B$. Hence, we get $\tilde{T}^{A \dot{\cup} B} \subseteq \tilde{T}^A \times \tilde{T}^B \subseteq \tilde{T}^A \times T^B \subseteq T^A \times T^B$. Let d^A, d^B denote the interfaces of the T -modules A and B respectively. By Lemma 6.3.1, it follows that $A \dot{\cup} B$ is a T -module with interface $d^A + d^B$.

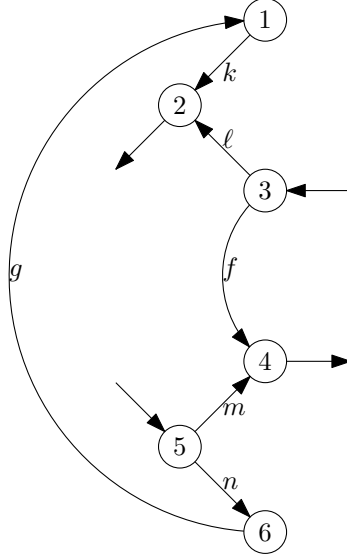


Figure 6.2: A module of the network shown in Fig. 6.1. The interface fluxes are represented by new exchange reactions (arcs that only involve one metabolite). All exchange (interface) fluxes have value 1. It can easily be seen that this network does not contain any internal cycles, hence every steady-state flux is thermodynamically feasible.

Let $x \in T^A, y \in T^B$ be arbitrary but fixed. Define $v' \in \mathbb{R}^{A \dot{\cup} B}$ by $v'_A := x$ and $v'_B := y$. Since A and B are T -modules, we have $S_{A \dot{\cup} B} v' = d^A + d^B$. Since $\ell_A \leq x \leq u_A$ and $\ell_B \leq y \leq u_B$, it follows that $v' \in F^{A \dot{\cup} B}$. Hence, $T^A \times T^B \subseteq F^{A \dot{\cup} B}$.

Now we continue with the case $\bar{\ell}_{\mathcal{C}} \leq 0$. Since $T \neq \emptyset$, there exists $w' \in T$. Define $w \in \mathbb{R}^{\mathcal{R}}$ by $w_{A \dot{\cup} B} := v'$ and $w_{\mathcal{R} \setminus (A \dot{\cup} B)} := w'_{\mathcal{R} \setminus (A \dot{\cup} B)}$. Since $A \dot{\cup} B$ is a T -module, it follows that $w \in F$, as defined in (6.3). Let $v = v^w$ be the flux vector obtained from w by subtracting all contained internal cycles using Alg. 4. Since $\bar{\ell}_{\mathcal{C}} \leq 0$, it follows by Thm. 4.5.1 that $v \in T$ and $\text{sign}(w - v) \subseteq \text{sign}(w)$. With $w_A = x$, this implies $\text{sign}(x - v_A) \subseteq \text{sign}(x)$.

Since A is a T -module, we have $S_A v_A = d^A$. From $S_A x - S_A v_A = d^A - d^A = 0$, we get $S_A(x - v_A) = 0$. If $x - v_A \neq 0$, it follows from $\text{sign}(x - v_A) \subseteq \text{sign}(x)$ that x would have contained an internal cycle. This is a contradiction and hence, $v_A = x$. By the same argument, we can show $v_B = y$. Since $v \in T$, we obtain $v' = v_{A \dot{\cup} B} \in \tilde{T}^{A \dot{\cup} B}$ and since we can do this for every $x \in T^A$ and $y \in T^B$, we get $\tilde{T}^{A \dot{\cup} B} \supseteq T^A \times T^B$. Therefore, $\tilde{T}^{A \dot{\cup} B} = T^A \times T^B$. ■

Corollary 6.3.1 *If $T \neq \emptyset$ with $\bar{\ell}_{\mathcal{C}} \leq 0$ and A is a T -module, then $T^A = \tilde{T}^A = \text{pr}_A(T)$. □*

PROOF By Lemma 6.3.1, $B := \mathcal{R} \setminus A$ is a T -module. With Lemma 6.3.2, it follows $\tilde{T}^{A \dot{\cup} B} = T = T^A \times T^B$ and by projection $T^A = \text{pr}_A(T) = \tilde{T}^A$. ■

Using Lemma 6.3.2, we can now show by induction that from a partition of the reaction

set \mathcal{R} into a set of T -modules, we can get a decomposition of the thermodynamically constrained flux space T .

Theorem 6.3.1 (Product Space from Modules) *Assume $T \neq \emptyset$. If $\mathcal{X} = \{A_1, \dots, A_n\}$ is a partition of \mathcal{R} into T -modules, then*

$$T \subseteq \prod_{A \in \mathcal{X}} T^A \subseteq F.$$

If $\bar{\ell}_{\bar{C}} \leq 0$, then

$$T = \prod_{A \in \mathcal{X}} T^A.$$

PROOF Define $B_1 := A_1$ and $B_i := A_i \cup B_{i-1}$, $i = 2, \dots, n$. It follows from the definition of module and Lemma 6.3.1 that A_i and B_i , $i = 1, \dots, n$, are T -modules. By Lemma 6.3.2 it follows that $\tilde{T}^{B_j} \subseteq \tilde{T}^{B_{j-1}} \times T^{A_j}$, for all $j = 2, \dots, n$. We already observed that $\tilde{T}^{B_1} \subseteq T^{B_1} = T^{A_1}$, hence it follows by induction that $\tilde{T}^{B_j} \subseteq \prod_{i=1}^j T^{A_i}$. Since $B_n = \mathcal{R}$ we obtain that $T \subseteq \prod_{i=1}^n T^{A_i}$.

To prove $\prod_{i=1}^n T^{A_i} \subseteq F$, let $v^i \in T^{A_i}$ be arbitrary but fixed. Since $T \neq \emptyset$, it follows that there exists a $w \in T$. Let d^i denote the interface of T -module A_i . We get $S_{A_i} w_{A_i} = d^i$, for all $i = 1, \dots, n$, which implies $b = Sw = \sum_{i=1}^n S_{A_i} w_{A_i} = \sum_{i=1}^n d^i$. Now define $v \in \mathbb{R}^{\mathcal{R}}$ with $v_{A_i} := v^i$, for all $i = 1, \dots, n$. It follows that $Sv = \sum_{i=1}^n S_{A_i} v^i = \sum_{i=1}^n d^i = b$. Clearly, v also satisfies $\ell \leq v \leq u$. We conclude $v \in F$ and thus $\prod_{i=1}^n T^{A_i} \subseteq F$.

If in addition, we have $\bar{\ell}_{\bar{C}} \leq 0$, then by Lemma 6.3.2 and Cor. 6.3.1 we also get the equalities $\tilde{T}^{B_j} = T^{B_j} = T^{B_{j-1}} \times T^{A_j}$ and hence, $T = \prod_{i=1}^n T^{A_i}$. \blacksquare

6.3.1 Uniqueness of the Decomposition

Next we study the existence and uniqueness of the decomposition of a network into minimal flux modules. When we talk about minimality, we have to exclude \emptyset as a proper module, since otherwise there is only one trivial minimal module.

Definition 6.3.2 (Proper flux module) *A P -module A is called proper if $A \neq \emptyset$. \square*

Definition 6.3.3 (Minimal Flux Module) *Let $P \subseteq \mathbb{R}^{\mathcal{R}}$ be a flux space. A proper P -module $A \subseteq \mathcal{R}$ is minimal if there exists no proper P -module B s.t. $B \subset A$. \square*

We will see that there always exists a unique decomposition into minimal modules and hence the question of finding such a decomposition (as discussed in Sec. 6.3.3) is well defined.

Theorem 6.3.2 (Uniqueness) *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0\}$. Then the partition of \mathcal{R} into minimal P -modules exists and is unique. \square*

We observe that we can always decompose a proper module into minimal modules.

Proposition 6.3.1 (Existence) *Let P be a flux space. Every proper P -module A can be partitioned into minimal P -modules, i.e., there exist minimal P -modules A_1, \dots, A_k s.t. $A = A_1 \dot{\cup} A_2 \dot{\cup} \dots \dot{\cup} A_k$. \square*

PROOF Assume the proposition is false. Then there exists a non-minimal P -module $A \subset \mathcal{R}$ that cannot be partitioned into smaller P -modules and a minimal P -module $B \subset A$. By Lemma 6.3.1, $C := A \setminus B$ is also a P -module. Thus $C \dot{\cup} B$ is a partition of A , which contradicts the assumption. \blacksquare

Note that this proposition holds for arbitrary flux spaces P . To obtain uniqueness of the decomposition we also have to require that the flux space satisfies the steady-state condition.

Originally, the following theorem was stated in a much more restricted form with $P = T$ under the condition that $\ell_C \leq 0 \leq u_C$. However, the recently found characterization of modules with matroid theory allows me to formulate the theorem in this more general way. I will give both proofs: The proof of the generalized theorem using matroid theory and the direct proof, as originally published in [100].

PROOF (PROOF OF THM. 6.3.2 USING MATROID THEORY) First of all, we observe that every $r \in \mathcal{R} \setminus V$ is a minimal module by itself by definition of V , where V is defined as in (6.2). We also observe that all other minimal modules only consist of reactions in V , since by Lemma 6.3.1 we can always subtract those reactions not in V and obtain a smaller module.

Since $A \subseteq V$ is a P -module if and only if A is a 0-module, it follows by Thm. 6.2.1 that A is a P -module if and only if A is a $\ker(S_V)$ -module and V is defined as in (6.2). Furthermore, we know this is equivalent by Thm. 6.2.3 to A being a separator of the linear matroid M on elements V generated by S_V .

We observe that the minimality condition for a module $A \subseteq V$ translates by Prop. 6.3.1 into the matroidal condition that the matroid $M|_A$ (the matroid obtained from M by restriction to A) is connected (Prop. 4.2.1 in [118]). By Proposition 4.1.3 in [118] we know that a matroid is connected if for any two elements x, y there exists a circuit that contains both. By Prop. 2.5.4 it follows that the decomposition in connected separators is unique and hence, also the decomposition into minimal modules is unique. \blacksquare

We will now prove the necessary lemmas for the proof without matroid theory. Note that we can extend the second proof easily also to full generality by using the same steps as in the beginning of the first proof.

The following lemma holds for every flux space that satisfies the steady-state assumption. In particular, it holds for the thermodynamically constrained flux space T .

Lemma 6.3.3 (Modules from Product Space) *Let $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$. Assume $P = \prod_{i=1}^n P_i$ with $P_i \subseteq \mathbb{R}^{A_i}$, where $A_i \subseteq \mathcal{R}$. Then for every $i = 1, \dots, n$, there exists a vector $b^i \in \mathbb{R}^{\mathcal{M}}$ s.t. $S_{A_i} v_{A_i} = b^i$, for all $v \in P$, i.e., A_i is a P -module. \square*

PROOF Assume the lemma is false. Then there exist $i \in \{1, \dots, n\}$ and $v, w \in P$ s.t. $S_{A_i} v_{A_i} \neq S_{A_i} w_{A_i}$. Define $w' \in \mathbb{R}^{\mathcal{R}}$ by $w'_{A_i} = w_{A_i}, w'_{\mathcal{R} \setminus A_i} = v_{\mathcal{R} \setminus A_i}$. Since $P = \prod_{i=1}^n P_i$, it follows that $w' \in P$. Since $S_{A_i} w_{A_i} \neq S_{A_i} v_{A_i}$, we get $S w' = S_{A_i} w_{A_i} + S_{\mathcal{R} \setminus A_i} v_{\mathcal{R} \setminus A_i} = S_{A_i} w_{A_i} + b - S_{A_i} v_{A_i} \neq b$. Thus $w' \notin P$, which is a contradiction. ■

To prove uniqueness of the decomposition into minimal T -modules (Thm. 6.3.2), we use Thm. 6.3.1 from the previous section. Given a partition of the reaction set \mathcal{R} into T -modules A , the thermodynamically constrained flux space T can be written as the product of the flux spaces T^A . Assuming that there exist two different partitions with minimal T -modules, we show that we can write T as a product of smaller factors (Lemma. 6.3.4). We then go in the reverse direction and show with Lemma 6.3.3 that from this we can obtain smaller T -modules, contradicting the minimality.

Lemma 6.3.4 *Let X, I be sets and $P \subseteq X^I$. Assume $P = \prod_{i=1}^n P_i = \prod_{j=1}^m Q_j$, where $P_i \subseteq X^{A_i}$ and $Q_j \subseteq X^{B_j}$, with $A_i, B_j \subseteq I$, for $i = 1, \dots, n, j = 1, \dots, m$. Then $P = \prod_{i=1}^n \prod_{j=1}^m R_{ij}$ with $R_{ij} = \text{pr}_{A_i \cap B_j} P$. □*

PROOF For every $j = 1, \dots, m$ it holds that

$$Q_j = \text{pr}_{B_j}(P) = \text{pr}_{B_j} \left(\prod_{i=1}^n P_i \right) = \prod_{i=1}^n \text{pr}_{A_i \cap B_j} P = \prod_{i=1}^n R_{ij}.$$

Since $P = \prod_{j=1}^m Q_j = \prod_{j=1}^m \prod_{i=1}^n R_{ij} = \prod_{i=1}^n \prod_{j=1}^m R_{ij}$, the claim follows. ■

PROOF (PROOF OF THM. 6.3.2 FOR $P = T$ WITH $\bar{\ell}_{\mathcal{C}} \leq 0$) Since \mathcal{R} is a T -module, it follows by Prop. 6.3.1 that there always exists a partition into minimal T -modules. Assume there exist two partitions $\mathcal{X} \neq \mathcal{Y}$ of \mathcal{R} into minimal T -modules. By Thm. 6.3.1 and Cor. 6.3.1, we can write

$$\prod_{x \in \mathcal{X}} \text{pr}_x(T) = \prod_{x \in \mathcal{X}} T^x = T = \prod_{y \in \mathcal{Y}} T^y = \prod_{y \in \mathcal{Y}} \text{pr}_y(T).$$

By Lemma 6.3.4 it follows that there exists a partition \mathcal{Z} of \mathcal{R} which is finer than \mathcal{X}, \mathcal{Y} , i.e., every $z \in \mathcal{Z}$ is contained in some $x \in \mathcal{X}$ and $y \in \mathcal{Y}$. The partition \mathcal{Z} also satisfies

$$T = \prod_{z \in \mathcal{Z}} \text{pr}_z(T).$$

It follows by Lemma 6.3.3 that every $z \in \mathcal{Z}$ is also a T -module. Since $\mathcal{X} \neq \mathcal{Y}$, there exists at least one T -module of \mathcal{Z} that is strictly contained in a T -module of \mathcal{X} . This contradicts the minimality of the T -modules in \mathcal{X} . ■

6.3.2 Decomposition Theorem for EFMs

An important consequence of the decomposition into T -modules is that we can describe the set of elementary flux modes in a more compact form. This result follows basically

directly from the product form of the flux space (Thm. 6.3.1). We recall that the set of *elementary flux modes* (EFM) of the thermodynamically constrained flux space T are the flux modes with minimal support (Def. 2.4.2), respectively minimal sign-support (Prop. 2.4.1):

$$\text{EFM}(P) := \{v \in P : \text{sign}(w) \not\subseteq \text{sign}(v) \forall w \in P \setminus \{0\}\}.$$

From this, we can derive the following relationships between the elementary flux modes of F^A and T^A :

Proposition 6.3.2 *Assume $T \neq \emptyset$. Let A be a T -module with interface $d \neq 0$. Let*

$$P^A := \{(v, x) \in \mathbb{R}^A \times \mathbb{R}^+ : S_A v - dx = 0, \ell_A \leq v \leq u_A\}.$$

Then

- a) $\text{EFM}(T^A) \subseteq \text{EFM}(F^A)$
- b) $\{v \in \mathbb{R}^A : (v, 1) \in \text{EFM}(P^A)\} \subseteq \text{EFM}(T^A)$
- c) *If $\bar{\ell}_{\bar{A} \cap \bar{C}} \leq 0$, then $\text{EFM}(T^A) = \text{EFM}(F^A)$ for $\bar{A} := A \cup \{-r : r \in A\}$.*
- d) *If $\ell_r \in \{-\infty, 0\}$ and $u_r \in \{0, \infty\}$ for all $r \in A$, then $\{v \in \mathbb{R}^A : (v, 1) \in \text{EFM}(P^A)\} = \text{EFM}(T^A)$. \square*

PROOF We show all statements separately:

- a) Let $v \in \text{EFM}(T^A)$. Assume $v \notin \text{EFM}(F^A)$. Since $T^A \subseteq F^A$, it follows that there exists $w \in F^A$ with $\text{sign}(w) \subset \text{sign}(v)$ and w is thermodynamically infeasible. By Thm. 2.6.1, there exists an internal cycle $c \in \mathbb{R}^A$ with $\text{sign}(c) \subseteq \text{sign}(w)$. It follows that $\text{sign}(c) \subseteq \text{sign}(v)$, contradicting $v \in T^A$.
- b) Let $(v, 1) \in \text{EFM}(P^A)$. Assume $v \notin \text{EFM}(T^A)$. Since $T^A \subseteq \text{pr}_A(P^A)$, it follows that $v \notin T^A$. By Thm. 2.6.1, there exists an internal cycle c with $\text{sign}(c) \subseteq \text{sign}(v)$. It follows that $\text{sign}(c) \subset \text{sign}(\{v, 1\})$. This contradicts $(v, 1) \in \text{EFM}(P^A)$.
- c) Assume $\bar{\ell}_{\bar{A} \cap \bar{C}} \leq 0$, and suppose there exists $v \in \text{EFM}(F^A) \setminus \text{EFM}(T^A)$. We conclude that $v \notin T^A$, hence v contains an internal cycle c . Since $\bar{\ell}_{\bar{A} \cap \bar{C}} \leq 0$, we can subtract all internal cycles using Alg. 4. By Thm. 4.5.1, we obtain $v' \in F^A$ with $\text{sign}(v') \subset \text{sign}(v)$, contradicting $v \in \text{EFM}(F^A)$.
Thus, it follows that $\text{EFM}(F^A) \subseteq \text{EFM}(T^A)$. Together with a) we get that $\text{EFM}(F^A) = \text{EFM}(T^A)$.
- d) We will now consider the case when $\ell_r \in \{-\infty, 0\}$ and $u_r \in \{0, \infty\}$ for all $r \in A$. Assume there exists $v \in \text{EFM}(T^A)$ with $(v, 1) \notin \text{EFM}(P^A)$. Since $(v, 1) \in P^A$, it follows that $(v, 1)$ is not minimal. Hence, there exists a $(w, x) \in P^A$ with $\text{sign}(w, x) \subset \text{sign}(v, 1)$.

If $x > 0$ we scale (w, x) to $(w', 1)$. Since the bounds are only 0 or infinity, the flux bounds will also be satisfied by w' . It follows that $w' \in T^A$. Since $\text{sign}(w') \subset \text{sign}(v)$, this is a contradiction to the minimality of v . Therefore, we only need to consider the case where $x = 0$.

Let $v' = v - \alpha w$, where α is chosen as large as possible such that $\text{sign}(v') \subset \text{sign}(v)$. Since $\ell_A \leq 0 \leq u_A$, it follows that $v' \in P^A$ and $\text{sign}(v') \subset \text{sign}(v)$. Since $(w, 0) \in P^A$, we have $Sw = 0$ and hence, $Sv' = d$. Since v was thermodynamically feasible, v' is also thermodynamically feasible, hence $v' \in T^A$, which contradicts the minimality of v . \blacksquare

We remark that Condition d) is necessary, because P^A does not restrict x to 1 and thus there can exist flux vectors in P^A that satisfy the bounds ℓ, u only by choosing $x \neq 1$. The formulation of P^A does not contain the constraint $x = 1$ to allow internal cycles with $x = 0$. Note that P^A under Condition d) is a flux cone, for which our definition of EFM coincides with the standard definition in [141]. Condition c) is also necessary, which can be seen on the following example.

Example 6.3.2 *We continue the Example 6.3.1, shown in Fig. 6.1.*

If we combine the thermodynamically feasible flux vector $v \in T^A$ with $v_g = v_k = v_n = v_f = 1, v_\ell = v_m = 0$ in the T -module $A = \{g, k, l, f, m, n\}$ (see Fig 6.2) with a feasible flux vector in the module formed by the other reactions $\{a, b, c, d, e\}$ (see Fig. 6.3), we still get a steady-state flux vector. However, this flux vector is not thermodynamically feasible (for example, it contains the cycle $\{e, g\}$). Thus, we conclude that we also require the condition $\bar{\ell}_{\bar{c}} \leq 0$ in Thm. 6.3.1.

We also observe that we cannot obtain all (including thermodynamically infeasible) steady-state flux vectors by combining steady-state flux vectors of the two modules. For example, it is impossible to obtain the steady-state flux vector $w \in F$ with $w_e = w_k = w_n = w_a = 1, w_g = 2, w_l = w_m = w_f = w_d = w_b = w_c = 0$. The interface flux enforces flux through either d or b . \square

We can now characterize the set of elementary modes of the whole network. Modules that have an interface flux of 0 will not contribute to the set of elementary modes. Hence, we call N the set of reactions that are contained in such modules and which will have a 0-flux in every elementary mode.

Theorem 6.3.3 *Let $b \neq 0$ and $T \neq \emptyset$. Let $\mathcal{X} = \{A_1, \dots, A_n\}$ be a partition of \mathcal{R} into T -modules, then*

$$\begin{aligned} \text{EFM}(T) &\subseteq 0^N \times \prod_{A \in \mathcal{X}: 0 \notin T^A} \text{EFM}(T^A), \\ \text{EFM}(T) &\subseteq \text{EFM}(F) \end{aligned}$$

where $0^N \in \mathbb{R}^N$ with $0_r^N = 0$ for all $r \in N = \bigcup_{A \in \mathcal{X}: 0 \in T^A} A$.

If $T = \prod_{A \in \mathcal{X}} T^A$, then the first inclusion becomes an equality.

If $\bar{\ell}_{\bar{c}} \leq 0$, then both inclusions become equalities. \square

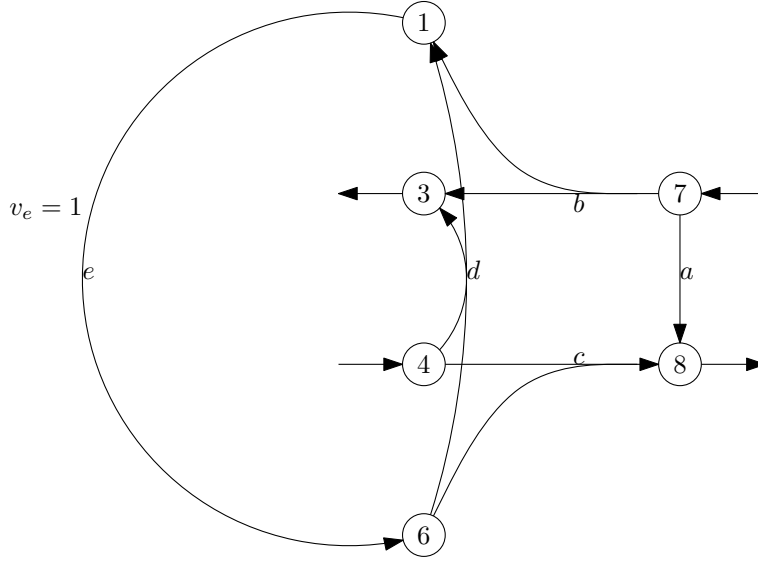


Figure 6.3: A module of the network shown in Fig. 6.1, which partitions the whole network together with the module shown in Fig. 6.2. The interface fluxes are represented by new exchange reactions (arcs that only involve one metabolite). All exchange (interface) fluxes have value 1. It can easily be seen that this network does not contain any internal cycles, hence every steady-state flux is thermodynamically feasible.

PROOF We start with the first inclusion. Let $v \in \text{EFM}(T)$ be fixed but arbitrary. Assume there exists $A \in \mathcal{X}$ with $0 \in T^A$ and $v_A \neq 0$. We define $w \in \mathbb{R}^{\mathcal{R}}$ by $w_B = v_B$, for all $B \in \mathcal{X} \setminus \{A\}$, and $w_A = 0$. It follows by Thm. 6.3.1 that $w \in F$. By construction, we have $\text{sign}(w) \subset \text{sign}(v)$. Since v is thermodynamically feasible, it follows that w is thermodynamically feasible, hence $w \in T$. Since $b \neq 0$, it follows that $w \neq 0$, which is a contradiction to the minimality of v .

Assume there exists an $A \in \mathcal{X}$ with $0 \notin T^A$ s.t. $v_A \notin \text{EFM}(T^A)$. Since $v_A \in T^A$ by Thm. 6.3.1, it follows that there exists $w^A \in T^A$ with $\text{sign}(w^A) \subset \text{sign}(v_A)$. We now define $w \in \mathbb{R}^{\mathcal{R}}$ by $w_B = v_B$, for all $B \in \mathcal{X} \setminus \{A\}$, and $w_A = w^A$. It follows by Thm. 6.3.1 that $w \in F$. By construction we have $\text{sign}(w) \subset \text{sign}(v)$. Since v is thermodynamically feasible, it follows that w is thermodynamically feasible, hence $w \in T$, which is a contradiction to the minimality of v . Therefore

$$\text{EFM}(T) \subseteq 0^N \times \prod_{A \in \mathcal{X}: 0 \notin T^A} \text{EFM}(T^A).$$

The second inclusion follows directly from Prop. 6.3.2 a) by choosing $A = \mathcal{R}$.

Now we consider the case where $T = \prod_{A \in \mathcal{X}} T^A$. Let $v^A \in \text{EFM}(T^A)$ be a fixed but arbitrary elementary flux mode for each $A \in \mathcal{X}$ with $0 \notin T^A$. By assumption, it follows that $v \in \mathbb{R}^{\mathcal{R}}$ defined by $v_A = v^A$, for $A \in \mathcal{X}$ with $0 \notin T^A$, and $v_A = 0$, for $A \in \mathcal{X}$ with

$0 \in T^A$, satisfies $v \in T$. Assume there exists a flux vector $w \in T$ with $\text{sign}(w) \subset \text{sign}(v)$. It follows that there exists a reaction $r \in \text{supp}(v) \setminus \text{supp}(w)$. Since the T -modules form a partition of \mathcal{R} , it follows that there exists a T -module $A \in \mathcal{X}$ with $r \in A$ and $0 \notin T^A$. It follows that $\text{sign}(w_A) \subset \text{sign}(v^A)$. By Thm. 6.3.1 it follows that $w_A \in T^A$, a contradiction to the minimality of v^A . Hence, we conclude that if $T = \prod_{A \in \mathcal{X}} T^A$, then

$$\text{EFM}(T) \supseteq 0^N \times \prod_{A \in \mathcal{X}: 0 \notin T^A} \text{EFM}(T^A),$$

which implies

$$\text{EFM}(T) = 0^N \times \prod_{A \in \mathcal{X}: 0 \notin T^A} \text{EFM}(T^A).$$

If we have $\bar{\ell}_c \leq 0$ and $T \neq \emptyset$, it follows by Thm. 6.3.1 that $T = \prod_{A \in \mathcal{X}} T^A$ and thus we get the first inclusion. The second inclusion follows from Prop. 6.3.2 c) by choosing $A = \mathcal{R}$. ■

6.3.3 Finding Flux Modules

We will now discuss methods for computing the decomposition of \mathcal{R} into minimal (w.r.t. set inclusion) T -modules. We have already shown that this decomposition is always well defined (Thm. 6.3.2).

There exist currently three methods to compute flux modules or variants of them. The first method, by Kelk et al. [75] was the starting point for research on flux modules, since they found that such things as flux modules actually exist in practice. For the computation they used a highly complicated and inefficient method. The followup works by myself then improved this method using additional theoretical findings. Although I consider it nowadays unlikely that anyone would want to implement any of the two previous methods, I will describe these methods here to give a complete overview.

6.3.3.1 The Vertex Correlation Method

This method was developed by Kelk et al. [75]. I did not contribute in the development of this method, it is just stated here as a comparison point to my other two works.

It is widely known that *flux balance analysis* (FBA) as introduced in Sec. 4.1 does not give unique solutions [95]. This is exhibited by *flux variability analysis* (FVA) which shows for each reaction the range of values that it can attain. However, FVA does not tell us anything about the dependencies between the variabilities.

In their pioneering work Kelk et al. [75] studied the alternative optimal FBA-solutions \bar{P} :

$$\begin{aligned} \bar{P} &:= \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, \ell \leq v \leq u, cv = \text{opt}\}, & \text{where} \\ \text{opt} &:= \max\{cv : Sv = 0, \ell \leq v \leq u\} \end{aligned}$$

In the following we will call \bar{P} the optimal flux space.

Therefore, they did something related to EFM enumeration: They tried to enumerate the vertices of the optimal flux space \bar{P} . However, it is very common that \bar{P} contains linealities (for example from thermodynamically infeasible cycles) so that \bar{P} is not pointed and hence, does not contain any vertices.

Because of this problem, they first eliminated all the linealities by deleting one of the involved reactions. Since the lineality can operate in any direction it follows that metabolic flow through the deleted reaction can always be substituted by the other reactions in the lineality. This way they obtained a flux space P' that was pointed.

To analyze P' , they then computed all the vertices of P' . Although the number of vertices also explodes with the network size, they were able to compute the vertices for genome-scale metabolic networks, since usually only a small portion of the reactions show variability in the optimal flux space and only these induce vertices.

With a correlation analysis they then discovered that there are clusters of reactions that are correlated and which are entirely uncorrelated to reactions outside the cluster. These clusters correspond to what I call flux modules.

We can summarize these findings as follows:

- The method by Kelk et al. [75] was not developed to find flux modules, but flux modules were actually just a by-product that was found by their novel vertex enumeration method.
- Due to the “curation”-step of the optimal flux space \bar{P} to P' , the computed clusters do not always coincide with the flux modules. There is another technical difficulty that we will discuss in a later section in more detail.

If we assume that the flux space P is already pointed (and hence we can omit the steps for eliminating linealities), then also this method can be considered an application of Thm. 6.3.1, or more precisely of the following corollary:

Corollary 6.3.2 *Let $P := \{v : Sv = b, \ell \leq v \leq u\}$ or $P := T$ with $P \neq \emptyset$ and $\ell_{\mathcal{I}} \leq 0 \leq u_{\mathcal{I}}$. Let \mathcal{X} be a partition of \mathcal{R} into P -modules. For every $A \in \mathcal{X}$ let $\Omega_A = \text{pr}_A(P)$ be the sample space of a fixed but arbitrary probability space. Let $B \in \mathcal{X}$, $r \in B$, $s \in \mathcal{R} \setminus B$ be arbitrary but fixed. Let*

$$E_1 = \{v \in P : v_r \in X\}, \quad E_2 = \{v \in P : v_s \in Y\}$$

be events, where $X, Y \subseteq \mathbb{R}$. Then, E_1, E_2 are independent in the product probability space $P = \prod_{A \in \mathcal{X}} \Omega_A$. \square

PROOF By Thm. 6.3.1, we have $P = \prod_{A \in \mathcal{X}} \Omega_A$. The independence follows directly from the definition of product probability space. ■

Let us consider the following discrete probability measure for each module A (under the assumption $b \neq 0$ it follows that $\text{EFM}(P)$ is finite):

$$\text{Pr}(v \in \text{pr}_A(P)) := \frac{|\{w \in \text{EFM}(P) : w_A = v\}|}{|\text{EFM}(P)|},$$

where $P = \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b, \ell \leq v \leq u\}$ and $b \neq 0$. It follows by Cor. 6.3.2 that the random variables $X_r, X_s : \mathbb{R}^{\mathcal{R}} \rightarrow \mathbb{R}$ with $X_r : v \mapsto v_r, X_s : v \mapsto v_s$ are independent if r and s belong to different modules. It follows that X_r and X_s can only be correlated if r and s belong to the same module.

By Thm. 6.3.3 it follows that the elementary flux modes are uniformly distributed in the product probability space P , i.e.,

$$\text{Pr}(v \in P) = \begin{cases} \frac{1}{|\text{EFM}(P)|} & v \in \text{EFM}(P) \\ 0 & \text{otherwise.} \end{cases}$$

Thus, X_r, X_s are exactly those flux variables that the vertex correlation method uses to compute flux correlations.

6.3.3.2 Using Flux Forcing and Several Runs of FVA

This method was developed by me and is published in [100]. It is repeated here for completeness' sake, although the algorithm in 6.3.3.3 is much more efficient.

This algorithm for computing T -modules is based on the insight that the flux space T can be written as the Cartesian product of the flux spaces for the T -modules (Thm. 6.3.1). From this, we can derive that if we fix the flux value of one reaction to a fixed value, this will have no influence on the flux variability of a reaction in a different T -module:

Corollary 6.3.3 *Let $P := \{v : Sv = b, \ell \leq v \leq u\}$ or $P := T$ with $\ell_{\mathcal{I}} \leq 0 \leq u_{\mathcal{I}}$. Assume $P \neq \emptyset$. Let \mathcal{R} be partitioned into P -modules. Let A be a P -module and $r \in A, s \in \mathcal{R} \setminus A$. Let x be a feasible flux rate for r , i.e., $x \in \text{pr}_r(P)$. Then*

$$\max\{cv_s : v \in P, v_r = x\} = \max\{cv_s : v \in P\} \quad \text{for all } c \in \mathbb{R}.$$

PROOF Since x is a feasible flux rate for r , there exists a flux vector $w \in P$ with $w_r = x$. Let v be a flux vector maximizing $\max\{cv_s : v \in P\}$. By Thm. 6.3.1 it follows that $v' \in \mathbb{R}^{\mathcal{R}}$, with $v'_A = w_A$ and $v'_{\mathcal{R} \setminus A} = v_{\mathcal{R} \setminus A}$, satisfies $v' \in P$. Since $s \notin A$, it follows that $cv_s = cv'_s$, showing $\max\{cv_s : v \in P, v_r = x\} \geq \max\{cv_s : v \in P\}$. The other direction is obvious. ■

We can now use this result to put together this method for computing minimal T -modules. Since we want to see dependencies between reactions in the same minimal T -module, we choose extreme values for x (minimal and maximal flux). This will likely cause big effects on the variability of reactions in the same minimal T -module, but by Cor. 6.3.3 we will see no effect on the variability in other reactions.

To compute candidate sets of T -modules, we compute a graph $G = (\mathcal{R}, E)$ using Alg. 6. For the algorithm to work, we assume that there exists no pathway that can carry unbounded flux with thermodynamic constraints (this is only possible if nutrient/energy uptake is unbounded). Alternatively, this can be achieved by assigning large upper and lower to the reaction flux rates.

Algorithm 6 Computation of candidate sets for T -modules.

1. Compute thermodynamic flux variability $v_r \in [v_r^{\min}, v_r^{\max}]$ for each reaction r in the network. Define $V := \{r : v_r^{\min} < v_r^{\max}\}$.
 2. Each reaction $r \notin V$ forms a T -module by itself.
 3. For each reaction $r \in V$ do the following
 - (a) Fix r to its maximal/minimal flux rate (which exists because of thermodynamic feasibility)
 - (b) Compute thermodynamic flux variability $v_s \in [v_s^{\min,r}, v_s^{\max,r}]$ for each reaction $s \in V$.
 - (c) If and only if $v_s^{\min,r} > v_s^{\min}$ or $v_s^{\max,r} < v_s^{\max}$, then we say that s is influenced by r and add the edge (r, s) to E .
 - (d) If and only if $v_s^{\min,r} > v_s^{\min} \geq 0$ or $v_s^{\max,r} < v_s^{\max} \leq 0$, we say that r forces flux through s .
 - (e) If and only if $v_s^{\min,r} = 0 = v_s^{\max,r}$, we say that r blocks flux through s .
 4. Compute the connected components $\mathcal{X} = \{A_1, \dots, A_n\}$ of G .
-

To run *thermodynamically constrained flux variability analysis* (tFVA), we use the fast-tFVA tool [101]. With it, we are able to run this algorithm on genome-scale networks like *E. coli* iAF1260 and *S. cerevisiae* iND750.

By Cor. 6.3.3 it follows that every connected component of G is a subset of a minimal T -module. To check if a subset $A \subset \mathcal{R}$ is indeed a T -module, we run Alg. 7 with $P = T$. This algorithm returns YES if and only if A is a P -module, because we individually minimize and maximize each component (metabolite) of the interface of the candidate P -module. If and only if A is a P -module, d is a fixed vector, i.e., for each metabolite the minimum and maximum must be the same. If there were a flux vector $v \in P$ with different interface, then also the maximum or minimum would be different.

Algorithm 7 Checks if a candidate set A is indeed a P -module. This algorithm works not only for the flux space $P = T$, but for arbitrary flux spaces P .

```

Input:  $A \subseteq \mathcal{R}$ 
 $M := \{m \in \mathcal{M} : \exists r \in A : S_{mr} \neq 0\}$ 
for  $m \in M$  do
   $d_{\min} := \min\{S_{mAv_A} : v \in P\}$ 
   $d_{\max} := \max\{S_{mAv_A} : v \in P\}$ 
  if  $d_{\min} \neq d_{\max}$  then
    return NO
  end if
end for
return YES

```

In practice, it rarely happens that the connected components are not T -modules. If, however, connected components are detected that are not T -modules, these have to be combined manually to form T -modules. This is an easy task if only two connected components A and B are not T -modules. Since the T -modules partition the set of all reactions, it follows that the union of A and B forms a minimal T -module. In general, however, there are exponentially many combinations possible.

Despite these theoretical problems the algorithm directly computed the modules in nearly all test cases. Only in two cases (*E. coli* iJR094 grown on *threonine* resp. *tryptophan*), two connected components of the interaction graph were not modules. The example in Fig. 6.4 with the following flux polytope shows how this can happen:

$$P = \left\{ v \in \mathbb{R}^8 \left| \begin{pmatrix} 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} v = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, v \geq 0 \right. \right\}.$$

For this reason, it is also important in practice to always check at the end of Alg. 6 if the computed connected components are indeed modules (Alg. 7). It follows from Thm. 6.3.2 that this problem is not an intrinsic property of minimal modules. Instead, it is caused by how we detect interactions between reactions. We also remark that the addition of thermodynamic constraints can also not be its sole cause, since the example network does not involve thermodynamic constraints.

Blocking Graph The blocking graph is a side-product with potential applications outside module detection. It visualizes reactions that are on alternative pathways. We define it as the directed graph $\mathcal{B} = (\mathcal{R}, E)$, where

$$E := \{(r, s) : (v_r = v_r^{\max} \rightarrow v_s = 0) \vee (v_r = v_r^{\min} \rightarrow v_s = 0) \ \forall v \in T\}.$$

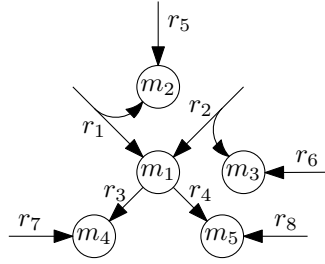


Figure 6.4: The proposed method computes that $\{r_1, r_5\}$, $\{r_2, r_6\}$, $\{r_3, r_7\}$, $\{r_4, r_8\}$ are minimal modules. However, this network only consists of exactly one minimal module.

We observe that these arcs are computed in step 3e of the Alg 6. It follows that no reactions of different T -modules will be connected in \mathcal{B} .

Hence, we will usually only look at the subgraph of the blocking interaction graph consisting of the nodes of a T -module. We will call this subgraph the *blocking graph* of the T -module. An example can be seen in Fig. 6.5.

6.3.3.3 Using Matroid Theory

This method was developed by me in a collaboration with Leen Stougie and is published in [102]

We recall Thm. 6.2.1 and Thm. 6.2.3, which applied to flux modules (0-modules) give:

Corollary 6.3.4 For $P \subseteq \{v \in \mathbb{R}^{\mathcal{R}} : Sv = b\}$ it holds for all $A \subseteq \mathcal{R}$ that

$$A \text{ is } P\text{-module} \Leftrightarrow A \cap V \text{ is } \ker(S_V)\text{-module.}$$

Corollary 6.3.5 $A \subseteq \mathcal{R}$ is a $\ker(S)$ -module if and only if A is a separator in the matroid represented by S . □

We recall further that V was defined as the set of reactions with flux variability:

$$\begin{aligned} V &:= \{r \in \mathcal{R} : v_r^{\max} \neq v_r^{\min}\}, \text{ where} & (6.2) \\ v_r^{\max} &:= \sup\{v_r : v \in P\} \\ v_r^{\min} &:= \inf\{v_r : v \in P\} \end{aligned}$$

The characterization of modules as separators of matroids allows us to compute the flux modules of a metabolic network efficiently. By the equivalence of modules and separators it follows from Thm. 6.3.2 that there exists a unique partition into minimal separators.

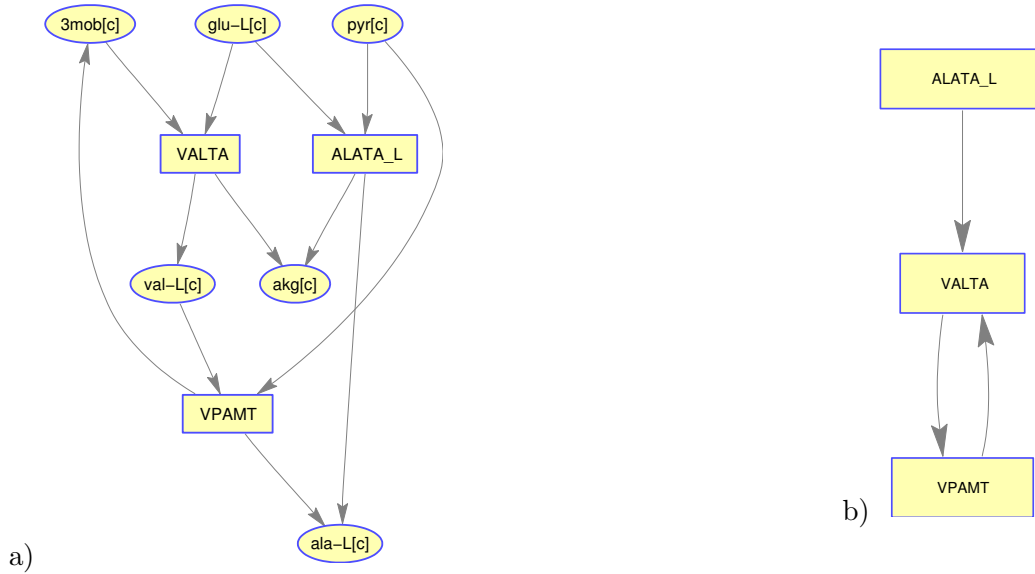


Figure 6.5: Graphical representations of a module of *E. coli* iAF1260.

a) The subnetwork of the module consisting of ALATA_L, VPAMT, and VALTA. The participating metabolites are drawn in ellipses. Stoichiometric coefficients are not shown.

b) The blocking graph of the module. An arc from reaction r_1 to r_2 means that if r_1 has minimal or maximal flux, then no flux is possible through r_2 . In this example it means that VPAMT and VALTA block each other when carrying optimal flux. However, VALTA does not block ALATA_L when carrying optimal flux. This shows that ALATA_L is also necessary for other pathways and crucial for optimal growth.

To understand the algorithm for finding the modules, we observe that the minimal non-trivial separators are the connected components of the matroid. In the context of graph-theory these are called 2-connected components (Note the inconsistency of the terminology between matroid and graph theory. The connected components in graph-theory are something different.) Formulated in matroid-terminology we recall the following graph-theoretic characterization of 2-connected component: For any 2 elements (columns of S in the linear matroid, edges in the graph) in the same connected component there exists a minimal dependent set (circuit) that contains them both. For pairs of elements of different connected components this is not true. It turns out that this also holds for matroids in general (Proposition 4.3.4 in [118]).

Theoretically, we could now build a graph $G = (V, E)$, where V is the set of reactions defined in (6.2) and there is an edge between two reactions (columns of S_V) if and only if there exists a circuit that contains both. The connected components (in the graph-theoretic sense) of G will be the minimal separators. Since the number of circuits explodes exponentially, it is not efficient to enumerate all circuits in order to compute the connected components of the graph G . Indeed, this is also not necessary and it suffices to look at a special set of circuits, so called *fundamental circuits* [161].

A set of fundamental circuits is obtained as follows: We start by finding a basis X of the matroid; i.e., a maximal independent set, which we compute by Alg. 8. Notice that, starting from the empty set, the algorithm grows X by adding elements only if this keeps X independent. Since we try to add all elements to X , it follows that at the end of the algorithm, X will be a basis of the linear matroid represented by S_V .

Let $Y := V \setminus X$. Clearly, for every $r \in Y$, adding r to X will create a cycle $C^r \subseteq X \cup \{r\}$. It is easy to see that C^r is actually a circuit, which is called fundamental circuit. In Alg. 8 the fundamental circuits are constructed simultaneously with constructing X . This gives us a so-called *partial representation*.

We now build, by Alg. 9, the graph $G' = (V, E')$, where two reactions are connected by an edge if there exists a fundamental circuit that contains both. Krogdahl and Cunningham showed that the connected components of G' , found by Alg. 9, are precisely the minimal separators of the matroid [28, 81].

To each circuit C there exists a flux vector v that is unique up to scaling with $C = \text{supp}(v)$, $Sv = 0$. If we enter for every circuit in B the corresponding flux values from v , we obtain a null-space matrix of S . Hence, this approach can be understood as computing a block-diagonalization of the null-space matrix. Approaches like this in the context of stoichiometric matrices have already been studied in [142]. However, [142] does not use matroid theory and it is unclear whether their method will always compute the finest block-diagonalization.

Algorithm 8 Computes a basis X and its set of fundamental circuits of a matroid represented by S

```
function ComputePartialRepresentation( $S$ )
 $\mathcal{F} = \emptyset$ 
 $X = \emptyset$ 
for  $r \in V$  do
  check feasibility of  $S_X v = -S_r$ 
  if feasible then
     $C := \text{supp}(v) \cup \{r\}$ 
     $\mathcal{F} := \mathcal{F} \cup \{C\}$ 
  else
     $X := X \cup \{r\}$ 
  end if
end for
return  $\mathcal{F}$ 
```

Here we recapitulate all the steps for finding the modules of the optimal flux space of a metabolic network.

1. Determine the optimal value by LP;
2. Set the objective function equal to the optimum value and add it as a constraint;

Algorithm 9 Computes the modules of $\{v : S_V v = 0\}$

```
function ComputeModules()
   $\mathcal{F} = \text{ComputePartialRepresentation}(S_V)$ 
  Build Graph  $G = \{V, E\}$  with  $(x, y) \in E$  iff there exists  $C \in \mathcal{C}$  with  $x, y \in C$ .
   $\mathcal{A} = \text{find connected components of } G$  (e.g. using depth-first search).
  return  $\mathcal{A}$ 
```

3. For each reaction r maximize and minimize the flux through r in the optimal flux space;
4. Determine the set V of reactions for which the maximum and the minimum are not equal;
5. Select the set of columns S_V corresponding to V of the stoichiometric matrix S and neglect the non-negativity constraints; i.e., irreversibilities, directions of the reactions;
6. Apply Alg. 9 to compute the minimal modules \mathcal{A} of $\{v \in \mathbb{R}^V : S_V v = 0\}$.
7. \mathcal{A} is the set of minimal modules that contain reactions in V . The reactions with fixed flux are all minimal modules by themselves.

We notice that steps 3 (and therefore 4) of the algorithm can be parallelized in a trivial way, reducing the computation times even further.

6.3.4 Comparison of the Methods

First of all we observe that since the FVA-based method (Sec. 6.3.3.2) and the matroid based method (Sec. 6.3.3.3) work using the same definitions, they also must (as proven) compute the same results. We also saw that we can see some aspects of the vertex correlation based method (Sec. 6.3.3.1) in our theoretical framework.

Indeed, we can even observe that the FVA-based method is not much different than the vertex correlation method. It also can be considered an application of Cor. 6.3.2. The main difference between the FVA method and the vertex correlation method can hence be considered in the choice of probability measures and events that are used to detect interactions between reactions.

Another difference between the FVA method and the vertex correlation method (besides complexity theoretic aspects) lies in how we deal with internal cycles. In the FVA method we employed NP-hard thermodynamic constraints, while in the vertex correlation method the vertices are directly computed and hence automatically the cycles (in form of rays) are omitted. Hence, both methods do not solve the original problem on the steady-state flux space, but slightly modified variants. In particular how the vertex correlation-based method is dealing with linealities can induce differences, as Example 6.3.3 shows.

Example 6.3.3 *Let us consider the metabolic network shown in Fig. 6.6. We observe that there is a reversible cycle involving the metabolites A, B, C, D , which is causing a lineality. This causes that the network cannot be partitioned into modules w.r.t. to its steady-state flux space.*

Assume now that we break the cycle by deleting the reaction $A \leftrightarrow D$. It can easily be seen that this implies that no flux through $C \leftrightarrow B$ is possible and we get 2 modules; the upper part and the lower part.

Let us now assume that we break the cycle by deleting the reaction $A \leftrightarrow B$. We observe that we can still use any of the other reactions, but do not have to (except the exchange reactions). Furthermore, we observe that in this case the network does not partition into two flux modules (in particular not into an upper and a lower part).

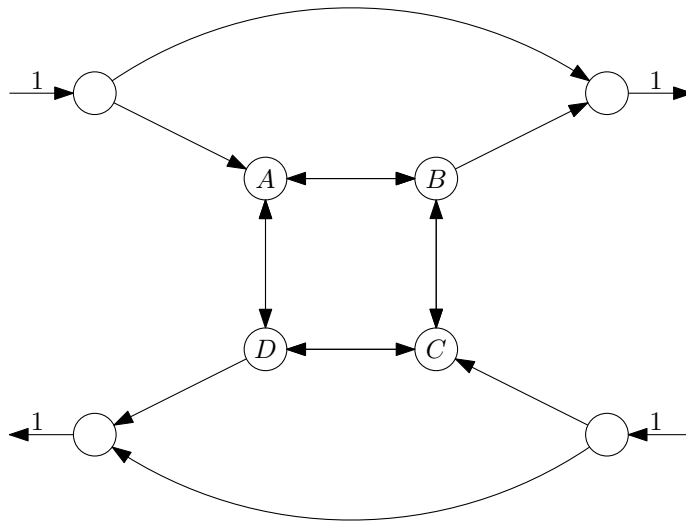


Figure 6.6: A metabolic network, where elimination of linealities by deletion causes partitions into modules that are not well-defined. The exchange fluxes are all fixed to a rate of 1. The flux rates of all the other reactions are unbounded.

We conclude that although the elimination of linealities by deletion can give us smaller (and hence more easily comprehensible) modules, the operation of deletion is highly unstable and the effect depends on which reaction of the cycle we delete. This effect is highly undesirable since in a biological analysis of the results, we may end up puzzling over an artifact caused by the deletion of a reaction in a cycle. Even worse, the results could end up being not reproducible. \square

The core observation of the matroid based method is however that we do not have to get rid of internal cycles at all. With this we found a polynomial time algorithm to compute a decomposition into minimal modules without requiring any restrictions at all. Additionally we can also apply it to arbitrarily restricted flux spaces - we only have to be able to identify reactions with flux variability and those without. The practical

efficiency can be seen in Tab. 6.1.

With the matroid-based method we can compute all flux modules for the optimal flux space of genome-scale networks in about the same time as is needed for conventional flux variability analysis. In Tab. 6.1 we see that the matroid-based method outperforms the previous methods by orders of magnitude. I used the metaopt toolbox [100] (see also Chapter 4) to solve the flux variability subproblems. Unfortunately, I did not have access to all the runtime data of [75] which is why some of the data is missing and the reported runtimes may be only from some of the steps in the pipeline. The computations for the matroid approach were obtained by computations on a 4-core desktop computer.

Table 6.1: Runtime comparison for module computation in the optimal flux space of genome-scale networks.

| Network | Vertex Correlation | FVA based | Matroid based | Matroid based with thermo. |
|-------------------------------|-----------------------|--------------|------------------|-------------------------------|
| <i>E. coli</i> iAF1260 | 133495 sec | 755 sec | 6.4 sec | 40.7 sec |
| <i>E. coli</i> iJR904 | 1906 sec | 162sec | 1.9 sec | 6.8 sec |
| <i>E. coli</i> iJO1366 | | | 8.4 sec | |
| <i>H. pylori</i> iT341 | | 55.5 sec | 0.8 sec | 2.3 sec |
| <i>H. sapiens</i> recon. 1 | | | 153.3 sec | |
| <i>H. sapiens</i> recon. 2 | | | 1131 sec | |
| <i>M. barkeri</i> iAF692 | 1088 sec | 941 sec | 1.4 sec | 7.3 sec |
| <i>M. tuberculosis</i> iNJ661 | 9317 sec | 1623 sec | 4.3 sec | 23.0 sec |
| <i>S. aureus</i> iSB619 | | 127.8 sec | 1.2 sec | 2.2 sec |
| <i>S. cerevisiae</i> iND750 | | | 3.0 sec | 28.1 sec |

I used the metaopt toolbox [99] (see also Ch. 4) to solve the flux variability subproblems.

In particular notice that large networks like *Human recon 2* can now also be analyzed. In addition, the new method is numerically much more stable. In the FVA-based method it often happens that error tolerances are chosen too small or too large, which causes that linear programs that should be feasible are detected as infeasible etc. This then usually caused the algorithm to abort and the tolerance sometimes needed to be adjusted according to the problem instance.

I experienced that the matroid based method is much more robust in this respect. The initial tolerances of 10^{-20} for the optimization step, 10^{-8} for the flux variability and 10^{-9} for the final module computation worked in all cases.

Note, that this comparison is slightly unfair, since the methods are solving slightly different problems as discussed above.

For example, the thermodynamic constraints have the following effect on *E. coli* iAF1260: 7 of the modules coincide with the steady-state modules and 2 modules contain reactions which have fixed flux under thermodynamic constraints. The other modules are found

in the steady-state flux space, but not in the thermodynamically constrained flux space since they contain only reactions that have fixed flux by thermodynamic constraints. Usually those modules are formed by a splitted pair of forward and backward reactions.

6.3.5 Computed Modules in Genome-Scale Metabolic Networks

In Table 6.2 we see a statistical summary over the modules computed for the optimal flux space of classical genome-scale metabolic networks. These results were obtained for the thermodynamically constrained flux space. Furthermore I tried to compute the optimal yield elementary modes from the computed decompositions into flux modules. However, due to the high sensitivity of the EFM solvers to rounding errors, I was not always able to compute all EFMs through each module. Those cases are indicated in Table 6.2 by a question mark. Detailed lists of the computed modules for different genome-scale metabolic networks can be found in Appendix A.1. In Appendix A.2, you can also find the set of optimal-yield EFMs for two selected networks.

Using the blocking graph, we may get additional information on interactions between the reactions inside of modules (see Sect. 6.3.3.2 and steps 3c, 3d, 3e of Alg. 6). Let us consider the module consisting of the reactions L-alanine transaminase (`ALATA_L`), valine transaminase (`VALTA`), and valine-pyruvate aminotransferase (`VPAMT`). This is depicted in Fig. 6.5a and the blocking graph is shown in Fig. 6.5b. By studying Fig. 6.5a, we might think that `VALTA` and `VPAMT` together form an alternative route to `ALATA_L`. However, a look at Fig. 6.5b reveals that if `VALTA` carries maximal flux, then `VPAMT` does not carry any flux and vice versa. The blocking interaction graph actually shows us that `VPAMT` and `ALATA_L` together form an alternative route to `VALTA`. Furthermore, we can derive that `ALATA_L` is also important for other pathways, since even maximal or minimal flux through `VALTA` cannot force flux through `ALATA_L` to zero.

The blocking graph may also give us information about which reactions may be subject to regulatory control in order to obtain a specific effect. For example, if we consider the module of *S. aureus* iSB619 shown in Fig. 6.7, we see that regulatory control on `LDH_D` or `LDH_L` will potentially influence what kind of lactate is produced.

6.3.5.1 Sensitivity to Growth Conditions in *E. coli*

When I analyzed *E. coli* iAF1260 grown on *glucose*, I discovered instead of the biggest module found by [75] three smaller modules, seen in Figs. 6.8, 6.9, 6.10, which mostly contain the same reactions. It turns out that the difference was actually not caused by the different analysis methods, but actually by slight modifications of the metabolic network. [75] used an uptake flux of at most 12.7777mmol/gDW/h (mmol per gram dry weight per hour) for *glucose*, while we used an uptake flux of 8mmol/gDW/h for *glucose* as originally given in the model. All other bounds on the network were essentially the same (they additionally allow uptake of *Cob(I)alamin*, which however is blocked in the network).

Table 6.2: Computed modules in genome-scale networks and the number of EFMs.

| Model | no. reactions | no. modules | no. efms | run time |
|---|---------------|-------------|--|----------|
| <i>E. coli</i> iJR904 on <i>Glucose</i> | 1075 | 8 | $2^6 \times 90 \times 96$ | 162s |
| <i>E. coli</i> iJR904 on <i>L-Threonine</i> | 1075 | 12 | $2^{10} \times 6 \times 90$ | 152s |
| <i>E. coli</i> iJR904 on <i>L-Arginine</i> | 1075 | 11 | $2^8 \times 6^2 \times 9$ | 139s |
| <i>E. coli</i> iJR904 on <i>Citrate</i> | 1075 | 11 | $2^8 \times 6 \times 15 \times 90$ | 138s |
| <i>E. coli</i> iJR904 on <i>Fumarate</i> | 1075 | 11 | $2^8 \times 6 \times 90$ | 144s |
| <i>E. coli</i> iJR904 on <i>L-glutamine</i> | 1075 | 10 | $2^7 \times 6 \times 90 \times 222$ | 145s |
| <i>E. coli</i> iJR904 on <i>Lactose</i> | 1075 | 8 | $2^6 \times 8 \times ?$ | 160s |
| <i>E. coli</i> iJR904 on <i>L-Malate</i> | 1075 | 10 | $2^7 \times 6 \times 15 \times 90$ | 135s |
| <i>E. coli</i> iJR904 on <i>L-Tryptophan</i> | 1075 | 7 | $2^5 \times 18 \times 96$ | 136s |
| <i>E. coli</i> iAF1260 on <i>Glucose</i> , aerobic | 2382 | 9 | $2^4 \times 3^2 \times 6 \times 54 \times 5184$ | 755s |
| <i>E. coli</i> iAF1260 on <i>Glucose</i> , anaerobic | 2382 | 9 | $2^6 \times 3 \times 6 \times 2592$ | 531s |
| <i>E. coli</i> iAF1260 on <i>Glucose</i> , limited oxygen | 2382 | 6 | $2^3 \times 3 \times 2592 \times ?$ | 976s |
| <i>E. coli</i> iAF1260 on <i>L-Threonine</i> , aerobic | 2382 | 9 | $2^4 \times 3 \times 8^2 \times 108 \times 4944$ | 6836s |
| <i>H. pylori</i> iT341 | 554 | 5 | $2^4 \times 18$ | 55.5s |
| <i>M. barkeri</i> iAF62 | 690 | 7 | $2^4 \times 3 \times 28 \times 156$ | 941s |
| <i>M. tuberculosis</i> iNJ661 | 1025 | 10 | $2^6 \times 3^3 \times ?$ | 1623s |
| <i>S. aureus</i> iSB619 | 743 | 10 | $2^7 \times 4 \times 192$ | 127.8s |
| <i>S. cerevisiae</i> iND750 | 1266 | 8 | $2^4 \times 5 \times 6 \times 80 \times ?$ | |

For each of the analyzed networks the table shows the number of computed modules in the optimal flux space with respect to the specified growth condition. If no growth condition is specified, the default from the BiGG-database [134] was used. I also computed the number of optimal elementary flux modes through each module. Since every combination of elementary flux modes of the modules gives an optimal elementary flux mode of the whole network, we did not compute the product and simply stated the factors. I use '?' to denote that we were not able to compute the elementary flux modes through a module, because there were too many elementary flux modes (≥ 300). I used the original networks from the BiGG-database, where no duplicate reactions were removed.

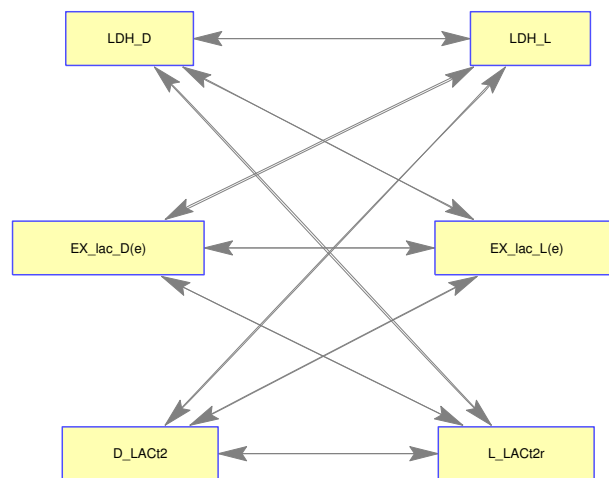


Figure 6.7: The blocking interaction graph shows clearly that (LDH_D, D_LACt2, EX_lac_D(e)) forms an alternative pathway to (LDH_L, D_LACt2r, EX_lac_L(e)).

A careful analysis of the network revealed that fluxes do not scale linearly with the uptake of *glucose* as assumed in [75]. This is caused by two reactions with small absolute flux bounds: The network requires a flux through *ATP maintenance* of 8.39mmol/gDW/h and maximal *oxygen* uptake of 18.5mmol/gDW/h. These values are fixed and do not scale with the *glucose* uptake.

If *E. coli* iAF1260 is allowed to have only an uptake flux of 8mmol/gDW/h for *glucose*, it will not consume all the *oxygen* to achieve optimal growth. However, with an uptake flux greater than 12mmol/gDW/h for *glucose*, it will require all the supplied *oxygen* to grow optimally. It follows that in this case the structure of the optimal flux space of *E. coli* iAF1260 will also change structurally. Consequently the optimal flux space gets partitioned into different modules.

To understand this structural change, we also analyzed anaerobic growth of *E. coli* iAF1260 under *glucose*. Interestingly, the modules shown in Figs. 6.8, 6.10 also existed under anaerobic growth conditions. This was unexpected since in the aerobic growth case with limited *oxygen* supply as studied by [75], these modules do not exist. Instead of the module in Fig. 6.9, we find a module consisting only of a subset of the reactions, as shown in Fig. 6.11.

A comparison of the modules in Fig. 6.9 (aerobic) and Fig. 6.11 revealed that the former is transforming *succinate* into *fumarate*, while the latter is doing the reverse transformation (see Tab. 6.3). Hence, the modules just look similar but actually perform a different metabolic function. This also explains why under limited *oxygen* supply these modules do not appear.

Furthermore, I found small changes regarding the second largest module found by [75]. Although it stays mostly the same, the reactions ACKr, ACS, ADK1, PTAr, R15BPK, R1PK

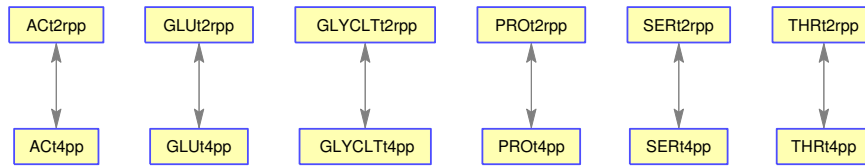


Figure 6.8: An arrow is drawn between two reactions in this module of *E. coli* iAF1260 (grown on *glucose*) if the reactions do not have a blocking interaction. This figure raises the assumption that we have 6 alternative pathways that realize the same function and that each of the pathways is realized by the two reactions connected by an edge. Indeed, this is the case: This module transports sodium from the periplasm to the cytosol in exchange to hydrogen.

leave and enter the module depending on the amount of *oxygen* supply. It is interesting to note that *ACKr* and *PTAr* are actually contained in the largest module from [75] (the module that decomposed into 3 smaller modules under high or no oxygen supply).

6.3.6 Visualizing the Interplay of Flux Modules

This method was developed together with Frank Bruggeman, Brett Olivier and Leen Stougie and is published in [102].

We observed that the decomposition of flux modules changes under different growth conditions and objective functions. One approach to understand these changes is to visualize the interplay of the modules to get a more intuitive understanding of the changes.

6.3.6.1 Method

By the definition of module, the reactions inside a module have together a fixed function (the interface flux). Hence, we can represent the module by a single reaction with a fixed flux in the genome-scale network. The stoichiometry of the representing reaction is precisely the interface flux of the module.

This way we can create a compressed network that contains all the reactions with fixed flux rates and artificial reactions that represent the modules. This compressed network has the following advantages:

- The number of reactions carrying flux is compressed (a module with many reactions, is represented by a single reaction).

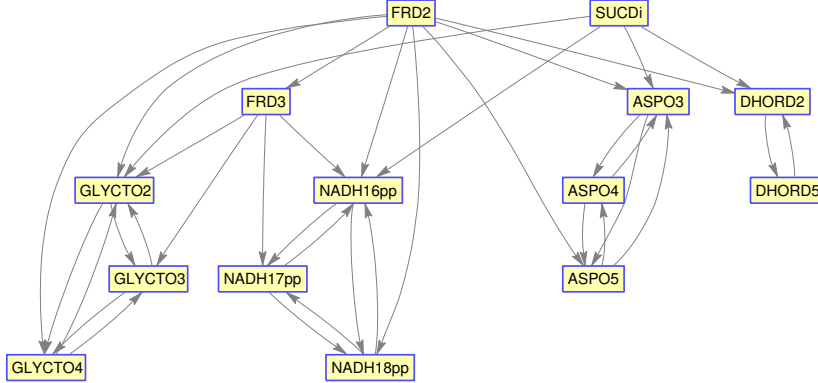


Figure 6.9: Blocking interaction graph of a module of *E. coli* iAF1260 (grown on *glucose*, aerobic). Here, we see that this module has 4 submodules which interact by the reactions FRD2, FRD3 and SUCDi.

- All the reactions in the compressed network have a fixed flux rate.

Unfortunately, the number of fixed reactions is still very large. This prevents automatic visualization of the network and the role of the modules containing variable reactions is obfuscated. However, reactions that have a fixed flux rate can also be grouped together into modules by Lemma. 6.3.1.

Theoretically, we could group all reactions with a fixed flux rate into 1 module. This would result in a compressed metabolic network consisting of $k + 1$ reactions, where k is the number of minimal modules containing reactions with variable flux rates. In particular, the module containing all fixed reactions will likely also contain the biomass- and nutrient-uptake reactions. If we want to understand the role of the modules for biomass production or nutrient uptake, this is not very useful. Moreover, modules of variable reactions may disconnect reactions with fixed flux rates from each other. Such disconnected reactions are important for the mediation between modules and should also be displayed separately. Hence, we decided to build a compressed network as follows:

1. Given: A collection Mod of interesting modules (selected by the user). Mod has to cover all reactions with variable flux rates. Typically Mod contains all minimal modules of variable reactions, a module containing the biomass reaction and modules containing the nutrient uptake reactions.
2. We compute the set $\mathcal{R}_{\text{Mod}} := \{r \in \mathcal{R} : r \in M \exists M \in \text{Mod}\}$ of reactions in interesting modules.

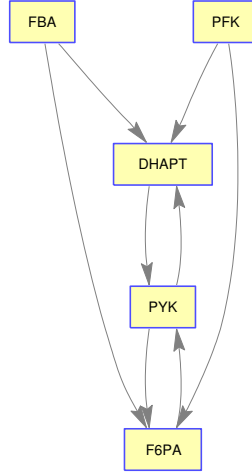


Figure 6.10: Blocking interaction graph of a module of *E. coli* iAF1260 (grown on *glucose*, aerobic). This diagram proposes the thesis that F6PA and DHAPT form an alternative pathway to FBA, PFK, and PYK. Indeed, both form two alternative pathways for transforming D-fructose 6-phosphate and phosphoenolpyruvate into dihydroxyacetone phosphate, pyruvate and glyceraldehyde 3-phosphate. The reason why the edges from FBA and PFK are not bidirectional is that these reactions are also used in other pathways.

3. We compute the set $\mathcal{R}_B := \{r \in \mathcal{R} \setminus \mathcal{R}_{\text{Mod}} : v_r = 0 \forall v \in P\}$ of blocked reactions.
4. We compute the set $\mathcal{M}_{\text{Mod}} := \{m \in \mathcal{M} : \exists r \in \mathcal{R}_{\text{Mod}} \text{ such that } m \in \text{supp}(S_r)\}$ of metabolites involved in the interesting modules.
5. We consider the metabolic network, where \mathcal{R}_{Mod} , \mathcal{R}_B and \mathcal{M}_{Mod} are removed. It is represented by the stoichiometric matrix $S' := S_{\mathcal{M} \setminus \mathcal{M}_{\text{Mod}}, \mathcal{R} \setminus (\mathcal{R}_{\text{Mod}} \cup \mathcal{R}_B)}$.
6. We compute the connected components Mod_F of S' . We do so by defining the incidence matrix of a bipartite graph, the nodes of which on one side of the bipartition correspond to the rows of S' , and the ones on the other side tot the columns of S' , and there is an edge between row-node i and column-node j if and only if $S'_{ij} \neq 0$. The column-nodes represent the reactions in $\mathcal{R} \setminus (\mathcal{R}_{\text{Mod}} \cup B)$, and the corresponding reactions of the connected components of this bipartite graph, whence Mod_F , forms a partition of $\mathcal{R} \setminus (\mathcal{R}_{\text{Mod}} \cup B)$. Clearly, every $A \in \text{Mod}_F$ is a module, since Mod_F only contains fixed reactions.
7. We represent each module in Mod, Mod_F by a single reaction with the corresponding interface flux. Let \mathcal{M}_0 be the set of metabolites that have a net interface flux of 0 in all these modules. We suppress \mathcal{M}_0 , since they would just show up as isolated metabolites. We obtain a metabolic network with metabolites $\mathcal{M}' := \mathcal{M} \setminus \mathcal{M}_0$ and reactions $\mathcal{R}' := \text{Mod} \cup \text{Mod}_F$.

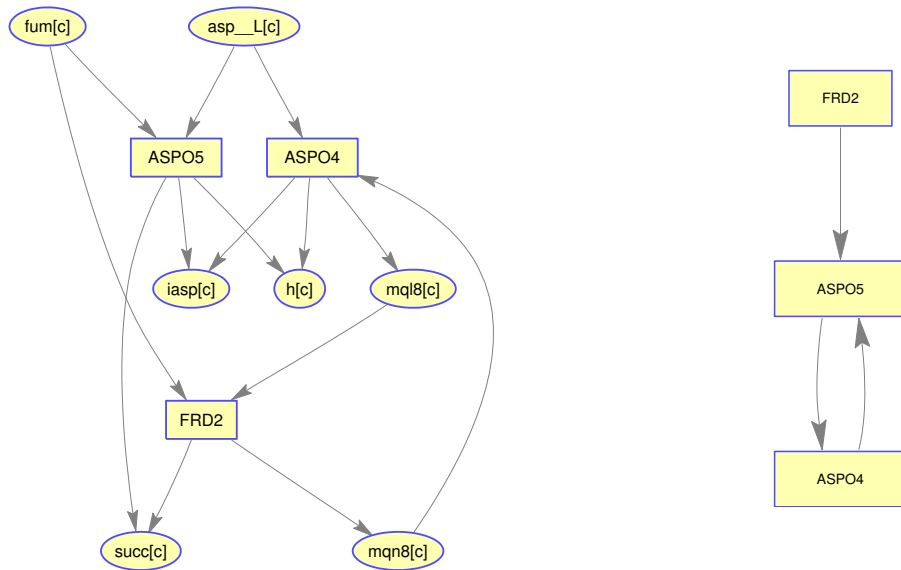


Figure 6.11: In anaerobic growth of *E.coli* iAF1260 under *glucose*, *fumarate reductase* (FRD2) and two kinds of *L-aspartate oxidase* (ASP04, ASP05) already form a module. Under aerobic growth, these are only part of a bigger module, which is shown in Fig. 6.9. This module essentially transforms *fumarate* and *L-aspartate* into *iminoaspartate* and *succinate*. FRD2 again is also used in a different pathway and additionally transforms small amounts of *menaquinol 8* into *menaquinone 8*.

8. We remove reactions disconnected from the network that contain the target reaction, e.g. because of modules that form thermodynamically infeasible cycles or otherwise have no role in the metabolism.

In practice, this results in medium-scale networks that can automatically be visualized with graph-drawing software like *GraphViz* [54].

6.3.6.2 An Anomaly of Flux Modules

Consider the metabolic network shown in Fig. 6.12. We observe that the upper three reactions $A = \{(m_1 \rightarrow m_3), (m_1 \rightarrow m_5), (m_5 \rightarrow m_3)\}$ form a flux module. The interface flux d_A of this flux module satisfies $d_1 = -1$, $d_3 = 1$ and $d_i = 0$ for all $i = 2, 4, 5$. If we simplify this network, we obtain a network as shown in Fig. 6.13. Note that the stoichiometry for metabolite m_5 in the compressed network in the new reactions representing the modules is actually 0 (hence, it is drawn with undirected edges).

According to the algorithm presented in Sec. 6.3.6.1 we would not draw these edges and even omit metabolite m_5 . This can lead to the misconception that there is no interplay between the two modules and thus to wrong interpretations. For example in

Table 6.3: Interface flux comparison of the Modules shown in Fig. 6.9 and Fig. 6.11.

| Metabolite | interface flux aerobic module | interface flux anaerobic module |
|----------------------------|----------------------------------|------------------------------------|
| <i>L-Aspartate</i> | -0.001678 | -0.0005999 |
| <i>Iminoaspartate</i> | 0.001678 | 0.0005999 |
| <i>Fumarate</i> | 3.729 | -0.08771 |
| <i>Succinate</i> | -3.729 | 0.08771 |
| H^+ | -114.2 | 0.0005999 |
| H^+ (<i>periplasm</i>) | 85.67 | 0 |
| <i>Glyoxylate</i> | 0.0004929 | 0 |
| <i>Glycolate</i> | -0.0004929 | 0 |
| <i>NAD</i> | 28.56 | 0 |
| <i>NADH</i> | -28.56 | 0 |
| <i>(S)-Dihydroorotate</i> | -0.2437 | 0 |
| <i>Orotate</i> | 0.2437 | 0 |
| <i>Ubiquinone-8</i> | -32.53 | 0 |
| <i>Ubiquinol-8</i> | 32.53 | 0 |
| <i>Menaquinol 8</i> | 0 | -0.08711 |
| <i>Menaquinone 8</i> | 0 | 0.08711 |

All metabolites are cytosolic except where stated otherwise. The flux units are in mmol/gDW/h (mmol per gram dry weight per hour). By definition of module these interface fluxes are constant for all optimal flux vectors under the corresponding growth condition.

^{13}C labeling experiments, because if metabolite m_1 is labeled, it can actually happen that also metabolite m_4 becomes labeled. If the reactions representing modules were actually proper reactions this would be impossible. Hence, we suggest that metabolites that are involved in a module but actually do not participate in the interface flux should also be marked in the compressed network, for example with such an undirected edge.

6.3.6.3 Results

We used the visualization method to create visualizations of the above mentioned genome-scale networks. In Tab. 6.4, we compare the original size of the networks with the size of the compressed networks that are used to visualize the interplay of the flux modules with variable flux rates. Each reaction of the compressed network is a flux module. Every minimal flux module containing reactions with variable flux rates is represented by exactly one reaction. Reactions with fixed flux rate are grouped together. It is interesting to see that although the networks have quite different sizes originally, the compressed sizes do not vary very much.

Visualizations of some of the example networks and their modules, using the tool dot

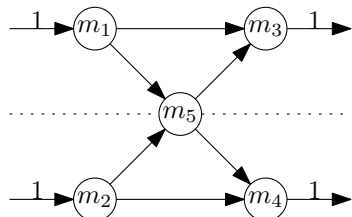


Figure 6.12: A metabolic network with two flux modules that share metabolites.

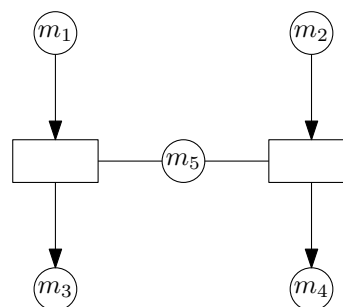


Figure 6.13: Compressed form of the metabolic network.

Table 6.4: Size of the compressed networks.

| Network | No. Metabolites (original) | No. Reactions (original) | No. Metabolites (compressed) | No. Reactions (compressed) |
|-------------------------------|-------------------------------|-----------------------------|---------------------------------|-------------------------------|
| <i>E. coli</i> iAF1260 | 1668 | 2382 | 46 | 25 |
| <i>E. coli</i> iJR904 | 761 | 1075 | 42 | 17 |
| <i>E. coli</i> iJO1366 | 1805 | 2583 | 49 | 27 |
| <i>H. pylori</i> iT341 | 485 | 554 | 32 | 20 |
| <i>M. barkeri</i> iAF692 | 628 | 690 | 35 | 13 |
| <i>M. tuberculosis</i> iNJ661 | 826 | 1025 | 58 | 26 |
| <i>S. aureus</i> iSB619 | 655 | 743 | 39 | 22 |
| <i>S. cerevisiae</i> iND750 | 1061 | 1266 | 57 | 24 |

[53] from the `GraphViz` toolbox, can be found on the website <https://sourceforge.net/projects/fluxmodules/>, which was created for the publication that described this method. The MATLAB scripts for module detection can be found there as well.

6.3.7 Conclusion

We observed that the decomposition into minimal flux modules highly depends on the growth conditions, like available nutrients and objective function. Also, it can be easily seen that as soon as we relax the optimality condition a bit to also allow suboptimality, we most likely lose all interesting modules.

In many metabolic networks something like the zero-flow is a feasible solution (usually there is a maintenance reaction, but for the sake of the argument we can forget about it). It follows that for many flux modules 0 has to be in the interface space (since $S_A v_A = 0$ if $v_A = 0$). Biologically this means that the flux module is isolated from the rest of the network and hence probably a modeling error.

As soon as we allow suboptimality, we allow convex combinations (to a certain degree) with the full flux space. It follows that the suboptimal flux space has the same dimension as the full flux space and hence, by Thm. 6.2.1 also the same flux modules. However,

biomass yield is almost never the only objective criterion which gives evolutionary advantage and most organisms usually do not exhibit optimal yield rates. Hence, it is very important to also consider suboptimal solutions.

There are now two remedies to the problem:

1. Compute modules for many different nutrient sources and objective functions and try to see a pattern. With the matroid based computation approach, we can now compute flux modules very efficiently and hence this method became a practical approach. The visualization method presented in Sec. 6.3.6 is one of the approaches with which this can be done. However, during the work with the visualization method, I observed that the visualization is very unstable even w.r.t. to small changes. Hence, further work will be needed to turn this into a more generally useful approach.
2. Generalize the notion of flux module to obtain something that is more robust to perturbations in the growth condition and in particular to suboptimality. A possible generalization are the already introduced k -modules.

6.4 Decomposition Theorem for Linear 1-Modules

Similar to the decomposition theorem for flux modules, we can derive a decomposition theorem for linear 1-modules. In the following, we restrict ourselves to polyhedra $P = \{v : Sv = b, v_{\text{Irrev}} \geq 0\} \neq \emptyset$.

We assume that we are given a partition \mathcal{X} of \mathcal{R} into linear 1-modules. If we consider P to be the flux space of a metabolic network, the decomposition theorem can be understood as follows: We build a compressed network that contains one reaction for each linear 1-module $A \in \mathcal{X}$. The stoichiometry of each reaction will be the variable interface d^A of the corresponding linear 1-module A . In the compressed network, we will also have irreversible reactions, which represent irreversible 1-modules:

Definition 6.4.1 (irreversible 1-Module) $A \subseteq \mathcal{R}$ is called an irreversible 1-module if there exists a $d \in \mathbb{R}^{\mathcal{M}}$ s.t. for every $v \in P$ there exists an $\alpha \geq 0$ s.t.

$$S_A v_A = d\alpha.$$

d is called the positive variable interface of A . We call a linear 1-module A reversible if it is not irreversible. □

We will assume that if a linear 1-module A is irreversible, then d^A will be the positive variable interface. This leads us to the following result which tells us that every feasible flux in the compressed network can easily be turned into a feasible flux of the whole network and vice versa:

Theorem 6.4.1 For every irreversible 1-module $A \in \mathcal{X}$, let $v^A \in \mathbb{R}^A$ be an arbitrary but fixed flux vector satisfying $S_A v^A = d^A$, $v_{\text{Irrev}}^A \geq 0$, where d^A is the positive variable interface of A . For every reversible $A \in \mathcal{X}$, take two flux vectors $v^A \in \mathbb{R}^A$ and $\bar{v}^A \in \mathbb{R}^A$, arbitrary but fixed, satisfying $S_A v^A = d^A$, $v_{\text{Irrev}}^A \geq 0$ and $S_A \bar{v}^A = -d^A$, $\bar{v}_{\text{Irrev}}^A \geq 0$, where d^A is the variable interface of A .

Then it holds that:

1. There exists $\alpha^A \in \mathbb{R}$ (and $\alpha^A \geq 0$, if A is irreversible) for every $A \in \mathcal{X}$ s.t.

$$\sum_{A \in \mathcal{X}} d^A \alpha^A = b.$$

2. For all $\alpha^A \in \mathbb{R}$ (and $\alpha^A \geq 0$, if A is irreversible) $A \in \mathcal{X}$, satisfying

$$\sum_{A \in \mathcal{X}} d^A \alpha^A = b$$

we have that $v \in P$ for v defined by

$$\begin{aligned} v_A &= \alpha^A v^A & \forall A \in \mathcal{X} \text{ with } \alpha^A \geq 0 \\ v_A &= -\alpha^A \bar{v}^A & \forall A \in \mathcal{X} \text{ with } \alpha^A < 0 \end{aligned}$$

PROOF We prove both statements separately.

1. Since $P \neq \emptyset$, we can choose a $w \in P$. For each linear 1-module $A \in \mathcal{X}$ there exists an $\alpha^A \in \mathbb{R}$ (and $\alpha^A \geq 0$, if A is irreversible) such that $S_A w_A = d^A \alpha^A$. It follows that

$$\sum_{A \in \mathcal{X}} d^A \alpha^A = \sum_{A \in \mathcal{X}} S_A w_A = b.$$

2. Let $\alpha^A \in \mathbb{R}$ (and $\alpha^A \geq 0$, if A is irreversible) arbitrary but fixed for every $A \in \mathcal{X}$ such that

$$\begin{aligned} b &= \sum_{A \in \mathcal{X}} \alpha^A d^A. \\ \Rightarrow b &= \sum_{A \in \mathcal{X}: \alpha^A \geq 0} \alpha^A S_A v^A - \sum_{A \in \mathcal{X}: \alpha^A < 0} \alpha^A S_A \bar{v}^A && \text{(by definition of } v^A, \bar{v}^A) \\ &= \sum_{A \in \mathcal{X}} S_A v_A && \text{(by definition of } v) \\ &= S v \end{aligned}$$

Since for every A v_A is a positive multiple of v^A or \bar{v}^A and $v_{\text{Irrev}}^A \geq 0$, $\bar{v}_{\text{Irrev}}^A \geq 0$, we have $v_{\text{Irrev}} \geq 0$ and thus $v \in P$. \blacksquare

We can use this theorem to also characterize the set of EFMs using linear 1-modules. For technical reasons, we will split all reversible reactions in the compressed network. For notation we will introduce symbolic functions $r^+(A), r^-(A)$ that simply denote the forward (resp. backward) reaction that represent a 1-module A . If A is irreversible then there exists only the forward representative $r^+(A)$. Summing up, the compressed network has the same metabolite set \mathcal{M} as the original network, it consists of reactions $\mathcal{R}' = \{r^+(A) : A \in \mathcal{X}\} \cup \{r^-(A) : A \in \mathcal{X}, A \text{ reversible}\}$ and its stoichiometric matrix D satisfies $D_{r^+(A)} = d^A, D_{r^-(A)} = -d^A$ for every $A \in \mathcal{X}$. All reactions in \mathcal{R}' are irreversible. For a flux vector $v \in \mathbb{R}^{\mathcal{R}'}$ and an irreversible module 1-module A , we define $v_r^-(A) := 0$ (note that the reaction $r^-(A)$ actually does not exist).

The following corollary then states that every EFM in the compressed network can be turned by simple substitution into an EFM of the whole network and that every EFM of the whole network can be produced in such a way.

Corollary 6.4.1 *Let*

- E be the set of elementary flux modes of P ,
- E_+^A be the set of elementary flux modes of $P_+^A := \{v_A \in \mathbb{R}^A : S_A v_A = d^A, v_{\text{Irrev}}^A \geq 0\}$,
- E_-^A be the set of elementary flux modes of $P_-^A := \{v_A \in \mathbb{R}^A : S_A v_A = -d^A, v_{\text{Irrev}}^A \geq 0\}$,
- F^A be the set of elementary flux modes of $Q^A := \{v_A \in \mathbb{R}^A : S_A v_A = 0, v_{\text{Irrev}}^A \geq 0\}$,
- E'' be the set of elementary flux modes of $P' := \{\alpha \in \mathbb{R}^{\mathcal{R}'} : D\alpha = b, \alpha \geq 0\}$, and
- $E' := \{a \in E'' : a_{r^+(A)} = 0 \vee a_{r^-(A)} = 0 \forall A \in \mathcal{X} \text{ reversible}\}$.

Then

$$E = \bigcup_{a \in E'} \left(\prod_{\substack{A \in \mathcal{X} \\ a_{r^-(A)} = 0}} a_{r^+(A)} E_+^A \times \prod_{\substack{A \in \mathcal{X} \\ a_{r^-(A)} > 0}} a_{r^-(A)} E_-^A \right) \quad (\text{if } b \neq 0)$$

$$E = \bigcup_{a \in E'} \left(\prod_{\substack{A \in \mathcal{X} \\ a_{r^-(A)} = 0}} a_{r^+(A)} E_+^A \times \prod_{\substack{A \in \mathcal{X} \\ a_{r^-(A)} > 0}} a_{r^-(A)} E_-^A \right) \cup \bigcup_{A \in \mathcal{X}} F^A \times 0_{\mathcal{R} \setminus A} \quad (\text{if } b = 0).$$

PROOF (COROLLARY 6.4.1) \subseteq : Let $e \in E$. By definition of linear 1-module, there exists an $\alpha^A \in \mathbb{R}$ such that $S_A e_A = d^A \alpha^A$ for every $A \in \mathcal{X}$. If $d^A = 0$, we assume w.l.o.g. that $\alpha^A = 0$. We observe that for each module one of the following cases happens:

Case 1: $\alpha^A = 0$ and $e_A \neq 0$. $e' \in \mathbb{R}^{\mathcal{R}}$ with $e'_{\mathcal{R} \setminus A} = e_{\mathcal{R} \setminus A}$ and $e'_A = 0$ satisfies $e' \in P$ and has strictly smaller support. Since e is an EFM, it follows that $e' = 0$. Hence, $e \in F^A \times 0_{\mathcal{R} \setminus A}$. This is only possible if $b = 0$.

Case 2: $\alpha^A = 0$ and $e_A = 0$.

Case 3: $\alpha^A > 0$. It follows that $f^A := \frac{1}{\alpha^A} e_A \in P_+^A$. Assume f^A would not have minimal support, then we can select an $f' \in P_+^A$ with smaller support and $e' \in \mathbb{R}^{\mathcal{R}}$ with $e'_A = \alpha^A f'$ and $e'_{\mathcal{R} \setminus A} = e_{\mathcal{R} \setminus A}$ satisfies $e' \in P$ and $\text{supp}(e') \subset \text{supp}(e)$. Since $d^A \neq 0$, it follows that $e' \neq 0$. Thus, this is a contradiction to e being an elementary flux mode. Hence, $f^A \in E_+^A$.

Case 4: $\alpha^A < 0$. It follows that $f^A := \frac{1}{-\alpha^A} e_A \in P_-^A$. Assume f^A would not have minimal support, then we can select an $f' \in P_-^A$ with smaller support and $e' \in \mathbb{R}^{\mathcal{R}}$ with $e'_A = -\alpha^A f'$ and $e'_{\mathcal{R} \setminus A} = e_{\mathcal{R} \setminus A}$ satisfies $e' \in P$ and $\text{supp}(e') \subset \text{supp}(e)$. Since $d^A \neq 0$, it follows that $e' \neq 0$. Thus, this is a contradiction to e being an elementary flux mode. Hence, $f^A \in E_-^A$.

We further observe that if case 1 happens, we have already shown the inclusion. Hence, we only need to consider cases 2, 3, 4. We observe that we can choose $a \in \mathbb{R}^{\mathcal{R}'}$ as follows

$$\begin{aligned} a_{r+(A)} &:= \max\{\alpha^A, 0\} \\ a_{r-(A)} &:= \max\{-\alpha^A, 0\} \end{aligned}$$

and obtain

$$Da = \sum_{\substack{A \in \mathcal{X} \\ a_{r-(A)}=0}} a_{r+(A)} d^A - \sum_{\substack{A \in \mathcal{X} \\ a_{r-(A)}>0}} a_{r-(A)} d^A = \sum_{A \in \mathcal{X}} \alpha^A d^A = \sum_{A \in \mathcal{X}} S_A e_A = Se = b.$$

It follows from the analysis of cases 2, 3, 4 that

$$e \in \prod_{\substack{A \in \mathcal{X} \\ a_{r-(A)}=0}} a_{r+(A)} E_+^A \times \prod_{\substack{A \in \mathcal{X} \\ a_{r-(A)}>0}} a_{r-(A)} E_-^A.$$

We only need to show that $a \in E''$. We observe that by construction $a_{r+(A)} = 0 \vee a_{r-(A)} = 0$ for all reversible $A \in \mathcal{X}$.

Since $a \geq 0$ by construction, we obtain $a \in P'$. Assume $a \notin E'$, i.e., there exists an elementary mode $b \in E'$ with smaller support. By Theorem 6.4.1 it follows that there exists a flux mode $f \in P$ with

$$\begin{aligned} f_A &:= \frac{b_{r+(A)} - b_{r-(A)}}{a_{r+(A)} - a_{r-(A)}} e_A && \text{for all } A \in \mathcal{X} \text{ with } b_{r+(A)} \neq 0 \text{ or } b_{r-(A)} \neq 0 \\ f_A &:= 0_A && \text{for all } A \in \mathcal{X} \text{ with } b_{r+(A)} = 0 \wedge b_{r-(A)} = 0. \end{aligned}$$

Since $\emptyset \neq \text{supp}(b) \subset \text{supp}(a)$ and $d^A \neq 0$ for every $A \in \mathcal{X}$ with $b_{r+(A)} \neq 0$ or $b_{r-(A)} \neq 0$, we conclude that $\emptyset \neq \text{supp}(f) \subset \text{supp}(e)$, which is a contradiction. Hence, a must already have been an elementary mode of P' .

\supseteq : Clearly, if $b = 0$ then $F^A \times 0_{\mathcal{R} \setminus A}$ is also a set of elementary modes of P .

Hence, we only need to consider the product. Let $e_+^A \in E_+^A$ for every $A \in \mathcal{X}$, $e_-^A \in E_-^A$ for every reversible A , and let $a \in E''$. Let $v \in \mathbb{R}^{\mathcal{R}}$ be defined as

$$\begin{aligned} v_A &:= a_{r+(A)} e_+^A && \text{for all } A \text{ with } a_{r-(A)} = 0 \\ v_A &:= -a_{r-(A)} e_-^A && \text{for all } A \text{ with } a_{r-(A)} > 0 \end{aligned}$$

By Thm. 6.4.1 it follows that $v \in P$.

Assume v is not an elementary mode of P . It follows that there exists an $e \in E$ with $\emptyset \neq \text{supp}(e) \subset \text{supp}(v)$. Hence, there exists an $A \in \mathcal{X}$ s.t. $\text{supp}(e) \cap A \subset \text{supp}(v) \cap A$. Since A is a linear 1-module, it follows that there exists an $\alpha \in \mathbb{R}$ such that $S_A e_A = d^A \alpha$.

Case $e_A = \emptyset$: It follows that we can choose $\alpha = 0$. For every $B \in \mathcal{X}$ there exists a $\beta^B \in \mathbb{R}$ s.t. $d^B \beta^B = S_B e_B$. Let $a' \in \mathbb{R}^{\mathcal{R}'}$ with

$$\begin{aligned} a'_{r+(B)} &= \max\{\beta^B, 0\} \\ a'_{r-(B)} &= \max\{-\beta^B, 0\} \end{aligned}$$

It follows that $a' \in P'$. If $\text{supp}(a') = \emptyset$, it follows that there exists an $B \in \mathcal{X}$ with $S_B e_B = 0$ and $\text{supp}(e_B) \neq \emptyset$. It follows that $e_B \in F^B$, a contradiction to the minimality of either e_+^B or e_-^B .

Hence, $\emptyset \neq \text{supp}(a') \subset \text{supp}(a)$. However, this is a contradiction to the minimality of a .

Case $\alpha > 0$: It follows that $e_A \in P_+^A$. This is a contradiction to the elementarity of e_+^A , since $\emptyset \neq \text{supp}(e_A) = \text{supp}(e) \cap A \subset \text{supp}(v) \cap A = \text{supp}(e_+^A)$.

Case $\alpha < 0$: It follows that $e_A \in P_-^A$. This is a contradiction to the elementarity of e_-^A , since $\emptyset \neq \text{supp}(e_A) = \text{supp}(e) \cap A \subset \text{supp}(v) \cap A = \text{supp}(e_-^A)$. \blacksquare

6.4.1 Computation of Linear 1-Modules

Similar to flux modules, we can employ the connection to matroid theory to efficiently compute a decomposition into 1-modules. In this case it translates to computing a decomposition of the matroid into 2-separators.

Therefore, I used an algorithm by Bixby and Cunningham [13] to test 3-connectivity in matroids. This algorithm assumes that the matroid is simple, i.e., that it does not contain parallel or coparallel elements. Translated to metabolic networks this means that our network must not contain parallel or fully-coupled reactions (see Sec. 6.4.2 for a precise definition). As we will see in the following section (Sec. 6.4.2) sets of parallel and fully-coupled reactions also form 1-modules, which allows us to compress such reaction

sets to single reactions and this way obtain a metabolic network that does not contain parallel or fully-coupled reactions. To find parallel and fully-coupled reactions, I used a modification of the algorithm implemented by Laszlo David [31] (by matroid duality the algorithm is also applicable to find parallel reactions). Also, the matroid must be 2-connected. Translated to metabolic networks this means that the network must consist out of a single flux module. This is easily achieved by application of the flux module detection algorithm presented in Sec. 6.3.3.3.

The algorithm by Bixby and Cunningham [13] is rather involved and technical, which is why I will not explain its details here but only explain how I used it to compute a decomposition into 1-modules. By the previous considerations we can assume that our metabolic network with flux space $P \subseteq \{v : Sv = b\}$ consists of only one flux module (0-module). Since we are interested in linear 1-modules, it follows by Thm. 6.2.2 that we can assume w.l.o.g. that the flux space of the metabolic network is of the form $P \subseteq \{v : (S|b)v = 0\}$, where an additional reaction with stoichiometry b was added (if $b \neq 0$, otherwise we don't have to do anything).

In the case that the matroid is not 3-connected, the algorithm outputs a 2-separator A , which is exactly such a 1-module that we are interested in. We note that if A is a 2-separator then also $\mathcal{R} \setminus A$ is a 2-separator and, interpreted as 1-modules, both have the same variable interface flux d^A .

We can now replace the 1-module by a single reaction with the function (interface) of the 1-module, or equivalently consider the matroid generated by $(S_{\mathcal{R} \setminus A} | d^A)$. Since a matroid is 3-connected iff it does not contain any non-trivial 2-separator, we can recursively apply this algorithm to compute a decomposition into 2-separators, i.e., 1-modules. The algorithm will terminate, when all computed 1-modules cannot be decomposed further.

6.4.2 Linear 1-Modules in Practice

The simplest linear 1-modules are sets of parallel and fully-coupled reactions. Two reactions r, s are called parallel if their stoichiometries are multiples of each other, i.e., $S_r = \lambda S_s$ with $\lambda \in \mathbb{R}$. They are called fully coupled if it holds for all fluxes $v \in P$ that $v_r = \lambda v_s$ with $\lambda \in \mathbb{R}$ [21]. A set of parallel reactions is forming a linear 1-module since they all have the same stoichiometry which then also defines the variable interface of the linear 1-module (see Fig. 6.14). In the case of fully-coupled reactions, we also obtain a linear 1-module. There, the variable interface is obtained by $S_r + \lambda S_s$ (see Fig. 6.15).

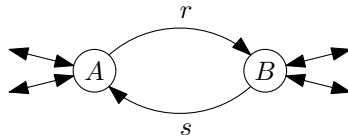


Figure 6.14: The reaction $A \rightarrow B$ and $B \rightarrow A$ are parallel ($S_r = -S_s$). Together, they form a reversible linear 1-module.

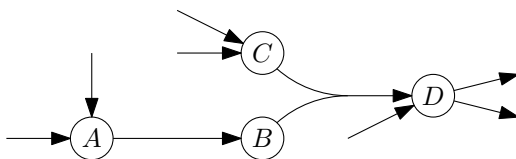


Figure 6.15: The reactions $A \rightarrow B$ and $B + C \rightarrow D$ are fully coupled. Together, they form a linear 1-module with variable interface of $A + C \rightarrow D$.

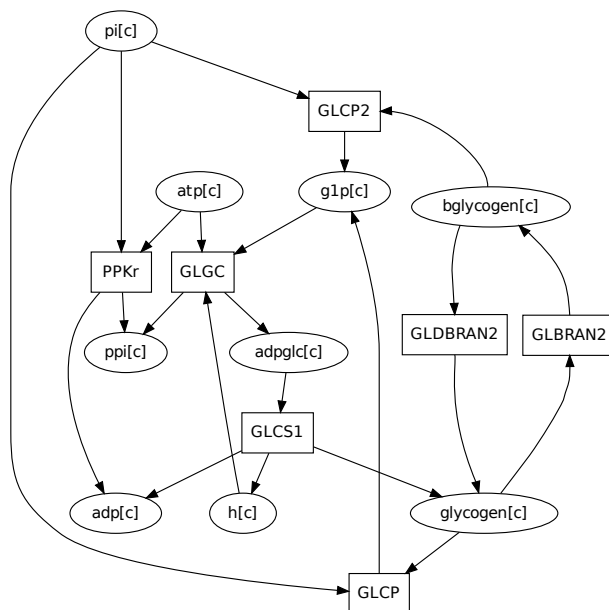


Figure 6.16: This is a 1-module in the full flux space of *E. coli* iAF1260. It can be represented by the reaction $\text{atp}[c] + \text{pi}[c] \rightarrow \text{adp}[c] + \text{ppi}[c]$. This module is a chain of parallel and fully coupled reactions. It can easily be seen that **GLBRAN2** and **GLDBRAN2** are performing the same function only with different directions. Hence, together they form a module. Together, they are fully coupled to **GLCP2** with which they form an alternative to **GLCP**. Finally, these 4 reactions are again fully coupled to **GLCS1** and **GLGC** with which they form an alternative to **PPKr**.

By definition of linear 1-module, each module can be replaced by a single reaction. Hence, also chains of parallel and fully-coupled reactions form linear 1-modules. I found that most of the linear 1-modules in genome-scale metabolic networks can indeed be obtained by chaining parallel and fully coupled reactions together. An example is shown in Fig. 6.16. In *E. coli* iAF1260 grown on glucose, for example, we find the linear 1-modules shown in Fig. 6.17 and Fig. 6.18. These linear 1-modules are not formed by chains of parallel and fully coupled reactions.

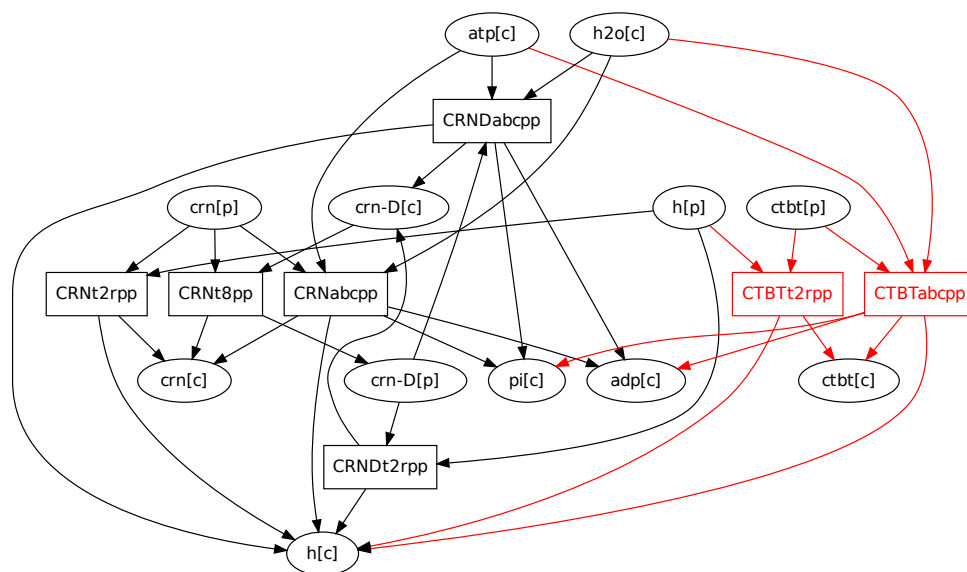


Figure 6.17: This is a 1-module in the full flux space of *E. coli* iAF1260. It can be represented by the reaction $atp[c] + h2o[c] \rightarrow adp[c] + pi[c] + h[p]$. This module contains two fully coupled reactions which are marked in red.

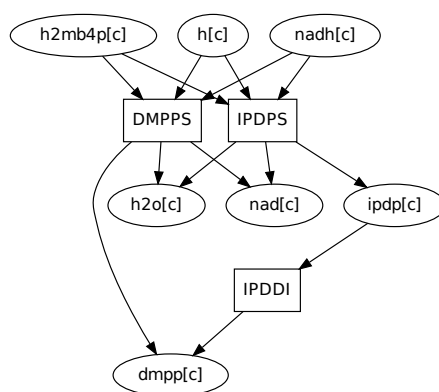


Figure 6.18: This is a 1-module in the full flux space of *E. coli* iAF1260. It turns $h2mb4p[c]$, $h[c]$, and $nadh[c]$ into $dmp[c]$, $h2o[c]$, $ipdp[c]$, and $nad[c]$. This module is fully coupled to biomass production.

I used linear 1-modules to compress existing genome-scale metabolic networks. An overview of the results can be seen in Tab 6.5.

Table 6.5: Network compression of genome-scale networks.

| Network | Number reactions | Number modules | Size biggest module | Size after compression | run time modules | run time compression |
|-------------------------------|------------------|----------------|---------------------|------------------------|------------------|----------------------|
| <i>E. coli</i> iAF1260 | 2382 | 1 | 1531 | 923 | 28sec | 1085sec |
| <i>E. coli</i> iJR904 | 1075 | 2 | 664 | 404 | 4.7sec | 31sec |
| <i>E. coli</i> iJO1366 | 2583 | 2 | 1703 | 1019 | 35sec | 1781sec |
| <i>H. pylori</i> iT341 | 554 | 3 | 431 | 165 | 3.4sec | 4.7sec |
| <i>H. sapiens</i> recon. 1 | 3742 | 6 | 2450 | 1467 | 85sec | 7110sec |
| <i>H. sapiens</i> recon. 2 | 7440 | 13 | 5795 | | | |
| <i>M. barkeri</i> iAF692 | | | | | | |
| <i>M. tuberculosis</i> iNJ661 | 1025 | 3 | 735 | 347 | 4.6sec | 53sec |
| <i>S. aureus</i> iSB619 | 743 | 1 | 450 | 191 | 2.1sec | 12sec |
| <i>S. cerevisiae</i> iND750 | 1266 | 2 | 628 | 353 | 6.3sec | 27sec |

As expected there are not many flux modules in the full flux space of genome-scale metabolic networks. It was always the case that there existed one big module that contained nearly all the unblocked reactions and a few small modules with about 2 – 3 reactions. These small modules usually consist of reactions that form internal cycles that would be blocked if internal cycles would not be allowed to carry flux. Although the computation of modules did not give a big improvement in the reduction, its computation is also not very demanding and it is a prerequisite for the computation of 1-modules.

In nearly all networks the network size could be reduced by about half. Although most of the reduction is due to fully coupled reactions, the detection of parallel reaction plays an important role, too. In networks like *H. sapiens* recon 1 I found more than 50 parallel reactions. It follows that the number of EFMs in the compressed network can be up to a factor $2^{50} \approx 10^{15}$ (quadrillion) smaller. Using the decomposition theorem for linear 1-modules it follows that all the EFMs of the uncompressed network can be obtained from the EFMs of the compressed network.

6.5 Decomposition Theorem for k -Modules

This result is submitted to *Discrete and Computational Geometry*.

Here, I present a method for EFM enumeration resp. vertex enumeration of polyhedra using k -modules. While vertex enumeration has been shown to be **NP**-hard for unbounded polyhedra [76] (there exists no enumeration algorithm with polynomial time in input and output unless $\mathbf{P} = \mathbf{NP}$), the problem is still open for bounded polyhedra.

We first observe that enumeration of EFMs of the flux cone is equivalent to vertex enumeration of bounded polyhedra (Prop. 6.5.1). The enumeration of EFMs that contain

a given reaction however, already corresponds to vertex enumeration of an unbounded polyhedron [2].

6.5.1 Elementary Flux Mode Enumeration and Vertex Enumeration

The following result is formulated in terms of vertex enumeration of polyhedra. However, we can reduce the EFM enumeration problem to vertex enumeration using nearly no additional overhead as follows:

Proposition 6.5.1 *Let $\mathcal{N} = (\mathcal{M}, \mathcal{R}, S)$ be a metabolic network with irreversible reactions Irrev . Let*

$$P := \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, v_{\text{Irrev}} \geq 0\}$$

$$Q := \{(v^+, v^-) \in \mathbb{R}^{\mathcal{R}} \times \mathbb{R}^{\mathcal{R} \setminus \text{Irrev}} : Sv^+ - S_{\mathcal{R} \setminus \text{Irrev}} v^- = 0, v^+ \geq 0, v^- \geq 0, \mathbb{1}v^+ + \mathbb{1}v^- = \mathbb{1}\}$$

and define the function $f : P \setminus \{0\} \rightarrow Q$ by

$$v \mapsto \frac{1}{\|v\|_1} (\max\{v, 0\}, -\min\{v_{\mathcal{R} \setminus \text{Irrev}}, 0\})$$

Then it holds that

1. for every vertex (v^+, v^-) of Q it either holds that there exists a $r \in \mathcal{R}$ with $v_r^+ = v_r^-$ or for all $v \in P \setminus \{0\}$ with $f(v) = (v^+, v^-)$ it holds that $v \in \text{EFM}(P)$.
2. for each elementary mode $e \in \text{EFM}(P)$ it holds that $f(e)$ is a vertex of Q .

PROOF We observe that $\text{supp}(v) = \text{supp}(v^+) \dot{\cup} \text{supp}(v^-)$ for $(v^+, v^-) = f(v)$. We now show the two statements:

1. It is easy to see that for every $r \in \mathcal{R} \setminus \text{Irrev}$ it holds that v^+, v^- with $v_r^+ = \frac{1}{2} = v_r^-$, $v_s^+ = 0$ for all $s \in \mathcal{R} \setminus \{r\}$ and $v_s^- = 0$ for all $s \in \mathcal{R} \setminus (\text{Irrev} \cup \{r\})$ has a maximal amount of inequalities satisfied by equality and hence, (v^+, v^-) is a vertex of Q .

It follows that if (v^+, v^-) is a vertex of Q and not of this type then it holds for each $r \in \mathcal{R} \setminus \text{Irrev}$ that $v_r^+ = 0$ or $v_r^- = 0$. We now define $v \in \mathbb{R}^{\mathcal{R}}$ by $v_{\mathcal{R} \setminus \text{Irrev}} = v_{\mathcal{R} \setminus \text{Irrev}}^+ - v_{\mathcal{R} \setminus \text{Irrev}}^-$ and $v_{\text{Irrev}} = v_{\text{Irrev}}^+$. It follows that $f(v) = (v^+, v^-)$ and for every $w \in P \setminus \{0\}$ with $f(w) = (v^+, v^-)$ there exists an $\alpha > 0$ such that $w = \alpha v$. Hence, it suffices to show that $v \in \text{EFM}(P)$.

Assume v is not an elementary mode. Then, there exists a $x \in P \setminus \{0\}$ with $\text{supp}(x) \subset \text{supp}(v)$. It follows that $\text{supp}(f(x)) \subset \text{supp}((v^+, v^-))$ which is a contradiction to (v^+, v^-) being a vertex of Q .

2. Let $e \in \text{EFM}(P)$ and assume that $f(e)$ is not a vertex of Q . We easily verify that $f(e) \in Q$, hence there must exist a vertex $x = (x^+, x^-) \in Q$ that satisfies more inequalities than $f(e)$ with equality. Hence, $\text{supp}(x) \subset \text{supp}(f(e))$. By 1., it follows that there exists a $x' \in P \setminus \{0\}$ with $x = f(x')$. By the above observation it follows that $\text{supp}(x') \subset \text{supp}(e)$, which is a contradiction to $e \in \text{EFM}(P)$. ■

We observe that Q is bounded and hence, if we can enumerate the vertices of polytopes efficiently, we can also enumerate elementary modes very efficiently. Of course, vertex enumeration questions do not only arise in the context of elementary modes, but also from a lot of different sources. Hence, I chose to formulate this result in the universal language of vertex enumerations, since this way it is more accessible to a wider audience. Therefore, I will use x to denote the vector of variables in the rest of this chapter as it is the more common notation in the area of polytope theory.

Although I only managed to prove a run time bound for polytopes, the algorithm can also be applied to enumerate vertices of polyhedra in general. For these cases the algorithm may be improved by adding additional tests that check whether a face of a module will only result in unbounded faces of the whole polyhedron etc.

6.5.2 Branch-Width

First, we relate the existence of a certain decomposition of a polyhedron $P = \{x \in \mathbb{R}^{\mathcal{R}} : Sx = b, x \geq 0\} \neq \emptyset$ to the branch-width of the linear matroid ([67, 130]) with elements \mathcal{R} , the columns of S .

Definition 6.5.1 (branch-width) *Let M be a matroid on a set \mathcal{R} of elements. Let $\rho(A) := \text{rank}(A) + \text{rank}(\mathcal{R} \setminus A) - \text{rank}(\mathcal{R}) + 1$ be the connectivity function.*

- *A branch decomposition (T, τ) consists of a tree T with nodes of degree 3 and 1 and a bijective map τ that maps the leaves of T onto \mathcal{R} . For short hand notation we write for sets A of leaves: $\tau(A) := \{\tau(a) : a \in A\}$.*
- *The width of an edge e of T is $\rho(\tau(A_e))$, where (A_e, B_e) is the partition of the leaves of T given by $T \setminus e$. Observe that deleting an edge of T splits T into two connected components, one with leaves A_e and the other with leaves B_e . This is also well defined, since $\rho(A) = \rho(\mathcal{R} \setminus A)$.*
- *The width of a branch decomposition is the maximal width of an edge $e \in T$.*
- *The branch-width of M is the minimal width of all possible branch-decompositions.*

□

Observe that $\rho(A) \leq k$ if and only if A is a k -separator of the matroid M (Def. 6.2.1), which is again the case if and only if A is a P - k -module.

We now interpret the branch-decomposition as a hierarchical structure of k -modules. This enables us to apply recursive divide-and-conquer algorithms, like the vertex enumeration algorithm that we are going to present in this section.

In what follows we use notation $A \dot{\cup} B$ to indicate that sets A and B are disjoint and to denote their union. We call a family W of subsets of \mathcal{R} *binary rooted* if it satisfies the following properties:

(P1) For each $A \in W$, $A \neq \mathcal{R}$ there exists exactly one $B \in W$ with $A \dot{\cup} B \in W$.

(P2) For each $C \in W$, $|C| \geq 2$ there exist exactly one pair $A, B \in W$ with $A \dot{\cup} B = C$.

This describes a binary rooted tree with root \mathcal{R} and leaves all the single element sets of \mathcal{R} .

To facilitate the exposition, we assume in this section that for polytope P the set of variables with constant value is empty; i.e, the set V as defined in (6.2) is equal to \mathcal{R} .

Proposition 6.5.2 *Let $P = \{x \in \mathbb{R}^{\mathcal{R}} : Sx = b, x \geq 0\} \neq \emptyset$ and assume $V = \mathcal{R}$. Let M be the linear matroid generated by the columns of S . There exists a binary rooted family Mod of k -modules of P if and only if M has branch-width at most $k + 1$.*

PROOF \Rightarrow : Define the tree $T = (\text{Mod} \setminus \{\mathcal{R}\}, E)$ with vertex set the sets of Mod except for \mathcal{R} and an edge (A, B) between two sets (vertices) A and B if

- there exists $C \in \text{Mod}$ with $A = B \dot{\cup} C$ or $B = A \dot{\cup} C$ or
- $A \dot{\cup} B = \mathcal{R}$.

Clearly, this tree defines a branch decomposition. The leaves of T are sets containing a single element and hence the map τ of the branch decomposition has $\tau(\{i\}) = i$ for all $i \in \mathcal{R}$. We observe that if we delete an edge $e = (A, B)$ from T that either A or B is the union of the leaves of its subtree. Let us assume w.l.o.g. that this is A . Since A is a k -module, it follows by Theorem 6.2.3 and Theorem 6.2.1 that A is a $k + 1$ separator of M . Hence, (T, τ) has width at most $k + 1$ and thus, the branch-width of M is at most $k + 1$.

\Leftarrow : Let (T, τ) be a branch-decomposition of M with width at most $k + 1$. We obtain a rooted binary tree T' from T by choosing an arbitrary edge $e = (a, b)$, removing e and adding a root c with children a, b , and direct all edges away from c . For each node a of T' we define the set $A(a) := \{\tau(i) : i \text{ is leaf under } a\}$. By definition of branch-decomposition, we observe for each node a of T' that $A(a)$ is a $(k+1)$ -separator and hence, a k -module by Theorem 6.2.3 and Theorem 6.2.1. It follows that $\text{Mod} = \{A(a) : a \text{ node of } T'\}$ is a family of k -modules that satisfies properties (P1) and (P2). ■

As we have seen, the last part of the proof of this proposition is constructive, in the sense that it gives us a binary rooted family Mod of k -modules from a branch decomposition of width at most $k + 1$ of the linear matroid defined by the columns of S . We will now show how such a family Mod can be used to develop a recursive algorithm for vertex enumeration.

6.5.3 Decomposition Theorem for Vertex Enumeration

For each module $A \in \text{Mod}$ let $D^A \in \mathbb{R}^{\mathcal{M} \times k}$ denote the variable interface and $d^A \in \mathbb{R}^{\mathcal{M}}$ the constant interface and define

$$P^A := \{x \in \mathbb{R}^A : S_A x = D^A \alpha + d^A, x \geq 0, \exists \alpha \in \mathbb{R}^k\}. \quad (6.8)$$

In what follows we will study faces of P ; i.e., intersections of P with separating hyperplanes. Notice that every face F is completely characterized by the variables that have value 0; i.e., $x_F = 0$ for all $x \in F$. Therefore we take the liberty to use the name ‘faces’ for subsets of variables. Note that not for every subset F of variables there exists a face where only the variables in F have value 0. Other variables may indirectly be also forced to 0. To capture this issue, we speak about feasible and infeasible faces, even though by its proper definition every face is feasible.

Definition 6.5.2 (feasible A-face) For $A \subseteq \mathcal{R}$ a set $F \subseteq A$ is called a feasible A-face if there exists a $x \in P$ with $x_F = 0$ and $x_{A \setminus F} > 0$. \square

Definition 6.5.3 (vertex feasible A-face) For $A \subseteq \mathcal{R}$ a set $F \subseteq A$ is called vertex feasible A-face if there exists a vertex v of P with $v_F = 0$ and $v_{A \setminus F} > 0$. \square

We note, that whereas testing for $A \subseteq \mathcal{R}$ if a subset F defines a feasible A-face can be done easily by linear programming (see Prop. 6.5.7), testing however if $F \subseteq A$ is a vertex feasible A-face is NP-hard [52].

To approximate vertex feasible faces, we introduce the relaxed notion of minimal A-face:

Definition 6.5.4 (minimal A-face) For $A \in \text{Mod}$ a set $F \subseteq A$ is a minimal A-face if there exist no distinct $y, z \in \{x \in P^A : x_F = 0\}$ with $S_A y = S_A z$, i.e., S_A is injective on $\{x \in P^A : x_F = 0\}$. \square

Algorithm 10 enumerates all minimal feasible C -faces for a given k -module $C \in \text{Mod}$, by recursively enumerating all minimal A-faces and all minimal B-faces for the two k -modules $A, B \in \text{Mod}$ that constitute C ; i.e., $C = A \dot{\cup} B$.

That this is correct follows from the following theorem.

Theorem 6.5.1 Algorithm 10 computes all the minimal feasible C -faces for a given $C \in \text{Mod}$. \square

To prove Thm. 6.5.1, we observe the following decomposition property of minimal faces:

Lemma 6.5.1 Let $A, B, C \in \text{Mod}$ with $C = A \dot{\cup} B$. For every minimal feasible C -face F^C there exists a minimal feasible A-face F^A and a minimal feasible B-face F^B with $F^C = F^A \cup F^B$.

Algorithm 10 Algorithm to compute all minimal feasible C -faces for $C \in \text{Mod}$. For $C = \mathcal{R}$, this algorithm will compute all vertices.

```

function  $\mathcal{F} = \text{getMinimalFeasibleFaces}(C)$ 
  if  $|C| = 1$  then
     $\mathcal{F} := \emptyset$ .
    if  $\emptyset$  feasible for  $C$  then
       $\mathcal{F} := \mathcal{F} \cup \{\emptyset\}$ .
    end if
    if  $C$  minimal feasible for  $C$  then
       $\mathcal{F} := \mathcal{F} \cup \{C\}$ .
    end if
  else
    Let  $A, B \in \text{Mod}$  with  $C = A \dot{\cup} B$ .
     $\mathcal{F}^A := \text{getMinimalFeasibleFaces}(A)$ 
     $\mathcal{F}^B := \text{getMinimalFeasibleFaces}(B)$ 
     $\mathcal{F} := \{F^A \cup F^B : F^A \in \mathcal{F}^A, F^B \in \mathcal{F}^B\}$ .
    for  $F \in \mathcal{F}$  do
      if  $F$  not minimal for  $C$  or  $F$  not feasible for  $C$  then
         $\mathcal{F} := \mathcal{F} \setminus \{F\}$ .
      end if
    end for
  end if
end if
    
```

PROOF Let F^C be an arbitrary but fixed minimal C -face. Hence, there exists a $x \in P$ with $x_{F^C} = 0$ and $x_{C \setminus F^C} > 0$.

Define $F^A := F^C \cap A$ and $F^B := F^C \cap B$. Since $C = A \dot{\cup} B$, it follows that $F^C = F^A \cup F^B$. Furthermore, F^A is a feasible A -face and F^B is a feasible B -face, since it follows from $x_{F^C} = 0$ that $x_{F^A} = 0, x_{F^B} = 0$ and from $x_{C \setminus F^C} > 0$ that $x_{A \setminus F^A} > 0$ and $x_{B \setminus F^B} > 0$.

We only have to show that F^A and F^B are also minimal. For proof by contradiction assume that F^A is not minimal (the case F^B non minimal is analogous). It follows that there exist distinct $y, z \in P^A$ with $y_{F^A} = 0, z_{F^A} = 0$ and $S_A y = S_A z$. Define $w \in \mathbb{R}^C$ by $w_A = y - z$ and $w_B = 0$. We observe that $S_C w = S_A w_A = 0$ and $\text{supp}(w) \subseteq \text{supp}(x_C)$. It follows that there exists an $\alpha > 0$ such that $x_C + \alpha w \in P^C$ and $x_C - \alpha w \in P^C$. This is a contradiction to minimality of F^C , since $y \neq z$ and hence, $w \neq 0$. ■

PROOF (THM. 6.5.1) First notice that the only two possible faces of a k -module C with $|C| = 1$ are \emptyset , and C itself.

Any k -module $C \in \text{Mod}$ with $|C| \geq 2$ is constituted by two disjoint k -modules $A, B \in \text{Mod}$: $C = A \dot{\cup} B$. By Lemma. 6.5.1 we know that for any minimal feasible C -face F^C , there exist a minimal feasible A -face F^A and a minimal feasible B -face F^B , such that $F^C = F^A \cup F^B$. Since Algorithm 10 tests every possible pair consisting of a minimal A -

face and a minimal B -face on feasibility and minimality for C , this implies the theorem for any set $C \in \text{Mod}$ with $|C| \geq 2$. ■

That Algorithm 10 eventually outputs all vertices of P is a corollary of the following theorem:

Theorem 6.5.2 *Let $A \in \text{Mod}$ be a 0-module. Then, a feasible A -face is minimal if and only if it is vertex feasible.* □

Corollary 6.5.1 *The minimal feasible \mathcal{R} -faces are the vertices of P and Algorithm 10 applied to $C = \mathcal{R}$ computes all vertices of P .*

PROOF By Theorem 6.5.1 all computed \mathcal{R} -faces are minimal feasible and thus, by Theorem 6.5.2 vertex feasible, since \mathcal{R} is a 0-module.

Since there are no further variables, every vertex feasible \mathcal{R} -face is a vertex. ■

Before we prove Thm. 6.5.2, we observe the following property that allows us to relate minimal feasible faces and vertex feasible faces.

Proposition 6.5.3 *Every vertex feasible A -face F for $A \in \text{Mod}$ is a minimal A -face.*

PROOF By definition of vertex feasible A -face, there exists a vertex v of P with $v_F = 0$ and $v_{A \setminus F} > 0$. For proof by contradiction, we assume now that F is not a minimal A -face. It follows that there exist distinct $y, z \in \{x \in P^A : x_F = 0\}$ with $S_A y = S_A z$. We observe that $w' := y - z$ satisfies $S_A w' = S_A y - S_A z = 0$. Define $w \in \mathbb{R}^{\mathcal{R}}$ by $w_A = w'_A$ and $w_{\mathcal{R} \setminus A} = 0$. We observe that $\text{supp}(w) \subseteq \text{supp}(v)$. Hence, there exists an $\alpha > 0$ such that $v - \alpha w \in P$ and $v + \alpha w \in P$. This is a contradiction to the assumption that v is a vertex. ■

PROOF (THM. 6.5.2) Proposition 6.5.3 shows that every vertex feasible A -face is a minimal A -face.

Let F be minimal feasible A -face. Since A is a 0-module it holds for all $x \in P^A$ that $S_A x_A = d^A$. It follows by definition of minimal A -face that $x \in P^A$ with $x_F = 0$ is unique. Since F is also feasible such x exists and satisfies $x_{A \setminus F} > 0$.

We observe that by construction P is pointed. Since F is a feasible A -face, it follows that $\hat{P} := \{x \in P : x_F = 0\}$ is a non-empty pointed polyhedron. Hence, there exists a vertex y of \hat{P} . Clearly, y is also a vertex of P and satisfies $y_A \in P^A$ and $y_F = 0$. Thus, $y_A = x_A$ and F is a vertex feasible A -face. ■

Now we will show that in case P is a polytope (i.e., P bounded) the existence of a set Mod of k -modules makes Algorithm 10 run in polynomial time, for fixed k . Therefore, we will do a number of observations that allow us to bound the number of minimal feasible faces and finally obtain the runtime bound. The following proposition still also holds for unbounded polyhedra:

Proposition 6.5.4 *For every minimal A -face F for $A \in \text{Mod}$ holds that $\dim f \leq k$ with $f = \{x \in P^A : x_F = 0\}$.*

PROOF Since A is a k -module, it follows that S_A maps every point in f into a k -dimensional space. If $\dim f > k$, it follows that S_A is not injective on f and hence, F would not be minimal. ■

We observe the following corollary, which may give another intuition for the final complexity bound.

Corollary 6.5.2 *Every vertex feasible A -face F for $A \in \text{Mod}$ satisfies $\dim\{x \in P^A : x_F = 0\} \leq k$.*

PROOF Directly from Prop. 6.5.3 and Prop. 6.5.4. ■

Lemma 6.5.2 *Assume P is bounded. Suppose for $A \in \text{Mod}$ that F is a feasible A -face and let $h = \dim f$ with $f = \{x \in P^A : x_F = 0\}$. Then there exist a set of $\ell \leq h + 1$ vertex feasible A -faces F^1, \dots, F^ℓ such that $F = F^1 \cap F^2 \cap \dots \cap F^\ell$.*

PROOF Since F is a feasible A -face, there exists a $y \in P$ with $y_F = 0$ and $y_{A \setminus F} > 0$. Therefore, y lies in the face $\{x \in P : x_F = 0\}$ of P . Since P is bounded, there exist a set of vertices V^f of P such that $y \in \text{conv}(V^f)$ and $w_F = 0$ for all $w \in V^f$.

It follows that $y_A \in \text{conv}(\text{pr}_A V^f)$ and $\text{pr}_A V^f \subseteq P^A$. Since $\dim\{x \in P^A : x_F = 0\} = h$, there exist by Carathéodory's theorem [139] $\ell \leq h + 1$ points $w^1, w^2, \dots, w^\ell \in \text{pr}_A V^f$ with $y_A \in \text{conv}(w^1, \dots, w^\ell)$.

Clearly $F^i = \{j \in \mathcal{R} : w_j^i = 0\}$ is a vertex feasible A -face and $F \subseteq F^i$ for each $i = 1, \dots, \ell$. For every $j \in A \setminus F$ with $j \in F^i$ for all $i = 1, \dots, \ell$ it follows that $y_j = 0$, since $y_A \in \text{conv}(w_A^1, \dots, w_A^\ell)$, and hence, $F \supseteq F^1 \cap F^2 \cap \dots \cap F^\ell$. Thus, $F = F^1 \cap F^2 \cap \dots \cap F^\ell$. ■

Proposition 6.5.5 *If P is bounded then holds for all $A \in \text{Mod}$ that*

$$|\{F \subseteq A : F \text{ minimal feasible } A\text{-face}\}| \leq |\{F \subseteq A : F \text{ vertex feasible } A\text{-face}\}|^{k+1}.$$

PROOF By Prop. 6.5.4 every minimal feasible A -face has dimension at most k . Hence, by Lemma 6.5.2 there exists an injective map that assigns to each minimal feasible A -face a non-empty set of at most $k + 1$ vertex feasible A -faces. Let c_{vert} denote the number of vertex feasible A -faces. There are at most

$$\sum_{i=1}^{k+1} \binom{c_{\text{vert}}}{i}$$

non-empty subsets of at most $k + 1$ elements. For $c_{\text{vert}} = 1$, we have $\sum_{i=1}^{k+1} \binom{c_{\text{vert}}}{i} = 1 = c_{\text{vert}}^{k+1}$ and for $c_{\text{vert}} \geq 2$, we can estimate

$$\sum_{i=1}^{k+1} \binom{c_{\text{vert}}}{i} \leq \sum_{i=1}^{k+1} \frac{c_{\text{vert}}^i}{i!} \leq \sum_{i=1}^{k+1} \frac{2^{k+1-i} c_{\text{vert}}^i}{k+1} \leq (k+1) \frac{c_{\text{vert}}^{k+1}}{k+1} = c_{\text{vert}}^{k+1},$$

since $2^{k+1-i} \geq \frac{k+1}{i!}$ for all $i \leq k \in \mathbb{N}$.

By injectivity it follows that this is also a bound on the number of minimal feasible A -faces. \blacksquare

Proposition 6.5.6 *For $A \in \text{Mod}$ holds that*

$$|\{F \subseteq A : F \text{ vertex feasible } A\text{-face}\}| \leq |\{v \in \mathbb{R}^{\mathcal{R}} : v \text{ is a vertex of } P\}|.$$

PROOF Let F^1, F^2 be distinct vertex feasible A -faces and let v^1, v^2 be representing vertices. It follows that $\text{supp}(v_A^1) = A \setminus F^1 \neq A \setminus F^2 = \text{supp}(v_A^2)$. Hence, $v^1 \neq v^2$ and the result follows. \blacksquare

Theorem 6.5.3 *Assume P is bounded. Let $A, B, C \in \text{Mod}$ with $C = A \dot{\cup} B$. Assume the set of minimal feasible A -faces \mathcal{F}^A and the set of minimal feasible B -faces \mathcal{F}^B are given. Then the set of minimal feasible C -faces \mathcal{F}^C can be computed in time*

$$O(|\mathcal{V}|^{2k+2}t),$$

where \mathcal{V} is the set of vertices of P and t is the time needed to check if a face is feasible and minimal.

PROOF By Lemma 6.5.1 every minimal feasible C -face can be obtained from a combination of one minimal feasible A -face and a minimal feasible B -face. It follows that we have to consider at most $|\mathcal{F}^A| \cdot |\mathcal{F}^B|$ combinations. By Proposition 6.5.6 and Proposition 6.5.5 it follows that

$$|\mathcal{F}^A| \cdot |\mathcal{F}^B| \leq |\mathcal{V}|^{k+1} \cdot |\mathcal{V}|^{k+1} \leq |\mathcal{V}|^{2k+2}.$$

For each candidate we have to check if it is feasible and minimal, which gives the final runtime bound. \blacksquare

We now observe that the time for checking if a face is minimal feasible can be done in polynomial time, which then leads us to the final result on the runtime of Alg. 10.

Proposition 6.5.7 *Given $A \in \text{Mod}$, it can be checked in input polynomial time if a A -face F is minimal feasible.*

PROOF To check feasibility, we just have to solve the following LP:

$$\begin{aligned}
 & \max z \\
 & \text{s.t. } Sx = b \\
 & \quad x_F = 0 \\
 & \quad x_i - z \geq 0 \quad \forall i \in A \setminus F \\
 & \quad x \geq 0
 \end{aligned}$$

If the LP is unbounded or the optimal value is greater than 0 then we have found a solution $x^* \in P$ with $x_F^* = 0$ and $x_{A \setminus F}^* > 0$, which proves feasibility of F . If F is feasible, there exists a solution of the LP with $z > 0$ and hence, the optimal value of the LP has to be positive.

To check minimality, do the following: By minimizing / maximizing each $i \in A$ using linear programming, we compute the following set G :

$$\begin{aligned}
 G & := \{i \in A : u_i > \ell_i\}, \text{ where} \\
 u_i & := \sup\{x_i : x \in P^A, x_F = 0\} & \forall i \in A \\
 \ell_i & := \inf\{x_i : x \in P^A, x_F = 0\} & \forall i \in A
 \end{aligned}$$

Observe that $G \cap F = \emptyset$.

Claim 6.5.1 F is minimal if and only if $S_G : \mathbb{R}^G \rightarrow \mathbb{R}^M$ is injective.

PROOF \Rightarrow : Assume S_G is not injective, i.e., there exists a $x' \in \mathbb{R}^G \setminus \{0\}$ with $S_G x' = 0$. Define $x \in \mathbb{R}^A$ by $x_G = x'$ and $x_{A \setminus G} = 0$. By construction of G and convexity, there exists a $v \in P^A$ with $v_F = 0$ and $\ell_G < v < u_G$. Hence, there exists an $\alpha > 0$ such that $v + \alpha x \geq 0$ and $v - \alpha x \geq 0$. We observe further that $S_A(v \pm \alpha x) = S_A v \pm \alpha S_A x = S_A v$ and hence, $v \pm \alpha x \in P$ with $(v \pm \alpha x)_F = 0$. This is a contradiction to minimality of F .

\Leftarrow : Assume F is not minimal. Hence, there exist $x, y \in P^A$ with $x_F = y_F = 0$ and $S_A x = S_A y$. It follows that $w := x - y$ satisfies $S_A w = 0$ and by definition of G , we have $w_{A \setminus G} = 0$. Hence, $S_G w_G = S_A w = 0$, which is a contradiction to injectivity of S_G . ■

Since we can check injectivity of matrices by computing the nullspace matrix in polynomial time, we can also check minimality of F in polynomial time. ■

Theorem 6.5.4 *If P is a polytope and Mod is a family of k -modules satisfying Properties (P1) and (P2) for constant k , then Algorithm 10 runs in total polynomial time.*

PROOF As mentioned before, Mod can be represented as a binary tree, rooted at \mathcal{R} , with leaves the single element modules. We observe that the time spend for the leaves (modules $A \in \text{Mod}$ with $|A| = 1$) is $O(\mathcal{R}t)$, where t is the time needed to check if the corresponding face is minimal and feasible. Let \mathcal{C} be the set of interior nodes, then the total time needed for determining the minimal feasible faces of all modules corresponding to interior nodes is, by Theorem 6.5.3,

$$O\left(|\mathcal{C}||\mathcal{V}|^{2k+1}t\right),$$

where t is the time needed to check if a given face is minimal and feasible. By Prop. 6.5.7 t grows polynomially in the input size. Since the number of internal nodes $|\mathcal{C}|$ of a binary tree is linear in the number of leaves $|\mathcal{R}|$, the result follows. ■

6.6 Conclusion

In this chapter I presented methods to decompose metabolic networks into smaller, more easily understandable parts. We saw that in the case of the optimal flux space, this works very well: We can compute the flux modules very efficiently and most genomes-scale networks also decompose rather nicely. Most of the flux modules can themselves be visualized using standard graph-drawing applications and we saw how metabolic interactions inside a flux module can be understood using the blocking graph. Also, it is possible to compress the whole metabolic network using flux modules into a network that fits on one page and can again be drawn using standard graph-drawing tools.

However, the practical applicability of flux modules always requires that we work on the optimal yield flux space. If we do not, it is very likely that we do not find any flux modules at all. This is a problem, since yield-optimality is a very artificial criterion that is usually not attained in nature. Hence, we also looked at k -modules, which generalize flux modules. The strength of k -modules, their flexibility, of course also makes it harder to find them and to derive nice decomposition theorems for them. In the case of 1-modules, we can state a decomposition theorem that at least resembles the original decomposition theorem for flux modules. In the more general case for k -modules, we had to follow a new approach by recursively splitting the network into 2 pieces. There, the concept of *branch-width* allowed us to derive a novel parametrized complexity-theoretic result.

Chapter 7

Sublinear Growth & Flux Forcing Reactions

Abstract *Chlamydomonas reinhardtii* is a eukaryotic green algae that is commonly studied in the context of biofuel production. Green algae, like *C. reinhardtii* and also cyanobacteria essentially grow on light and carbon dioxide. Hence, those organisms are highly interesting for the biofuel industry, since the nutrients needed to feed these organisms are basically available for free.

When growing *C. reinhardtii* in a bioreactor, we observe an effect that does not occur for cyanobacteria: Very quickly the total growth rate in the bioreactor is reaching the limit predicted by *flux balance analysis* (FBA). However, after some time the total growth rate decreases, although the only parameter that is changing over time is the density of cells in the bioreactor.

In this chapter, we analyze this effect using metabolic network analysis. We observe that this effect must be caused by reactions through which the flux cannot decrease proportionally as in the rest of the network.

To detect potential sets of reactions that can cause such effects, we will develop a method to solve bilevel optimization problems with mixed integer problems as inner problems to optimality.

This work is the result of a collaboration with Guillaume Cogne, who presented the problem to me in a series of meetings. Unfortunately, we did not yet manage to bring this project close to publication yet.

7.1 Motivation

The group of G. Cogne is analyzing the growth of the green algae *C. reinhardtii* in a torus photo-bioreactor as shown in Fig. 7.1 [25]. It is supplied with a one-sided light source that illuminates the cells and supplies them with the necessary light for growth. Other nutrient concentrations (like CO_2) are kept constant using a chemostat mechanism. Additionally the cells are moved around, so that the system stays well mixed. This has the effect that cells move between the illuminated (front) part of the bio reactor to the dark (back) part in a few seconds. The only parameter that is growing over time is the number of cells in the bioreactor. The changing number of cells of course also has an effect on which parts of the bioreactor stay illuminated by light, i.e., the higher the cell density, the shorter becomes the illuminated zone at the front of the bioreactor.



Figure 7.1: A torus photobioreactor. The figure is taken from http://www.mazenalamir.fr/ANR_CLPP/bioreacteur.html.

In this experimental setup, G. Cogne and coworkers (according to oral communication with G. Cogne) observed that the growth rate is decreasing over time, i.e., with increasing density of the cells (see Fig. 7.2). This is surprising, since the amount of energy (in form of light) supplied to the system stays constant and hence, the total metabolic capacity should also stay constant. They also found that the largest growth rate that they measured also corresponds to the FBA optimum (see Sec. 4.1) that they computed from their metabolic network reconstructions. In contrast to this, if instead the cyanobacterium *A. platensis* is grown in the same experimental setup, this effect is observed and no decrease in growth-rate is observed (see Fig. 7.3).

When looking for explanations for this effect we have to consider two potential causes:

1. The effect is caused because of the time. It is for example known that too much

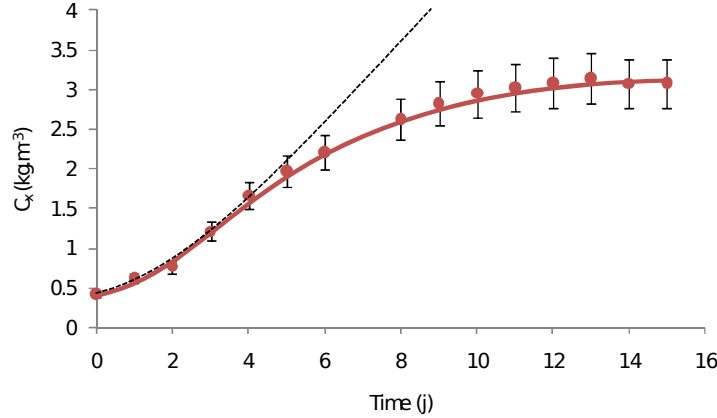


Figure 7.2: Cell density C_x of *C. reinhardtii* 137AH over time. We observe that after some time, the growth rate decreases. Note that the growth rate is the gradient of the curve.

light damages the plant cells and thus, inhibits light uptake and hence, growth. This effect is known as *photo-inhibition*. However, we can rather safely exclude this effect, since the cells are stirred and frequently moved to the dark side of the bioreactor. Indeed, the effect should be strong at the beginning and diminish for later time points, since the dark region increases with time. Hence, we would expect that it is working against the effect that we observe.

2. The effect is caused by the changing cell concentration and hence, by the decreasing amount of light that each cell can take up.

Since we could not find a reasonable explanation of why a change in time might cause the effect, we focus on understanding the effects of a changing cell concentration. Therefore, I built a mathematical model that models the local light intensity for each cell with the following modeling assumptions (see Fig. 7.4):

1. We model the container as the 1-dimensional interval $[0, X]$, the light input is at 0. This means, that we assume that only the distance to the front is relevant.
2. We assume that the cells are uniformly distributed in the bioreactor with density $D \in \mathbb{R}^+$. The density D is a variable.
3. We assume that the light input is fixed to intensity ℓ^{in} .
4. We model the light intensity at position $x \in [0, X]$ by $\ell_D(x)$, where D denotes the cell density introduced above. We assume that ℓ_D satisfies that
 - $\ell_D(x) \geq 0$ for all $x \in \mathbb{R}^+$.

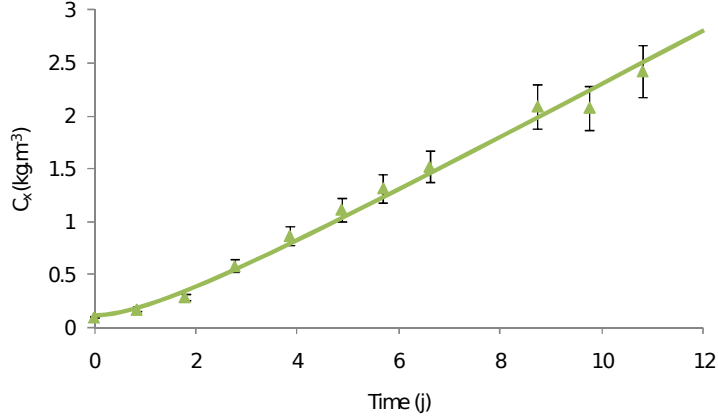


Figure 7.3: Cell density C_x of the cyanobacterium *A. platensis* PCC 8005 over time. We observe that the growth rate does not decrease. Note that the growth rate is the gradient of the curve.

- $\ell_D(x) = \ell_1(Dx)$ for all $D \in \mathbb{R}^+$, since with density D there are D times more cells between 0 and x than with density 1.
 - ℓ_D is continuous.
5. We assume that the growth rate of each cell only depends on the light intensity at the cell, i.e., there exists a continuous function $g : \mathbb{R}^+ \rightarrow \mathbb{R}$ where $g(\ell)$ gives the growth rate for local light intensity $\ell \in \mathbb{R}^+$.
 6. On any interval $[x, x + \Delta]$ are ΔD cells and the light intensity for each cell in this interval ranges between $g(\ell_D(x))$ and $g(\ell_D(x + \Delta))$. Approximating the cells as a continuous medium, we obtain by continuity of g and ℓ_D a total growth rate of

$$G = \int_0^X g(\ell_D(x)) D dx.$$

We observe that the total light energy supplied to the bioreactor by the light function increases with increasing density, since

$$\int_0^X \ell_D(x) D dx = \int_0^X \ell_1(Dx) D dx = \int_0^{DX} \ell_1(x) dx.$$

Furthermore, we can characterize the relationship between growth-rate and cell density as follows:

Theorem 7.1.1 $\frac{\partial G(D)}{\partial D} = Dg(\ell_D(X))$ for all $D \geq 0$.

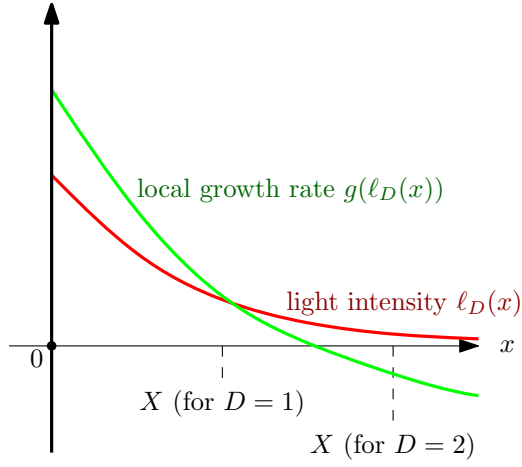


Figure 7.4: Schematic drawing of the bioreactor as assumed in the mathematical model.

PROOF Define

$$G_1(T) := \int_0^T g(\ell_1(x)) dx.$$

We observe that

$$G'_1(T) = g(\ell_1(T))$$

and it holds that

$$\begin{aligned} \frac{\partial G(D)}{\partial D} &= \frac{\partial}{\partial D} \int_0^X g(\ell_D(x)) D dx \\ &= \frac{\partial}{\partial D} \int_0^X g(\ell_1(Dx)) D dx \\ &= \frac{\partial}{\partial D} \int_0^{DX} g(\ell_1(x)) dx \\ &= \frac{\partial}{\partial D} G_1(DX) \\ &= DG'_1(DX) \\ &= Dg(\ell_1(DX)) \\ &= Dg(\ell_D(X)), \end{aligned}$$

which had to be proven. ■

The following corollary follows directly

Corollary 7.1.1 *If $g(\ell) \geq 0$ for all $\ell \geq 0$, then $\frac{\partial G}{\partial D} \geq 0$ for all $D \geq 0$.* □

We conclude that if there exist densities D' for which we observe negative $\frac{\partial G}{\partial D}(D')$, then this can only be due to negative growth rate. If ℓ_D is monotonically decreasing and we observe the effect for high densities, this means that $g(\ell) < 0$ for small enough ℓ .

Therefore, under our model assumptions (which are not very strong), it follows that to observe the effect the cells must exhibit sublinear (actually even negative) growth for low light intensities. Since each cell passes through bright and dark areas, there is no cell that effectively has negative growth (which would be physically impossible). However, we can interpret the result as follows: While the cell is in the bright area it is accumulating resources that are then used up in the dark area. This way, we still end up with a positive growth rate for long-term averages.

We also remark that the sublinear growth is most likely also not characterized by the fact that the cell cannot grow more efficiently, but that the regulatory state, which also has to be efficient for the illuminated zone, does not permit a higher growth rate. If the green algae would be subject to constant low light conditions, it would very likely adapt using its regulatory mechanisms and grow better than under changing light conditions (with the same average light uptake). In particular, it would be no contradiction if *C. reinhardtii* can survive for long times in absolute darkness.

We also observe that we only have to measure the light intensity at the back of the container $\ell_D(X)$, the density D , and the change in growth-rate $\frac{\partial G(D)}{\partial D}$ to obtain the function $g(\ell)$, since by Thm. 7.1.1 we have

$$g(\ell_D(X)) = \frac{1}{D} \frac{\partial G(D)}{\partial D}$$

In particular, the assumption that for high light levels the cells grow at the FBA-optimal level can be verified.

7.2 Flux-Forcing Reactions can Explain the Effect

In the previous section we concluded the following effect for *C. reinhardtii*: If sufficient amounts of light (inflow through reaction $h\nu$), say $y_2 = 20$, is supplied, then the measured growth rate matches with the results from FBA. However, if not enough light is supplied, say $y_1 = 10$, then only suboptimal growth (compared to FBA) is observed.

In the following the precise values of y_2, y_1 will not be important. We will only require $0 < y_1 < y_2$.

We now want to understand the effect using the methods from FBA. Such an approach may, at the beginning, sound counter-intuitive, since the cells are moving rather quickly from the illuminated zone to the dark zone and back and hence, we can exclude a gene regulatory control of the metabolism. Such a control however is assumed for FBA optimality.

However, since oscillatory behavior from illuminated areas to dark areas is also common

in nature (turbulences in the water), we can expect that the organism has evolved to deal with such an environment. Therefore we would expect that the metabolism is calibrated in such a way that it can quickly react to rapid changes of light intensities. In the case of cyanobacteria this seems to be indeed the case. For *C. reinhardtii* this adaption does not seem to work so smoothly. This could be due to the fact that *C. reinhardtii* is a higher life form (eukaryote) than cyanobacteria (prokaryotes) and hence employs more sophisticated (and more efficient) mechanisms that on the other hand are also not so flexible. By applying optimality-based approaches, we may be able to identify these mechanisms (or rather the reactions participating therein).

Hence, we want to characterize those flux spaces that allow efficient adaptation to lower light intensities y_1 from an optimal light intensity y_2 .

For the following, let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be the metabolic network describing *C. reinhardtii*.

Let $P \subseteq \mathbb{R}^{\mathcal{R}}$ be an arbitrary but fixed feasible flux space. Let $s, t \in \mathcal{R}$ be fixed reactions, where s is the source reaction $h\nu$ (light) and t is the target reaction (biomass). We define for a pseudo-reaction $r \in \overline{\mathcal{R}}$ (Def. 2.1.2) and $y \in \mathbb{R}$ the following short-hand notation for the optimal values of the flux optimization problem through r :

$$\begin{aligned} \lambda^r(y) &:= \max v_r : v \in P, v_s \leq y && \text{(full flux space)} \\ \lambda_t^r(y) &:= \max v_r : v \in P, v_s \leq y, v_t = \lambda^t(y) && \text{(optimal yield flux space)} \end{aligned}$$

Throughout this chapter, we will assume that the above optimization problems are well defined:

Assumption 7.2.1 *The optimization problems*

$$\begin{aligned} \max\{v_t : v \in P, v_s \leq y_1\} \\ \max\{v_t : v \in P, v_s \leq y_2\} \end{aligned}$$

are feasible and bounded. □

We observe that we can often simplify this assumption as follows:

Observation 7.2.1 *If P is polyhedral and $\max\{v_t : v \in P, v_s \leq y_1\}$ is feasible and bounded, then $\max\{v_t : v \in P, v_s \leq y_2\}$ is feasible and bounded.*

PROOF Let $P = \{v : Av \leq b\}$. Feasibility is clear, we only have to show boundedness. Assume it is unbounded. It follows that there exists a ray, i.e., a vector w satisfying

$$Aw \leq 0, w_s \leq 0, w_t > 0.$$

Let $v \in P$ with $v_s \leq y_1$. It follows that $v + \alpha w \in P$ for all $\alpha > 0$. This is a contradiction to the boundedness of $\max\{v_t : v \in P, v_s \leq y_1\}$. ■

It follows that under Assumption 7.2.1 the optimization problems for $\lambda^r(y_1), \lambda^r(y_2), \lambda_t^r(y_1), \lambda_t^r(y_2)$ are always well defined and feasible for $r \in \overline{\mathcal{R}}$.

Theorem 7.2.1 *Let $P \subseteq \mathbb{R}^{\mathcal{R}}$ be a flux space with the property that $v \in P$ and $0 \leq \alpha \leq 1$ imply $\alpha v \in P$. Then the described effect is not observable, i.e.,*

$$\frac{\lambda^t(y_2)}{y_2} \leq \frac{\lambda^t(y_1)}{y_1}.$$

PROOF Let v be an optimal solution vector for the optimization problem of $\lambda^t(y_2)$. Chose $\alpha = \frac{y_1}{y_2}$. It follows that $\alpha v \in P$ with $\alpha v_s \leq \alpha y_2 = y_1$. Hence, αv is a feasible solution for the optimization problem of $\lambda^t(y_1)$. It follows that

$$\lambda^t(y_1) \geq \alpha v_t = \alpha \lambda^t(y_2) = \frac{y_1}{y_2} \lambda^t(y_2).$$

Division by y_1 on both sides of the inequality completes the proof. □

This result tells us that if the flux space is a cone (i.e., we can uniformly scale down the flux through all reactions), the cell can always adapt to lower light concentrations linearly. Since we do not observe this behavior for *C. reinhardtii*, we conclude that there must be an obstruction. This obstruction could simply be a reaction r that due to its kinetic behavior has a flux rate more or less independent from the substrate concentrations. Because the other reactions would attain lower flux rates due to the lower supply of substrates, this reaction r would divert the flux away from the other reactions. For example, the flux could be diverted into a futile cycle or into the production of by-products and not be used for biomass production anymore. However, not every reaction with such a diverting behavior can cause a reduction of biomass production. For example if the reaction r is coupled to the biomass reaction, it at some point will anyway starve out and reduce its flux rate. Hence, we are interested in finding only those reactions that with a diverting behavior can produce the observed effect (decrease of biomass production).

A rough approximation of this behavior can be modeled using fixed positive lower bounds or negative upper bounds, i.e., by adding constraints that force flux through the reaction. For the beginning we will work with a fixed positive lower bound for flux through one pseudo-reaction $r \in \overline{\mathcal{R}}$ (Def. 2.1.2). This lower bound should not restrict optimal flux through the target reaction t at y_2 , but restrict flux through t at y_1 . In the following we will only consider positive lower bounds, because the case for negative upper bounds is analogous (we only have to reverse the reaction).

The maximal effect that can be achieved using flux forcing for reaction r is the solution of the following bilevel optimization problem:

$$\begin{aligned} \min_k \max \{ & v_t : v \in P, v_s \leq y_1, v_r \geq k \} & (7.1) \\ \text{s.t. } \max \{ & v_t : v \in P, v_s \leq y_2, v_r \geq k \} = \lambda^t(y_2) \\ & \{v_t : v \in P, v_s \leq y_1, v_r \geq k\} \neq \emptyset \end{aligned}$$

We require that the inner problem $\max\{v_t : v \in P, v_s \leq y_2, v_r \geq k\}$ attains the FBA optimum, since this is a property that we observe in the experiments. Also, $\max\{v_t : v \in P, v_s \leq y_1, v_r \geq k\}$ must be feasible, because otherwise the infeasibility will starve out the flux forcing reaction r and hence r would be forced to a lower flux. This would then again reduce (or even cancel the effect).

Definition 7.2.1 (Diverting reaction) A pseudo-reaction $r \in \overline{\mathcal{R}}$ is called diverting, if the optimal value of (7.1) is smaller than $\max\{v_t : v \in P, v_s \leq y_1\}$. \square

However, we do not have to apply complicated algorithms for bilevel optimization, since:

Theorem 7.2.2 The optimization problem (7.1) is minimized by $k = \min\{\lambda_t^r(y_2), \lambda^r(y_1)\}$.

PROOF Since $\lambda^t(y_2) = \max\{v_t : v \in P, v_s \leq y_2\}$, we have

$$\begin{aligned} \max\{v_t : v \in P, v_s \leq y_2, v_r \geq k\} &= \lambda^t(y_2) \\ \Leftrightarrow \exists v \in P : v_s \leq y_2, v_r \geq k, v_t &= \lambda^t(y_2) \\ \Leftrightarrow \max\{v_r : v \in P, v_s \leq y_2, v_t &= \lambda^t(y_2)\} \geq k \\ \Leftrightarrow \lambda_t^r(y_2) &\geq k \quad (\text{by def. of } \lambda_t^r) \end{aligned}$$

We also have

$$\begin{aligned} \exists v \in P : v_s \leq y_1, v_r \geq k &\Leftrightarrow \max\{v_r : v \in P, v_s \leq y_1\} \geq k \\ &\Leftrightarrow \lambda^r(y_1) \geq k. \end{aligned}$$

Thus, the optimization problem (7.1) is feasible if and only if $k \leq \lambda_t^r(y_2)$ and $k \leq \lambda^r(y_1)$.

Looking at the objective function, we see that the feasible domain of

$$\max\{v_t : v \in P, v_s \leq y_1, v_r \geq k\} \tag{7.2}$$

is monotonically shrinking (w.r.t. inclusion) for growing k . Thus, the objective value of (7.2) is shrinking monotonically for growing k . Hence, the optimum of (7.1) is attained for the largest feasible k .

We conclude that (7.1) is minimized by $k = \min\{\lambda_t^r(y_2), \lambda^r(y_1)\}$. \blacksquare

We can now reformulate the condition for a diverting reaction as follows:

Corollary 7.2.1 A pseudo-reaction $r \in \overline{\mathcal{R}}$ is diverting if and only if $\lambda_t^r(y_1)$ is bounded and

$$\lambda_t^r(y_1) < \min\{\lambda_t^r(y_2), \lambda^r(y_1)\}.$$

PROOF If $\lambda_t^r(y_1)$ is unbounded, it follows that no constraint of the form $v_r \geq k$ is restricting and hence, r cannot be diverting.

Assume that $\lambda_t^r(y_1)$ is bounded and $k := \min\{\lambda_t^r(y_2), \lambda^r(y_1)\}$ is unbounded. It follows that for finite $k' > \lambda_t^r(y_1)$ it holds that

$$\begin{aligned} \max\{v_t : v \in P, v_s \leq y_2, v_r \geq k'\} &= \lambda^t(y_2) && \text{(since } \lambda_t^r(y_2) \text{ unbounded)} \\ \{v : v \in P, v_s \leq y_1, v_r \geq k'\} &\neq \emptyset && \text{(since } \lambda^r(y_1) \text{ unbounded)} \\ \max\{v_t : v \in P, v_s \leq y_1, v_r \geq k'\} &< \lambda^t(y_1) && \text{(since } k' > \lambda_t^r(y_1)) \end{aligned}$$

and hence, r is flux forcing.

For the remaining case where k is finite, we observe that by Def. 7.2.1 and Thm. 7.2.2 a pseudo-reaction $r \in \overline{\mathcal{R}}$ is flux forcing if and only if

$$\begin{aligned} \lambda^t(y_1) &= \max\{v_t : v \in P, v_s \leq y_1\} > \max\{v_t : v \in P, v_s \leq y_1, v_r \geq k\} \\ &\Leftrightarrow \{v : v \in P, v_s \leq y_1, v_t = \lambda^t(y_1), v_r \geq k\} = \emptyset \\ &\Leftrightarrow \lambda_t^r(y_1) = \max\{v_r : v \in P, v_s \leq y_1, v_t = \lambda^t(y_1)\} < k = \min\{\lambda_t^r(y_2), \lambda^r(y_1)\}. \end{aligned}$$

This completes the proof. ■

7.3 Sets of Diverting Reactions

In nature, however, it is not guaranteed (and indeed unlikely) that the observed effect is caused by a single reaction. Hence we will also want to investigate sets of reactions that may cause the effect. There are practically two different kinds of flux forcing for a set of reactions.

7.3.1 Diverting Set of Type 1

Let $A \subseteq \overline{\mathcal{R}}$ be a set of pseudo-reactions. Then we can add the flux forcing constraint (of type 1) $kv_A \geq 1$, for a $k \in \mathbb{R}_+^A$.

Definition 7.3.1 (Diverting set of type 1) *We call $A \subseteq \overline{\mathcal{R}}$ a diverting set of type 1, if there exists a $k \in \mathbb{R}_+^A$ such that*

$$\begin{aligned} \max\{v_t : v \in P, v_s \leq y_1, kv_A \geq 1\} &< \max\{v_t : v \in P, v_s \leq y_1\} = \lambda^t(y_1) \\ \max\{v_t : v \in P, v_s \leq y_2, kv_A \geq 1\} &= \lambda^t(y_2). \end{aligned}$$

By Cor. 7.2.1 we can reformulate this condition by defining

$$\begin{aligned} \lambda^{kA}(y) &:= \max\{kv_A : v \in P, v_s \leq y\} \\ \lambda_t^{kA}(y) &:= \max\{kv_A : v \in P, v_s \leq y, v_t = \lambda^t(y)\}. \end{aligned}$$

Proposition 7.3.1 $A \subseteq \overline{\mathcal{R}}$ is a diverting set of type 1 if and only if there exists a $k \in \mathbb{R}_+^A$ with $\lambda_t^{kA}(y_1)$ bounded and

$$\lambda_t^{kA}(y_1) < \min\{\lambda^{kA}(y_1), \lambda_t^{kA}(y_2)\}$$

PROOF Define a flux space $Q \subseteq \mathbb{R}^{\mathcal{R} \cup x}$ with an additional reaction x by

$$Q := \{v \in \mathbb{R}^{\mathcal{R} \cup x} : v_{\mathcal{R}} \in P, v_x = kv_A\}.$$

It follows that $\lambda^{kA}(y) = \lambda^x(y)$, $\lambda_t^{kA}(y) = \lambda_t^x(y)$ when applied on Q instead of P and it holds for all $\ell \geq 0$ that

$$\max\{v_t : v \in Q, v_s \leq y_1, kv_A \geq \ell\} = \max\{v_t : v \in Q, v_s \leq y_1, v_x \geq \ell\}.$$

The result follows by Cor. 7.2.1 applied to the new reaction x in the flux space Q . ■

We observe on the example of Fig. 7.5 that the notion of diverting set of type 1 is necessary, since it is not always possible to reduce such a diverting set to a smaller diverting set. There, the set $A = \{r_1, r_2\}$ is a diverting set with the constraint $v_{r_1} + v_{r_2} \geq 2$. For $s \geq 2$, we observe that we can produce t with the FBA optimal rate of 2. For $s = 1$ on the other hand, the only feasible solution is attained with $v_{r_1} = v_{r_2} = 1$. It follows that no flux through t is possible.

However, if we only choose $\{r_1\}$ as the diverting set with $v_{r_1} \geq k$, then the optimal attainable flux rate for t is $v_t = v_s - k$. This is not the FBA optimum. If we chose $\{r_2\}$ as the diverting set with $v_{r_2} \geq k$, we observe that for $v_s \geq k$, we can have FBA-optimal v_t , but as soon as $v_s < k$, the system becomes infeasible.

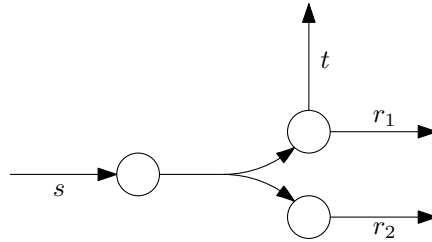


Figure 7.5: A minimal diverting set $A = \{r_1, r_2\}$ of type 1. Note that A is not a diverting set of type 2.

7.3.2 Diverting Set of Type 2

Let $A \subseteq \mathcal{R}$ be a set of reactions. Instead of summing up the fluxes through the reactions, we can also consider separate flux forcing bounds for each of the reactions, i.e., we add the flux forcing constraint $v_A \geq k$, for a $k \in \mathbb{R}_+^A$.

Definition 7.3.2 (Diverting set of type 2) We call $A \subseteq \overline{\mathcal{R}}$ a diverting set of type 2, if there exists a $k \in \mathbb{R}_+^A$ such that

$$\begin{aligned} \max\{v_t : v \in P, v_s \leq y_1, v_A \geq k\} &< \max\{v_t : v \in P, v_s \leq y_1\} = \lambda^t(y_1) \\ \max\{v_t : v \in P, v_s \leq y_2, v_A \geq k\} &= \lambda^t(y_2). \end{aligned}$$

Again, we can reformulate this definition using Cor. 7.2.1. Therefore we define

$$\begin{aligned} \lambda^{A \geq k}(y) &:= \max\{w : v \in P, v_s \leq y, v_A \geq kw\} \\ \lambda_t^{A \geq k}(y) &:= \max\{w : v \in P, v_s \leq y, v_t = \lambda^t(y), v_A \geq kw\}. \end{aligned}$$

Proposition 7.3.2 $A \subseteq \overline{\mathcal{R}}$ is a diverting set of type 2 if and only if there exists a $k \in \mathbb{R}_+^A$ with $\lambda_t^{A \geq k}(y_1)$ bounded and

$$\lambda_t^{A \geq k}(y_1) < \min\{\lambda^{A \geq k}(y_1), \lambda_t^{A \geq k}(y_2)\}.$$

PROOF Define a flux space $Q \subseteq \mathbb{R}^{\mathcal{R} \cup x}$ with an additional reaction x by

$$Q := \{v \in \mathbb{R}^{\mathcal{R} \cup x} : v_{\mathcal{R}} \in P, v_A \geq kv_x\}.$$

It follows that $\lambda^{A \geq k}(y) = \lambda^x(y)$, $\lambda_t^{A \geq k}(y) = \lambda_t^x(y)$ when applied on Q instead of P and for all $w \geq 0$

$$\max\{v_t : v \in Q, v_s \leq y_1, v_A \geq kw\} = \max\{v_t : v \in Q, v_s \leq y_1, v_x \geq w\}.$$

The result follows by Cor. 7.2.1 applied to the new reaction x in the flux space Q . ■

Similar to the case of flux forcing sets of type 1, there also exist flux forcing sets out of several reactions that cannot be reduced. An example can be seen in Fig. 7.6. Due to the metabolites X_1, Y_1 and X_2, Y_2 it follows that $v_{r_1} \leq v_s$ and $v_{r_2} \leq v_s$ for all steady-state flux distributions. It follows that neither r_1 or r_2 alone can induce the flux forcing effect. Together however with $v_{r_1}, v_{r_2} \geq k$, they can produce a demand of $2k$ that for $v_s = k$ leads to only one feasible solution with $v_t = 0$. A similar example can be observed in Fig. 7.7

We observe that not every diverting set of type 1 is also a diverting set of type 2, like in the example shown in Fig. 7.5. However, for polyhedral flux spaces P , every diverting set of type 2 is also a diverting set of type 1. To prove this result, we first show 2 auxiliary results:

Lemma 7.3.1 Let $P \subseteq \mathbb{R}^{\mathcal{R}}$, $A \subseteq \mathcal{R}$ and $k \in \mathbb{R}_+^A$. Then it holds for all row vectors $k' \in \mathbb{R}_+^A$ with $k'k = 1$ that

$$\sup\{w : v \in P, kw - v_A \leq 0\} \leq \sup\{k'v_A : v \in P\}.$$

PROOF Let (\bar{v}, \bar{w}) be a feasible solution of $\max\{w : v \in P, kw - v_A \leq 0\}$. Clearly, \bar{v} is a feasible solution of $\max\{k'v_A : v \in P\}$. We observe that $k\bar{w} \leq \bar{v}_A$ and hence, $k'k\bar{w} \leq k'\bar{v}_A$, which implies $\bar{w} \leq k'\bar{v}_A$. The result follows. ■

Lemma 7.3.2 *Let $P \subseteq \mathbb{R}^{\mathcal{R}}$ be a non-empty polyhedral space, $A \subseteq \mathcal{R}$ and $k \in \mathbb{R}_+^A$. If $\max\{w : v \in P, kw - v_A \leq 0\}$ is bounded then there exists a row vector $k' \in \mathbb{R}_+^A$ with $k'k = 1$ such that*

$$\max\{w : v \in P, kw - v_A \leq 0\} = \max\{k'v_A : v \in P\}$$

PROOF Since P is polyhedral, there exists a matrix B and a vector b such that $P = \{v : Bv \leq b\}$.

Since $\max\{w : v \in P, kw - v_A \leq 0\}$ is feasible and bounded by assumption, it holds by LP duality that the dual of

$$\text{opt} := \max\{w : Bv \leq b, kw - v_A \leq 0\}$$

satisfies

$$\begin{aligned} \text{opt} &= \min \alpha b \\ \text{s.t. } \alpha B_A - \beta &= 0 \\ \alpha B_{\mathcal{R} \setminus A} &= 0 \\ \beta k &= 1 \\ \alpha, \beta &\geq 0 \end{aligned}$$

For an optimizer $\bar{\alpha}, \bar{\beta}$ it follows that

$$\begin{aligned} \text{opt} &= \min \alpha b \\ \text{s.t. } \alpha B_A &= \bar{\beta} \\ \alpha B_{\mathcal{R} \setminus A} &= 0 \\ \alpha &\geq 0 \end{aligned}$$

and again, by LP-duality

$$\text{opt} = \max\{\bar{\beta}v_A : Bv \leq b\}$$

Hence, the result follows by choosing $k' = \bar{\beta}$, since $\bar{\beta}$ satisfies $\bar{\beta} \geq 0$ and $\bar{\beta}k = 1$. ■

Theorem 7.3.1 *Let $P = \{v : Bv \leq b\}$ be a polyhedral flux space and let $A \subseteq \mathcal{R}$ be a diverting set of type 2. Then A is also a diverting set of type 1.*

PROOF Since A is a diverting set of type 2, it follows (Prop. 7.3.2) that there exists a $k \in \mathbb{R}_+^A$ with

$$\lambda_t^{A \geq k}(y_1) < \min\{\lambda^{A \geq k}(y_1), \lambda_t^{A \geq k}(y_2)\}$$

and $\lambda_t^{A \geq k}(y_1)$ is bounded.

By Lemma 7.3.2 it follows that there exists a row vector $k' \in \mathbb{R}_+^A$ with $k'k = 1$ and

$$\lambda_t^{k'A}(y_1) = \lambda_t^{A \geq k}(y_1).$$

By Lemma 7.3.1 it follows that

$$\begin{aligned} \lambda^{A \geq k}(y_1) &\leq \lambda^{k'A}(y_1) \\ \lambda_t^{A \geq k}(y_2) &\leq \lambda_t^{k'A}(y_2). \end{aligned}$$

Hence, we can conclude that

$$\lambda_t^{k'A}(y_1) < \min\{\lambda^{k'A}(y_1), \lambda_t^{k'A}(y_2)\}.$$

By Prop. 7.3.1 it follows that A is a diverting set of type 1. ■

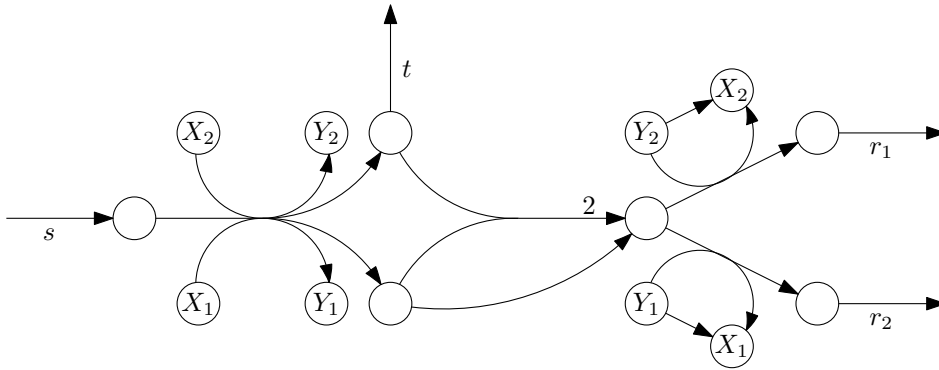


Figure 7.6: The diverting set $A = \{r_1, r_2\}$ of type 2 with $k = (1, 1)$ for $y_1 = 1, y_2 = 2$ cannot be decomposed into smaller diverting sets. It is easy to see that $v_{r_1} \leq v_s$ and $v_{r_2} \leq v_s$, but $v_{r_1} = v_{r_2} = v_s$ is possible.

7.3.3 Finding Flux Forcing Coefficients for Diverting Sets of Type 2

In the previous section we saw how we can easily check if a set of reactions A with known coefficients k is diverting. However, the coefficients k are usually unknown. Hence, we now want to compute these coefficients if only the set A of reactions is given.

Although, we have seen that diverting sets of type 1 are more general for polyhedral flux spaces, we will focus here on diverting sets of type 2. This seems to be the mathematically

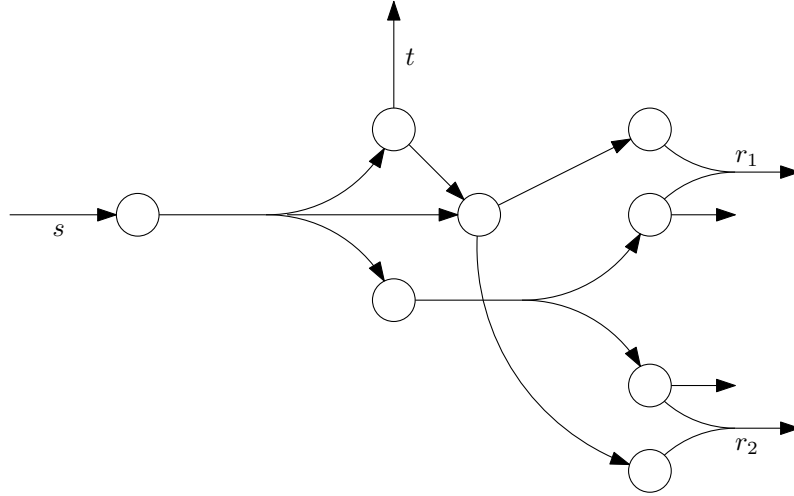


Figure 7.7: Another example for a minimal diverting set $A = \{r_1, r_2\}$ of type 2.

easier case, since otherwise kv_A becomes a quadratic term for variable k . To find the k for which the diversion effect is maximal, we recall that we need to solve

$$\begin{aligned} \min_k \max\{v_t : v \in P, v_s \leq y_1, v_A \geq k\} \\ \text{s.t. } \max\{v_t : v \in P, v_s \leq y_2, v_A \geq k\} = \lambda^t(y_2) \\ \{v : v \in P, v_s \leq y_1, v_A \geq k\} \neq \emptyset. \end{aligned}$$

This can also be rewritten as the following bilevel (min-max) programming problem:

$$\begin{aligned} \min_{k,w,v'} \max\{v_t : v \in P, v_s \leq y_1, v_A \geq k\} \\ \text{s.t. } w \in P, w_s \leq y_2, w_A \geq k, w_t = \lambda^t(y_2) \\ v' \in P, v'_s \leq y_1, v'_A \geq k \end{aligned} \quad (7.3)$$

We call $\max\{v_t : v \in P, v_s \leq y_1, v_A \geq k\}$ the *inner* optimization problem in contrast to the *outer* optimization problem, which controls k, w, v' and contains the inner problem as a constraint / objective.

If P is convex we can use the *Karush-Kuhn-Tucker* (KKT) conditions to reformulate this problem as a *mathematical program with equilibrium constraints* (MPEC) [93]. If P is a polytope, this MPEC will have linear constraints and complementarity constraints. Hence, it can be solved by reformulation as a MILP [26].

If P is not convex, solving the bilevel problem becomes difficult. In particular, if we add thermodynamic constraints we will have a non-convex P .

It is important to consider also thermodynamic (loop-law) constraints, because without these constraints we will never be able to have diverting reactions contained in internal cycles. Hence, we will now look at thermodynamically constrained P .

7.3.4 Thermodynamically Constrained P

In Chapter 4 we discussed several ways on how to formulate thermodynamic constraints. Although we saw that the MILP formulation is not the best formulation for the problem, it still uses the most standardized theoretical framework. This is an important feature, since the chances are high that bilevel optimization for MILPs has already been studied.

Indeed, there exists a lot of (theoretical and practical) work in the area of bilevel optimization. However, most of the work is focusing on problems where either all variables are continuous (and KKT conditions can be applied to obtain criteria for local optimality) or where all variables are discrete and a lot of techniques from game theory are available.

For the case of MILP, I found a work by Gümüş and Floudas [61] and a work by Faísca et al. [41], which study how general bilevel problems with MILP in the inner problem can be solved to optimality. Other works in the literature either only allow integer variables in the outer problem [165] or are only properly developed for inner problems, where the outer variables appear only in the objective of the inner problem [34] (Remark: For diverting sets of type 1 this would be sufficient).

Gümüş and Floudas [61] present an algorithm that does not only work for MILPs but also allows certain types of nonlinear constraints. They exploit the fact that the inner MILP can be described by its integer hull. This way, they obtain a convex inner problem, which they then can reformulate using KKT conditions into a nonlinear one-level optimization problem. This nonlinear optimization problem they then solve with an MINLP solver to global optimality.

Faísca et al. [41] approach the problem from a different perspective. They apply a method by Dua and Pistikopoulos [37] for *multi-parametric MILP* (mpMILP) to solve the inner problem for all possible parameters from the outer problem. The mpMILP solution will be a piece-wise affine linear function. For each linear piece (also called *critical region*), they then evaluate the outer problem and take the minimum.

In our case, the method proposed in [41] is the more promising approach. In our application the inner problem only depends on k and hence, is of low dimension. Since the method is formulated in a very general sense, we can even apply it to thermodynamically constrained fluxes without using the MILP formulation.

In the next section we will discuss how we can solve the parametrized problem for our special case of thermodynamically constrained FBA.

7.4 Multi-Parametric Thermodynamically Constrained Flux Balance Analysis (mpTFBA)

Although the area of *multi-parametric linear programming* (mpLP) is quite well researched, there are only few works on mpMILP. Of these works the method by Dua and

Pistikopoulos [37] looks the most promising.

7.4.1 The Algorithm by Dua and Pistikopoulos

Before we look at our special case of thermodynamically constrained FBA, let us recapitulate the idea of the algorithm by Dua and Pistikopoulos [37]. Let us assume that we want to solve the following parametrized optimization on parameters $k \in Q \subseteq \mathbb{R}^\ell$, where Q is a polyhedral parameter space, g, h are row-vectors, A, B, C are matrices and d is a column vector with corresponding dimensions, where m, n are integers:

$$\hat{v}(k) := \max\{gx + hy : Ax + By \leq Ck + d, x \in \mathbb{R}^m, y \in \{0, 1\}^n\}$$

The central idea is to reduce the problem to *multi-parametric linear programming* (mpLP) by fixing integer variables. However, we use MILP to find good assignments for the integer values, to keep the number of mpLP solves minimal. For mpLP several efficient algorithms, like the simplex-based criss-cross algorithm, which walks from facet to facet of the polyhedron, exist. By fixing the integer variables y to \tilde{y}^1 , we obtain a polyhedral subset of the feasible domain. On this subset, we can compute a parametrized solution using standard mpLP algorithms.

$$\hat{v}^1(k) := h\tilde{y}^1 + \max\{gx : Ax \leq Ck + d - B\tilde{y}^1, x \in \mathbb{R}^m\} \forall k \in Q$$

Since this solution is surely feasible, it is a lower bound on the solution set ($\hat{v}^1(k) \leq \hat{v}(k)$). We observe that the critical regions $CR^i \subseteq \mathbb{R}^\ell$ (linear piece of the parametrized solution) of this mpLP solution are polyhedral and partition the whole parameter space Q . For each critical region CR^i , they check if there exists another solution (with a different assignment of the integer variables) that is at least as good by solving a MILP:

$$\begin{aligned} (\bar{x}, \bar{y}, \bar{k}) &:= \arg \max gx + hy \\ Ax + By - Ck &\leq d \\ gx + hy - \hat{v}^1(k) &\geq 0 \\ y^i(1 - y) + (1 - y^i)y &\geq 1 \quad \forall \text{ previously computed sol. } y^i \text{ (no-good cut)} \\ x &\in \mathbb{R}^m \\ y &\in \{0, 1\}^n \end{aligned}$$

If they find a solution, they check for all critical regions, whether it can be improved with the assignment \bar{y} . If the assignment yields larger values ($g\bar{x} + h\bar{y} > \hat{v}^1(\bar{k})$), they use the improved assignment of the integer variables $\tilde{y}^2 = \bar{y}$ and repeat the same procedure on the critical region:

$$\hat{v}^2(k) := h\tilde{y}^2 + \max\{gx : Ax \leq Ck + d - B\tilde{y}^2, x \in \mathbb{R}^m\} \forall k \in CR^i$$

When no better solution can be found, the critical region is solved to optimality and the algorithm continues with the other critical regions.

7.4.2 Adaptation to Thermodynamically Constrained FBA

In the case of thermodynamically constrained FBA (TFBA), we observed (Thm. 4.5.1) that we do not have to fix all the integer variables (reaction directions) to guarantee that the LP solution value equals the MILP solution value. We will exploit this property also for multi-parametric thermodynamically constrained FBA (mpTFBA).

Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network. Let $G \subseteq \mathcal{R}$ be a set of reactions, for which additional linear constraints are defined by an affine linear function $g : \mathbb{R}^G \rightarrow \mathbb{R}^n$, $n \in \mathbb{N}$. Let $\ell \leq 0$ be flux lower bounds and $u \geq 0$ flux upper bounds.

We consider the steady-state flux space P and the flux space T with additional thermodynamic constraints.

$$P = \left\{ v \in \mathbb{R}^{\mathcal{R}} \left| \begin{array}{l} Sv = 0 \\ \ell \leq v \leq u \\ g(v_G) \leq 0 \end{array} \right. \right\}$$

$$T = \left\{ v \in \mathbb{R}^{\mathcal{R}} \left| \begin{array}{l} v \in P \\ \text{sign}(v) \not\subseteq C \quad \forall \text{ internal circuits } C \end{array} \right. \right\}$$

Let $A \subseteq G$ be a set of reactions that we want to parametrize over. Let $K \subseteq \mathbb{R}^A$ be a convex set of parametrizations. Given a cost function $c \in \mathbb{R}^G$ we want to find a function $\hat{v} : K \rightarrow \mathbb{R}$ s.t.

$$\hat{v}(k) = \max cv_G : v \in T, v_A \geq k \quad (7.4)$$

if $\max\{cv_G : v \in T, v_A \geq k\}$ is feasible and we write $\hat{v}(k) = -\infty$ otherwise.

Note that if we want to parametrize on upper bounds, we can simply change the direction of the corresponding reactions.

To compute the optimal function, we run Algorithm 11. It differs from the method by Dua and Pistikopoulos [37] in the following points:

- Instead of fixing all decision variables, we only fix a subset, a so-called blocking set:

Definition 7.4.1 (Blocking Set) *Let $(\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S)$ be a metabolic network with lower and upper bounds ℓ, u and objective function $c \in \mathbb{R}^{\mathcal{R}}$. A set $Z \subseteq \overline{\mathcal{R}}$ is called a blocking set w.r.t. c if there exists no $v \in \mathbb{R}^{\mathcal{I}}$ with*

$$S_{\mathcal{I}}v = 0, v_{\text{Irrev}} \geq 0, v_Z \geq 0, c_{\mathcal{I}}v > 0,$$

where Irrev denotes the set of irreversible pseudo-reactions as defined in Def. 2.1.7. \square

We use a function $\mathcal{Z} : \mathbb{R}^{\mathcal{R}} \rightarrow 2^{\overline{\mathcal{R}}}$ that computes for a given flux distribution $w \in T$ a set $Z = \mathcal{Z}(w) \subseteq \overline{\mathcal{R}}$ with $w_Z \geq 0$ which is a blocking set w.r.t. the objective function c , for each of the additional constraints g , and for each of the flux forcing constraints (including the parametrization $v_A \geq k$).

It follows that when we add the additional constraint $v_Z \geq 0$ to the FBA problem (without thermodynamic constraints), every flux vector v can be turned into a thermodynamically feasible flux vector without violating flux through any objective or reaction with flux fixed by parametrization (Thm. 4.5.1). The condition $w_Z \geq 0$ enforces that w stays feasible. Since w is a thermodynamically feasible flux such a Z can always be computed (by fixing the sign of all reactions).

Often, we can obtain Z (partially) from the variable fixings of the MILP solver for the computation of w . Ideally, we want to compute a Z , which contains as few elements as possible, because this allows the greatest amount of flexibility in the LP. We will discuss the computation of Z in more detail in Sec. 7.4.3.

- Since no-good cuts are a rather expensive MILP technique, I turn the constraint $gx + hy \geq v^1(k)$ into the objective. In terms of the general MILP formulation this corresponds to solving the MILP:

$$(\bar{x}, \bar{y}, \bar{k}) := \arg \max \{ gx + hy - \hat{v}^i(k) : Ax + By - Ck \leq d, x \in \mathbb{R}^m, y \in \mathbb{Z}^n, k \in CR^i \}$$

In the special case of TFBA, we can formulate this as:

$$(\bar{v}, \bar{k}) := \arg \max \{ cv_G - \hat{v}_G^i(k) : v \in T, v_A \geq k, k \in CR^i \}$$

We observe that if there exists a point where the objective value is bigger than the current LP solution $\hat{v}^i(k)$, then this point is found. If no such point exists, we can see this also immediately by an objective value of 0. Since we maximize the MILP, it follows that the computed optimizer (\bar{v}) cannot be improved further and hence, the critical region which contains \bar{v} will also be part of the final solution.

A drawback of this approach is that I do not enumerate all optimal integer-assignments, but only one for each critical region. Since we are however, only interested in the optimal value, this problem is negligible. Also, I do not extend the mpLP solution for CR^i to other critical regions, where it also may provide an improvement. However, such a step also comes with the cost of solving the mpLP problem on a larger domain. If this is really worth the effort is unclear to me and should be investigated experimentally.

Theorem 7.4.1 *After Algorithm 11 has finished, we have computed a list of critical regions CR^i with value functions \hat{v}^i for $i = 1 \dots n$ such that*

$$\hat{v}(k) = \{ \hat{v}^i(k) : k \in CR^i \wedge i \text{ has no children} \}$$

is single-valued and the value is a solution to (7.4) for each $k \in K$. □

Algorithm 11 Compute critical regions for tFBA

```

 $CR^0 := K, \hat{v}^0 := -\infty$ 
 $i := 0$ 
 $n := 1$ 
while  $i < n$  do
  if  $\hat{v}^i = -\infty$  then
    solve  $(w, \bar{k}) := \arg \max\{cv_G : v \in T, v_A \geq k, k \in CR^i\}$ 
    if  $w$  exists then
      compute  $Z := \mathcal{Z}(w)$ 
       $\{(CR^n, \hat{v}^n), \dots, (CR^{n+l-1}, \hat{v}^{n+l-1})\} :=$  critical regions computed by the following mpLP for  $k \in CR^i$ :
        
$$\hat{v}(k) = \max\{cv_G : v \in P, v_A \geq k, v_Z \geq 0\}$$

      for  $j = n$  to  $n + l - 1$  do register  $j$  as child of  $i$ .
      end for
       $n := n + l$ 
    end if
  else
    solve  $(w, \bar{k}) := \arg \max\{cv_G - \hat{v}_G^i(k) : v \in T, v_A \geq k, k \in CR^i\}$ 
    if  $cv_G > 0$  then
      Compute  $Z := \mathcal{Z}(w)$ 
       $\{(CR^n, \hat{v}^n), \dots, (CR^{n+l-1}, \hat{v}^{n+l-1})\} :=$  critical regions computed by the following mpLP for  $k \in CR^i$ :
        
$$\hat{v}(k) = \max\{cv_G : v \in P, cv_G \geq \hat{v}^i(k), v_A \geq k, v_Z \geq 0\}$$

      for  $j = n$  to  $n + l - 1$  do
        if  $\hat{v}^j = -\infty$  ( $CR^j$  infeasible) then
           $\hat{v}^j := \hat{v}^i$ 
        end if
        register  $j$  as child of  $i$ .
      end for
       $n := n + l$ 
    end if
  end if
   $i := i + 1$ 
end while

```

To prove this theorem we will first have to show a couple of simple lemmas.

Lemma 7.4.1 *Let $i \in \{1, \dots, n\}$ that has no children and $\hat{v}^i = -\infty$. Then $\{v \in T : v_A \geq k\} = \emptyset$ for all $k \in CR^i$.*

PROOF Since critical regions computed by mpLP always partition the space of the problem, it follows that the only reason why i has no children is because $\max\{cv_G : v \in T, v_A \geq k, k \in CR^i\}$ is infeasible (by the if-statement of the algorithm). Hence $\{(v, k) : v \in T, v_A \geq k, k \in CR^i\} = \emptyset$. It follows that $\{v : v \in T, v_A \geq k\} = \emptyset$ for every fixed $k \in CR^i$. ■

Lemma 7.4.2 *Let $i \in \{1, \dots, n\}$ with $\hat{v}^i \neq -\infty$. For every $k \in CR^i$ there exists a $v \in T$ with $v_A \geq k$ and $cv_G = \hat{v}^i(k)$.*

PROOF By construction we know that there exists a $w \in T$ with $Z = \mathcal{Z}(w)$ such \hat{v}^i is the solution on CR^i of the mpLP

$$\max\{cv_G : v \in P, v_A \geq k, v_Z \geq 0\}.$$

By definition of Z and Thm. 4.5.1 it follows that

$$\hat{v}^i(k) = \max\{cv_G : v \in P, v_A \geq k, v_Z \geq 0\} = \max\{cv_G : v \in T, v_A \geq k, v_Z \geq 0\},$$

which was to be shown. ■

Lemma 7.4.3 *Let $i \in \{1, \dots, n\}$ that has no children and $\hat{v}^i \neq -\infty$. Then $\hat{v}^i(k) = \max\{cv_G : v \in T, v_A \geq k\}$ for all $k \in CR^i$.*

PROOF Since critical regions computed by mpLP always partition the space of the problem, it follows that the only reason why i has no children is because $\max\{cv_G - \hat{v}_G^i(k) : v \in T, v_A \geq k, k \in CR^i\} \leq 0$ (by the if-statement in the algorithm).

It follows that $\hat{v}_G^i(k) \geq \max\{cv_G : v \in T, v_A \geq k\}$ for every $k \in CR^i$. Since $\hat{v}^i(k)$ is a feasible solution for every $k \in CR^i$ by Lemma 7.4.2, this lemma follows. ■

PROOF (OF THEOREM 7.4.1) From the structure of the algorithm it is easy to see that every critical region is either partitioned into smaller critical regions or not. If a critical region CR^i is partitioned into smaller critical regions, it has children. Since it is a partition, there will exist for every $k \in CR^i$ exactly one child critical region CR^j with $k \in CR^j$.

If a region is not partitioned, it does not have children.

Hence, $\hat{v}(k)$ is single-valued for every $k \in K$. By Lemma 7.4.1 the value of $\hat{v}(k)$ is $-\infty$ if there exists no solution and by Lemma 7.4.3 the returned value is optimal for any given k with $\hat{v}(k) \neq -\infty$. ■

Theorem 7.4.2 *Algorithm 11 terminates in finite time.*

PROOF We show that the value of \hat{v} is monotonically increasing during the algorithm. Since there are only finitely many assignments of reaction directions, it follows that there can only be finitely many increases and hence, the algorithm terminates in finite time.

If $\hat{v}^i = -\infty$ on a critical region CR^i , then the value of \hat{v} can only increase on CR^i for any subdivision of CR^i .

If $\hat{v}^i \neq -\infty$, we recall that we solve the following mpLP problem to obtain the child-regions:

$$\hat{v}(k) = \max\{cv_G : v \in P, cv_G \geq c\hat{v}^i(k), v_A \geq k, v_Z \geq 0\}$$

By the constraint $cv_G \geq c\hat{v}^i(k)$ we enforce that every feasible solution of this mpLP is at least as good as the original solution. Regions, where the mpLP would otherwise be worse become infeasible. Child critical regions that are infeasible just take the solution of the parent critical region. Hence, the objective value of \hat{v} is also increasing in this case. ■

7.4.3 Finding Cut Sets for Blocking Internal Cycles

Now, we discuss how to compute a minimal blocking set Z w.r.t. an objective function c . We recall that c can either be the real objective function, or it can be an artificial objective function that simply forces flux through one of the flux forcing reactions. In either case, the theory for finding such a blocking set is similar.

We recall that a blocking set Z does not really block reactions entirely, but only one direction of the reaction. The effect must be that there exists no internal circulation $v \in \mathbb{R}^{\mathcal{I}}$ for which the objective value cv becomes positive. Additionally, we have a few reactions for which we are not allowed to block certain directions, because the thermodynamically feasible MILP solution w must stay feasible, i.e., Z has to satisfy $w_Z \geq 0$.

We observe that this problem is related to the cut set problem [79]:

Problem 7.4.1 (MinimumCut)

Given: *Metabolic network $(\mathcal{M}, \mathcal{R}, S)$, a reaction $r \in \mathcal{R}$*

Want: *Minimum cardinality set $X \subseteq \mathcal{R}$ such that $Sv = 0, v \geq 0, v_X = 0$ implies $v_r = 0$.*

□

We see the similarity to the computation of a blocking set by considering the metabolic network $(\mathcal{M}, \mathcal{I}, S_{\mathcal{I}})$. The only difference between the two problems is that for blocking sets, we are also allowed to block single directions of a reaction and some directions we are not allowed to block.

Acuña et al. [1] showed that the Problem 7.4.1 is APX-hard and that there is no $o(\log n)$ approximation unless $\mathbf{P} = \mathbf{NP}$. This result also transfers to finding minimal blocking sets:

Proposition 7.4.1 *There exists no algorithm (unless $\mathbf{P} = \mathbf{NP}$) that computes for a given metabolic network $(\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$, irreversible reactions $\text{Irrev} \subseteq \overline{\mathcal{R}}$, $c \in \mathbb{R}^{\mathcal{R}}$, and $w \in T$ a blocking set $Z \subseteq \overline{\mathcal{R}}$ with $w_Z \geq 0$ that is at most a factor $o(\log |\mathcal{R}|)$ bigger than the smallest such blocking set.*

PROOF We reduce Problem 7.4.1 to finding a minimal blocking set Z . Let $(\mathcal{M}, \mathcal{R}, S)$ be a metabolic network and $r \in \mathcal{R}$. Define $\mathcal{I} = \text{Irrev} = \mathcal{R}$, $c \in \mathbb{R}^{\mathcal{R}}$ with $c_r = 1$ and $c_s = 0$ for all $s \neq r$. It follows that $w_Z \geq 0$ is no restriction and blocking a reaction direction is equivalent to blocking the reaction completely. It follows that computing a minimal blocking set is precisely the same as computing a minimal-cut set. \blacksquare

However, we remark that for the correct functioning of Alg. 11 we do not have to compute a minimum blocking set. It is only advantageous for the run time to compute small blocking sets. Hence, it is sufficient to rely on suboptimal heuristics or approximation algorithms for computing blocking sets.

Acuña et al. [1] propose for MINIMCUT an iterative algorithm that computes fluxes in the network and then blocks the reaction in that mode with the smallest weight. They show that this algorithm is a λ -approximation, where λ is the length of the longest elementary mode. This approach can also be extended for computing blocking sets.

Here, however, I follow a different approach (more closely related to the approaches suggested in [87, 7]), for which I do not even have a weak quality guarantee like Acuña et al., but which was simpler to implement. The idea is that if we maximize flux through the objective reaction, the reduced costs of the dual linear problem tell us which reactions are rate limiting. If we block all these reactions, then no flux through the target reaction is possible anymore. We assume w.l.o.g. for the following theorem that all reactions are oriented in such a way that their bounds allow positive flux.

Theorem 7.4.3 *Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ be a metabolic network with lower and upper bounds $\ell \leq u > 0$. Let v be a thermodynamically feasible flux and $c \in \mathbb{R}^{\mathcal{R}}$ an objective function.*

The following LP is always feasible and bounded:

$$\begin{array}{l} \max cx \\ S_{\mathcal{I}}x = 0 \\ \left. \begin{array}{l} a_r := 0 \quad \ell_r \geq 0 \\ a_r := -1 \quad v_r \geq 0, \ell_r < 0 \\ a_r := -\infty \quad \text{else} \end{array} \right\} \leq x_r \leq \left\{ \begin{array}{ll} b_r := 1 & v_r \leq 0 \\ b_r := \infty & \text{else} \end{array} \right. \quad \forall r \in \mathcal{I} \end{array}$$

Let x^ be the maximizer. Let \bar{c} be the corresponding reduced costs, i.e., $\bar{c} = c - \mu S_{\mathcal{I}}$, where μ is the dual variable to the constraints $S_{\mathcal{I}}x = 0$.*

Then $Z := \{-r : x_r^* = 1 \wedge \bar{c}_r > 0, r \in \overline{\mathcal{R}}\}$ is a blocking set w.r.t. c with $v_Z \geq 0$.

PROOF The LP is feasible, since $x = 0$ is a feasible solution. Assume it were not bounded. It follows that there exists an $x \in \mathbb{R}^{\mathcal{I}}$ with $cx > 0$, $S_{\mathcal{I}}x = 0$, $x_r \geq 0$ for all $r \in \mathcal{I}$ with $v_r \geq 0$ and $x_r \leq 0$ for all $r \in \mathcal{I}$ with $v_r \leq 0$. It follows that $\text{sign}(x) \subseteq \text{sign}(v)$ and $x \neq 0$. This is a contradiction to v being thermodynamically feasible (Thm. 2.6.1).

Since the LP is bounded, there exists a feasible optimal dual solution μ, α, β , where μ is the dual variable for $S_{\mathcal{I}}x = 0$, $\alpha \leq 0$ is the dual variable for $x \geq a$ and $\beta \geq 0$ is the dual variable for $x \leq b$. Central to the following observation is the dual constraint

$$\mu S_{\mathcal{I}} + \alpha + \beta = c.$$

By complementary slackness, we know for a feasible optimal primal solution x^* that it holds for all $r \in \mathcal{R}$ that

$$x_r^* = 1 \Rightarrow x_r^* > 0 \Rightarrow x_r^* > a_r \Rightarrow \alpha_r = 0 \Rightarrow \bar{c}_r = c - \mu S_r \geq 0 \quad (7.5)$$

$$x_r^* = -1 \Rightarrow x_r^* < 1 \Rightarrow x_r^* < b_r \Rightarrow \beta_r = 0 \Rightarrow \bar{c}_r = c - \mu S_r \leq 0. \quad (7.6)$$

We conclude that (let Irrev be defined as in Def. 2.1.7)

- $x_r^* > 1$ implies $\bar{c}_r = 0$ for all $r \in \overline{\mathcal{R}}$ by complementary slackness.
- $x_r^* = 1$ implies $\bar{c}_r = 0$ for all $r \in \overline{\mathcal{R}}$ with $-r \notin Z$ since by definition of Z we have $\bar{c}_r \leq 0$ and by (7.5) (resp. (7.6)) it holds that $\bar{c}_r \geq 0$.
- $x_r^* < 1$ implies $\bar{c}_r \leq 0$ for all $r \in \text{Irrev}$, since $\text{Irrev} \subseteq \overline{\mathcal{R}}$.
- $-1 < x_r^* < 1$ implies $\bar{c}_r = 0$ for all $r \in \mathcal{R} \setminus \text{Irrev}$ by complementary slackness.
- $x_r^* = -1$ implies $\bar{c}_r \leq 0$ for all $r \in \overline{\mathcal{R}}$ by (7.6) (resp. (7.5)).
- $x_r^* < -1$ implies $\bar{c}_r = 0$ for all $r \in \overline{\mathcal{R}}$ by complementary slackness.

Assume there exists a $y \in \mathbb{R}^{\mathcal{I}}$ with $cy > 0$, $S_{\mathcal{I}}y = 0$, $y_Z \geq 0$ and $y_{\text{Irrev}} \geq 0$. We observe that for each $r \in \text{sign}(y)$ we have $-r \notin Z$ and $-r \notin \text{Irrev}$. Hence, $\bar{c}_{\text{sign}(y)} \leq 0$ and thus,

$$0 \geq \bar{c}y = cy - \mu S_{\mathcal{I}}y = cy > 0,$$

a contradiction. Hence, Z is a blocking set. ■

7.5 Tight Integration of Bilevel Programming into Parametric Programming

When we want to check whether a set of reactions A is diverting, it is sufficient to compute one optimal (minimal) solution of the bilevel program (7.3). In particular, if we know that a certain critical region cannot yield anything better than the current best (incumbent) solution, then we do not have to split up the critical region any further.

During the mpMILP algorithm, we frequently compute optimal solutions to the inner problem. Each such solution gives rise to a feasible solution of the outer problem, hence a candidate for an incumbent solution.

We can obtain lower bounds for a critical region by assuming a collaborative inner problem, i.e., by dropping the optimality condition of the inner problem.

7.5.1 Min-Max Problems

If the bilevel problem is a min-max problem, like in our case of (7.3), we can even derive stronger lower bounds.

We know that during the mpTFBA algorithm the solution \hat{v} increases monotonically with each subdivision (since the inner problem is a maximization problem). Since the objective value of the outer problem is the objective value of the inner problem, it follows that also the objective value of the outer problem increases. It follows that for any critical region CR^i $\min_{k \in CR^i} c_G \hat{v}_G^i(k)$ is a lower bound.

7.6 Implementation

For the analysis of *C. reinhardtii* 137AH, I computed the parametrized solutions using the multi parametric toolbox (MPT) [84]. The MPT toolbox offers an implementation for multi-parametric linear programming and other tools for computational geometry. However, the MPT toolbox only works with full dimensional polyhedra. Since, the polyhedra obtained from FBA problems contain a lot of direct equality-constraints (steady-state assumption) and several indirect equality constraints by flux bounds that fix reaction fluxes, I wrote additional code to remove unnecessary dimensions. In particular the identification and elimination of indirect inequality constraints introduced numerical instabilities and it occurred several times that the LP solver falsely reported that no solution exists for LPs that were constructed to be feasible.

I did not implement the bilevel optimization method, since the parametrized result can also be used to generate pretty pictures, which may give a deeper insight into the functions behind the effect.

7.7 Results

For the analysis, Guillaume Cogne supplied me with a core metabolic network and a genome-scale metabolic network. Since the genome-scale metabolic network contains many internal cycles, it was not possible to run even ordinary thermodynamic flux variability analysis on it. Hence, we focused on the analysis of the core metabolic network. In addition to the steady-state constraints of the core network, we also added two additional energetic constraints derived from phosphate-oxygen ratios (P/O-ratios) that could be formulated using linear equality constraints (v_r denotes the flux through reaction r) [25]:

$$\begin{aligned} \text{P/O}_{(2e^-)} \left(\frac{1}{2} v_{\text{PSI}} - v_{\text{FQR}} \right) &= v_{\text{CF-ATPase}}, & \text{P/O}_{(2e^-)} &= 1.4273 \\ \text{P/O}_{(\text{NADH})} v_{\text{NADH}} - \text{P/O}_{(\text{Succ})} (v_{\text{NADH}} + v_{\text{QCR}}) &= v_{\text{MF-ATPase}}, & \text{P/O}_{(\text{NAD})} &= 1.76, \text{P/O}_{(\text{Succ})} = 1.17 \end{aligned}$$

Since G. Cogne thought that *NADH-Glutamine oxoglutarate aminotransferase* (NADH-GOGAT) might be a trouble maker, I ran computations for the network with NADH-GOGAT and without NADH-GOGAT (i.e., by blocking NADH-GOGAT).

The maximal growth rate under the low light intensity of $v_{\text{h}\nu} = 9.5059$ computed by FBA is 0.0292, both with NADH-GOGAT and without.

I found that *ferredoxin NADPH reductase* (FNR) is the only reaction that by itself can divert so much flux that the cell would stop growing for the lower light intensity. Interestingly, this effect only kicks in on a very small segment of the range of feasible flux values (see Fig. 7.8). Other reactions that have large diverting effect are *transketolase* (TRK1, TRK2), *ribulose-phosphate 3-epimerase* (RPE), *ribulose 5-phosphate isomerase* (RPI), *phosphoribulokinase* (PRK), *ribulose biphosphate carboxylase* (Rubisco), *phosphoglycerate kinase* (PGK). Tab. 7.1 gives an overview of all reactions where flux forcing could reduce the maximal yield of 0.0292 under the low light intensity ($v_{\text{h}\nu} = 9.5059$), while still allowing FBA optimal growth for the high light intensity ($v_{\text{h}\nu} = 19.0118$).

For some of the reactions, the diversion effect can be increased by allowing two flux forcing reactions. Interestingly, all combinations that I found consist of reactions with already rather high flux diversion, like RPE, RPI, TRK1, TRK2 and a reaction with low flux diversion like *glyceraldehyde 3-phosphate dehydrogenase* (GAPDH) or the hydrogen transport (TH: $\text{NAD} + \text{NADPH} \leftrightarrow \text{NADH} + \text{NADP}$). The maximal effect can be read from Tables 7.2, 7.3 and can also be seen in Figs. 7.11, 7.12.

We observe that it does not make much of a difference if NADH-GOGAT is part of the network. We can conclude that if the effect is caused by flux forcing reactions as modeled in this computational study, then NADH-GOGAT likely does not play a central role.

Furthermore, we observe that GAPDH and TH induce exactly the same effect. When we however consider the flux modules (see Chapter 6) of the metabolic network, we see that GAPDH together with GAPDhr form an alternative to TH. This the case, both in the network

Table 7.1: Single diverting reactions and their maximal effect.

| Diverting reaction | without NADH-GOGAT | with NADH-GOGAT | Diverting reaction | without NADH-GOGAT | with NADH-GOGAT |
|--------------------|--------------------|-----------------|--------------------|--------------------|-----------------|
| PGI | 0.0268 | 0.0268 | PFK | 0.0269 | 0.0269 |
| FBPase | 0.0262 | 0.0262 | FBA | 0.0244 | 0.0244 |
| TPI | 0.0211 | 0.0211 | GAPDH | 0.0268 | 0.0265 |
| GAPDHr | 0.0222 | 0.0217 | PGK | 0.0203 | 0.0203 |
| PGM | 0.0238 | 0.0238 | ENO | 0.0238 | 0.0238 |
| PK | 0.0267 | 0.0267 | PD | 0.0261 | 0.0261 |
| CS | 0.0266 | 0.0266 | ACON | 0.0266 | 0.0266 |
| IDH | 0.0266 | 0.0266 | OGDH | 0.0269 | 0.0269 |
| SCS | 0.0269 | 0.0269 | SDH | 0.0269 | 0.0269 |
| FH | 0.0268 | 0.0268 | MDH | 0.0267 | 0.0267 |
| G6PDH | 0.0269 | 0.0269 | PGL | 0.0269 | 0.0269 |
| GND | 0.0269 | 0.0269 | RPE | 0.0208 | 0.0208 |
| RPI | 0.0211 | 0.0211 | TRK1 | 0.0209 | 0.0209 |
| TAL | 0.0262 | 0.0262 | TRK2 | 0.0207 | 0.0207 |
| PRK | 0.0209 | 0.0209 | Rubisco | 0.0209 | 0.0209 |
| SBA | 0.0254 | 0.0254 | SBPase | 0.0254 | 0.0254 |
| ICL | 0.0268 | 0.0268 | MS | 0.0268 | 0.0268 |
| PEPC | 0.0268 | 0.0268 | PEPCK | 0.0269 | 0.0269 |
| MME1 | 0.0267 | 0.0267 | MME2 | 0.0267 | 0.0267 |
| PPDK | 0.0269 | 0.0269 | ADK | 0.0267 | 0.0267 |
| FNR | 0.0000 | 0.0000 | NADH | 0.0256 | 0.0256 |
| QCR | 0.0256 | 0.0256 | COX | 0.0256 | 0.0256 |
| UCP | 0.0256 | 0.0256 | MF-ATPase | 0.0256 | 0.0256 |
| ATPM | 0.0269 | 0.0269 | TH | 0.0268 | 0.0265 |
| PPase | 0.0266 | 0.0266 | | | |

The effect is measured in the maximal possible biomass yield under low light conditions when the corresponding reaction is maximally diverting, i.e., it is the objective value of the bilevel problem (7.3). The results were computed with the reaction NADH-GOGAT included in the network and without it.

Table 7.2: Maximal diversion for pairs of reactions with large effect for the network without NADH-GOGAT.

| | single | RPE | RPI | TRK1 | TRK2 |
|--------|--------|--------|--------|--------|--------|
| single | | 0.0208 | 0.0211 | 0.0209 | 0.0207 |
| GAPDH | 0.0268 | 0.0163 | 0.0171 | 0.0166 | 0.0161 |
| TH | 0.0268 | 0.0163 | 0.0171 | 0.0166 | 0.0161 |

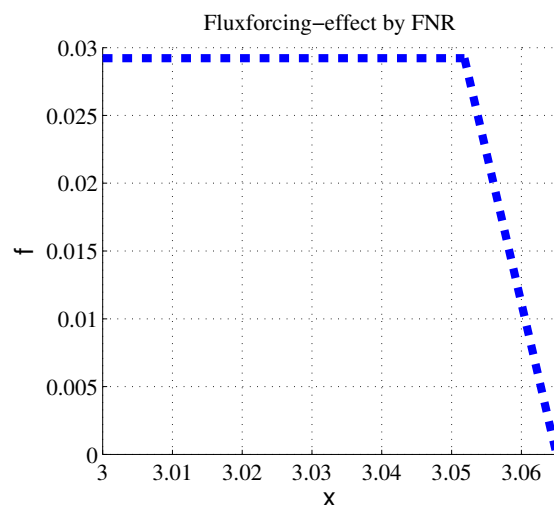


Figure 7.8: Flux forcing effect of reaction *ferredoxin NADPH reductase* (FNR) in the network without NADH-GOGAT. The x -axis specifies the flux forcing parameter and the f -axis gives the maximal flux rate through biomass possible with $v_{\text{FNR}} \geq x$.

Table 7.3: Maximal diversion for pairs of reactions with large effect for the network with NADH-GOGAT.

| | single | RPE | RPI | TRK1 | TRK2 |
|--------|--------|--------|--------|--------|--------|
| single | | 0.0208 | 0.0211 | 0.0209 | 0.0207 |
| GAPDH | 0.0265 | 0.0153 | 0.0161 | 0.0156 | 0.0151 |
| TH | 0.0265 | 0.0153 | 0.0161 | 0.0156 | 0.0151 |

with NADH-GOGAT (Fig. 7.9) and without (Fig. 7.10).

7.8 Discussion

First I want to remark a technical detail: The careful reader may have observed that in the flux forcing plots with the GAPDH-reaction (Fig. 7.11), the reaction RPE (for example) alone seems to be able to achieve a large diversion and reduce flux through biomass to a value less than 0.018 and not only to 0.0208 as predicted in Tab. 7.1. If on the other hand, we look at Fig. 7.12, we see that indeed RPE alone cannot cause a reduction to 0.018. The effect is probably caused by thermodynamic constraints. An infinitesimally small flux through GAPDH already prohibits flux through other reactions and hence the flux diversion becomes larger. Since the parameter space for which $v_{\text{GAPDH}} = 0$ is allowed has measure 0, it is not drawn in the figure and hence, the misleading impression is produced.

Biologically, the results do not answer our hope of identifying eukaryotic mechanisms

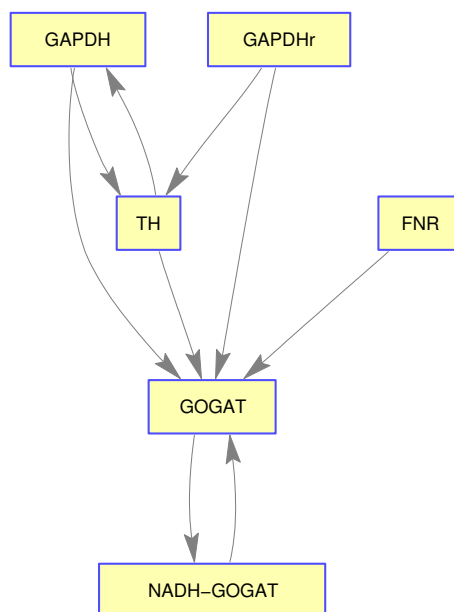


Figure 7.9: Module containing TH and GAPDH without NADH-GOGAT blocked. From this blocking graph, we can see that TH and GAPDH also here are blocking each other and hence, form alternatives.

that could be responsible for the different growth behavior of *C. reinhardtii* compared to the cyanobacterium *A. platensis*. All reactions with large diversion effect that we found are catalyzed by enzymes that are also available in the cyanobacterium *A. platensis*. On the contrary, the reaction *NADH-GOGAT*, which does not exist in cyanobacteria, only has very little effect even on the flux forcing behavior of the other reactions.

Since flux diversion is also a systemic effect, it seems at this point important to do the same computations also on a model of a cyanobacterium like *A. platensis*. It could be that a reaction common to both *C. reinhardtii* and *A. platensis* becomes flux diverting because of other additional (or missing) reactions. In this case an analysis to where the flux will be diverted will give the necessary insights into the mechanism. If the diversion leads to the production of by-products, then this would generate a hypothesis that could be verified experimentally. In the best case such predictions can even be exploited in biotechnological applications. In the case of *C. reinhardtii* it is for example not necessarily a bad effect that the growth rate decreases. If with decreasing growth rate the production of carbohydrates increases, this would be positive, because the carbohydrates can more easily be turned into bio-fuel than other biomass compounds.

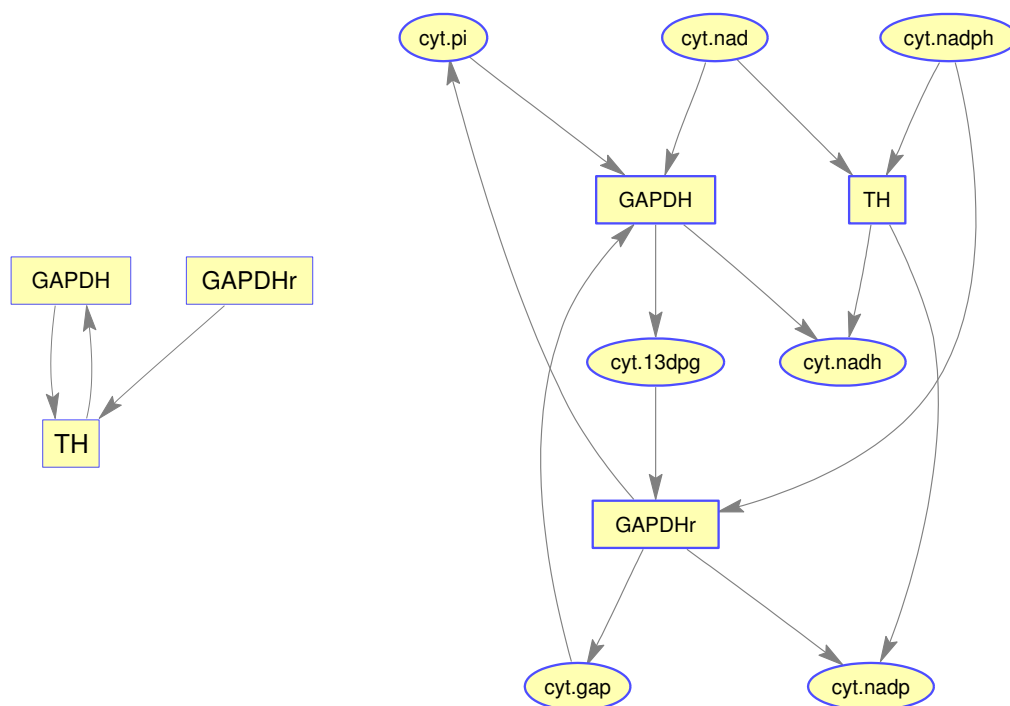


Figure 7.10: Module containing TH and GAPDH with NADH-GOGAT blocked. On the left the blocking graph (see Sec. 6.3.3.2) of the module is shown and on the right we see the subnetwork itself.

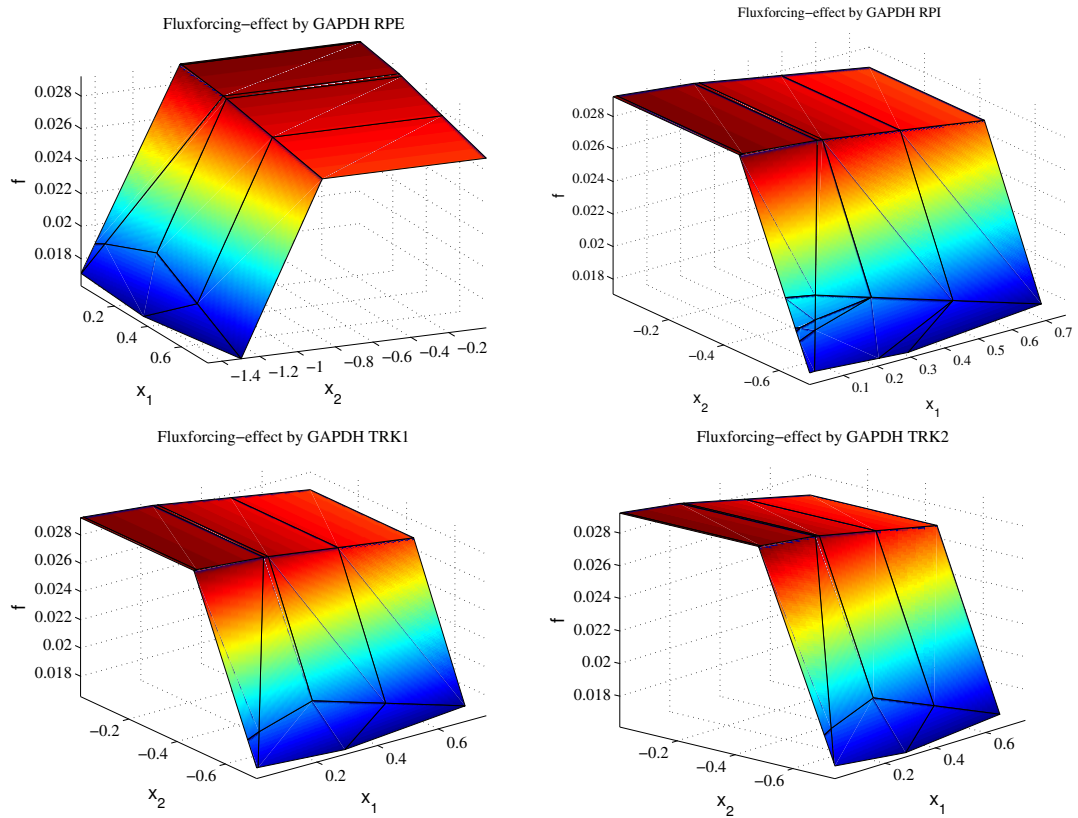


Figure 7.11: Flux diverting sets containing *glyceraldehyde-3-phosphate dehydrogenase* (GAPDH), whose flux forcing parameter is always marked down on the axis x_1 . The flux forcing parameter of the other reaction is given on x_2 . The results are for the network without NADH-GOGAT.

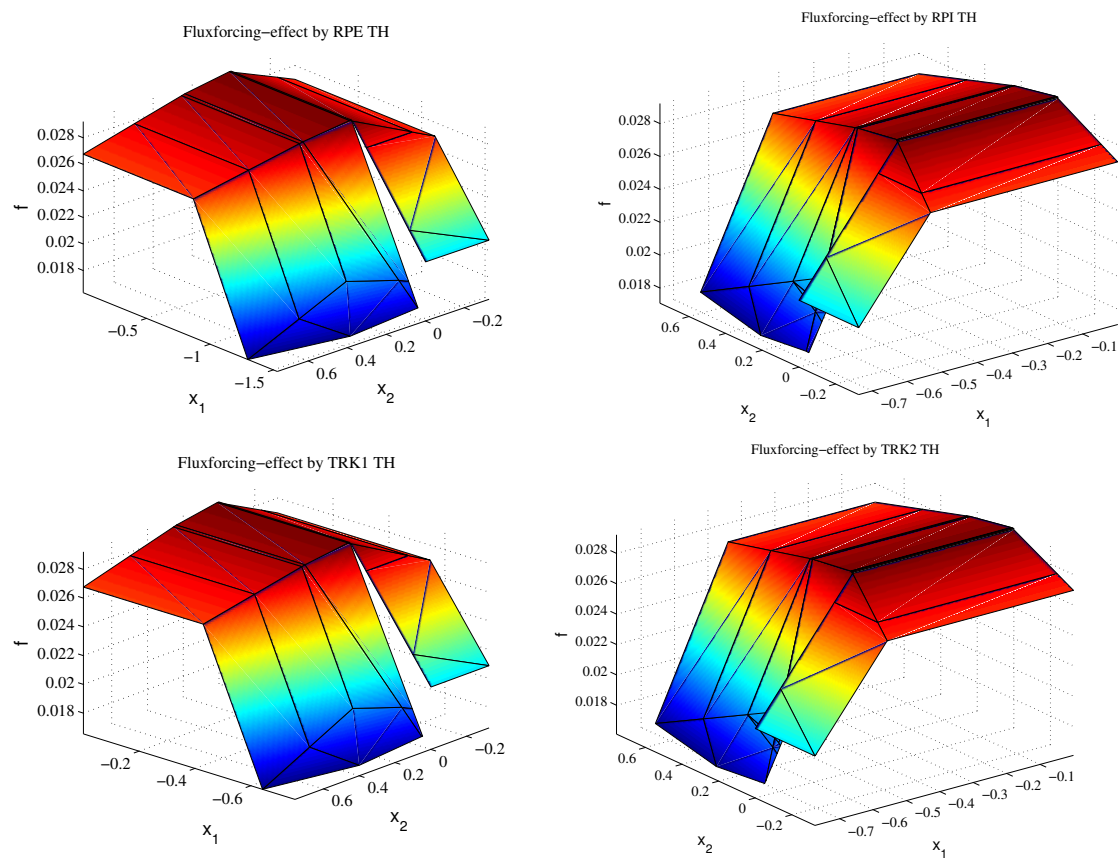


Figure 7.12: Flux diverting sets containing a hydrogen transporter (TH: $\text{NAD} + \text{NADPH} \leftrightarrow \text{NADH} + \text{NADP}$), whose flux forcing parameter is always marked down on the axis x_2 . The flux forcing parameter of the other reaction is given on x_1 . The results are for the network without NADH-GOGAT.

Chapter 8

Flux Coupling Analysis with Thermodynamic Constraints

Abstract *Flux coupling analysis* (FCA) has proven to be a useful tool for aiding metabolic reconstructions and guiding genetic manipulations. It originally has been introduced for constraint-based models under the steady-state assumption. Recently, it has been shown that the steady-state assumption can be replaced by much weaker lattice theoretic properties. This allowed the extension of efficient algorithms for FCA also to certain classes of qualitative models.

In this work, we further generalize FCA to arbitrary qualitative models and present an efficient computation method. We illustrate this on the example of steady-state metabolic networks with loop-law thermodynamic constraints, which do not satisfy the lattice theoretic properties required in the previous methods.

We will discuss how thermodynamic constraints will alter the coupling results obtained by ordinary FCA theoretically and practically on the example of a set of genome-scale metabolic networks.

This is work done together with Yaron Goldstein and Alexander Bockmayr. Yaron worked on applications of lattice theory to metabolic networks and flux coupling analysis in particular [57]. Although thermodynamic constraints actually break the lattice theoretic axioms Yaron was working with, we found a way to apply his concepts also to the metabolic networks with thermodynamic constraints and in the end to arbitrary qualitative models. This is submitted as “Qualitative and thermodynamic flux coupling analysis” to *Journal of IEEE Transactions on Computational Biology and Bioinformatics* [127]

8.1 Introduction

One of the central applications of FBA (see Chapter 4) is to check if a certain gene-respective reaction-knockout will be lethal. However, it is often the case that not all reactions can carry flux independently from each other. For instance, if one reaction is knocked out, we may implicitly also disable flux through another reaction. This is useful information for identifying drug and knock-out targets, because some reactions may be easier to manipulate than others [63] and hence, this information can be used to reduce the number of wet lab experiments. Such coupling information can also be used to validate and check consistency of metabolic network reconstructions [19] and to find co-regulated reactions [109].

Flux coupling analysis (FCA) was introduced by Burgard et al. [21] to comprehensively analyze these kinds of dependencies between reactions. They introduced the following three types of reaction coupling between two reactions r, s (see also Sec. 2.1.2). Note that I use here the notation introduced in [90] to not confuse this with directed flux coupling.

- r and s are *directionally coupled* ($r \xrightarrow{=0} s$) if for each steady-state flux vector v zero flux through r implies zero flux through s ($v_r = 0 \Rightarrow v_s = 0$) but not necessarily the reverse.
- r is *partially coupled* to s if $r \xrightarrow{=0} s$ and $s \xrightarrow{=0} r$.
- r is *uncoupled* to s if neither $r \xrightarrow{=0} s$ nor $s \xrightarrow{=0} r$.

While methods like *elementary flux mode analysis* [141] can be used to obtain even more comprehensive coupling information between reactions, combinatorial explosion makes it prohibitive for application on genome-scale networks. FCA on the other hand convinces with its performance even on genome-scale metabolic networks with thousands of reactions. With the theoretical results by Larhlimi et al. [86, 85, 31, 90] we are now able to run FCA in a couple of seconds for genome-scale metabolic networks.

We observe that in the definition of flux coupling analysis we do not consider the quantity of flux through reactions, but only the question whether there is some flux is of importance. Hence, FCA is a purely qualitative analysis. Hence, one may be tempted to also apply FCA to qualitative flux models of metabolic networks as introduced in [150, 23]. These models are motivated by the fact that the steady-state assumption is rather strong, which may not hold for short time scales.

Goldstein and Bockmayr [58] generalized FCA to constraint-based models that do not necessarily have to satisfy the steady-state constraints as long as they satisfy certain axioms of lattice theory. The generalization to lattices already allows us to analyze a wide range of network models. For example, we may use lattices to also analyze models where bounds on the flux rates are given.

The following example however shows that sometimes the lattice assumption is too strong and that we actually want to generalize FCA to models that do not even satisfy lattice theoretic axioms. Let us consider the network shown in Fig. 8.1. There, reaction 3 is not coupled to reaction 1, because $v = (0, 0, 1, 1)$ is a steady-state flux. However, this flux is an internal circulation, which violates the second law of thermodynamics (see Sec. 2.6). We already see at this example that with thermodynamic constraints reaction 3 is fully coupled to 1 and thus, we get a stronger result than without thermodynamic constraints.

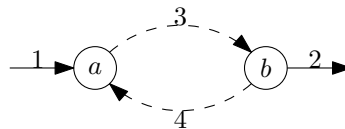


Figure 8.1: Without thermodynamic constraints reaction 3 is not directionally coupled to reaction 1, but it is with thermodynamic constraints, since reactions 3 and 4 form an internal cycle (dashed arrows).

The main property of a lattice of metabolic pathways is that you can combine the pathways by taking the union of the sets of involved reactions. However, this is not always possible if the pathways have to satisfy thermodynamic constraints. For example, in the network shown in Fig. 8.2 the pathways $\{r, b, s, d, e\}$, $\{r, a, s, c, e\}$ are thermodynamically feasible, but the combination $\{r, a, b, s, c, d, e\}$ is not thermodynamically feasible, since it contains the internal cycle $\{r, a, d\}$.

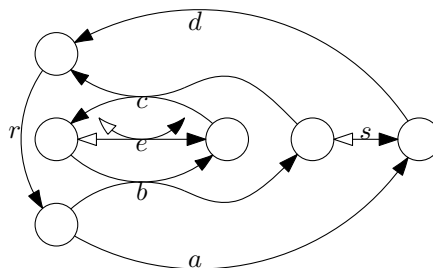


Figure 8.2: The only exchange reaction in this network is e . If r carries flux, then s must also carry flux. But s can carry both positive or negative flux. All reactions are irreversible except those with two arrows (black and white), where the white arrow indicates the reverse direction.

Here, we show why we can circumvent this problem and incorporate thermodynamic constraints into FCA. More generally, we show that the same algorithm as proposed in [58] can be applied on effectively any qualitative model. In particular, this implies that the algorithm poses no restrictions on what kinds of constraints may be added to the stoichiometric model. As an example, we show how to solve thermodynamically constrained FCA (tFCA).

To do this, we extend the flux coupling relation $\xrightarrow{=0}$ to lattices in general and define a lattice L^T that represents the thermodynamically feasible fluxes. The application of the flux coupling relation on L^T gives then the desired thermodynamically constrained FCA.

8.2 Lattice Theory

Lattices generalize the concept of flux pathways in a qualitative way. Although lattice theory is a much more general concept (see [29]), we only work here with lattices L represented by a collection of reaction-sets. Each of the reaction-sets can be interpreted as a pathway in the metabolic network. In the case of flux conservation pathways, the lattice of flux conserving pathways L^F is defined as the set of supports of steady-state fluxes, i.e.,

$$L^F := \{\text{supp}(v) : Sv = 0, v_{\text{Irrev}} \geq 0\}$$

As introduced in [58], we can generalize the notion of flux coupling of two unblocked reactions r, s as follows: Reactions r, s are directionally coupled ($r \xrightarrow{=0} s$) in a lattice L if and only if

$$\forall a \in L : r \notin a \Rightarrow s \notin a$$

In the case of $L = L^F$, this corresponds to ordinary flux coupling on

$$F := \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, v_{\text{Irrev}} \geq 0\}.$$

Clearly, we can apply this definition also to the supports of thermodynamically feasible fluxes in

$$\begin{aligned} L^T &:= \{\text{supp}(v) : v \in T\}, \text{ where} \\ T &:= \{v \in F : v \text{ thermo feas. w.r.t. } Q = \mathbb{R}^{\mathcal{M}} \text{ by Def. 2.6.4}\}. \end{aligned}$$

However, L^T is not a lattice in general and thus we cannot use the results of [58] directly. Every lattice is closed with respect to taking unions, i.e., if $a_1, a_2 \in L$, it follows that $a_1 \cup a_2 \in L$. As already shown in the introduction, using Fig. 8.2, L^T is not a lattice.

8.2.1 FCA for arbitrary P

Here, we will discuss a generalization of FCA that not only works on $L^T \subseteq L^F$ but on arbitrary $P \subseteq 2^{\mathcal{R}}$ with $P \neq \emptyset$. We define

$$r \xrightarrow{=0}_P s \Leftrightarrow \forall a \in P : r \notin a \Rightarrow s \notin a$$

and observe:

Observation 8.2.1 Let $\tilde{P} \subseteq \mathbb{R}^{\mathcal{R}}$ and $P = \{\text{supp}(v) : v \in P\}$. Then it holds that

$$r \xrightarrow{=0}_P s \Leftrightarrow r \xrightarrow{=0}_{\tilde{P}} s.$$

To show that we can use the method introduced in [58] to find the coupling pairs, we first need to define the *irreducible* elements $\mathcal{J}(P)$ of a set of reaction-sets:

$$\mathcal{J}(P) := \left\{ b \in P \setminus \{\emptyset\} : \forall A \subseteq P : b = \bigcup_{a \in A} a \Rightarrow b \in A \right\}.$$

Next we define the *closure* of P :

$$\langle P \rangle = \left\{ \bigcup_{a \in A} a : A \subseteq P \right\}.$$

It is easy to see that $\langle P \rangle$ is the smallest lattice that contains P . We say P is a *generator* of $\langle P \rangle$.

We observe that for each lattice L , $\mathcal{J}(L)$ is the unique minimal generator: $L = \langle \mathcal{J}(L) \rangle$. Thus, $\langle \mathcal{J}(P) \rangle = \langle P \rangle$ and $\mathcal{J}(P) = \mathcal{J}(\langle P \rangle)$.

Now we are prepared to prove our main result:

Theorem 8.2.1 Let $\emptyset \neq P \subseteq 2^{\mathcal{R}}$ and $B \subseteq 2^{\mathcal{R}}$ with $\mathcal{J}(P) \subseteq B \subseteq \langle P \rangle$.

Then the following are equivalent:

(a) $r \xrightarrow{=0}_P s,$

(b) $r \xrightarrow{=0}_{\langle P \rangle} s,$

(c) $r \xrightarrow{=0}_B s,$

(d) $r \xrightarrow{=0}_{\langle B \rangle} s.$

□

PROOF It is $\mathcal{J}(P) \subseteq B \subseteq \langle P \rangle$ and therefore $\langle P \rangle = \langle B \rangle$.

Thus, it is sufficient to prove (c) \Leftrightarrow (d).

\Rightarrow : Assume $r \not\xrightarrow{=0}_{\langle B \rangle} s$. It follows by definition that there exists an $a \in \langle B \rangle$ such that $r \not\subseteq a \ni s$. Since $\mathcal{J}(\langle B \rangle)$ is a generator of $\langle B \rangle$ it follows that there exists a $b \in \mathcal{J}(\langle B \rangle)$ with $r \not\subseteq b \ni s$. Since $b \in \langle B \rangle$, it follows that there exists by definition of $\langle B \rangle$ an $A \subseteq B \subseteq \langle B \rangle$ with $b = \bigcup_{a \in A} a$. Since b is irreducible, it follows that $b \in A$ and thus, $b \in B$. This proves that r is not directionally coupled to s in B .

\Leftarrow : Assume r is not directionally coupled to s in B . It follows that there exists an $a \in B$ such that $r \not\subseteq a \ni s$. Since $B \subseteq \langle B \rangle$, we have $a \in \langle B \rangle$. It follows that r is not directionally coupled to s in $\langle B \rangle$. ■

8.2.2 Algorithm for FCA in P

In [58] an algorithm that works on any lattice L was introduced. To use it we only need to implement a test method $\text{isCoupled}(r, s)$ that returns a lattice element $a \in L$ with $r \not\leq a \leq s$ if such an element exists, otherwise \emptyset . Now the same holds for our flux coupling in L^T or P respectively.

By Thm. 8.2.1 it is sufficient to define $\text{isCoupled}_B(r, s)$ in a way that selects $a \in B$ with $\mathcal{J}(P) \subseteq B \subseteq \langle P \rangle$.

Theorem 8.2.2 *Let L be a lattice, $P \subseteq L$ and B with $\mathcal{J}(P) \subseteq B \subseteq \langle P \rangle$.*

Let further isCoupled_B be a function that fulfills the following conditions:

$$\text{isCoupled}_B(r, s) = \begin{cases} a & \exists a \in B : r \not\leq a \leq s, \\ \emptyset & \text{otherwise.} \end{cases}$$

Then $r \xrightarrow{=0}_P s$ if and only if $\text{isCoupled}_B(r, s) = \emptyset$. □

PROOF Because of Thm. 8.2.1 we know that $r \xrightarrow{=0}_P s$ holds if and only if $r \xrightarrow{=0}_B s$.

By definition of isCoupled_B we further know that $\text{isCoupled}_B(r, s) = \emptyset$ if and only if $r \xrightarrow{=0}_B s$. ■

Corollary 8.2.1 *The algorithm introduced in [58] implemented with usage of isCoupled_B performs FCA for P . □*

8.3 FCA in T

For our implementation of thermodynamically constrained FCA, we do not operate on the space of thermodynamically feasible fluxes directly, but we employ the result of Thm. 8.2.1 and work on the space

$$B := \left\{ \text{supp}(v) : \begin{array}{l} Sv = 0, v_{\text{Irrev}} \geq 0, \\ \nexists w \neq 0 : S_{\mathcal{I}}w = 0, w_{\text{Irrev}} \geq 0, \text{supp}(w) \subseteq \text{supp}(v) \end{array} \right\}.$$

We observe that $B \subseteq L^T$, because by Thm. 2.6.1 it holds that

$$T := \left\{ v : \begin{array}{l} Sv = 0, v_{\text{Irrev}} \geq 0, \\ \nexists w \neq 0 : S_{\mathcal{I}}w = 0, \text{sign}(w) \subseteq \text{sign}(v) \end{array} \right\}.$$

We further observe that B can be strictly smaller than L^T . For example in L^T we allow flux through parallel reactions, which is not allowed in B , because parallel reactions together form an internal cycle. However, we will now show that all irreducible elements of L^T are contained in B . Therefore, we use the following result:

Proposition 8.3.1 (Cor. 5 in [99]) *Let $\mathcal{N} = (\mathcal{M}, \mathcal{R} = \mathcal{I} \cup \mathcal{E}, S)$ be a metabolic network with irreversible reactions Irrev and potential space $Q \subseteq \mathbb{R}^{\mathcal{M}}$. Then it holds for the space T of thermodynamically feasible fluxes w.r.t. Q that*

$$\text{conv}(T) = \text{conv}\{e \text{ elementary mode} : e \in T\}.$$

Now we can state the desired result:

Proposition 8.3.2 $\mathcal{J}(L^T) \subseteq B$

PROOF By Prop. 8.3.1, every $e \in \mathcal{J}(L^T)$ is an elementary mode and hence, minimal in L^F . Thus, there is no $a \in L^F \setminus \{\emptyset\}$ with $a \subsetneq e$. Assume $e \notin B$. Then there exists $w \in \mathbb{R}^{\mathcal{I}} \setminus \{0\}$ with $S_{\mathcal{I}}w = 0$, $w_{\mathcal{I} \cap \text{Irrev}} \geq 0$, and $\text{supp}(w) \subseteq e$. If $\text{supp}(w) \subsetneq e$, then e is not minimal in L^F . If $\text{supp}(w) = \text{supp}(e)$, then e is the support of the internal circulation w , and it follows $e \notin L^T$. In both cases, we get a contradiction, hence $e \in B$. ■

Hence, $\mathcal{J}(L)^T \subseteq B \subseteq L^T \subseteq \langle L \rangle^T$ and by Thm. 8.2.1 it follows that the FCA results on B are the same as the FCA results on L^T .

To check if two elements are coupled in B , we implemented the method `isCoupled` in which we solve the following MILP, where \mathcal{C} denotes the set of unoriented circuits of the internal circuit matroid (Def. 2.5.6) and M is a sufficiently large constant (typically 1000):

$$\begin{aligned} & \min 0v && (8.1) \\ \text{s.t.} & \quad Sv = 0 \\ & \quad v_{\text{Irrev}} \geq 0 \\ & \quad -Ma_i \leq v_i \leq Ma_i && (a_i = 0 \Rightarrow v_i = 0) \\ & \quad \sum_{i \in c} a_i \leq |c| - 1 \quad \forall c \in \mathcal{C} && (\text{circuit-constraints}) \\ & \quad v_r = 0 \\ & \quad v_s = 2b - 1 && (v_s \in \{-1, 1\}) \\ & \quad a_i, b \in \{0, 1\} \quad \forall i \in \mathcal{R} \end{aligned}$$

The idea of this MILP is the following: The variables a denote the support of the flux vector v . However, we require only the relation $a_i = 0 \rightarrow v_i = 0$, because the a_i are only used in the circuit-constraints. Violated circuit-constraints cannot be turned feasible by setting additional $a_i = 1$. The decision variable b is used to force positive or negative flux through reaction s .

Algorithm 12 Implementation of `isCoupled(r, s)` for the flux space B .

```

function isCoupled(r, s)
  solve (8.1)
  if the MILP is infeasible then
    return  $\emptyset$ 
  else
    return  $\{i \in \mathcal{R} : v_i \neq 0\}$ .
  end if

```

8.4 Implementation

The high speed of the algorithm introduced in [58] results from a search via nested intervals approach: The reactions *not coupled* to r in a lattice L are the elements of

$$\mathbf{max}_r := 1_{L_{\perp\{r\}}} := \bigcup_{A \in L_{\perp\{r\}}} A,$$

where $L_{\perp\{r\}}$ is the lattice defined by

$$L_{\perp\{r\}} := \{a \in L : r \notin a\}.$$

The algorithm searches for \mathbf{max}_r via lower and upper bounds, where $\mathbf{lb} \subseteq \mathbf{max}_r$ is union of known (witness) pathways that do not contain r and an upper bound $\mathbf{ub} \supseteq \mathbf{max}_r$ with reactions missing who are known to be blocked or to be knocked out by r . Thus, only the difference reactions $s \in \mathbf{ub} \setminus \mathbf{lb}$ have to be tested.

Traditional FCA tests the feasibility of $\{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, v_{\text{Irrev}} \geq 0, v_r = 0, |v_s| \geq 1\}$. Supports of feasible solution vectors extend the lower bound (by at least adding s), infeasible tests lead to an update $\mathbf{ub} \leftarrow \mathbf{ub} \setminus \{s\}$.

One could now implement tFCA be just replacing these LP feasibility tests by the function `isCoupled` as mentioned above. Note, that in `isCoupled` we have to solve an NP-hard problem due to the thermodynamic constraints (see Ch. 3). Since solving `isCoupled` is computationally hard, we decided to use our knowledge of the lattice structure to minimize the number of `isCoupled` tests by introducing a relaxation as pre-processing: It is $\langle T \rangle \subseteq L^F$, thus we can start with $\mathbf{ub} = 1_{L^T} \cap 1_{L_{\perp\{r\}}^F}$, where 1_{L^T} are the unblocked reactions under thermodynamic constraints and $1_{L_{\perp\{r\}}^F}$ the set of reactions that are uncoupled to r in traditional FCA [58].

Our software is implemented in Java and alternates between traditional FCA and tFCA, where the results of the FCA calculation steps are used to deduce tFCA properties as often as possible.

To run traditional FCA and to solve `isCoupled` we use CPLEX to solve the LPs and MILPs. The internal circuits of the network are computed using a variant of the WW-algorithm [168] using the `efmtool` by Terzer et al. [155]. All the networks analyzed in

this study have a low number of internal cycles, which made this approach feasible and easy to implement.

The current implementation does not respect concentration information that may be available. An integration of concentration information can also be done in a similar fashion using the results from Sec. 2.6.3.3 about infeasible sets or the methods from Chapter. 4. Note however that the transformation from Section 8.3 cannot be applied directly since we used specific properties of the infeasible sets. Here, a method that solves the thermodynamically constrained flux problem in full generality is needed, like it is used in [69, 25].

8.5 Minimal Representation of Coupling Data

This is work done together with Alexandra Grigore and Yaron Goldstein [128].

When we analyze all coupled pairs of reactions, we have a lot of redundant information. For example, if we have the couplings $a \xrightarrow{=0} b, b \xrightarrow{=0} c, a \xrightarrow{=0} c$ then the third coupling can be inferred from the previous two couplings. In practice this can lead to a quadratic blow-up of redundant couplings and hence the numbers of coupled pairs does not really show the gained information. Hence, we computed a minimum set of couplings from which all couplings can be deduced (for the case of FCA) and a minimum extension set of couplings from which all additional couplings in tFCA can be deduced.

Since the number of couplings can grow quadratically simply by transitive closure, we only compared the size of minimal generators / extension of the transitive closure. Here, we will formally define our minimality condition and present the algorithms used in computing the size of such minimal generators / extensions.

We assume that the coupling data of the FCA results is given as a directed graph $D_{\text{FCA}} = (\mathcal{R}, A_{\text{FCA}})$, where $(a, b) \in A_{\text{FCA}}$ if and only if $a \xrightarrow{=0} b$. The results of tFCA are given as a directed graph $D_{\text{tFCA}} = (\mathcal{R}, A_{\text{tFCA}})$, where $(a, b) \in A_{\text{tFCA}}$ if and only if $a \xrightarrow{=0} b$ in T .

Our minimality condition is an extension of transitive reductions [4] and minimal equivalent subgraphs [98]. A *transitive reduction* of a digraph $D = (V, A)$ is defined to be a smallest digraph $D^{\text{min}} = (V, A^{\text{min}})$ that has the same transitive closure as D [4]. Similarly, a *minimal equivalent subgraph* of a digraph $D = (V, A)$ is defined to be a smallest digraph $D^{\text{min}} = (V, A^{\text{min}})$ with $A^{\text{min}} \subseteq A$ that has the same transitive closure and D [98]. We observe that these two notions are equivalent if D is transitively closed.

It is easy to see that if one additional arc (one piece of information) is added, the size of A^{min} will increase by at most 1. Here, we now address the question of how the additional information of a digraph D compared to a smaller digraph D' can be measured and computed.

We write $V^2 := V \times V$ to denote the set of ordered 2-tuples of a set V . Furthermore, we write $D|_X$ for digraphs $D = (V, A)$ to denote the induced subgraph $(X, X^2 \cap A)$. In the following we always assume that D is simple, i.e., that it does not contain loops and parallel edges must have different orientation.

8.5.1 Minimal Extensions

For digraphs $D = (V, A)$, $D' = (V, A')$ with $A' \subseteq A$ we define $f(D, D')$ as the minimum number of arcs from D that have to be added to D' so that the transitive closures become the same. Formally,

$$f(D, D') := \min_{E \subseteq A} \{ |E| : \langle D \rangle = \langle (V, A' \cup E) \rangle \}.$$

We call a minimizer $E \subseteq A$ a *minimal extension*. A set $E \subseteq A$ with $\langle D \rangle = \langle (V, A' \cup E) \rangle$ is called an *extension*.

We observe that for a digraph D , $f(D, \emptyset)$ gives the size of a minimum equivalent subgraph. Computing $f(D, \emptyset)$ is NP-hard [98]. If, however, D is transitively closed, this corresponds to computing the transitive reduction of D , which can be done in polynomial time [4].

We note that in general $f(D, D') \neq f(D, \emptyset) - f(D', \emptyset)$, see Figure 8.3 for an example. There, $D = (\{A, B, C\}, \{1, 2, 3\})$ and $D' = (\{A, B, C\}, \{2, 3\})$. Then $f(D, \emptyset) = 2$ (the set $\{1, 2\}$), $f(D', \emptyset) = 2$ (the set $\{2, 3\}$), but $f(D, D') = 1$ (the set $\{1\}$).

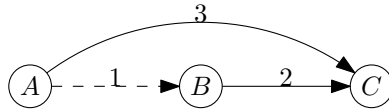


Figure 8.3: Example why $f(D, D') = f(D, \emptyset) - f(D', \emptyset)$ does not always hold. The set D' is drawn with continuous edges, while D also contains the dashed edge.

8.5.2 Computation of Minimal Extensions

We now discuss how to compute $f(D, D')$ for digraphs $D = (V, A)$, $D' = (V, A')$ with $A' \subseteq A$. Let $\mathcal{K} \subseteq 2^V$ be the set of maximal strongly connected components (represented as the corresponding sets of vertices) of D .

We now define digraphs \tilde{D}, \tilde{D}' , where the strongly connected components of D are contracted to single nodes (see Fig. 8.4 for an example):

$$\begin{aligned} (\mathcal{K}, \tilde{A}) &= \tilde{D} := D/\mathcal{K} \\ (\mathcal{K}, \tilde{A}') &= \tilde{D}' := D'/\mathcal{K}, \end{aligned}$$

where for graphs $G = (W, B)$ and partitions \mathcal{P} of W we define

$$G/\mathcal{P} := (\mathcal{P}, \tilde{B}) \quad \text{with}$$

$$\tilde{B} := \{(C_1, C_2) \in \mathcal{P}^2 : \exists(a, b) \in B, a \in C_1, b \in C_2\}.$$

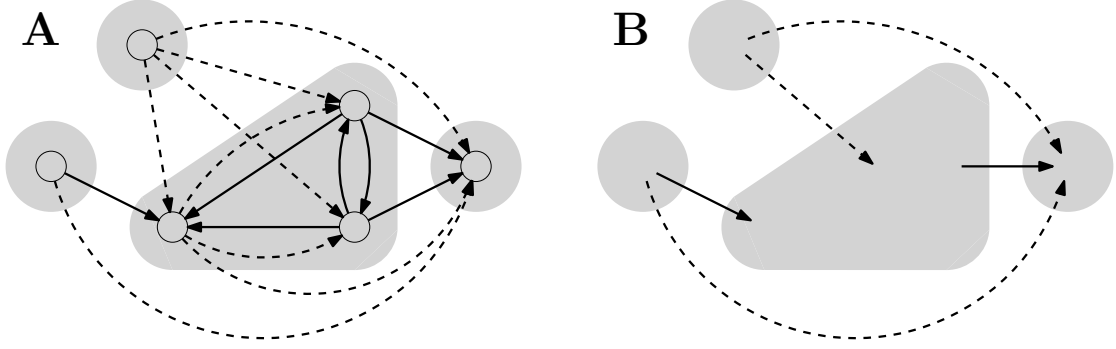


Figure 8.4: **A)** D is the digraph containing both the continuous and the dashed edges, while D' is the subgraph of D that contains only the continuous edges. The individual connected components of \mathcal{K} are shaded in grey. **B)** \tilde{D} is the digraph drawn with continuous and dashed edges, while \tilde{D}' is the one drawn only with continuous edges.

Proposition 8.5.1 \tilde{D} and \tilde{D}' are acyclic.

PROOF If \tilde{D} would contain a cycle, this would contradict the maximality of the components in \mathcal{K} .

Since $\tilde{A}' \subseteq \tilde{A}$, if X is a cycle in \tilde{A}' , then X is also a cycle in \tilde{A} . This contradicts the acyclicity of \tilde{D} proven above. Hence, \tilde{D}' is also acyclic. ■

Proposition 8.5.2 If D is transitively closed, then \tilde{D} is transitively closed.

PROOF Assume $(C_1, C_2) \in \tilde{A}$ and $(C_2, C_3) \in \tilde{A}$. By definition, there exists $a \in C_1, b \in C_2, b' \in C_2, c \in C_3$ with $(a, b) \in A$ and $(b', c) \in A$. Since $D|_{C_2}$ is strongly connected and transitively closed, $b, b' \in C_2$ implies that $(b, b') \in A$. Therefore, there exists a path (a, b, b', c) from a to c , and, from the transitivity of A , there must exist also an edge $(a, c) \in A$. It then follows that $(C_1, C_3) \in \tilde{A}$. ■

Lemma 8.5.1 Let $D = (V, A)$ and $D' = (V, A')$ be digraphs with $A' \subseteq A$ and \mathcal{P} a partition of V . Then it holds for $D'/\mathcal{P} = (\mathcal{P}, \tilde{B}')$ and $E \subseteq A$ that

$$\langle D \rangle = \langle (V, A' \cup E) \rangle \Rightarrow \langle D/\mathcal{P} \rangle = \langle (\mathcal{P}, \tilde{B}' \cup \tilde{E}) \rangle,$$

where $\tilde{E} := \{(C_1, C_2) \in \mathcal{P}^2 : s \in C_1, t \in C_2, (s, t) \in E\}$.

PROOF Let $D/\mathcal{P} = (\mathcal{P}, \tilde{B})$ and $G = (\mathcal{P}, B) := \langle (\mathcal{P}, \tilde{B}' \cup \tilde{E}) \rangle$. Let $e = (C_1, C_2) \in \tilde{B}$ be arbitrary but fixed. Since $\langle D \rangle = \langle (V, A' \cup E) \rangle$ it follows that there exists $s \in C_1, t \in C_2$ with $(s, t) \in A$ and a path $(s = p_1, p_2, \dots, p_n = t)$, $n \geq 2$ using only edges in $A' \cup E$ from s to t . Let $Q := (q_1, q_2, \dots, q_n)$ with q_i being the set of \mathcal{P} that contains p_i , $i = 1, \dots, n$. Note that $q_1 = C_1$ and $q_n = C_2$. It follows that Q is a path in $(\mathcal{P}, \tilde{B}' \cup \tilde{E})$. Hence, $e \in B$ and $\tilde{B} \subseteq B$. Since $\tilde{B}' \cup \tilde{E} \subseteq \tilde{B}$, the lemma follows. ■

Theorem 8.5.1 *It holds that*

$$f(D, D') = f(\tilde{D}, \tilde{D}') + \sum_{C \in \mathcal{K}} f(D|_C, D'|_C).$$

PROOF \geq : Let $E \subseteq A$ minimal s.t. $\langle D \rangle = \langle (V, A' \cup E) \rangle$. Clearly, each edge of E either lies inside a single strongly connected component of D , or connects two different strongly connected components of D .

Claim 8.5.1 $\langle (C, (A' \cup E) \cap C^2) \rangle = \langle D|_C \rangle$ for all $C \in \mathcal{K}$.

PROOF Let B be the edges of the graph $G := \langle (C, (A' \cup E) \cap C^2) \rangle$. Let $e = (s, t) \in A \cap C^2$ arbitrary but fixed. Since $\langle D \rangle = \langle (V, A' \cup E) \rangle$ it follows that there exists a path P that only uses edges in $A' \cup E$ from s to t . Since C is a strongly connected component, it follows that all the edges in P are contained in C . Hence, the path also exists in G and thus, $e \in B$ and $A \cap C^2 \subseteq B$. Since $(A' \cup E) \cap C^2 \subseteq A \cap C^2$, the claim follows. ■

From the Claim 8.5.1 it follows that E is an extension for each connected component $C \in \mathcal{K}$ of size $|E \cap C \times C| \geq f(D|_C, D'|_C)$ and by Lemma 8.5.1 (with $\mathcal{P} = \mathcal{K}$) it also induces a valid extension \tilde{E} for \tilde{D}' to \tilde{D} of size $|\{(s, t) \in E : s \in C_1 \in \mathcal{K}, t \in C_2 \in \mathcal{K}, C_1 \neq C_2\}| \geq |\tilde{E} \cap \mathcal{K} \times \mathcal{K}| \geq f(\tilde{D}, \tilde{D}')$. Since an edge is either inside a connected component or between two connected components, the \geq relation of the formula follows.

\leq : Let $E_C \subseteq A$ be an extension for each connected component $C \in \mathcal{K}$, i.e., $\langle (C, (A' \cap C \times C) \cup E_C) \rangle = \langle D|_C \rangle$. Let $\tilde{E} \subseteq \tilde{A}$ be a valid extension from \tilde{D}' to \tilde{D} , i.e. $\langle \tilde{D} \rangle = \langle (\mathcal{K}, \tilde{A}' \cup \tilde{E}) \rangle$.

We now construct a valid extension for D' to D with $\sum_{C \in \mathcal{K}} |E_C| + |\tilde{E}|$ edges. For each $e = (C_1, C_2) \in \tilde{A}'$ choose an arbitrary but fixed $s(e) \in C_1, t(e) \in C_2$ with $(s(e), t(e)) \in A'$, and for each $e = (C_1, C_2) \in \tilde{E}$ choose an arbitrary but fixed $s(e) \in C_1, t(e) \in C_2$ with $(s(e), t(e)) \in A$. Define $\bar{E} := \{(s(e), t(e)) : e \in \tilde{E}\}$ and $E := \bar{E} \cup \bigcup_{C \in \mathcal{K}} E_C$.

Claim 8.5.2 $\langle D \rangle = \langle (V, A' \cup E) \rangle$

PROOF Let $G = (V, B) := \langle (V, A' \cup E) \rangle$.

Let $s \in C_1 \in \mathcal{K}$, $t \in C_2 \in \mathcal{K}$ with $(s, t) \in A$. By definition of \tilde{E} there exists a path $Q = (C_1 = q_1, q_2, \dots, q_n = C_2)$, $n \geq 1$ using edges in \tilde{A}' and \tilde{E} from C_1 to C_2 . Let $P := (s, s(q_1, q_2), t(q_1, q_2), s(q_2, q_3), t(q_2, q_3), \dots, s(q_{n-1}, q_n), t(q_{n-1}, q_n), t)$. By construction, there exist edges from $s(q_i, q_{i+1})$ to $t(q_i, q_{i+1})$, $i = 1, \dots, n - 1$ in $\tilde{E} \cup A' \subseteq B$.

We observe that $\{s, s(q_1, q_2)\}$, $\{t(q_{i-1}, q_i), s(q_i, q_{i+1})\}$ for $i = 1, \dots, n - 1$, and $\{t(q_{n-1}, q_n), t\}$ are each in the same connected component. Since $E_C \subseteq E$ is a valid extension for each connected component $C \in \mathcal{K}$, there exists for all $a, b \in C$ a path using edges in $A' \cup E_C$ from a to b . Hence, $(a, b) \in B$. Hence, it follows that $(s, s(q_1, q_2)) \in B$, $(t(q_{i-1}, q_i), s(q_i, q_{i+1})) \in B$ for $i = 1, \dots, n - 1$, and $(t(q_{n-1}, q_n), t) \in B$.

It follows that P is a path connecting s to t using only edges in B . Since G is transitively closed, it follows that $(s, t) \in B$. Thus, $B \supseteq A$ and since $A' \cup E \subseteq A$, we have $G = \langle D \rangle$. ■

We observe that $|E| = |\tilde{E}| + \sum_{C \in \mathcal{K}} |E_C|$ and the theorem follows. ■

By Thm. 8.5.1 it follows that we only need to be able to compute $f(D, D')$ for digraphs that are acyclic and for digraphs where D forms one strongly connected component. We first analyze the case of acyclic digraphs. The following proposition characterizes the arcs in a minimal extension. It should be noted that it does not apply for digraphs in general, since in general there exists no unique minimal extension.

Proposition 8.5.3 (Locality) *Assume D, D' are acyclic and let E be minimal s.t. $\langle D \rangle = (V, B) = \langle (V, A' \cup E) \rangle$. Then it holds for $e = (a, b) \in A$ that $e \in E$ if and only if there exists no $c \in V$ with $(a, c) \in B$ and $(c, b) \in B$.*

PROOF \Rightarrow : Assume there exists $(a, c), (c, b) \in B$. Due to the acyclicity of D it follows that there exists no path in A that connects a to c and uses (a, b) and there exists no path in A that connects c to b and uses (a, b) . Thus, there exists no path in $A' \cup E$ that connects a to c or c to b and uses (a, b) . It follows that there exists a path from a to b that does not use (a, b) . Hence, $(a, b) \notin E$.

\Leftarrow : Assume there exists no $c \in V$ with $(a, c) \in B$ and $(c, b) \in B$. Since $\langle D \rangle$ is closed under transitivity, it follows that there exists no path from a to b in A that does not use (a, b) . Hence $(a, b) \in E$. ■

Note that computing the minimal extension for \tilde{D}, \tilde{D}' is equivalent to finding the transitive reduction of \tilde{D} , which has already been shown to be unique in Thm. 1 of [4]. Therefore, we can compute a minimal extension for \tilde{D}, \tilde{D}' in time $O(|\mathcal{K}|^\alpha)$, if two $|\mathcal{K}| \times |\mathcal{K}|$ matrices can be multiplied in time $O(|\mathcal{K}|^\alpha)$ [4].

Alternatively, we observe from the locality property (Prop. 8.5.3) that we can easily compute a minimal extension (hence $f(\tilde{D}, \tilde{D}')$) by computing the length of the longest simple path between each pair of vertices. This is possible in time $O(|\mathcal{K}| \cdot |\tilde{A}|)$ using topological sort. We now only have to show how to compute the size of a valid extension for each of the strongly connected components.

Theorem 8.5.2 (Full extension) *Let $D = (V, A)$, $V \neq \emptyset$ be the complete digraph and $D' = (V = S \dot{\cup} T, A')$ be a bipartite digraph with edges only going from the sources S to the sinks T and assume each vertex is used by at least one edge. Then $f(D, D') = \max\{|S|, |T|\}$.*

PROOF Since each node in S has at least one ingoing arc from the extension and each node in T has at least one outgoing arc from the extension, it can be easily seen that $f(D, D') \geq \max\{|S|, |T|\}$. Hence, we only show the other direction.

Define $F \subseteq A'$ by iteratively removing one edge $e = (s, t)$ and its vertices (including incident edges) from $G = D'$ and adding e to F until it is not possible anymore. Let V' be the vertices used by edges in F . We observe that (V', F) consists of $|F|$ weakly connected components (each a single edge). It follows that there exist $|F|$ edges $E \subseteq (V' \cap T) \times (V' \cap S)$ such that $\langle (V', E \cup F) \rangle$ is one strongly connected component.

We further observe that for each $s \in S$ there exists $t \in V'$ with $(s, t) \in A'$ and for each $t \in T$ there exists $s \in V'$ with $(s, t) \in A'$ (otherwise we could add another edge to F .) It follows that in $(V, A' \cup E)$ we can reach from every $s \in S$ every $v \in V'$, and every $t \in T$ can be reached from every $v \in V'$.

We now iteratively add edges (t, s) to E where $t \in T$ and $s \in S$ are not yet used by any edge in E . We observe that after this operation $|E| = \min\{|S|, |T|\}$ and either only sources with no in-arc or sinks with no out-arc are left. For each of these nodes we now add an edge to E to connect it to a node that has in-arcs and out-arcs. It follows that finally $|E| = \max\{|S|, |T|\}$.

It can be easily seen that now every source can also be reached from at least one sink (and hence from any node) and from each sink we can reach a source and hence, the whole network. ■

Usually, however, the connected components $C \in \mathcal{K}$ do not have the form as in Thm. 8.5.2, i.e., $D|_C$ is not bipartite. Hence, we have to bring them into that form first. As a first step, we show that for acyclic D' it is sufficient to look at sources and sinks (see Fig. 8.5 B and C).

Corollary 8.5.1 *Let $D = (V, A)$, $V \neq \emptyset$ be a complete digraph and D' be a directed acyclic graph. Then*

$$\begin{aligned} f(D, D') &= 0 && \text{if } |V| = 1 \\ f(D, D') &= \max\{|S|, |T|\}, && \text{otherwise} \end{aligned}$$

where S is the set of sources (nodes without in-arc) and T is the set of sinks (nodes without out-arc) of D' .

PROOF If $|V| = 1$, we obviously do not have to add any arcs.

We observe that in D' every node that is not a source can be reached from a source, and from each node that is not a sink we can reach a sink. It follows that for each $E \subseteq A$ where every source is reachable from every sink and vice versa in $(V, A' \cup E)$ it holds that $(V, A' \cup E)$ is strongly connected. Hence, we can ignore nodes that are neither sources nor sinks (see Fig. 8.5 C).

Since D' contains more than one node, it follows that we have to add one in-arc and one out-arc for each isolated node (which is sink and source at the same time) in D' . Hence, we can split up each isolated node into a sink and a source without having to invest more edges to build a valid extension (see Fig. 8.5 D). Finally, Thm. 8.5.2 applied to D' yields $f(D, D') = \max\{|S|, |T|\}$. ■

To deal with strongly connected components \mathcal{K}' of D' , we contract them to single nodes (see Fig. 8.5 A and B) and reconstruct an extension in the component graph D'/\mathcal{K}' to D' (see Fig. 8.5 E and F).

Lemma 8.5.2 *Let $D = (V, A)$, $V \neq \emptyset$ be a complete digraph, $D' = (V, A')$ be a digraph, and $\bar{D}' = (\mathcal{K}', \bar{A}') := D'/\mathcal{K}'$ be the component graph of D' , where \mathcal{K}' are the strongly connected components of D' . Then $E \subseteq A$ is an extension of D' to D if and only if*

$$\bar{E} := \{(C_1, C_2) \in \mathcal{K}'^2 : a \in C_1, b \in C_2, (a, b) \in E\}$$

is an extension of \bar{D}' to the complete digraph on \mathcal{K}' .

PROOF \Rightarrow : Follows directly from Lemma 8.5.1 with $\mathcal{P} = \mathcal{K}'$.

\Leftarrow : Let $G = (V, B) := \langle (V, A' \cup E) \rangle$. By construction of \bar{A}' , we can choose for each $e = (C_1, C_2) \in \bar{A}'$ arbitrary but fixed $s(e) \in C_1, t(e) \in C_2$ with $(s(e), t(e)) \in A'$. Similarly, by construction of \bar{E} , we can choose for each $e = (C_1, C_2) \in \bar{E}$ arbitrary but fixed $s(e) \in C_1, t(e) \in C_2$ with $(s(e), t(e)) \in E$.

Let $s \in C_1 \in \mathcal{K}', t \in C_2 \in \mathcal{K}'$ with $(s, t) \in A$. Since \bar{E} is an extension, there exists a path $Q = (C_1 = q_1, q_2, \dots, q_n = C_2)$, $n \geq 1$ using edges in \bar{A}' and \bar{E} from C_1 to C_2 . Let $P := (s, s(q_1, q_2), t(q_1, q_2), s(q_2, q_3), t(q_2, q_3), \dots, s(q_{n-1}, q_n), t(q_{n-1}, q_n), t)$. By construction there exist edges from $s(q_i, q_{i+1})$ to $t(q_i, q_{i+1})$, $i = 1, \dots, n - 1$ in $E \cup A' \subseteq B$.

We observe that $\{s, s(q_1, q_2)\}, \{t(q_{i-1}, q_i), s(q_i, q_{i+1})\}$ for $i = 1, \dots, n - 1$, and $\{t(q_{n-1}, q_n), t\}$ are each in the same strongly connected component of D' . Hence, it follows that $(s, s(q_1, q_2)) \in B$, $(t(q_{i-1}, q_i), s(q_i, q_{i+1})) \in B$ for $i = 1, \dots, n - 1$, and $(t(q_{n-1}, q_n), t) \in B$.

It follows that P is a path connecting s to t using only edges in B . Since G is transitively closed, it follows that $(s, t) \in B$. Thus, $B \supseteq A$ and since $A' \cup E \subseteq A$, we have $G = \langle D \rangle$. ■

Theorem 8.5.3 Let $D = (V, A)$, $V \neq \emptyset$ be a complete digraph, $D' = (V, A')$ be a digraph, and \bar{D}' be the component graph of D' . Then

$$\begin{aligned} f(D, D') &= 0 && \text{if } \langle D' \rangle = D \\ f(D, D') &= \max\{|S|, |T|\}, && \text{otherwise} \end{aligned}$$

where S is the set of sources (nodes without in-arc) in \bar{D}' and T is the set of sinks (nodes without out-arc) in \bar{D}' .

PROOF Let \bar{D} be the complete digraph on \mathcal{K}' , where \mathcal{K}' is the set of strongly connected components of D' . By Lemma 8.5.2, we have $f(D, D') = f(\bar{D}, \bar{D}')$. Since \bar{D}' is a directed acyclic graph, the result follows by Corollary 8.5.1. ■

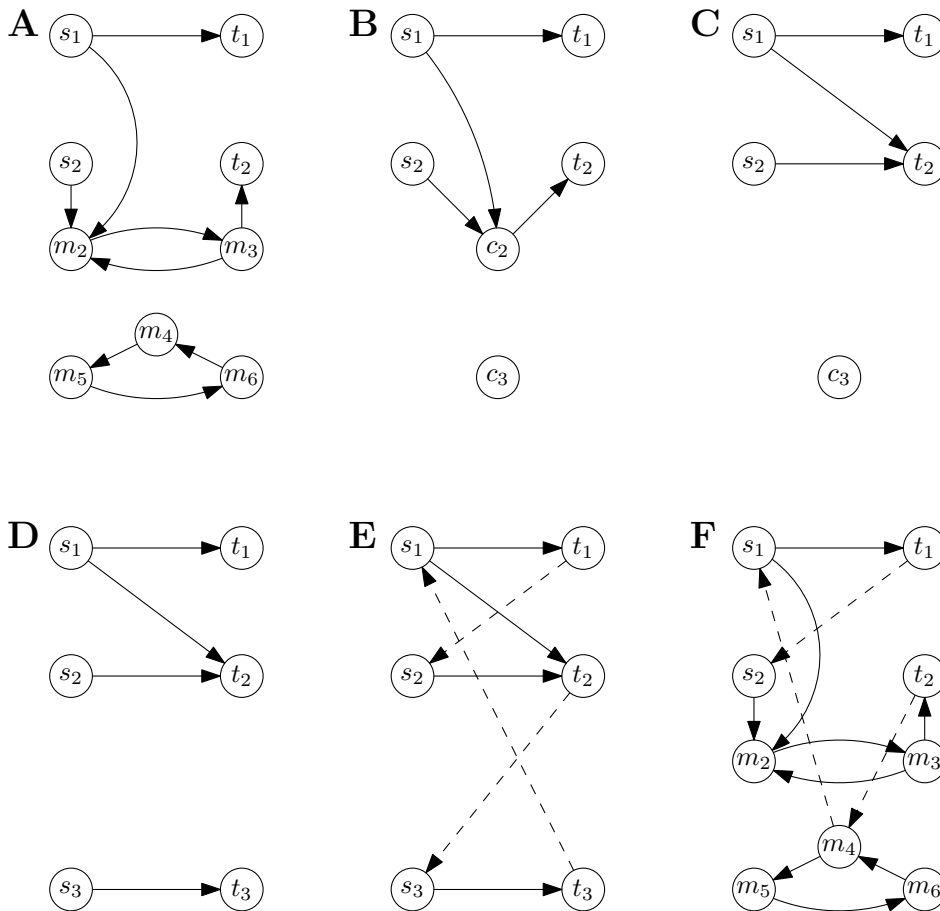


Figure 8.5: **A)** An example for D' . **B)** Compressed strongly connected components. **C)** Ignored nodes that are neither sources nor sinks. **D)** Splitted isolated nodes. **E)** Extension of D' (in dashed edges). **F)** Extension in the original digraph.

Note that in Thm. 8.5.1 and Prop. 8.5.3 we do not require that D is transitively closed. This property is however important for Thm. 8.5.2 because otherwise a minimal exten-

sion cannot be computed in polynomial time. This can be seen by studying $f(D, \emptyset)$, where $\langle D \rangle$ is the complete digraph. This corresponds to finding a Hamiltonian cycle in D [4, 98].

8.6 Discussion

8.6.1 Theoretical Differences

8.6.1.1 Old Couplings are Retained

If $r \xrightarrow{=0}_F s$, it also follows that $r \xrightarrow{=0}_T s$, since we cannot turn an infeasible system feasible by adding constraints. Thus, it follows that if two reactions are partially coupled, they are also partially thermodynamically coupled.

8.6.1.2 New Partial Couplings

In Fig. 8.2 we see an example where $r \xrightarrow{=0}_F s$, but $s \not\xrightarrow{=0}_F r$ and thus r is not partially coupled to s in F . However, $s \xrightarrow{=0}_T r$, hence r is partially thermodynamically coupled to s . Further examples are the pairwise uncoupled reactions a, b, c, d . With thermodynamic constraints however, a and c are partially coupled and b is partially coupled to d in T . In particular, we can deduce $v_a = v_c$ and $v_b = v_d$ for every thermodynamically feasible flux vector.

8.6.1.3 New Directional Couplings

In Fig. 8.6 we see an example where reactions a, b are uncoupled (because of flux vectors with supports $\{a, c, d\}$ and $\{b, d\}$). But, a and b are directionally thermodynamically coupled ($a \xrightarrow{=0}_T b$) since flux only through the cycle $\{b, c\}$ is thermodynamically infeasible.

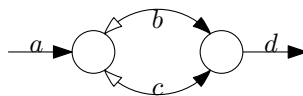


Figure 8.6: Example with uncoupled reactions that are thermodynamically coupled. The white arrow-heads indicate that the reactions are reversible.

8.6.2 Practical Differences

We ran the analysis on several genome-scale reconstructions from the BiGG-database [136]. A statistical overview on the results in *H. pylori* iTT341 can be seen in Fig. 8.7. The types of couplings are depicted in a set-diagram style:

- The set of reactions already blocked without thermodynamic constraints (blocked by FCA) is contained in the box of reactions blocked with thermodynamic constraints (blocked by tFCA). The number of reactions that are only blocked due to thermodynamic constraints is marked on the set difference. In the case of *H. pylori* one can observe that there are 6 reactions that are not blocked by steady-state constraints alone but blocked together with thermodynamic constraints.
- Since there are more reactions blocked with thermodynamic constraints than without, some of the coupled reactions that we found by ordinary FCA contain reactions that are blocked with thermodynamic constraints. This is why the set of reactions coupled by FCA intersects the set of thermodynamically blocked reactions and the set of thermodynamically coupled reactions. For both intersections we wrote down how many pairs of reactions fall into the respective category.
- In the set difference of the thermodynamically coupled reactions and the ordinarily coupled reactions we included the number of coupling pairs that fell into this category.

In the case of *H. pylori* we see that the couplings of FCA can be represented by 516 couplings of which 6 are actually blocked by thermodynamic constraints. Thermodynamic constraints give additional information on 10 couplings. Thermodynamic constraints also merged 6 groups of partially coupled reactions to 2 groups.

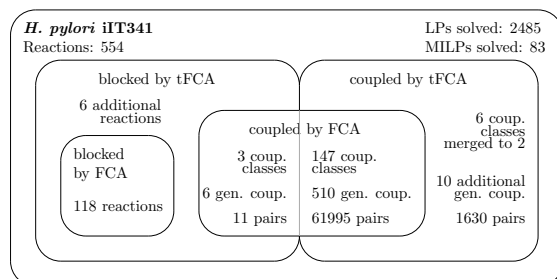


Figure 8.7: Results for *H. pylori* iT341. We found that *homoserine O-trans-acetylase* (HSERTA), *O-Acetyl-L-homoserine succinate-lyase (adding cysteine)* (METB1r) and *O-succinylhomoserine lyase* (SHSL1r, SHSL2r) are not only necessary for biomass production (as computed by ordinary FCA) but cannot work without this function. Together with the biomass reaction and all its partially coupled reactions they form one of the new coupling classes. Furthermore, the under tFCA partially coupled (but not fully coupled) SHSL2r and HSERTA were originally (in FCA) uncoupled.

We obtained that on all of these networks, we were able to find additional blocked and coupled reactions using tFCA. However, the influence of thermodynamic constraints depends heavily on the network (see Tab. 8.1). In average we found around 1% – 6% of additional couplings (in the minimal generator). In general however, it should be noted that the new directional couplings of previously uncoupled reactions were found most

often. On the other extreme, in only one case we found that previously uncoupled reactions became partially coupled. We also observed that the reaction *cystathionine g-lyase* (CYSTGL) was part of new directional couplings in *M. tuberculosis*, while in *S. cerevisiae* thermodynamic constraints blocked the reaction.

In the case of *H. pylori* iIT341 we observe that many new couplings are found because reactions are part of the main biomass production pathway, i.e., if one of the reactions in the pathway is blocked, the pathway becomes unfunctional and this induces couplings. If a reaction of the pathway, however, is part of an internal cycle, ordinary FCA allowed to operate the reaction via the cycle. This way ordinary FCA did not detect that the reaction is only part of the pathway.

The reactions *homoserine O-trans-acetylase* (HSERTA), *O-Acetyl-L-homoserine succinate-lyase (adding cysteine)* (METB1r) and *O-succinylhomoserine lyase* (SHSL1r, SHSL2r) in *H. pylori* iIT341 are an example for this effect.

Table 8.1: Comparison of results for different genome-scale networks.

| Model | | no. blocked reactions | generating couplings | runtime |
|-------------------------------|-----------|-----------------------|----------------------|---------|
| <i>E. coli</i> iAF1260 | FCA | 839 | 2101 | 36.35 s |
| | tFCA | 848 | 2128 | 47.74 s |
| | extension | 9 | 49 | |
| <i>S. cerevisiae</i> iND750 | FCA | 635 | 885 | 7.73 s |
| | tFCA | 640 | 935 | 12.43 s |
| | extension | 5 | 58 | |
| <i>M. tuberculosis</i> iNJ661 | FCA | 281 | 831 | 6.08 s |
| | tFCA | 287 | 834 | 9.46 s |
| | extension | 6 | 9 | |
| <i>S. aureus</i> iSB619 | FCA | 278 | 544 | 3.02 s |
| | tFCA | 279 | 546 | 3.95 s |
| | extension | 1 | 3 | |
| <i>H. pylori</i> iIT341 | FCA | 118 | 516 | 2.2 s |
| | tFCA | 124 | 515 | 5.05 s |
| | extension | 6 | 10 | |

The number of computed couplings is measured using the minimal number of couplings from which all couplings can be inferred by transitive closure (see Sec. 8.5).

We also observe (Tab. 8.1) that our new tFCA algorithm runs only slightly slower than the FCA algorithm introduced in [58]. This is caused by the pre-processing step that runs regular FCA. This way we already know many couplings (see Sec. 8.6.1.1) and many witnesses are found that prove some reactions to be also thermodynamically unblocked, or uncoupled. Only for those few cases, where we cannot already deduce any information from the previous run, we start the MILP solver.

8.7 Conclusion

We presented a refined version of FCA that finds more coupled reactions than ordinary FCA. Although thermodynamic constraints were used that are usually NP-hard, it was possible to also analyze genome-scale networks like *E. coli* iAF1260 in a few seconds. We observed that thermodynamic constraints do not only give additional blocked reactions but also additional coupled reactions. The impact however depends highly on the network that is analyzed.

We also observed in Thm. 8.2.1 that the presented approach is not only applicable to loop-law thermodynamic constraints but to any kind of restrictions of the flux space. Extensions of this method to also include concentration information or other constraints are straight-forward.

Chapter 9

Sampling the Thermodynamically Constrained Flux Space

Abstract Sampling methods have proven to be a very efficient and intuitive method to understand genome-scale metabolic networks. These methods are used to detect properties of the flux space by randomly computing feasible flux vectors.

Here we consider the problem of sampling thermodynamically feasible fluxes and focus on the variability of the fluxes through reactions. We show that there exists no polynomial time sampling algorithm that leads to reliable predictions for the flux variability of reactions unless $\mathbf{NP} = \mathbf{RP}$.

Methods based on sampling the thermodynamically constrained flux space are likely to be unreliable. This is illustrated on the example of flux correlation analysis.

9.1 Background

Sampling methods [135, 72] have been used in many practical cases [158, 17, 94, 6] to study the steady-state flux space, when deterministic methods like enumeration of elementary flux modes [141] or extreme pathways [138] fail due to the size of the networks. To study the set of solutions, sampling methods compute a random sample set of feasible solutions. Usually, a variant of the *Markov Chain Monte Carlo* (MCMC) method called *Artificial Centering Hit and Run* (ACHR) is used to compute the samples. ACHR seems to perform well for flux polytopes in practice [135], although convergence to a uniform distribution has not been proven as it has been done for MCMC [74]. Using statistical methods, we can then use this sample set to derive properties of the flux space itself. Typical properties that are analyzed are correlations between fluxes through different reactions [121, 132, 75, 169], or the distribution of flux rates through a given reaction [134].

This information can then be used to further constrain the solution space or to design

experiments that are likely to yield a high amount of new information.

Thermodynamic constraints are an additional source of constraints that have been used in the analysis of metabolic networks [10, 11, 43, 46, 71, 82, 83, 69, 64, 148] and were also used in sampling methods [134, 123, 25]. However, thermodynamic constraints are mathematically difficult, since they are non-convex. Here, we define the concept of *non-trivial polynomial time sampling algorithm* and show how it can be used to solve decision problems in randomized polynomial time. The complexity class of such problems is called **RP** [119]. We show that unless $\mathbf{NP} = \mathbf{RP}$, which is one of the major open problems in theoretical computer science, there exists no polynomial time method for sampling the thermodynamically constrained flux space without the risk of observing artifacts that do not exist in reality. In particular, we will study the artifact that all samples have zero flux through a given reaction, although a non-zero flux through the reaction is possible. We show that this indeed is likely to happen for any kind of sampling method on some networks. We discuss the consequences for other sampling-based methods like correlation analysis.

9.2 Theoretical Obstructions to Sampling

Let $\text{PROB} : \mathcal{I} \rightarrow \{0, 1\}$ be an **NP**-hard decision problem on a set \mathcal{I} of inputs (commonly we use the set of words over the alphabet $\{0, 1\}$ as input, i.e., $\mathcal{I} = \{0, 1\}^*$ and the length of an input is just the length of the word). To solve PROB by sampling, we require that the structure of the sampling space represents PROB in a certain way:

Definition 9.2.1 (Sampling space) *Given a decision problem $\text{PROB} : \mathcal{I} \rightarrow \{0, 1\}$, we call (\mathcal{X}, f) a sampling space for $\mathcal{X} : \mathcal{I} \rightarrow \mathbb{R}^n$ and $f : \mathcal{I} \times \mathbb{R}^n \rightarrow \mathbb{R}$ if $f(I, x)$ is continuous in x for all $x \in \mathbb{R}^n$, f can be computed in time polynomial in the encoding length of I and x , and*

$$\text{PROB}(I) = \begin{cases} 1 & \exists x \in \mathcal{X}(I) : f(I, x) > 0 \\ 0 & \text{otherwise.} \end{cases}$$

We observe that we can formulate the **THERMOFLUX** problem (Prob. 3.1.1) in such a form. There, a problem instance I encodes a metabolic network $(\mathcal{M}, \mathcal{R} = \mathcal{I} \dot{\cup} \mathcal{E}, S)$ and a target reaction $r(I) \in \mathcal{R}$. It follows that (\mathcal{X}, f) with $\mathcal{X}(I) := T(I)$ and $f(I, x) := \text{pr}_{r(I)}(x)$ for all $I \in \mathcal{I}$ is a sampling space, where $T(I)$ is the thermodynamically constrained flux space (Def. 2.6.4 with $Q = \mathbb{R}^{\mathcal{M}}$):

$$T(I) = \{v \in \mathbb{R}^{\mathcal{R}} : Sv = 0, v_{\text{Irrev}} \geq 0, v \text{ thermo. feas.}\}$$

Note, that this construction works with all **NP**-hard problems from Chapter 3.

Let (Ω, \mathcal{F}, P) be a probability space. It will serve us as the space from which we draw the seeds for the sampling algorithm. Here, we assume that the sampling method is

given as a function $\mathcal{S} : \mathcal{I} \times \mathbb{N} \times \Omega \rightarrow \mathbb{R}^n$, i.e., for every time point we get a sample. With this formalism we want to capture the behaviour of ACHR sampling methods that do a random walk through $\mathcal{X}(I)$ and can be run for arbitrarily long times to improve the sampling result. Classical sampling algorithms can also be captured by this formalism by iteratively running the sampling method and computing a consensus value. If the sampling algorithm did not produce a result for an (early) time point, it could simply return a default value. Since we will only consider asymptotic behavior, this will not be of any importance.

Definition 9.2.2 (Feasible Sampling Algorithm) $\mathcal{S} : \mathcal{I} \times \mathbb{N} \times \Omega \rightarrow \mathbb{R}^n$ is a feasible sampling algorithm, if there exists a polynomial $p : \mathbb{N} \rightarrow \mathbb{R}$ such that

$$\mathcal{S}(I, k, \omega) \in \mathcal{X}(I) \quad \forall k \geq p(|I|), I \in \mathcal{I}, \omega \in \Omega$$

Definition 9.2.3 (Polynomial Time Sampling Algorithm) $\mathcal{S} : \mathcal{I} \times \mathbb{N} \times \Omega \rightarrow \mathbb{R}^n$ is a polynomial time sampling algorithm if there exists a polynomial $q : \mathbb{N} \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ and for every $I \in \mathcal{I}$ a random variable $X : \Omega \rightarrow \mathbb{R}^n$ such that

- $\mathcal{S}(I, k, \omega)$ for $I \in \mathcal{I}$ and $\omega \in \Omega$ can be computed in time $O(k)$,
- $\mathcal{S}(I, k, \cdot) \rightarrow X$ in distribution for $k \rightarrow \infty$, and
- $\mathcal{S}(I, k, \cdot)$ converges to X in polynomial time, i.e., for every closed set $A \subseteq \mathbb{R}^n$ holds $|P(\mathcal{S}(I, k, \cdot) \in A) - P(X \in A)| < \varepsilon$ for all $k > q(|I|, \varepsilon^{-1})$.

Assume there exists such a sampling method $\mathcal{S} : \mathcal{I} \times \mathbb{N} \rightarrow \mathbb{R}^n$ that samples the feasibility space $\mathcal{X}(I)$ of the **NP**-hard optimization problem **PROB** for each given instance $I \in \mathcal{I}$ in a non-trivial way, i.e., without losing any features (represented by f):

Definition 9.2.4 (Non-trivial Sampling Algorithm) $\mathcal{S} : \mathcal{I} \times \mathbb{N} \times \Omega \rightarrow \mathbb{R}^n$ is a non-trivial sampling algorithm w.r.t. $f : \mathcal{I} \times \mathbb{R}^n \rightarrow \mathbb{R}$ if for every $I \in \mathcal{I}$ there exists a random variable $X : \Omega \rightarrow \mathbb{R}^n$ such that

- $\mathcal{S}(I, k, \cdot) \rightarrow X$ in distribution for $k \rightarrow \infty$.
- If $\exists x \in \mathcal{X}(I)$ with $f(I, x) > 0$, then $P(f(I, X) \leq 0) = t < 1$ (with $\frac{1}{1-t} \leq p(|I|)$ for a polynomial p).

We can then use \mathcal{S} to construct a probabilistic algorithm that will decide **PROB**. The probabilistic algorithm that we are going to construct will belong to the class **RP** [119].

Definition 9.2.5 (Complexity Class RP) A decision problem p is in **RP** if there exists a probabilistic algorithm that

- runs in polynomial time,

- if the answer to p is NO, it outputs NO, and
- if the answer to p is YES, it outputs YES with probability at least $\frac{1}{2}$. \square

Since $\mathbf{RP} = \mathbf{NP}$ is an open problem in theoretical computer science, it is very unlikely that a given probabilistic polynomial-time sampling algorithm of the thermodynamically constrained flux space actually solves the $\mathbf{RP} = \mathbf{NP}$ problem. Hence, it is much more likely that the sampling algorithm samples the feasible flux space incompletely.

Theorem 9.2.1 *Let $\text{PROB} : \mathfrak{J} \rightarrow \{0, 1\}$ be an \mathbf{NP} -hard decision problem with sampling space (\mathcal{X}, f) . Unless $\mathbf{RP} = \mathbf{NP}$, there exists no feasible, non-trivial, polynomial time sampling algorithm $\mathcal{S} : \mathfrak{J} \times \mathbb{N} \times \Omega \rightarrow \mathbb{R}^n$. \square*

PROOF Assume there exists such a sampling algorithm. We construct an algorithm in \mathbf{RP} for PROB .

For $I \in \mathfrak{J}$ define $t(I) := P(X \in A(I))$ for $A(I) := \{x \in \mathbb{R}^n : f(I, x) \leq 0\}$. Since $f(I, \cdot)$ is continuous, it follows that $A(I)$ is closed and measurable. Hence, $t(I)$ is well defined.

By Def. 9.2.2 and Def. 9.2.3 there exist polynomials $k_0 : \mathbb{N} \rightarrow \mathbb{R}^+$, $q : \mathbb{N} \times \mathbb{R}^+ \rightarrow \mathbb{R}^+$ with

$$\mathcal{S}(I, k, \omega) \in \mathcal{X}(I) \quad \forall k \geq k_0(|I|), I \in \mathfrak{J}, \omega \in \Omega. \quad (9.1)$$

$$P(\mathcal{S}(I, k, \cdot) \in A(I)) - P(X \in A(I)) < \varepsilon \quad \forall k > q(|I|, \varepsilon^{-1}) \quad (\text{since } A \text{ is closed}) \quad (9.2)$$

We assume w.l.o.g. that $q(m, \varepsilon) \geq k_0(m)$ for all $m \in \mathbb{N}, \varepsilon \in \mathbb{R}^+$.

Algorithm 13 Probabilistic Algorithm for PROB . k_0 is the polynomial from Def. 9.2.2 and q is the polynomial from Def. 9.2.3.

```

 $k = \max\{q(|I|, \frac{2}{1-t}), k_0(I)\}$ 
choose random  $\omega \in \Omega$ 
compute a sample  $X_k := \mathcal{S}(I, k, \omega)$ 
if  $f(I, X_k) \leq 0$  then
    return NO
else
    return YES
end if

```

Claim 9.2.1 *For a given input $I \in \mathfrak{J}$ and $t \geq t(I)$ Algorithm 13 returns NO with probability at most $\frac{t+1}{2}$ although $\text{PROB}(I) = 1$ and it always returns NO if $\text{PROB}(I) = 0$. \square*

PROOF Case: There exists a $x \in \mathcal{X}(I)$ with $f(x) > 0$: It follows by (9.2) that

$$P(f(\mathcal{S}(I, k, \cdot)) \leq 0) < t(I) + \varepsilon \leq t + \varepsilon \quad \forall k > q(|I|, \varepsilon^{-1}).$$

By choosing $\varepsilon = \frac{1-t}{2}$, we obtain $P(\mathcal{S}(I, k, \cdot) \in A) < \frac{t+1}{2}$. Thus, Alg. 13 will return NO although the correct answer is YES with probability at most $\frac{t+1}{2}$.

Case $f(x) \leq 0$ for all $x \in \mathcal{X}(I)$: It follows that $f(\mathcal{S}(I, k, \omega)) \leq 0$ for all $\omega \in \Omega, k \geq k_0(I)$ by Def. 9.2.2. Hence, the answer of the algorithm will always be NO, if the correct answer is NO. \blacksquare

To prove that the problem would be in **RP**, we still have to increase the probability of YES in the positive case. This can be done by re-running the algorithm.

By Def. 9.2.4 there exists a polynomial p with $\frac{1}{1-t(I)} \leq p(|I|)$. We choose $t := \frac{p(|I|)-1}{p(|I|)}$ and it follows that $\frac{1}{1-t} = p(|I|)$ and $t(I) \leq t$. Hence, we can apply Claim 9.2.1 without having to know $t(I)$.

By construction of Alg. 13 the computation of X_k takes time $O(q(|I|, \frac{2}{1-t}))$. We observe that the encoding for the computed sample X_k is bounded by the computation time $O(q(|I|, \frac{2}{1-t}))$. Hence, by Def. 9.2.1 there exists a polynomial g such that the runtime of Alg. 13 is bounded by $O\left(g\left(|I|, q\left(|I|, \frac{2}{1-t}\right)\right)\right)$.

To obtain a correct result if the correct answer is YES with probability at least $\frac{1}{2}$, we re-run the algorithm at least $\frac{1}{\log_2\left(\frac{2}{t+1}\right)}$ times with independent choice of $\omega \in \Omega$ for each run and return YES if one of the runs returned yes.

Since the probability of NO in one run is at most $\frac{t+1}{2}$, it follows that the probability for NO in all runs is at most

$$\left(\frac{t+1}{2}\right)^{\frac{1}{\log_2\left(\frac{2}{t+1}\right)}} = 2^{\frac{\log_2\left(\frac{t+1}{2}\right)}{\log_2\left(\frac{2}{t+1}\right)}} = 2^{-\frac{\log_2\left(\frac{2}{t+1}\right)}{\log_2\left(\frac{2}{t+1}\right)}} = \frac{1}{2}.$$

We can estimate the number of iterations by observing that

$$\begin{aligned} t &= \frac{p(|I|) - 1}{p(|I|)} \\ \Rightarrow \frac{2}{t+1} &= \frac{2}{\frac{p(|I|)-1}{p(|I|)} + 1} = \frac{2p(|I|)}{2p(|I|) - 1} \\ \Rightarrow \frac{1}{\log_2\left(\frac{2}{t+1}\right)} &= \frac{1}{\log_2\left(\frac{2p(|I|)}{2p(|I|)-1}\right)}. \end{aligned}$$

Using the Theorem of l'Hopital we have

$$\lim_{p \rightarrow \infty} \frac{p^{-1}}{\ln\left(\frac{p}{p-1}\right)} = \lim_{p \rightarrow \infty} \frac{p^{-1}}{\ln p - \ln(p-1)} = \lim_{p \rightarrow \infty} \frac{-p^{-2}}{\frac{1}{p} - \frac{1}{p-1}} = \lim_{p \rightarrow \infty} \frac{-p^{-2}}{\frac{p-1-p}{p(p-1)}} = \lim_{p \rightarrow \infty} p^{-2}(p^2 - p) = 1$$

Hence, we can bound the number of iterations by

$$\frac{1}{\log_2\left(\frac{2}{t+1}\right)} = \frac{1}{\log_2\left(\frac{2p(|I|)}{2p(|I|)-1}\right)} = O(p(|I|)).$$

Thus, we get a YES if the correct answer is YES with probability at least $\frac{1}{2}$ after a running time of

$$O\left(g\left(|I|, q\left(|I|, \frac{2}{1-t}\right)\right)\right) \frac{1}{\log_2\left(\frac{2}{t+1}\right)} \leq O\left(g\left(|I|, q(|I|, 2p(|I|))\right)p(|I|)\right).$$

We have shown under the assumption of the existence of a sampling algorithm with the given properties that PROB is in **RP**. Since PROB is also **NP**-hard, the existence of such a sampling algorithm implies **RP** = **NP**. Hence, no such sampling algorithm can exist if **RP** \neq **NP**. ■

9.3 Practical Obstructions to Sampling

9.3.1 Method

To verify the impact of the theoretical results, I implemented the following experiment to analyze the difference between sampling with thermodynamic constraints and without thermodynamic constraints. For a given metabolic network with flux space P (with or without thermodynamic constraints) I do the following:

1. Sample n points in the flux space P .
2. Run flux variability analysis (FVA) on P .
3. Define 4 sets of reactions
 - \mathcal{R}^C := reactions that are contained in an internal cycle (compute by blocking all exchange reactions and performing ordinary FVA on the remaining reactions),
 - \mathcal{R}^{NC} := $\mathcal{R} \setminus \mathcal{R}^C$ reactions that are not contained in an internal cycle,
 - \mathcal{R}_+ := reactions that can have positive flux (obtained from FVA),
 - \mathcal{R}_- := reactions that can have negative flux (obtained from FVA).
4. From these 4 sets of reactions, I define the following 4 sampling classes:
 - \mathcal{R}_+^C := $\mathcal{R}^C \cap \mathcal{R}_+$
 - \mathcal{R}_+^{NC} := $\mathcal{R}^{NC} \cap \mathcal{R}_+$
 - \mathcal{R}_-^C := $\mathcal{R}^C \cap \mathcal{R}_-$
 - \mathcal{R}_-^{NC} := $\mathcal{R}^{NC} \cap \mathcal{R}_-$
5. For each sampling class $A \subseteq \mathcal{R}$, I count the number of reactions for which I never sampled negative flux n_A^P and I compute the ratio $r_A^P := \frac{n_A}{|A|}$.

I do this for the steady-state flux space F (without thermodynamic constraints) and for the thermodynamically constrained flux space T . Since positive lower bounds and negative upper bounds for reactions in internal cycles make it already **NP**-hard to find a thermodynamically feasible flux distribution, I set all positive lower bounds and all negative upper bounds to 0.

For the sampling method I chose to use the method implemented in the COBRA toolbox [137], since it is one of the most advanced algorithms (ACHR) for sampling flux spaces. They also offer a flag to activate thermodynamic constraints. Unfortunately this flag has no effect in the current version (2.0.5). Hence, I implemented a simple post-processing step to turn thermodynamically infeasible fluxes into thermodynamically feasible fluxes by deleting internal cycles. By Thm. 4.5.1 this is possible in polynomial time. I decided not to implement the method by Schellenberger et al. [134], since it solves an MILP in the post-processing step. Note that because I selected the sampling method for the thermodynamically constrained flux space, the experimental result may not be statistically fair.

I selected a set of genome-scale metabolic networks based from the BiGG-database [136] as my test-set, since these networks are well curated. I did not select *Human Recon 1.*, since I am not able to run thermodynamically constrained flux variability analysis on it. Instead, I also added the more recent *E. coli* iJO1366 network [113]. For the *M. barkeri* network the `reduceModel` subset of the sampling algorithm from the COBRA toolbox did not work for me. Hence, I also excluded this network and ended up with the following test-networks:

- *E. coli* iAF1260
- *E. coli* iJO1366
- *H. pylori* iT341
- *M. tuberculosis* iNJ661
- *S. aureus* iSB619
- *S. cerevisiae* iND750

From our theoretical analysis we would expect the following results:

- With thermodynamic constraints more reactions (in internal cycles) are always sampled to have zero flux than when sampled without thermodynamic constraints, i.e., $r_{\mathcal{R}_+^C}^T > r_{\mathcal{R}_+^C}^F$ and $r_{\mathcal{R}_-^C}^T > r_{\mathcal{R}_-^C}^F$. Note that I compare the ratios because the number of reactions that can have positive/negative flux is different for T and F .
- When we look at the reactions that are not in internal cycles, we do not expect any differences, i.e., $r_{\mathcal{R}_+^{NC}}^T = r_{\mathcal{R}_+^{NC}}^F$ and $r_{\mathcal{R}_-^{NC}}^T = r_{\mathcal{R}_-^{NC}}^F$. Actually, this property

is already guaranteed by the experimental setup (the cycle subtraction step only manipulates fluxes through reactions in internal cycles).

- Since I do not deal with reactions in internal cycles in a special way without thermodynamic constraints, I expect that the ratio for reactions in cycles and not in cycles should be the same for the flux space F , i.e., $r_{\mathcal{R}_+^C}^F = r_{\mathcal{R}_+^{NC}}^F$ and $r_{\mathcal{R}_-^C}^F = r_{\mathcal{R}_-^{NC}}^F$.

9.3.2 Results

The computed results for 2000 and 10000 samples can be seen in Figs. 9.1, 9.2.

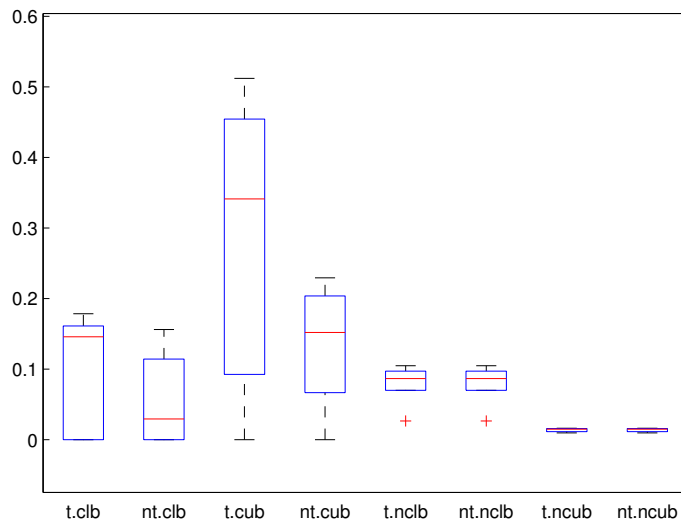


Figure 9.1: Sampling results with 2000 sample points. The y -axis shows the ratio for how many reactions that can have positive/negative flux the sampling method actually sampled such a flux at least once. A value of 0.1 for $t.clb$ for example means that for every tenth reaction that is contained in a cycle and can have negative flux with thermodynamic constraints no negative flux was sampled. For each sampling class we get several data points, because we ran the analysis on 6 genome-scale networks. The naming of the sampling classes is according to the following scheme: t denotes that thermodynamic constraints were used, nt denotes that no thermodynamic constraints were used. We use c to denote that the sampling class consists of the reactions in cycles and nc to denote that the sampling class consists of reactions not in cycles. Finally, lb means that the lower bound is negative and correspondingly ub that the upper bound is positive.

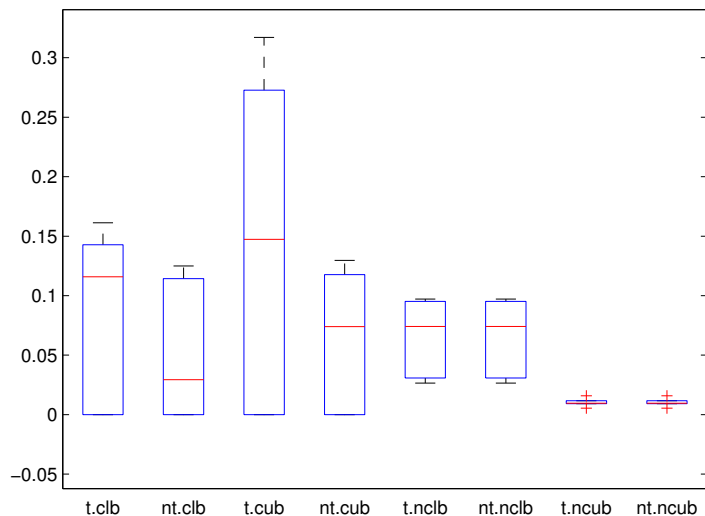


Figure 9.2: Sampling results with 10000 sample points. The labeling is the same as in Fig. 9.1.

We observe that indeed, the ratio of reactions where no positive/negative fluxes were sampled is larger for the case with thermodynamic constraints than without. Also, as expected, the results for \mathcal{R}^{NC} are the same. Moreover, the ratios in the non-thermodynamic case for reactions in internal cycles and reactions not in internal cycles seem to be very similar, although the number of test-cases is too small to get a sure result.

Another interesting observation is that even with 10000 sample points (even without thermodynamic constraints), we miss about 10% of all possible reaction directions.

9.4 Discussion

We observe that the conditions that we require for Thm. 9.2.1 on the sampling algorithm are very weak. We do not require uniform distribution, we only require that with some polynomially small probability we also sample fluxes unequal to zero in our target distribution and that we converge in polynomial time to this target distribution.

Assuming $\mathbf{RP} \neq \mathbf{NP}$, it follows that every sampling algorithm on the thermodynamic flux space has one of the following properties:

- The sampling algorithm does not converge in polynomial time to the target distribution, or

- the target distribution is trivial (i.e., the probability of sampling 0 is 1).

Of course, we may be lucky and the algorithm actually samples a non-trivial distribution for the input networks. However the result says that there are networks for which the sampling algorithm will only sample 0 fluxes for some reactions and indeed, we saw that this happens also in practice not only for sampling the thermodynamically constrained flux space but also the ordinary steady-state flux space. This is very critical, since we then may be led to the false assumption that the reaction is never used, although it actually could be. To make sure that such results are true, it is essential to verify them with a deterministic method. In the case of deciding whether flux is possible through a given reaction, we can decide this by solving some optimization problem [134, 101].

We have shown that sampling artifacts happen for the flux variability problem with thermodynamic constraints (and in practice they even happen without thermodynamic constraints). However, sampling is used to check a wide variety of different properties. Although the result does not directly imply that sampling results for these other properties are unreliable as well, caution is highly advised. For example, consider correlation / flux coupling analysis [169]. If a reaction always carries zero flux in all samples by an artifact, although it can also carry non-zero flux, it follows that this reaction seems uncorrelated to all other reactions. However, it may very well be correlated / coupled. In Fig. 9.3, we see such an example. Assume the flux space (see Fig. 9.4) is sampled using a uniform distribution. Then we will almost surely never sample non-zero flux through reaction r_1 . Correlation analysis would yield that flux through r_1 is uncorrelated (they are even independent) to flux through r_2 and r_3 , although the fluxes are actually exclusive (e.g. r_1 and r_2 cannot carry flux at the same time).

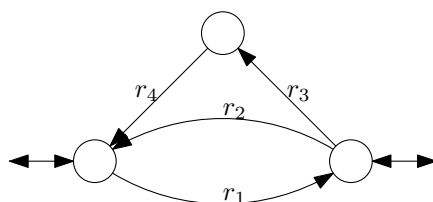


Figure 9.3: By thermodynamics, it is not possible to have non-zero flux through r_1 and also to have a non-zero flux through one of r_2, r_3 or r_4 at the same time.

9.5 Conclusion

Although sampling has been used successfully for the analysis of metabolic networks, the results obtained by sampling should be used with caution. In particular, if we sample thermodynamically feasible fluxes, we have seen (assuming $\mathbf{RP} \neq \mathbf{NP}$) that for every polynomial-time sampling method there exist networks for which the sampling method will produce artifacts. Hence, results obtained by sampling the thermodynamically constrained flux space should always be double checked by a different method.

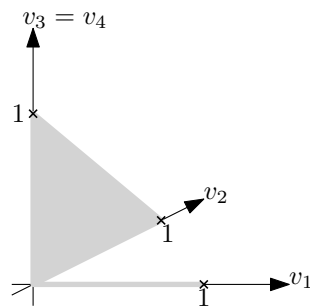


Figure 9.4: The gray area denotes the flux space. In this example it was assumed that input/output flux values are constrained to at most 1. It can be seen that flux v_1 through r_1 is exclusive to fluxes v_2 and v_3 through r_2 and r_3 respectively. Since fluxes through r_2 can be combined with fluxes through r_3 , the flux space with $v_2, v_3 > 0$ is two-dimensional, while the flux space with $v_1 > 0$ is only one-dimensional. Hence, a uniform sample of the flux space would almost surely have zero flux through r_1 .

Appendix A

Computational Results on Flux Modules

A.1 Flux Modules

A.1.1 Summary on *E. coli* iJR904 grown on L-tryptophan

| Module | Exchange flux (right hand side of the module) | EFMs |
|--|--|------|
| ACOATA, KAS14, KAS15 | accoa[c] = -0.106, actACP[c] = 0.106, co2[c] = 0.106, coa[c] = 0.106, h[c] = -0.106, malACP[c] = -0.106 | 2 |
| ADK1, ADK3, ADNK1, DADK, DGK1, DURIPP, GK1, GSNK, NDPK1, NDPK2, NDPK5, NDPK6, NDPK8, NTD1, NTD6, NTD8, PUNP1, PUNP2, PUNP3, PUNP4, PYNP2r, RNDR1, RNDR2, RNDR4, RNTR1, RNTR2, RNTR4, UMPK, URIDK2r, URIK2 | 2dr1p[c] = 0.04937, ade[c] = -0.002309, adp[c] = 5.691, amp[c] = -2.523, atp[c] = -3.174, datp[c] = 0.008149, dgtp[c] = 0.00838, dump[c] = 0.008149, gdp[c] = -0.1724, gmp[c] = -0.09185, gtp[c] = 0.2559, h2o[c] = 0.02468, h[c] = 0.05168, pi[c] = 0.05168, r1p[c] = -0.05168, trdox[c] = 0.07405, trdrd[c] = -0.07405, udp[c] = -0.01742, ump[c] = -0.1158, utp[c] = 0.1251 | 96 * |
| ALAR, ALARi | ala-D[c] = 0.01821, ala-L[c] = -0.01821 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.1326, akg[c] = 0.3209, ala-L[c] = 0.1883, glu-L[c] = -0.3209, pyr[c] = -0.1883, val-L[c] = 0.1326 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, SUCD1i, SUCD4 | asp-L[c] = -0.0009007, dhor-S[c] = -0.104, fum[c] = 5.021, glx[c] = 0.0165, glyclt[c] = -0.0165, iasp[c] = 0.0009007, orot[c] = 0.104, q8[c] = -5.142, q8h2[c] = 5.142, succ[c] = -5.021 | 18 |
| GALU, GALUi | g1p[c] = -0.006532, h[c] = -0.006532, ppi[c] = 0.006532, udpg[c] = 0.006532, utp[c] = -0.006532 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.06411, atp[c] = -0.06411, cdp[c] = -0.06411, ctp[c] = 0.05573, dctp[c] = 0.00838, h2o[c] = 0.00838, trdox[c] = 0.00838, trdrd[c] = -0.00838 | 2 |

* the MILP method sometimes missed up to 3 EFMs.

A.1.2 Summary on *E. coli* iJR904 grown on L-Threonine

| Module | Exchange flux (right hand side of the module) | EFM#s |
|--|--|-------|
| ACALDi, GLYATi, THRAr, THRD | accoa[c] = 7.755, coa[c] = -7.755, gly[c] = 7.755, h[c] = 7.755, nad[c] = -7.755, nadh[c] = 7.755, thr-L[c] = -7.755 | 2 |
| ACCOAL, PPCSCT, SUCOAS ^{new} | adp[c] = 0.2558, atp[c] = -0.2558, coa[c] = -0.2558, pi[c] = 0.2558, succ[c] = -0.2558, succoa[c] = 0.2558 | 2 |
| ACOATA, KAS14, KAS15 | accoa[c] = -0.1645, actACP[c] = 0.1645, co2[c] = 0.1645, coa[c] = 0.1645, h[c] = -0.1645, malACP[c] = -0.1645 | 2 |
| ADK1, ADK3, NDPK1, NDPK5, RNDR2, RNTR2 | adp[c] = 4.007, amp[c] = -1.799, atp[c] = -2.209, dgtp[c] = 0.013, gdp[c] = -0.41, gtp[c] = 0.397, h2o[c] = 0.013, trdox[c] = 0.013, trdrd[c] = -0.013 | 6 |
| ALAR, ALARi | ala-D[c] = 0.02826, ala-L[c] = -0.02826 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.2058, akc[c] = 0.498, ala-L[c] = 0.2922, glu-L[c] = -0.498, pyr[c] = -0.2922, val-L[c] = 0.2058 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FDH2, FDH3, FHL, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, HYD1, HYD2, HYD3, SUCD1i, SUCD4 | asp-L[c] = -0.001398, co2[c] = 8.574, dhor-S[c] = -0.1613, for[c] = -8.574, fum[c] = 4.764, glx[c] = 0.0256, glyclt[c] = -0.0256, h[c] = -25.72, h[e] = 17.15, iasp[c] = 0.001398, orot[c] = 0.1613, q8[c] = -13.53, q8h2[c] = 13.53, succ[c] = -4.764 | 90 |
| GALU, GALUi | g1p[c] = -0.01014, h[c] = -0.01014, ppi[c] = 0.01014, udpg[c] = 0.01014, utp[c] = -0.01014 | 2 |
| NAt3.1, THRt2r, THRt4 ^{new} | h[c] = 10, h[e] = -10, thr-L[c] = 10, thr-L[e] = -10 | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 | adp[c] = 0.1941, atp[c] = -0.1941, dudp[c] = 0.01264, h2o[c] = 0.01264, trdox[c] = 0.01264, trdrd[c] = -0.01264, udp[c] = -0.2068, utp[c] = 0.1941 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.09947, atp[c] = -0.09947, cdp[c] = -0.09947, ctp[c] = 0.08647, dctp[c] = 0.013, h2o[c] = 0.013, trdox[c] = 0.013, trdrd[c] = -0.013 | 2 |
| NDPK8, RNDR1, RNTR1 | atp[c] = -0.01264, datp[c] = 0.01264, h2o[c] = 0.01264, trdox[c] = 0.01264, trdrd[c] = -0.01264 | 2 |

^{new}This is only a module with thermodynamic constraints.

A.1.3 Summary on *E. coli* iJR904 grown on glucose

| Module | Exchange flux (right hand side of the module) | EFM _s |
|---|--|------------------|
| ACOATA, KAS14, KAS15 | accoa[c] = -0.2962, actACP[c] = 0.2962, co2[c] = 0.2962, coa[c] = 0.2962, h[c] = -0.2962, malACP[c] = -0.2962 | 2 |
| ADK1, ADK3, ADNK1, DADK, DGK1, DURIPP, GK1, GSNK, NDPK1, NDPK2, NDPK5, NDPK6, NDPK8, NTD1, NTD6, NTD8, PUNP1, PUNP2, PUNP3, PUNP4, PYNP2r, RNDR1, RNDR2, RNDR4, RNTR1, RNTR2, RNTR4, UMPK, URIDK2r, URIK2 | 2dr1p[c] = 0.4766, ade[c] = -0.006454, adp[c] = 6.387, amp[c] = -2.123, atp[c] = -4.281, datp[c] = 0.02277, dgtp[c] = 0.02342, dump[c] = 0.02277, gdp[c] = -0.4818, gmp[c] = -0.2567, gtp[c] = 0.715, h2o[c] = 0.06896, h[c] = 0.4831, pi[c] = 0.4831, r1p[c] = -0.4831, trdox[c] = 0.5456, trdrd[c] = -0.5456, udp[c] = -0.04868, ump[c] = -0.3237, utp[c] = 0.3496 | 96 |
| ALAR, ALARi | ala-D[c] = 0.05089, ala-L[c] = -0.05089 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.3706, akgl[c] = 0.8969, ala-L[c] = 0.5262, glu-L[c] = -0.8969, pyr[c] = -0.5262, val-L[c] = 0.3706 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FDH2, FDH3, FHL, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, HYD1, HYD2, HYD3, SUCD1i, SUCD4 | asp-L[c] = -0.002517, co2[c] = 0.3387, dhor-S[c] = -0.2905, for[c] = -0.3387, fum[c] = 4.022, glx[c] = 0.0461, glyclt[c] = -0.0461, h[c] = -1.016, h[e] = 0.6774, iasp[c] = 0.002517, orot[c] = 0.2905, q8[c] = -4.7, q8h2[c] = 4.7, succ[c] = -4.022 | 90 |
| DHAPT, F6PA, FBA, PFK, PYK | adp[c] = 6.586, atp[c] = -6.586, dhap[c] = 7.003, f6p[c] = -7.003, g3p[c] = 7.003, h[c] = 6.586, pep[c] = -0.4173, pyr[c] = 0.4173 | 2 |
| GALU, GALUi | g1p[c] = -0.01825, h[c] = -0.01825, ppi[c] = 0.01825, udpg[c] = 0.01825, utp[c] = -0.01825 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.1791, atp[c] = -0.1791, cdp[c] = -0.1791, ctp[c] = 0.1557, dctp[c] = 0.02342, h2o[c] = 0.02342, trdox[c] = 0.02342, trdrd[c] = -0.02342 | 2 |

A.1.4 Summary on *E. coli* iJR904 grown on L-Arginine

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|---|------|
| ACCOAL, PPCSCT, SUCOAS ^{new} | adp[c] = 3.219, atp[c] = -3.219, coa[c] = -3.219, pi[c] = 3.219, succ[c] = -3.219, succoa[c] = 3.219 | 2 |
| ACOATA, KAS14, KAS15 | accoa[c] = -0.2135, actACP[c] = 0.2135, co2[c] = 0.2135, coa[c] = 0.2135, h[c] = -0.2135, malACP[c] = -0.2135 | 2 |
| ADK1, ADK3, NDPK1, NDPK5, RNDR2, RNTR2 | adp[c] = 3.229, amp[c] = -1.348, atp[c] = -1.88, dgtp[c] = 0.01688, gdp[c] = -0.5323, gtp[c] = 0.5154, h2o[c] = 0.01688, trdox[c] = 0.01688, trdrd[c] = -0.01688 | 6 |
| ALAR, ALAR _i | ala-D[c] = 0.03669, ala-L[c] = -0.03669 | 2 |
| ALATA-L, VALTA, VPAMT | 3mob[c] = -0.2672, akc[c] = 0.6465, ala-L[c] = 0.3794, glu-L[c] = -0.6465, pyr[c] = -0.3794, val-L[c] = 0.2672 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FRD2, SUCD1i, SUCD4 | asp-L[c] = -0.001814, dhor-S[c] = -0.2094, fum[c] = 6.898, iasp[c] = 0.001814, orot[c] = 0.2094, q8[c] = -7.109, q8h2[c] = 7.109, succ[c] = -6.898 | 6 |
| ASPT, ASPTA, FUM, GLUDy, IDOND, IDOND2, MDH, ME1, ME2, NADTRHD, PPCK, PYK | adp[c] = 3.566, akc[c] = 16.36, asp-L[c] = 1.701, atp[c] = -3.566, co2[c] = 8.071, fum[c] = -7.408, glu-L[c] = -16.36, h2o[c] = -22.06, h[c] = 17.56, nad[c] = -13.82, nadh[c] = 13.82, nadp[c] = -8.245, nadph[c] = 8.245, nh4[c] = 14.66, oaa[c] = -2.364, pep[c] = 3.566, pyr[c] = 4.504 | 9 |
| GALU, GALU _i | g1p[c] = -0.01316, h[c] = -0.01316, ppi[c] = 0.01316, udp[c] = 0.01316, utp[c] = -0.01316 | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 | adp[c] = 0.252, atp[c] = -0.252, dudp[c] = 0.01642, h2o[c] = 0.01642, trdox[c] = 0.01642, trdrd[c] = -0.01642, udp[c] = -0.2684, utp[c] = 0.252 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.1291, atp[c] = -0.1291, cdp[c] = -0.1291, ctp[c] = 0.1123, dctp[c] = 0.01688, h2o[c] = 0.01688, trdox[c] = 0.01688, trdrd[c] = -0.01688 | 2 |
| NDPK8, RNDR1, RNTR1 | atp[c] = -0.01642, datp[c] = 0.01642, h2o[c] = 0.01642, trdox[c] = 0.01642, trdrd[c] = -0.01642 | 2 |

^{new}This is only a module with thermodynamic constraints.

A.1.5 Summary on *E. coli* iJR904 grown on citrate

| Module | Exchange flux (right hand side of the module) | EFMs |
|--|--|------|
| ACOATA, KAS14, KAS15 | accoa[c] = -0.2048, actACP[c] = 0.2048, co2[c] = 0.2048, coa[c] = 0.2048, h[c] = -0.2048, malACP[c] = -0.2048 | 2 |
| ADK1, ADK3, NDPK1, NDPK5, RNDR2, RNTR2 | adp[c] = 3.455, amp[c] = -1.472, atp[c] = -1.983, dgtp[c] = 0.01619, gdp[c] = -0.5106, gtp[c] = 0.4944, h2o[c] = 0.01619, trdox[c] = 0.01619, trdrd[c] = -0.01619 | 6 |
| ALAR, ALARi | ala-D[c] = 0.03519, ala-L[c] = -0.03519 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.2563, akg[c] = 0.6202, ala-L[c] = 0.3639, glu-L[c] = -0.6202, pyr[c] = -0.3639, val-L[c] = 0.2563 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FDH2, FDH3, FHL, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, HYD1, HYD2, HYD3, SUCD1i, SUCD4 | asp-L[c] = -0.00174, co2[c] = 0.03636, dhor-S[c] = -0.2009, for[c] = -0.03636, fum[c] = 10.29, glx[c] = 0.03188, glyclt[c] = -0.03188, h[c] = -0.1091, h[e] = 0.07271, iasp[c] = 0.00174, orot[c] = 0.2009, q8[c] = -10.56, q8h2[c] = 10.56, succ[c] = -10.29 | 90 |
| ASPT, ASPTA, FUM, GLUDy, HPYRI, HPYRRx, HPYRRy, IDOND, IDOND2, MDH, ME1, ME2, NADTRHD, PPCK, PYK, TRSAR | 2h3oppan[c] = -0.01594, adp[c] = 3.471, akg[c] = -3.561, asp-L[c] = 1.811, atp[c] = -3.471, co2[c] = 8.149, fum[c] = -10.96, glu-L[c] = 3.561, glyc-R[c] = 0.01594, h2o[c] = -5.585, h[c] = 0.8904, nad[c] = -10.85, nadh[c] = 10.85, nadp[c] = 5.28, nadph[c] = -5.28, nh4[c] = -5.372, oaa[c] = 0.9963, pep[c] = 3.471, pyr[c] = 4.679 | 15 |
| FUMt2.2, SUCct2.2, SUCFUMt | h[c] = 20, h[e] = -20, succ[c] = 10, succ[e] = -10 | 2 |
| GALU, GALUi | g1p[c] = -0.01262, h[c] = -0.01262, ppi[c] = 0.01262, udpg[c] = 0.01262, utp[c] = -0.01262 | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 | adp[c] = 0.2417, atp[c] = -0.2417, dudp[c] = 0.01575, h2o[c] = 0.01575, trdox[c] = 0.01575, trdrd[c] = -0.01575, udp[c] = -0.2575, utp[c] = 0.2417 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.1239, atp[c] = -0.1239, cdp[c] = -0.1239, ctp[c] = 0.1077, dctp[c] = 0.01619, h2o[c] = 0.01619, trdox[c] = 0.01619, trdrd[c] = -0.01619 | 2 |
| NDPK8, RNDR1, RNTR1 | atp[c] = -0.01575, datp[c] = 0.01575, h2o[c] = 0.01575, trdox[c] = 0.01575, trdrd[c] = -0.01575 | 2 |

A.1.6 Summary on *E. coli* iJR904 grown on fumarate

| Module | Exchange flux (right hand side of the module) | EFM#s |
|--|---|-------|
| ACOATA, KAS14, KAS15 | accoa[c] = -0.1294, actACP[c] = 0.1294, co2[c] = 0.1294, coa[c] = 0.1294, h[c] = -0.1294, malACP[c] = -0.1294 | 2 |
| ADK1, ADK3, NDPK1, NDPK5, RNDR2, RNTR2 | adp[c] = 2.183, amp[c] = -0.9302, atp[c] = -1.253, dgtp[c] = 0.01023, gdp[c] = -0.3226, gtp[c] = 0.3124, h2o[c] = 0.01023, trdox[c] = 0.01023, trdrd[c] = -0.01023 | 6 |
| ALAR, ALARi | ala-D[c] = 0.02223, ala-L[c] = -0.02223 | 2 |
| ALATA-L, VALTA, VPAMT | 3mob[c] = -0.1619, akg[c] = 0.3918, ala-L[c] = 0.2299, glu-L[c] = -0.3918, pyr[c] = -0.2299, val-L[c] = 0.1619 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FDH2, FDH3, FHL, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, HYD1, HYD2, HYD3, SUCD1i, SUCD4 | asp-L[c] = -0.0011, co2[c] = 0.02297, dhor-S[c] = -0.1269, for[c] = -0.02297, fum[c] = 3.864, glx[c] = 0.02014, glyclt[c] = -0.02014, h[c] = -0.06891, h[e] = 0.04594, iasp[c] = 0.0011, orot[c] = 0.1269, q8[c] = -4.035, q8h2[c] = 4.035, succ[c] = -3.864 | 90 |
| ASPT, ASPTA, FUM, GLUDy, HPYRI, HPYRRx, HPYRRy, IDOND, IDOND2, MDH, ME1, ME2, NADTRHD, PPCK, PYK, TRSAR | 2h3oppa[c] = -0.01007, adp[c] = 2.193, akg[c] = -2.25, asp-L[c] = 1.144, atp[c] = -2.193, co2[c] = 8.831, fum[c] = -14.29, glu-L[c] = 2.25, glyc-R[c] = 0.01007, h2o[c] = -10.89, h[c] = 4.245, nad[c] = -11.58, nadh[c] = 11.58, nadp[c] = 0.7001, nadph[c] = -0.7001, nh4[c] = -3.394, oaa[c] = 4.312, pep[c] = 2.193, pyr[c] = 6.638 | 15 |
| FUMt2.2, SUCct2.2, SUCFUMt | fum[c] = 10, fum[e] = -10, h[c] = 20, h[e] = -20 | 2 |
| GALU, GALUi | g1p[c] = -0.007975, h[c] = -0.007975, ppi[c] = 0.007975, udpg[c] = 0.007975, utp[c] = -0.007975 | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 | adp[c] = 0.1527, atp[c] = -0.1527, dudp[c] = 0.009949, h2o[c] = 0.009949, trdox[c] = 0.009949, trdrd[c] = -0.009949, udp[c] = -0.1627, utp[c] = 0.1527 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.07826, atp[c] = -0.07826, cdp[c] = -0.07826, ctp[c] = 0.06803, dctp[c] = 0.01023, h2o[c] = 0.01023, trdox[c] = 0.01023, trdrd[c] = -0.01023 | 2 |
| NDPK8, RNDR1, RNTR1 | atp[c] = -0.009949, datp[c] = 0.009949, h2o[c] = 0.009949, trdox[c] = 0.009949, trdrd[c] = -0.009949 | 2 |

A.1.7 Summary on *E. coli* iJR904 grown on L-glutamine

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|--|------|
| ACOATA, KAS14, KAS15 | accoa[c] = -0.195, actACP[c] = 0.195, co2[c] = 0.195, coa[c] = 0.195, h[c] = -0.195, malACP[c] = -0.195 | 2 |
| ADK1, ADK3, NDPK1, NDPK5, RNDR2, RNTR2 | adp[c] = 3.29, amp[c] = -1.402, atp[c] = -1.888, dgtp[c] = 0.01542, gdp[c] = -0.4862, gtp[c] = 0.4707, h2o[c] = 0.01542, trdox[c] = 0.01542, trdrd[c] = -0.01542 | 6 |
| ALAR, ALARi | ala-D[c] = 0.0335, ala-L[c] = -0.0335 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.244, akg[c] = 0.5904, ala-L[c] = 0.3464, glu-L[c] = -0.5904, pyr[c] = -0.3464, val-L[c] = 0.244 | 2 |
| ASNS1, ASNS2, ASPT, ASPTA, FUM, G6PDA, GF6PTA, GLUDy, GLUN, GLUSy, HPYRI, HPYRRx, HPYRRy, IDOND, IDOND2, MDH, ME1, ME2, NADTRHD, PPCK, PYK, TRSAR | 2h3oppan[c] = -0.01517, adp[c] = 3.304, akg[c] = 6.61, amp[c] = 0.139, asn-L[c] = 0.139, asp-L[c] = 1.585, atp[c] = -3.443, co2[c] = 8.238, f6p[c] = -0.0437, fum[c] = -11.39, gam6p[c] = 0.0437, gln-L[c] = -8.889, glu-L[c] = 2.28, glyc-R[c] = 0.01517, h2o[c] = -25.12, h[c] = 11.47, nad[c] = -12.25, nadh[c] = 12.25, nadp[c] = -4.014, nadph[c] = 4.014, nh4[c] = 13.59, oaa[c] = 1.428, pep[c] = 3.304, ppi[c] = 0.139, pyr[c] = 4.934 | 222 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FDH2, FDH3, FHL, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, HYD1, HYD2, HYD3, SUCD1i, SUCD4 | asp-L[c] = -0.001657, co2[c] = 0.03461, dhor-S[c] = -0.1913, for[c] = -0.03461, fum[c] = 10.75, glx[c] = 0.03035, glyclt[c] = -0.03035, h[c] = -0.1038, h[e] = 0.06923, iasp[c] = 0.001657, orot[c] = 0.1913, q8[c] = -11.01, q8h2[c] = 11.01, succ[c] = -10.75 | 90 |
| GALU, GALUi | g1p[c] = -0.01202, h[c] = -0.01202, ppi[c] = 0.01202, udpg[c] = 0.01202, utp[c] = -0.01202 | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 | adp[c] = 0.2302, atp[c] = -0.2302, dudp[c] = 0.01499, h2o[c] = 0.01499, trdox[c] = 0.01499, trdrd[c] = -0.01499, udp[c] = -0.2451, utp[c] = 0.2302 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.1179, atp[c] = -0.1179, cdp[c] = -0.1179, ctp[c] = 0.1025, dctp[c] = 0.01542, h2o[c] = 0.01542, trdox[c] = 0.01542, trdrd[c] = -0.01542 | 2 |
| NDPK8, RNDR1, RNTR1 | atp[c] = -0.01499, datp[c] = 0.01499, h2o[c] = 0.01499, trdox[c] = 0.01499, trdrd[c] = -0.01499 | 2 |

A.1.8 Summary on *E. coli* iJR904 grown on Lactose

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|---|---------|
| ACCOAL, PPCSCT, SUCOAS ^{new} | adp[c] = 0.6389, atp[c] = -0.6389, coa[c] = -0.6389, pi[c] = 0.6389, succ[c] = -0.6389, succoa[c] = 0.6389 | 2 |
| ACKr, ACt2r, ADK1, ADK3, ADNK1, ASAD, ASPK, ASPTA, ATPS4r, CO2t, DADK, DGK1, DHAPT, DRPA, DURIPP, ENO, EX_ac(e), EX_co2(e), EX_for(e), EX_h2o(e), EX_h(e), F6PA, FBA, FOrt, FTHFD, G6PDH2r, GAPD, GHMT2, GK1, GND, GSNK, H2ot, HSDy, HSK, MTHFC, MTHFD, NDPK1, NDPK2, NDPK5, NDPK6, NDPK8, NTD1, NTD6, NTD8, PDH, PFK, PFL, PGCD, PGI, PGK, PGL, PGM, PPC, PPM, PPM2, PSERT, PSP_L, PTAr, PUNP1, PUNP2, PUNP3, PUNP4, PYK, PYNP2r, RNDR1, textttRNDR2, RNDR4, RNTR1, RNTR2, RNTR4, RPE, RPI, TALA, THD2, THRAr, THRS, TKT1, TKT2, TPI, TRDR, UMPK, URIDK2r, URIK2 | 10fthf[c] = 1.337, ac[c] = -0.7325, accoa[c] = 6.131, ade[c] = -0.008951, adp[c] = -98.79, akg[c] = 5.697, amp[c] = -2.944, asp-L[c] = 2.333, aspsa[c] = 0.4522, atp[c] = 101.7, co2[c] = -3.501, coa[c] = -6.131, datp[c] = 0.03158, dgtp[c] = 0.03248, dhap[c] = 0.2146, dump[c] = 0.03158, e4p[c] = 0.5256, f6p[c] = 0.09207, for[c] = -0.07293, g3p[c] = -0.06905, g6p[c] = -19.78, gdp[c] = -0.6682, glu-L[c] = -5.697, gly[c] = 1.355, gmp[c] = -0.356, gtp[c] = 0.9917, h2o[c] = 33.88, h[c] = 177.2, h[e] = -227.5, hom-L[c] = 0.1867, mlthf[c] = 0.2823, nad[c] = -37.14, nadh[c] = 37.14, nadp[c] = -19.44, nadph[c] = 19.44, oaa[c] = 0.07953, pep[c] = 1.14, pi[c] = -90.4, pyr[c] = 3.407, r5p[c] = 1.202, ru5p-D[c] = 0.05373, s7p[c] = 0.03222, ser-L[c] = 0.7087, thf[c] = -1.62, thr-L[c] = 0.6611, trdox[c] = -0.3305, trdrd[c] = 0.3305, udp[c] = -0.06752, ump[c] = -0.449, utp[c] = 0.4849 | ≥ 300 * |
| ACOATA, KAS14, KAS15 | accoa[c] = -0.4109, actACP[c] = 0.4109, co2[c] = 0.4109, coa[c] = 0.4109, h[c] = -0.4109, malACP[c] = -0.4109 | 2 |
| ALAR, ALARi | ala-D[c] = 0.07059, ala-L[c] = -0.07059 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.514, akg[c] = 1.244, ala-L[c] = 0.7299, glu-L[c] = -1.244, pyr[c] = -0.7299, val-L[c] = 0.514 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FRD2, SUCD1i, SUCD4 | asp-L[c] = -0.003491, dhor-S[c] = -0.4029, fum[c] = -3.836e-06, iasp[c] = 0.003491, orot[c] = 0.4029, q8[c] = -0.4064, q8h2[c] = 0.4064, succ[c] = 3.836e-06 | 8 |
| GALU, GALUi | g1p[c] = -0.02532, h[c] = -0.02532, ppi[c] = 0.02532, udpg[c] = 0.02532, utp[c] = -0.02532 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.2485, atp[c] = -0.2485, cdp[c] = -0.2485, ctp[c] = 0.216, dctp[c] = 0.03248, h2o[c] = 0.03248, trdox[c] = 0.03248, trdrd[c] = -0.03248 | 2 |

* both tools `metatool` and the MILP enumeration failed to enumerate all EFMs. We aborted the MILP computation after the computation of 300 elementary modes.

^{new} This is only a module with thermodynamic constraints.

A.1.9 Summary on *E. coli* iJR904 grown on L-malate

| Module | Exchange flux (right hand side of the module) | EFMs |
|--|---|------|
| ACOATA, KAS14, KAS15 | accoa[c] = -0.1294, actACP[c] = 0.1294, co2[c] = 0.1294, coa[c] = 0.1294, h[c] = -0.1294, malACP[c] = -0.1294 | 2 |
| ADK1, ADK3, NDPK1, NDPK5, RNDR2, RNTR2 | adp[c] = 2.183, amp[c] = -0.9302, atp[c] = -1.253, dgtp[c] = 0.01023, gdp[c] = -0.3226, gtp[c] = 0.3124, h2o[c] = 0.01023, trdox[c] = 0.01023, trdrd[c] = -0.01023 | 6 |
| ALAR, ALARi | ala-D[c] = 0.02223, ala-L[c] = -0.02223 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.1619, akg[c] = 0.3918, ala-L[c] = 0.2299, glu-L[c] = -0.3918, pyr[c] = -0.2299, val-L[c] = 0.1619 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FDH2, FDH3, FHL, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, HYD1, HYD2, HYD3, SUCD1i, SUCD4 | asp-L[c] = -0.0011, co2[c] = 0.02297, dhor-S[c] = -0.1269, for[c] = -0.02297, fum[c] = 3.864, glx[c] = 0.02014, glyclt[c] = -0.02014, h[c] = -0.06891, h[e] = 0.04594, iasp[c] = 0.0011, orot[c] = 0.1269, q8[c] = -4.035, q8h2[c] = 4.035, succ[c] = -3.864 | 90 * |
| ASPT, ASPTA, FUM, GLUDy, HPYRI, HPYRRx, HPYRRy, IDOND, IDOND2, MDH, ME1, ME2, NADTRHD, PPCK, PYK, TRSAR | 2h3oppan[c] = -0.01007, adp[c] = 2.193, akg[c] = -2.25, asp-L[c] = 1.144, atp[c] = -2.193, co2[c] = 8.831, fum[c] = -4.286, glu-L[c] = 2.25, glyc-R[c] = 0.01007, h2o[c] = -0.8925, h[c] = 4.245, mal-L[c] = -10, nad[c] = -11.58, nadh[c] = 11.58, nadp[c] = 0.7001, nadph[c] = -0.7001, nh4[c] = -3.394, oaa[c] = 4.312, pep[c] = 2.193, pyr[c] = 6.638 | 15 |
| GALU, GALUi | g1p[c] = -0.007975, h[c] = -0.007975, ppi[c] = 0.007975, udpg[c] = 0.007975, utp[c] = -0.007975 | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 | adp[c] = 0.1527, atp[c] = -0.1527, dudp[c] = 0.009949, h2o[c] = 0.009949, trdox[c] = 0.009949, trdrd[c] = -0.009949, udp[c] = -0.1627, utp[c] = 0.1527 | 2 |
| NDPK3, NDPK7, RNDR3, RNTR3 | adp[c] = 0.07826, atp[c] = -0.07826, cdp[c] = -0.07826, ctp[c] = 0.06803, dctp[c] = 0.01023, h2o[c] = 0.01023, trdox[c] = 0.01023, trdrd[c] = -0.01023 | 2 |
| NDPK8, RNDR1, RNTR1 | atp[c] = -0.009949, datp[c] = 0.009949, h2o[c] = 0.009949, trdox[c] = 0.009949, trdrd[c] = -0.009949 | 2 |

* we observed that sometimes the MILP method was missing one EFM

A.1.10 Summary on *E. coli* iAF1260 grown on glucose, aerobic

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|---|--------|
| ACKr, ACS, ADK1, ADK3, ADNK1, ADPT, FLDR, GRXR, GTHOr, NDPK1, NDPK2, NDPK3, NDPK5, NDPK6, NDPK7, NDPK8, PAPSR, PAPSR2, PPKr, PPM, PRPPS, PTAr, PUNP1, R15BPK, R1PK, RNDR1, RNDR1b, RNDR2, RNDR2b, RNDR3, RNDR3b, RNDR4, RNDR4b, RNTR1c, RNTR2c, RNTR3c, RNTR4c, TRDR ^{new} | ac[c] = -0.428, accoa[c] = 0.428, ade[c] = -0.0003286, adp[c] = 2.476, amp[c] = -0.9813, atp[c] = -1.513, cdp[c] = -0.1333, coa[c] = -0.428, ctp[c] = 0.1134, datp[c] = 0.01928, dctp[c] = 0.01991, dgtp[c] = 0.01991, dudp[c] = 0.01928, gdp[c] = -0.5803, gtp[c] = 0.5604, h2o[c] = 0.07837, h[c] = 0.7897, nadp[c] = 0.2598, nadph[c] = -0.2598, pap[c] = 0.1815, paps[c] = -0.1815, pi[c] = 2.724, ppi[c] = -2.296, prpp[c] = 0.6863, r5p[c] = -0.6866, so3[c] = 0.1815, udp[c] = -0.3133, utp[c] = 0.294 | 5184 * |
| ACOATA, KAS14, KAS15 | accoa[c] = -0.2621, actACP[c] = 0.2621, co2[c] = 0.2621, coa[c] = 0.2621, h[c] = -0.2621, malACP[c] = -0.2621 | 2 |
| Act2rpp, Act4pp, GLUt2rpp, GLUt4pp, GLYCLTt2rpp, GLYCLTt4pp, PROt2rpp, PROt4pp, SERT2rpp, SERT4pp, THRT2rpp, THRT4pp ^{new} | h[c] = -0.00349, h[p] = 0.00349, na1[c] = 0.00349, na1[p] = -0.00349 | 6 |
| ALATA-L, VALTA, VPAMT | 3mob[c] = -0.3118, akg[c] = 0.7412, ala-L[c] = 0.4294, glu-L[c] = -0.7412, pyr[c] = -0.4294, val-L[c] = 0.3118 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FRD2, FRD3, GLYCTO2, GLYCTO3, GLYCTO4, NADH16pp, NADH17pp, NADH18pp, SUCDi | asp-L[c] = -0.001678, dhor-S[c] = -0.2437, fum[c] = 3.729, glx[c] = 0.0004929, glyclt[c] = -0.0004929, h[c] = -114.2, h[p] = 85.67, iasp[c] = 0.001678, nad[c] = 28.56, nadh[c] = -28.56, orot[c] = 0.2437, q8[c] = -32.53, q8h2[c] = 32.53, succ[c] = -3.729 | 54 |
| DHAPT, F6PA, FBA, PFK, PYK | adp[c] = 5.164, atp[c] = -5.164, dhap[c] = 6.191, f6p[c] = -6.191, g3p[c] = 6.191, h[c] = 5.164, pep[c] = -1.027, pyr[c] = 1.027 | 2 |
| DMPPS, IPDDI, IPDPS | dmpp[c] = 0.0002048, h2mb4p[c] = -0.00176, h2o[c] = 0.00176, h[c] = -0.00176, ipdp[c] = 0.001555, nad[c] = 0.00176, nadh[c] = -0.00176 | 3 |
| EX_fe2(e), EX_fe3(e), EX_h2o(e), EX_h(e), EX_o2(e), FE2tex, FE3tex, FEROpp, H2Otex, Htex, O2tex | fe2[p] = 0.005564, fe3[p] = 0.005235, h2o[p] = -37.24, h[p] = -6.766, o2[p] = 16.27 | 3 ** |
| GLCtex, GLCtexi | glc-D[e] = -8, glc-D[p] = 8 | 2 |

* computed with metatool only, MILP method took too long.

** computed with MILP method only. metatool did not find any pathways.

^{new} This is only a module with thermodynamic constraints.

A.1.11 Summary on *E. coli* iAF1260 grown on glucose, anaerobic

| Module | Exchange flux (right hand side of the module) | EFMs |
|--|---|--------|
| DHAPT, F6PA, FBA, PFK, PYK | adp[c] = 1.92, atp[c] = -1.92, dhap[c] = 12.55, f6p[c] = -12.55, g3p[c] = 12.55, h[c] = 1.92, pep[c] = -10.63, pyr[c] = 10.63 | 2 |
| ACT2rpp, ACT4pp, GLUT2rpp, GLUT4pp, GLYCLTt2rpp, GLYCLTt4pp, PROt2rpp, PROt4pp, SERT2rpp, SERT4pp, THRT2rpp, THRT4pp ^{new} | ac[c] = -10.84, ac[p] = 10.84, glyclt[c] = -0.0001762, glyclt[p] = 0.0001762, h[c] = -10.84, h[p] = 10.84, nai[c] = 0.001247, nai[p] = -0.001247 | 6 |
| DMPPS, IPDDI, IPDPS | dmp[c] = 7.321e - 05, h2mb4p[c] = -0.0006291, h2o[c] = 0.0006291, h[c] = -0.0006291, ipdp[c] = 0.0005559, nad[c] = 0.0006291, nadh[c] = -0.0006291 | 3 |
| ADK1, ADK3, ADNK1, ADPT, FLDR, GRXR, GTHOr, NDPK1, NDPK2, NDPK3, NDPK5, NDPK6, NDPK7, NDPK8, PAPSR, PAPSR2, PPKr, PPM, PRPPS, PUNP1, R15BPK, R1PK, RNDR1, RNDR1b, RNDR2, RNDR2b, RNDR3, RNDR3b, RNDR4, RNDR4b, RNTR1c, RNTR2c, RNTR3c, RNTR4c, TRDR ^{new} | ade[c] = -0.0002349, adp[c] = 0.7322, amp[c] = -0.3508, atp[c] = -0.3881, cdp[c] = -0.04764, ctp[c] = 0.04052, datp[c] = 0.006892, dctp[c] = 0.007116, dgtp[c] = 0.007116, dudp[c] = 0.006892, gdp[c] = -0.2075, gtp[c] = 0.2003, h2o[c] = 0.02802, h[c] = 0.2824, nadp[c] = 0.09288, nadph[c] = -0.09288, pap[c] = 0.06486, paps[c] = -0.06486, pi[c] = 0.8211, ppi[c] = -0.8209, prpp[c] = 0.2453, r5p[c] = -0.2456, so3[c] = 0.06486, udp[c] = -0.112, utp[c] = 0.1051 | 2592 * |
| ASPO4, ASP05, FRD2 | asp_L[c] = -0.0005999, fum[c] = -0.08771, h[c] = 0.0005999, iasp[c] = 0.0005999, mql8[c] = -0.08711, mqn8[c] = 0.08711, succ[c] = 0.08771 | 2 |
| ACCOAL, PPCSCT, SUCOAS ^{new} | adp[c] = 0.1382, atp[c] = -0.1382, coa[c] = -0.1382, pi[c] = 0.1382, succ[c] = -0.1382, succoa[c] = 0.1382 | 2 |
| ACOATA, KAS14, KAS15 | accoa[c] = -0.09371, actACP[c] = 0.09371, co2[c] = 0.09371, coa[c] = 0.09371, h[c] = -0.09371, malACP[c] = -0.09371 | 2 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.1114, akg[c] = 0.265, ala_L[c] = 0.1535, glu_L[c] = -0.265, pyr[c] = -0.1535, val_L[c] = 0.1114 | 2 |

* computed with `metatool` only, MILP method took too long.

^{new} This is only a module with thermodynamic constraints.

The module {GLCtex, GLCtexi} is not listed, since we ran the analysis on a network, where GLCtexi was removed, since it is just a duplicate of GLCtex.

A.1.12 Summary on *E. coli* iAF1260 grown on L-Threonine

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|---|--------|
| ACALD, GLYAT, THRAi, THRD | accoa[c] = 7.762, coa[c] = -7.762, gly[c] = 7.762, h[c] = 7.762, nad[c] = -7.762, nadh[c] = 7.762, thr-L[c] = -7.762 | 2 |
| ACKr, ACS, ADK1, ADK3, ADNK1, ADPT, FLDR, GRXR, GTHOr, NDPK1, NDPK2, NDPK3, NDPK5, NDPK6, NDPK7, NDPK8, PAPSR, PAPSR2, PPKr, PPM, PRPPS, PTAr, PUNP1, R15BPK, R1PK, RNDR1, RNDR1b, RNDR2, RNDR2b, RNDR3, RNDR3b, RNDR4, RNDR4b, RNTR1c, RNTR2c, RNTR3c, RNTR4c, TRDR ^{new} | ac[c] = -0.2545, accoa[c] = 0.2545, ade[c] = -0.0001953, adp[c] = 4.277, amp[c] = -1.986, atp[c] = -2.302, cdp[c] = -0.07923, coa[c] = -0.2545, ctp[c] = 0.06739, datp[c] = 0.01146, dctp[c] = 0.01183, dgtp[c] = 0.01183, dudp[c] = 0.01146, gdp[c] = -0.345, gtp[c] = 0.3332, h2o[c] = 0.04659, h[c] = 0.4695, nadp[c] = 0.1545, nadph[c] = -0.1545, pap[c] = 0.1079, paps[c] = -0.1079, pi[c] = 1.62, ppi[c] = -1.365, prpp[c] = 0.408, r5p[c] = -0.4082, so3[c] = 0.1079, udp[c] = -0.1863, utp[c] = 0.1748 | 4944 * |
| ACOATA, KAS14, KAS15 | accoa[c] = -0.1559, actACP[c] = 0.1559, co2[c] = 0.1559, coa[c] = 0.1559, h[c] = -0.1559, malACP[c] = -0.1559 | 2 |
| ACt2rpp, ACt4pp, CA2t3pp, CAt6pp, GLUt2rpp, GLUt4pp, GLYCLTt2rpp, GLYCLTt4pp, NAT3pp, PROt2rpp, PROt4pp, SERt2rpp, SERt4pp, THRt2rpp, THRt4pp ^{new} | ca2[c] = 0.002075, ca2[p] = -0.002075, h[c] = 7.998, h[p] = -7.998, thr-L[c] = 8, thr-L[p] = -8 | 8 |
| ALATA_L, VALTA, VPAMT | 3mob[c] = -0.1854, akg[c] = 0.4407, ala-L[c] = 0.2553, glu-L[c] = -0.4407, pyr[c] = -0.2553, val-L[c] = 0.1854 | 2 |
| ASPO3, ASPO4, ASPO5, DHORD2, DHORD5, FDH4pp, FDH5pp, FRD2, FRD3, GLYCT02, GLYCT03, GLYCT04, NADH16pp, NADH17pp, NADH18pp, SUCDi | asp-L[c] = -0.0009977, co2[c] = 0.001074, dhor-S[c] = -0.1449, for[p] = -0.001074, fum[c] = 5.281, glx[c] = 0.000293, glyclt[c] = -0.000293, h[c] = -83.35, h[p] = 62.51, iasp[c] = 0.0009977, nad[c] = 20.84, nadh[c] = -20.84, orot[c] = 0.1449, q8[c] = -26.27, q8h2[c] = 26.27, succ[c] = -5.281 | 108 |
| DMPPS, IPDDI, IPDPS | dmpp[c] = 0.0001218, h2mb4p[c] = -0.001046, h2o[c] = 0.001046, h[c] = -0.001046, ipdp[c] = 0.0009246, nad[c] = 0.001046, nadh[c] = -0.001046 | 3 |
| DSERDHr, LSERDHr, SERD_D, SERD_L, TRPAS2, TRPS1, TRPS2, TRPS3 | 3ig3p[c] = -0.0249, g3p[c] = 0.0249, h2o[c] = 0.0249, nh4[c] = 3.104, pyr[c] = 3.104, ser-L[c] = -3.129, trp-L[c] = 0.0249 | 8 |
| EX_fe2(e), EX_fe3(e), EX_h2o(e), EX_h(e), EX_o2(e), FE2tex, FE3tex, FEROpp, H2Otex, Htex, O2tex | fe2[p] = 0.003308, fe3[p] = 0.003112, h2o[p] = -17.6, h[p] = 3.977, o2[p] = 13.13 | 2 ** |

* computed with `metatool` only, MILP method took too long. Number of computed EFMs varies from run to run.

** computed with MILP method only. `metatool` did not find any pathways.

^{new} This is only a module with thermodynamic constraints.

A.1.13 Summary on *H. pylori* iIT341

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|--|------|
| ACOATA, KAS14, KAS15 | accoa[c] = -0.2317, actACP[c] = 0.2317, co2[c] = 0.2317, coa[c] = 0.2317, h[c] = -0.2317, malACP[c] = -0.2317 | 2 |
| ALAt2, ALAt4, EX_nh4(e), EX_orn(e), EX_proL(e), G5SADr, GLUDy, NAT3_1, NH4t, ORNTA, ORNt2r, P5CD, P5CR, PROt2r, PROt4r, PUTA3 | akg[c] = -2.731, ala-L[c] = 10, ala-L[e] = -10, glu-L[c] = 3.376, h2o[c] = 1.441, h[c] = 8.068, h[e] = -8.863, nad[c] = -0.6452, nadh[c] = 0.6452, nadp[c] = 2.086, nadph[c] = -2.086, nh4[c] = -21.78, orn[c] = -1.927, pro-L[c] = 0.1455 | 18 |
| FTHFLi, GARFTi, GART | 10fthf[c] = -0.3514, adp[c] = 0.02359, atp[c] = -0.02359, fgam[c] = 0.375, for[c] = -0.02359, gar[c] = -0.375, h[c] = 0.375, pi[c] = 0.02359, thf[c] = 0.3514 | 2 |
| H2CO3D, H2CO3D2, HCO3E | co2[c] = -2.233, h2o[c] = -2.233, h[c] = 2.233, hco3[c] = 2.233 | 2 |
| TRPS1, TRPS2, TRPS3 | 3ig3p[c] = -0.03741, g3p[c] = 0.03741, h2o[c] = 0.03741, ser-L[c] = -0.03741, trp-L[c] = 0.03741 | 2 |

A.1.14 Summary on *M. barkeri* iAF692 grown on methanol

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|--|------|
| ACGK, ACOTA, AGPR, DROPPRx, DROPPRy, G5SADr, G5SD2, GAPD, GAPD_nadp-, GLU5K, GLUDxi, GLUDyi, IPDDI3x, IPDDI3y, MDH, MDHi2, MDHy, ORNCD, ORNTAC, P5CRx, SLDx, SLDxi2, SLDy | 13dpg[c] = -0.06378, 25dhpp[c] = -0.00015, 25dthpp[c] = 0.00015, 3spyr[c] = 0.0005603, adp[c] = 0.01638, akg[c] = -0.22, atp[c] = -0.01638, dmpp[c] = 0.002206, f420-2[c] = -0.002206, f420-2h2[c] = 0.002206, g3p[c] = 0.06378, glu-L[c] = 0.2036, h2o[c] = 0.2364, h[c] = -0.2022, ipdp[c] = -0.002206, mal-L[c] = -0.1162, nad[c] = 0.007163, nadh[c] = -0.007163, nadp[c] = 0.195, nadph[c] = -0.195, nh4[c] = -0.2298, oaa[c] = 0.1162, orn[c] = 0.009791, pi[c] = 0.08016, pro-L[c] = 0.006592, sl-L[c] = -0.0005603 | 156 |
| ADK1, ADK2, ADK3, ADK4, H4MPTGL_atp-, H4MPTGL_gtp-, NDPK1, NDPK5, NDPK9, PPK2, RNDR2, RNTR2 | adp[c] = 0.3672, amp[c] = -0.175, atp[c] = -0.1922, dgtp[c] = 0.0005867, gdp[c] = -0.01674, glu-L[c] = -0.0006436, gtp[c] = 0.01616, h2o[c] = 0.0005867, h4mpt[c] = -0.0006436, h4spt[c] = 0.0006436, h[c] = 0.0006436, pi[c] = 0.0006436, ppi[c] = 0.0001292, pppi[c] = -0.0001292, trdox[c] = 0.0005867, trdrd[c] = -0.0005867 | 28 |
| ADKd, DADK, NDPK8, RNDR1, RNTR1 | atp[c] = -0.0009025, datp[c] = 0.0009025, h2o[c] = 0.0009025, trdox[c] = 0.0009025, trdrd[c] = -0.0009025 | 3 * |
| F4D, F4NH, F4RHi, H2td | f420-2[c] = 2.534, f420-2h2[c] = -2.534, h2[c] = -0.7356, h[c] = -5.885, h[e] = 5.885, mphen[c] = -3.269, mphenh2[c] = 3.269 | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 ^{new} | adp[c] = 0.008677, atp[c] = -0.008677, dudp[c] = 0.001489, h2o[c] = 0.001489, trdox[c] = 0.001489, trdrd[c] = -0.001489, udp[c] = -0.01017, utp[c] = 0.008677 | 2 |
| TRPS1, TRPS2, TRPS3 | 3ig3p[c] = -0.001695, g3p[c] = 0.001695, h2o[c] = 0.001695, ser-L[c] = -0.001695, trp-L[c] = 0.001695 | 2 |
| UMPk, URIDK1 | adp[c] = 0.009305, atp[c] = -0.009305, udp[c] = 0.009305, ump[c] = -0.009305 | 2 |

* metatool loses sometimes one EFM

^{new} This is only a module with thermodynamic constraints.

A.1.15 Summary on *M. tuberculosis* iNJ661

| Module | Exchange flux (right hand side of the module) | EFMs |
|--|---|--------------|
| ACACCT, ACGK, ACKr, ACONT, ACOTA, ADK1, ADK2, ADK3, ADK4, ADNK1, ADNK3, ADNK4, AGPR, AICART, AICART2, ATPS4r, CITLr, CO2t, CY01a, CY01b, CYSTGL, CYTBD, CYTBD2, DHORD2, DHORD3, DHORD5, ENO, EX_co2(e), EX_h2(e), EX_h2co3(e), EX_h2o(e), EX_h(e), EX_lac.L(e), EX_ppa(e), EX_succ(e), FHL, FORMCOAL, FRD, FRD2, FRD3, FRD5, FRD02r, FRD03r, FTHFL, FUM, G5SD, G5SD2, GCCa, GCCb, GCCc, GHMT2, GLU5K, GLUDx, GLX01, GLYCL, H2CO3D, H2CO3TP, H2Ot, H2td, HSDx, HSDy, HSK, HSST, ICL, LDH.L, L.LACD2, L.LACD3, L.LACT2r, MALS, MDH, ME1, MMCD, MME, MMM2r, MTHFC, MTHFD, MTHFD2, NADH10, NADH2r, NADH5, NADH9, NADTRHD, NDPK1, NDPK5, NDPK9, OCOAT1r, ORNTA, ORNTAC, OXACOAL, OXCDC, P5CR, P5CRx, PEPCK_re, PFL, PGCD, PGM, PTHCLR1, PTHCLR2, PPAtr, PPKr, PPCSCT, PPK2, PROD2, PSERT, PSP.L, PTAr, PYK, QRr, SHSL1r, SHSL4r, SUCCT2r, SUCCOAS, THD1, THRD_Lr, THRS _{new} | 10fthf[c] = 0.02051, 1pyr5c[c] = -0.007205, 2obut[c] = 0.00458, 3pg[c] = -2.817, aacoa[c] = -0.0001514, ac[c] = -0.00297, acac[c] = 0.0001514, accoa[c] = 0.4068, adn[c] = -0.003235, adp[c] = -3.891, aicar[c] = -0.02263, akgl[c] = -0.8243, amp[c] = -0.07882, aspsa[c] = -0.01356, atp[c] = 3.973, co2[c] = -0.718, coa[c] = -0.4244, cys-L[c] = -0.001821, cyst-L[c] = 0.001821, dhor-S[c] = -0.02065, fprica[c] = 0.02263, fum[c] = -0.03006, gdp[c] = -0.00956, glu5sa[c] = 0.007205, glu-L[c] = 0.8108, gly[c] = 0.03803, gtp[c] = 0.00956, h2o[c] = 3.325, h[c] = -9.37, h[e] = -0.6588, icit[c] = 0.6822, mlthf[c] = 0.006381, mmcoa-S[c] = 0.01328, nad[c] = 3.831, nadh[c] = -3.831, nadp[c] = -0.09375, nadph[c] = 0.09375, nh4[c] = -0.8758, o2[c] = -0.9784, oaa[c] = 0.07757, orn[c] = 0.006286, orot[c] = 0.02065, pep[c] = 0.01389, pi[c] = -1.265, pro-L[c] = 0.007205, pyr[c] = 0.07894, ser-L[c] = 0.0118, succ[c] = -0.004519, succoa[c] = 0.004519, thf[c] = -0.02689, thr-L[c] = 0.007157 | ≥ 300 * |
| ARI, DCPDPP, DCPDPP2, DCPE, DCPT, DCPT2 | dec_d.tb[c] = -0.002938, decda.tb[c] = 0.002938, h2o[c] = -0.002938, pi[c] = 0.002938, ppi[c] = 0.002938, prpp[c] = -0.002938 | 2 |
| DCPDPS, DPPS, UDCPDPS, UDPDPS, UDPDPS2 | frdp[c] = -0.0002459, ipdp[c] = -0.001967, ppi[c] = 0.001967, udcpdp[c] = 0.0002459 | 3 |
| DESAT16, FACOAL160, FACOAL161, FAS160, FAS161 | amp[c] = 0.0004313, atp[c] = -0.0004313, co2[c] = 0.02898, coa[c] = 0.02855, h2o[c] = 0.03013, h[c] = -0.08752, hdca[c] = 0.02798, hdcea[c] = 0.0005711, malcoa[c] = -0.02898, nadp[c] = 0.05854, nadph[c] = -0.05854, o2[c] = -0.0005711, pmtcoa[c] = 0.0004313, ppi[c] = 0.0004313, ttdca[c] = -0.02898 | 2 |

Continued on next page.

APPENDIX A. COMPUTATIONAL RESULTS ON FLUX MODULES

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|---|------|
| DESAT18, FACOAL180, FACOAL181, FAS180, FAS181 | $\text{amp}[c] = 7.318e - 05$, $\text{atp}[c] = -7.318e - 05$, $\text{co2}[c] = 0.01147$, $\text{coa}[c] = 0.0114$, $\text{h2o}[c] = 0.01265$, $\text{h}[c] = -0.03501$, $\text{hdca}[c] = -0.01147$, $\text{malcoa}[c] = -0.01147$, $\text{nadp}[c] = 0.02353$, $\text{nadph}[c] = -0.02353$, $\text{o2}[c] = -0.0005873$, $\text{ocdca}[c] = 0.01089$, $\text{ocdcea}[c] = 0.0005141$, $\text{odecoa}[c] = 7.318e - 05$, $\text{ppi}[c] = 7.318e - 05$ | 3 |
| DMPPS, IPDDI, IPDPS | $\text{dmpp}[c] = 0.0002459$, $\text{h2mb4p}[c] = -0.002704$, $\text{h2o}[c] = 0.002704$, $\text{h}[c] = -0.002704$, $\text{ipdp}[c] = 0.002459$, $\text{nad}[c] = 0.002704$, $\text{nadh}[c] = -0.002704$ | 3 |
| G16MTM2, G16MTM6, MANAT1, MANAT2 | $\text{Ac1PIM1}[c] = 8.775e - 05$, $\text{Ac1PIM2}[c] = 0.0002771$, $\text{PIM1}[c] = -0.0007399$, $\text{PIM2}[c] = 0.0003751$, $\text{coa}[c] = 0.0003648$, $\text{gdp}[c] = 0.0006522$, $\text{gdpmann}[c] = -0.0006522$, $\text{h}[c] = 0.0006522$, $\text{pmtcoa}[c] = -0.0003648$ | 2 |
| NDPK2, NDPK6, RNDR4, RNTR4 | $\text{adp}[c] = 0.0211$, $\text{atp}[c] = -0.0211$, $\text{dudp}[c] = 0.0001907$, $\text{h2o}[c] = 0.0001907$, $\text{trdox}[c] = 0.0001907$, $\text{trdrd}[c] = -0.0001907$, $\text{udp}[c] = -0.02129$, $\text{utp}[c] = 0.0211$ | 2 |
| TRPS1, TRPS2, TRPS3 | $\text{3ig3p}[c] = -0.00105$, $\text{g3p}[c] = 0.00105$, $\text{h2o}[c] = 0.00105$, $\text{ser-L}[c] = -0.00105$, $\text{trp-L}[c] = 0.00105$ | 2 |
| UMPK, URIDK1 | $\text{adp}[c] = 0.016$, $\text{atp}[c] = -0.016$, $\text{udp}[c] = 0.016$, $\text{ump}[c] = -0.016$ | 2 |

* both tools `metatool` and the MILP enumeration failed to enumerate all EFMs. We aborted the MILP computation after the computation of 300 elementary modes.

^{new} This is only a module with thermodynamic constraints.

A.1.16 Summary on *S. aureus* iSB619

| Module | Exchange flux (right hand side of the module) | EFMs |
|---|--|------|
| AGMT, ARGDC, ARGN, ORNDC | arg-L[c] = -13.72, co2[c] = 0.001374, h2o[c] = -13.72, h[c] = -0.001374, orn[c] = 13.71, ptrc[c] = 0.001374, urea[c] = 13.72 | 2 |
| AKP1, DNMPPA, DNTPPA, PPA | ahdt[c] = -0.00206, dhnp[c] = 0.00206, h2o[c] = -137.9, h[c] = 137.9, pi[c] = 275.9, ppi[c] = -137.9 | 2 |
| ALAR, ALAt2r, DALAt2r, EX_ala_D(e), EX_ala_L(e) | ala-D[c] = -29.6, ala-L[c] = 0.4876, h[c] = -29.12, h[e] = 29.12 | 2 |
| CAT, CO2t, DKMPPD, DKMPPD2, EX_co2(e), EX_for(e), EX_h(e), FDHr, FORt3, PDH, PFLr, POX2, PTAr | 2kmb[c] = 0.0006868, accoa[c] = 491.9, actp[c] = -0.1889, co2[c] = -516, coa[c] = -491.9, dkmp[c] = -0.0006868, for[c] = 0.5947, h2o[c] = 172.3, h[c] = -139.9, h[e] = -457.2, nad[c] = -286.4, nadh[c] = 286.4, o2[c] = -86.16, pi[c] = 0.1895, pyr[c] = -491.7 | 4 |
| CYTD, CYTDK2, CYTK1, CYTK2, DCMPDA, MEVK1, MEVK2, MEVK3, MEVK4, NDPK1, NDPK2, NDPK3, NDPK5, NDPK6, NDPK7, NDPK9, RNDR2, RNDR3, RNDR4, RNTR2, RNTR3, RNTR4, UMPK, URIDK2r, URIK1, URIK2, URIK3 | 5pmev[c] = 0.00206, adp[c] = 210.8, atp[c] = -210.8, cmp[c] = -83.29, ctp[c] = 83.44, cytd[c] = -0.524, dcmp[c] = 0.0364, dgdpc[c] = 0.04121, dump[c] = 0.05013, gdp[c] = -0.6016, gtp[c] = 0.5604, h2o[c] = -0.2088, h[c] = 0.1895, mev-R[c] = -0.00206, nh4[c] = 0.3365, trdox[c] = 0.1277, trdrd[c] = -0.1277, udp[c] = -26.36, ump[c] = -7.938, utp[c] = 34.58 | 192 |
| D.LACT2, EX_lac_D(e), EX_lac_L(e), LDH_D, LDH_L, L.LACT2r | h[c] = -892.1, h[e] = 446, nad[c] = 446, nadh[c] = -446, pyr[c] = -446 | 2 |
| MDH, ME1_rev, PPK, PYK | adp[c] = -403.5, atp[c] = 403.5, co2[c] = 55.82, h[c] = -230.9, mal-L[c] = -228.4, nad[c] = -228.4, nadh[c] = 228.4, oaa[c] = 172.6, pep[c] = -403.5, pyr[c] = 459.3 | 2 |
| NAt3, NAt3.1 | h[c] = 372, h[e] = -372, na1[c] = -372, na1[e] = 372 | 2 |
| NDPK8, RNDR1, RNTR1 | adp[c] = -0.1044, atp[c] = 0.05219, dadp[c] = 0.1044, datp[c] = -0.05219, h2o[c] = 0.05219, trdox[c] = 0.05219, trdrd[c] = -0.05219 | 2 |

A.1.17 Summary on *S. cerevisiae* iND750 grown on D-glucose

| Module | Exchange flux (right hand side of the module) | EFMs |
|--|--|---------|
| 2DDA7Ptm, 34HPPt2m, AASAD1, AASAD2, ACGKm, ACOAH, ACOTaim, ACRNtm, ADHAPR_SC, AGPRim, AHSERL2, ALDD2y, ASADi, ASPKi, ASPTA, ASPTAm, ASPt2m, ATPtm_H, CITtam, CITbm, CO2tm, CRNCARtm, CRNtim, CSNAT, CSNATim, CYSS, CYSTL, DDPa, DDPAm, DHAPtm, DHFRi, DHFRim, DHFtm, DICtm, E4Ptm, ENO, FDH, FDNG, FRDcm, FRDm, FTHFL, FUM, FUMm, G3PD1ir, G3PD1irm, G3PDm, G5SD, G5SD2, G6PI, GAT1_SC, GAT2_SC, GCC2am, GCC2bim, GCC2cm, GCCam, GCCbim, GCCcm, GHMT2r, GHMT2rm, GLU5K, GLUDxi, textttGLUDy, GLUK, GLUt2m, GLYC3Ptm, GLYCLm, GLYt2m, H2Otm, HCO3E, HEX1, HEX7, HSDxi, HSDyi, HSERTA, ICDHxm, ICDHym, MALtm, MDH, MDHm, MLTHFtm, MTHFC, MTHFD, MTHFD2, NADH2_u6cm, NADH2_u6m, NH4tm, OAAt2m, ORNTA, ORNTACim, ORNt3m, PC, PDHcm, PDHm, PGCD, PGI, PGM, Pit2m, PPND, PPND2, PSERT, PSP_L, PYK, PYRDC, PYRt2m, SBT_D2, SBTR, SERATi, SERt2m, SHSL1, SHSL4r, SUCCtm, SUCD1m, SUCD2_u6m, SUCD3_u6m, SUCFUMtm, THFtm, THRA, THRD_L, TYRTA, TYRTAi, TYRTAm, TYRt2m ^{new} | 10fthf[c] = 0.02553, 1ag3p_SC[c] = 2.404e - 05, 2dda7p[c] = 0.02572, 3pg[c] = -1.428, L2aadp6sa[c] = 0.02785, L2aadp[c] = -0.02785, accoa[c] = 0.07685, accoa[m] = 0.127, adp[c] = -6.283, adp[m] = 5.439, akc[c] = -0.1932, akc[m] = 0.02785, amp[c] = 0.02785, asp-L[c] = 0.08607, atp[c] = 6.255, atp[m] = -5.439, co2[c] = 2.417, co2[m] = -0.1022, coa[c] = -0.07444, coa[m] = -0.127, cys-L[c] = 0.0006423, dcacoa[c] = -4.808e - 05, ddcacoa[c] = -0.0001442, dhap[c] = -0.002404, dhf[c] = -0.0003504, e4p[c] = -0.02572, f6p[c] = 0.8337, for[c] = -0.0002141, fum[c] = -0.03646, g6p[c] = 0.1663, glc-D[c] = -1, glu5sa[c] = 0.01603, glu-L[c] = 0.2328, gly[c] = 0.0378, h2o[c] = 10.26, h2o[m] = -10.24, h2s[c] = -0.005577, h[c] = -13.98, h[m] = 10.98, hco3[c] = 0.0651, hcys-L[c] = 0.004934, hdcoa[c] = -0.0004087, icit[m] = -0.09915, mlthf[c] = 0.007105, nad[c] = 1.502, nad[m] = 0.02785, nadh[c] = -1.502, nadh[m] = -0.02785, nadp[c] = -0.1855, nadp[m] = -0.07335, nadph[c] = 0.1855, nadph[m] = 0.07335, nh4[c] = -0.4601, nh4[m] = -0.01875, oaa[m] = 0.09915, ocdycacoa[c] = -0.0002163, odecoc[c] = -0.0005769, orn[c] = 0.01564, pep[c] = 0.02572, pi[c] = -5.881, pi[m] = 5.439, pmtcoa[c] = -0.000649, pphn[c] = -0.009927, ppi[c] = 0.02785, pyr[c] = 0.04189, pyr[m] = 0.128, q6[m] = -4.846, q6h2[m] = 4.846, ser-L[c] = 0.022, stcoa[c] = -0.0001202, tdcoa[c] = -0.0002404, thf[c] = -0.03228, thr-L[c] = 0.03738, tyr-L[c] = 0.009927 | ≥ 300 * |
| ACONT, ACONTm, CITtcm | cit[m] = -0.09915, icit[m] = 0.09915 | 2 |

Continued on next page.

A.1. FLUX MODULES

| Module | Exchange flux (right hand side of the module) | EFM _s |
|---|--|------------------|
| ADK1, ADK3, ADK4, MEVK1, MEVK2, MEVK3, MEVK4, NDPK1, NDPK2, NDPK3, NDPK5, NDPK6, NDPK9, RNDR2, RNDR4, RNTR2, RNTR4 ^{new} | $5pmev[c] = 0.001285$, $adp[c] = 0.4597$, $amp[c] = -0.1018$, $atp[c] = -0.3579$, $cdp[c] = 0.002647$, $ctp[c] = -0.002647$, $dgdpc[c] = 0.0002336$, $dudpc[c] = 0.0005839$, $gdpc[c] = -0.09014$, $gtp[c] = 0.08991$, $h2o[c] = 0.0008175$, $h[c] = 0.001285$, $mev-R[c] = -0.001285$, $trdox[c] = 0.0008175$, $trdrd[c] = -0.0008175$, $udp[c] = -0.1681$, $utp[c] = 0.1675$ | 80 |
| DESAT18, FACOAL180, FACOAL181, FAS180, FAS181, FACOAL140, FACOAL160, FACOAL180, FAS160, FAS160COA, FAS180, FAS180COA | $co2[c] = 0.00654$, $coa[c] = 0.00654$, $h2o[c] = 0.00654$, $h[c] = -0.01962$, $malcoa[c] = -0.00654$, $nadp[c] = 0.01308$, $nadph[c] = -0.01308$, $pmtcoa[c] = 0.002398$, $stcoa[c] = 0.002071$, $tdcoa[c] = -0.004469$ | 5 |
| FBA, FBA3, PFK, PFK_3, TALA | $adp[c] = 0.7271$, $atp[c] = -0.7271$, $dhap[c] = 0.7271$, $e4p[c] = -0.001171$, $f6p[c] = -0.7282$, $g3p[c] = 0.7282$, $h[c] = 0.7271$, $s7p[c] = 0.001171$ | 2 |
| GALT, GALU, UGLT | $g1p[c] = -0.1632$, $h[c] = -0.1632$, $ppi[c] = 0.1632$, $udpg[c] = 0.1632$, $utp[c] = -0.1632$ | 2 |
| GBEZ, GLCS2, GLYGS | $glycogen[c] = 0.05046$, $h[c] = 0.05046$, $udp[c] = 0.05046$, $udpg[c] = -0.05046$ | 2 |
| GK1, GK2, NDPK8, RNDR1, RNTR1 | $adp[c] = -0.0001168$, $atp[c] = -0.0002336$, $dadpc[c] = 0.0003504$, $gdpc[c] = 0.0002336$, $gmp[c] = -0.0002336$, $h2o[c] = 0.0003504$, $trdox[c] = 0.0003504$, $trdrd[c] = -0.0003504$ | 6 |

* both tools `metatool` and the MILP enumeration failed to enumerate all EFM_s. We aborted the MILP computation after the computation of 300 elementary modes.

^{new} This is only a module with thermodynamic constraints.

A.2 Optimal-Yield Elementary Flux Modes

A.2.1 Elementary Flux Modes of *E. coli* iJR904 grown on *L-Arginine*

Only the reactions with variable flux rate are listed:

$$\begin{aligned}
 \text{EFM} = & \left\{ \begin{array}{l} \{\text{SUCOAS}\}, \\ \{\text{ACCOAL, PPCSCT}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{KAS15}\}, \\ \{\text{ACOATA, KAS14}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{ADK3, NDPK1, RNTR2}\}, \\ \{\text{ADK1, NDPK1, RNTR2}\}, \\ \{\text{ADK1, ADK3, RNTR2}\}, \\ \{\text{ADK3, NDPK1, NDPK5, RNDR2}\}, \\ \{\text{ADK1, ADK3, NDPK5, RNDR2}\}, \\ \{\text{ADK1, NDPK1, NDPK5, RNDR2}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{ALAR}_i\}, \\ \{\text{ALAR}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{ALATA}_L, \text{VALTA}\}, \\ \{\text{ALATA}_L, \text{VPAMT}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{ASPO5, DHORD2, SUCD1}_i, \text{SUCD4}\}, \\ \{\text{ASPO3, DHORD2, SUCD1}_i, \text{SUCD4}\}, \\ \{\text{ASPO4, DHORD5, FRD2, SUCD1}_i, \text{SUCD4}\}, \\ \{\text{ASPO3, DHORD5, FRD2, SUCD1}_i, \text{SUCD4}\}, \\ \{\text{ASPO5, DHORD5, FRD2, SUCD1}_i, \text{SUCD4}\}, \\ \{\text{ASPO4, DHORD2, FRD2, SUCD1}_i, \text{SUCD4}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{ASPT, ASPTA, FUM, GLUD}_y, \text{MDH, PPCK, PYK}\}, \\ \{\text{ASPTA, FUM, GLUD}_y, \text{MDH, ME1, NADTRHD, PPCK}\}, \\ \{\text{ASPTA, FUM, GLUD}_y, \text{MDH, NADTRHD, PPCK, PYK}\}, \\ \{\text{ASPT, ASPTA, FUM, GLUD}_y, \text{MDH, ME1, PPCK}\}, \\ \{\text{ASPTA, FUM, GLUD}_y, \text{MDH, ME2, NADTRHD, PPCK}\}, \\ \{\text{ASPT, ASPTA, FUM, GLUD}_y, \text{MDH, ME2, PPCK}\}, \\ \{\text{ASPTA, FUM, GLUD}_y, \text{IDOND, IDOND2, MDH, PPCK, PYK}\}, \\ \{\text{ASPTA, FUM, GLUD}_y, \text{IDOND, IDOND2, MDH, ME2, PPCK}\}, \\ \{\text{ASPTA, FUM, GLUD}_y, \text{IDOND, IDOND2, MDH, ME1, PPCK}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{GALU}_i\}, \\ \{\text{GALU}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{NDPK2, RNDR4}\}, \\ \{\text{NDPK2, NDPK6, RNTR4}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{NDPK3, RNTR3}\}, \\ \{\text{NDPK3, NDPK7, RNDR3}\} \end{array} \right\} \\
 & \times \left\{ \begin{array}{l} \{\text{RNTR1}\}, \\ \{\text{NDPK8, RNDR1}\} \end{array} \right\}
 \end{aligned}$$

Appendix B

Notation

Here I summarize all the notation I use in the thesis:

- \mathbb{R} denotes the set of real numbers.
- \mathbb{N} denotes the set of natural numbers (excluding 0).
- $\mathbb{R}_\infty := \mathbb{R} \cup \{-\infty, \infty\}$.
- \mathbf{P} denotes the complexity class of problem solvable in polynomial time
- \mathbf{NP} denotes the complexity class of problems solvable in non-deterministic polynomial time
- \mathbf{RP} denotes the complexity class of problems solvable with a probabilistic algorithm in polynomial time
- v_i denotes the i .th element of a vector v .
- v_A denotes the vector containing only the elements of the index set A .
- S_{ij} denotes the element at row i and column j of S .
- S_{i*} denotes the i .th row of matrix S .
- S_{*i} denotes the i .th column of matrix S .
- S_A denotes the matrix containing only the elements of the index set $A \subset \mathbb{N}^2$. If it is clear from the context, A may also be of one dimension and then S_A only contains the columns respectively the rows indexed by A .
- $(S|T)$ for matrices S, T denotes the horizontal concatenation of S and T .
- $\ker(S)$ for a matrix S denotes the null space of S ($\ker(S) := \{x : Sx = 0\}$)
- pr_A is the linear map $v \mapsto v_A$.
- $\text{pr}_A P := \text{pr}_A(P) := \{v_A : v \in P\}$
- $P \times Q := \{v : v_A \in P, v_B \in Q\}$ for $P \subseteq \mathbb{R}^A, Q \subseteq \mathbb{R}^B$
- $\prod_{i \in I} P^{A_i} := \{v : v_{A_i} \in P^{A_i} \text{ for all } i \in I\}$
- $\mathbb{1}$ denotes the all ones vector of appropriate size.

- 0^A denotes the all zero vector for components A . If the dimension is clear, I also just write 0.
- $X \in \{-, 0, +\}^E$ is called a signed subset of E .
- (X^+, X^-) with $X^+ \cap X^- = \emptyset$ is another notation of a signed set $X \in \{-, 0, +\}^E$, where

$$X^+ = \{i \in E : X_i = +\} \text{ and } X^- = \{i \in E : X_i = -\}.$$
- $\underline{X} := X^+ \dot{\cup} X^-$ is the support of a signed set (X^+, X^-) .
- $\text{sign} : \mathbb{R}^E \rightarrow \{-, 0, +\}^E$ is the component wise sign operation, see Definition 2.5.4.
- $(X^+, X^-) \subseteq (Y^+, Y^-) \Leftrightarrow X^+ \subseteq Y^+ \wedge X^- \subseteq Y^-$.
- $\text{supp}(v)$ denotes the support of v .
- \bar{A} for a $A \subseteq \mathbb{R}^n$ denotes the closure of A in the topological sense.
- $\text{aff}(A)$ for $A \subseteq \mathbb{R}^n$ denotes the affine hull of A
- $\text{conv}(A)$ for $A \subseteq \mathbb{R}^n$ denotes the convex hull of A
- $\text{cone}(A)$ for $A \subseteq \mathbb{R}^n$ denotes the conical hull of A
- $\text{span}(A)$ for $A \subseteq \mathbb{R}^n$ denotes the linear space spanned by A
- $\langle G \rangle$ for a digraph G denotes the transitive closure of G
- $\langle L \rangle$ for a lattice L denotes the closure of the lattice L
- $-r$ for a (pseudo-)reaction $r \in \bar{\mathcal{R}}$ denotes the reversed reaction (Def. 2.1.2)
- $r^+ := \{m \in \mathcal{M} : S_{mr} > 0\}$ for $r \in \bar{\mathcal{R}}$ is the set of metabolites produced by r
- $r^- := \{m \in \mathcal{M} : S_{mr} < 0\}$ for $r \in \bar{\mathcal{R}}$ is the set of metabolites consumed by r
- $\underline{r} := \{m \in \mathcal{M} : S_{mr} \neq 0\}$ for $r \in \bar{\mathcal{R}}$ is the set of metabolites involved in r
- $\bar{\ell} \in \mathbb{R}^{\bar{\mathcal{R}}}$ for given lower and upper bounds ℓ, u as defined by $\bar{\ell}_r = \ell_r, \bar{\ell}_{-r} = -u_r$ for all $r \in \mathcal{R}$.
- $\text{EFM}(P)$ denotes the set of elementary modes of the flux space P (Def. 2.4.2)
- \mathcal{N}^s for a sign-vector $s \in \{-, 0, +\}^{\mathcal{I}}$ denotes the subnetwork according to Def. 2.6.5
- 2^A for a set A denotes the powerset of A .
- $M \setminus A$ for a (oriented) matroid M denotes the (oriented) matroid obtained by deletion of A from M

-
- $M|_A$ for a (oriented) matroid M denotes the (oriented) matroid obtained by restriction of M to A
 - $r(A)$ for a set of elements A in a matroid denotes the rank of A

Usual names of variables are:

- \mathcal{R} for the set of reactions,
- $\overline{\mathcal{R}}$ for the set of pseudo-reactions (Def. 2.1.2)
- \mathcal{M} for the set of metabolites,
- \mathcal{I} for the set of internal reactions,
- $\mathcal{E} := \mathcal{R} \setminus \mathcal{I}$ for the set of exchange reactions,
- S for the stoichiometric matrix,
- \mathcal{C} set of circuits (either internal or all),
- $\overline{\mathcal{C}}$ denotes the set of reactions in internal cycles,
- Irrev for the set of irreversible reactions,
- v for the flux vector,
- μ for the potentials,
- μ^0 for the equilibrium constants of formation
- $\Delta\mu$ for the potential differences,
- $\Delta\mu^0$ for the equilibrium constants of reactions
- c for the cost function (sometimes also concentrations),
- ℓ for lower bounds,
- u for upper bounds,
- ℓ^v for lower bounds on the fluxes,
- u^v for upper bounds on the fluxes,
- ℓ^μ for lower bounds on the potentials,
- u^μ for the upper bounds on the potentials,
- P is a flux space
- F is the steady-state flux space (Def. 6.3.1)

- T is the thermodynamically constrained flux space (Def. 6.3.1)
- F^A for a flux module A is the steady-state flux space with the interface flux of module A . (Def. 6.3.1)
- T^A for a flux module A is the thermodynamically constrained flux space with the interface flux of module A . (Def. 6.3.1)
- $\tilde{T}^A = \text{pr}_A(T)$ (Def. 6.3.1)
- P^A for a k -module A is the flux space of the module A .
- \mathcal{P} is the set of paths (each consumed metabolite is also produced and vice versa),
- \mathcal{P}_s is the set of steady-state paths
- \mathcal{F} denotes the set of fundamental circuits
- \mathcal{K} denotes the set of strongly connected components
- Mod is a family of modules

Bibliography

- [1] 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*, 95:51–60, 2009. [187]
- [2] V. Acuña, A. Marchetti-Spaccamela, M.-F. Sagot, and L. Stougie. A note on the complexity of finding and enumerating elementary modes. *BioSystems*, 99:210–214, 2010. [154]
- [3] R. Adadi, B. Volkmer, R. Milo, M. Heinemann, and T. Shlomi. Prediction of microbial growth rate versus biomass yield by a metabolic network with kinetic parameters. *PLoS Computational Biology*, 8(7):e1002575, 2012. [7]
- [4] A. V. Aho, M. R. Garey, and J. D. Ullman. The transitive reduction of a directed graph. *SIAM Journal on Computing*, 1(2):131–137, 1972. [205, 206, 209, 213]
- [5] R. A. Alberty. *Thermodynamics of Biochemical Reactions*. Massachusetts Institute of Technology, Cambridge, MA, 2003. [5, 26, 31, 101]
- [6] E. Almaas, B. Kovács, T. Vicsek, Z. Oltvai, and A.-L. Barabási. Global organization of metabolic fluxes in the bacterium escherichia coli. *Nature*, 427:839–843, 2004. [217]
- [7] K. Ballerstein, A. von Kamp, S. Klamt, and U.-U. Haus. Minimal cut sets in metabolic networks are elementary modes in a dual network. *Bioinformatics*, 28(3):381–387, 2012. [19, 187]
- [8] 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]
- [9] D. A. Beard, E. Babson, E. Curtis, and H. Qian. Thermodynamic constraints for biochemical networks. *Journal of Theoretical Biology*, 228:327–333, 2004. [2, 8, 23, 27, 28, 53]
- [10] D. A. Beard, S. dan Liang, and H. Qian. Energy balance for analysis of complex metabolic networks. *Biophysical Journal*, 83:79–86, 2002. [5, 26, 218]

- [11] D. A. Beard and H. Qian. Thermodynamic-based computational profiling of cellular regulatory control in hepatocyte metabolism. *American Journal of Physiology - Endocrinology and Metabolism*, 288:E633–E644, 2005. [1, 7, 30, 218]
- [12] B. D. Bennet, E. H. Kimball, M. Gao, R. Osterhout, S. J. Van Dien, and J. D. Rabinowith. Absolute metabolite concentrations and implied enzyme active site occupancy in escherichia coli. *Nature Chemical Biology*, 5:593–599, 2009. [30, 97]
- [13] R. E. Bixby and W. H. Cunningham. Matroids, graphs, and 3-connectivity. In J. A. Bondy and W. T. Tutte, editors, *Graph Theory and Related Topics*. Acad. Press. [114, 149, 150]
- [14] R. E. Bixby and W. H. Cunningham. Matroid optimization and algorithms. Technical report, Rice University, 1990. www.caam.rice.edu/caam/trs/90/TR90-15.pdf. [114]
- [15] A. Björner, M. L. Vergnas, B. Sturmfels, N. White, and G. M. Ziegler. *Oriented Matroids*. Cambridge University Press, 1999. [8, 21, 22, 24, 25]
- [16] B. A. Boghigian, H. Shi, K. Lee, and B. A. Pfeifer. Utilizing elementary mode analysis, pathway thermodynamics, and a genetic algorithm for metabolic flux determination and optimal metabolic network design. *BMC Systems Biology*, 4:49, 2010. [36]
- [17] A. Bordbar, N. E. Lewis, J. Schellenberger, B. O. Palsson, and N. Jamshidi. Insight into human alveolar macrophage and m. tuberculosis interactions via metabolic reconstructions. *Molecular Systems Biology*, 6:422, 2010. [217]
- [18] A. Brøndsted. *An Introduction to Convex Polytopes*. Graduate Texts in Mathematics. Springer, 1982. [19]
- [19] J. G. Bundy, B. Papp, R. Harmston, R. A. Browne, E. M. Clayson, N. Burton, R. J. Reece, S. G. Oliver, and K. M. Brindle. Evaluation of predicted network modules in yeast metabolism using nmr-based metabolite profiling. *Genome Research*, 17(4):510–519, 2007. [198]
- [20] A. Burgard, P. Pharkya, and C. Maranas. Optknock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. *Biotechnology and Bioengineering*, 84:647–657, 2003. [1, 7, 8]
- [21] A. P. Burgard, N. E. V., C. H. Schilling, and C. D. Maranas. Flux coupling analysis of genome-scale metabolic network reconstructions. *Genome Research*, 14(2):301–312, 2004. [8, 15, 150, 198]
- [22] 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*, 17:791–797, 2001. [7]

-
- [23] P. Carbonell, D. Fichera, S. B. Pandit, and J.-L. Faulon. Enumerating metabolic pathways for the production of heterologous target chemicals in chassis organisms. *BMC Systems Biology*, 6:10, 2012. [198]
- [24] G. Codato and M. Fischetti. Combinatorial Benders' cuts for mixed-integer linear programming. *Operations Research*, 54(4):756–766, 2006. [73]
- [25] G. Cogne, M. Rügen, A. Bockmayr, M. Tital, C.-G. Dussap, J.-F. Cornet, and J. Legrand. A model-based method for investigating bioenergetic processes in autotrophically growing eukaryotic microalgae: Application to the green algae *Chlamydomonas reinhardtii*. *Biotechnol Progress*, 27(3):631–640, 2011. [7, 30, 53, 69, 70, 166, 190, 205, 218]
- [26] B. Colson, P. Marcotte, and G. Savard. An overview of bilevel optimization. *Annals of Operations Research*, 153:235–256, 2007. [179]
- [27] M. W. Covert, C. H. Schilling, and B. O. Palsson. Regulation of gene expression in flux balance models of metabolism. *Journal of Theoretical Biology*, 213(1):73–88, 2001. [4]
- [28] W. H. Cunningham. *A combinatorial decomposition theory*. PhD thesis, University of Waterloo, 1973. [42, 114, 132]
- [29] B. Davey and H. Priestley. *Introduction to Lattices and Order*. Cambridge University Press, 2 edition, 2002. [200]
- [30] L. David and A. Bockmayr. Constrained flux coupling analysis. In *Workshop on Constraint based Methods for Bioinformatics, WCB'13, Uppsala*, pages 75–83, 2013. [8]
- [31] 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*, 12:236, 2011. [8, 150, 198]
- [32] D. de Martino, F. Capuani, M. Mori, A. de Martino, and E. Marinari. Counting and correcting thermodynamically infeasible flux cycles in genome-scale metabolic networks. arXiv:1310.3693 [q-bio.MN]. [53]
- [33] D. De Martino, M. Figliuzzi, A. De Martino, and E. Marinari. A scalable algorithm to explore the gibbs energy landscape of genome-scale metabolic networks. *PLoS Computational Biology*, 8(6):e1002562, 2012. [53]
- [34] S. Dempe. Discrete bilevel optimization problems. Technical report, Institut für Wirtschaftsinformatik, Universität Leipzig, 2001. [180]
- [35] K. Dettmer, P. A. Aronov, and B. D. Hammock. Mass spectrometry-based metabolomics. *Mass Spectrom Rev*, 26:51–78, 2007. [3, 30]

- [36] H. Driouch, G. Melzer, and C. Wittmann. Integration of in vivo and in silico metabolic fluxes for improvement of recombinant protein production. *Metabolic Engineering*, 14(1):47–58, 2012. [19]
- [37] V. Dua and E. N. Pistikopolous. An algorithm for the solution of multiparametric mixed integer linear programming problems. *Annals of Operations Research*, 99:123–139, 2000. [180, 181, 182]
- [38] W. B. Dunn, N. J. Bailey, and H. E. Johnson. Measuring the metabolome: current analytical technologies. *Analyst*, 130:606–625, 2005. [3, 30]
- [39] M. Durot, P.-Y. Bourguignon, and V. Schachter. Genome-scale models of bacterial metabolism: reconstruction and applications. *FEMS Microbiology Reviews*, 33:164–90, 2009. [1]
- [40] J. S. Edwards, R. U. Ibarra, and B. O. Palsson. In silico predictions of *Escherichia coli* metabolic capabilities are consistent with experimental data. *Nature Biotechnology*, 19:125–130, 2001. [7, 18, 52]
- [41] N. P. Faísa, V. Dua, B. Rustem, P. M. Saraiva, and E. N. Pistikopolous. Parametric global optimisation for bilevel programming. *Journal of Global Optimization*, 38(4):609–623, 2007. [180]
- [42] M. Feinberg and F. J. Horn. Dynamics of open chemical systems and the algebraic structure of the underlying reaction network. *Chemical Engineering Science*, 29:775–787, 1974. [17]
- [43] A. M. Feist, C. S. Henry, J. L. Reed, M. Krummenacker, A. R. Joyce, P. D. Karp, L. J. Broadbelt, V. Hatzimanikatis, and B. O. Palsson. A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Molecular Systems Biology*, 3:121, 2007. [1, 6, 30, 66, 97, 218]
- [44] A. M. Feist and B. O. Palsson. The biomass objective function. *Current Opinion in Microbiology*, 13:344–349, 2010. [2]
- [45] O. Fiehn. Metabolomics – the link between genotypes and phenotypes. *Plant Molecular Biology*, pages 155–171, 2002. [3, 30]
- [46] R. Fleming, I. Thiele, and H. Nasheuer. Quantitative assignment of reaction directionality in constraint-based models of metabolism: Application to *Escherichia coli*. *Biophysical Chemistry*, 145:47–56, 2009. [2, 6, 30, 218]
- [47] R. Fleming, I. Thiele, G. Provan, and H. Nasheuer. Integrated stoichiometric, thermodynamic and kinetic modelling of steady state metabolism. *Journal of Theoretical Biology*, 264:683–692, 2010. [53]

-
- [48] R. M. Fleming and I. Thiele. von Bertalanffy 1.0: a COBRA toolbox extension to thermodynamically constrain metabolic models. *Bioinformatics*, 27(1):142–143, 2011. [42, 43]
- [49] S. S. Fong, A. P. Burgard, C. D. Herring, E. M. Knight, F. R. Blattner, C. D. Maranas, and P. B. Ø. In silico design and adaptive evolution of escherichia coli for production of lactic acid. *Biotechnology and Bioengineering*, 91(5):643–648, 2005. [1]
- [50] S. Fortune, J. Hopcroft, and J. Wyllie. The directed subgraph homeomorphism problem. *Theoretical Computer Science*, 10:111–121, 1980. [43]
- [51] C. Francke, R. J. Siezen, and B. Teusink. Reconstructing the metabolic network of a bacterium from its genome. *Trends in Microbiology*, 13:550–558, 2005. [1]
- [52] K. Fukuda, T. M. Liebling, and F. Margot. Analysis of backtrack algorithms for listing all vertices and all faces of a convex polyhedron. *Computational Geometry*, 8(1):1–12, 1997. [157]
- [53] E. R. Gansner, E. Koutsofios, S. C. North, and K.-p. Vo. A technique for drawing directed graphs. *IEEE Transactions on Software Engineering*, 19(3):214–230, 1993. [144]
- [54] E. R. Gansner and S. C. North. An open graph visualization system and its applications to software engineering. *Software - Practice and Experience*, 30(11):1203–1233, 2000. [142]
- [55] A. Gevorgyan, M. G. Poolman, and D. A. Fell. Detection of stoichiometric inconsistencies in biomolecular models. *Bioinformatics*, 24(19):2245–2251, 2008. [2]
- [56] M. A. Goberna, V. Jornet, and M. M. Rodríguez. On linear systems containing strict inequalities. *Linear Algebra and its Applications*, 360:151–171, 2003. [70]
- [57] Y. Goldstein. *Untersuchung metabolischer Netzwerke mit Verbandstheorie*. Doctoral thesis, Freie Universität Berlin, 2012. [197]
- [58] Y. A. B. Goldstein and A. Bockmayr. A lattice-theoretic framework for metabolic pathway analysis. In A. Gupta and T. A. Henzinger, editors, *Computational Methods in Systems Biology*, volume 8130 of *Lecture Notes in Computer Science*, pages 178–191. Springer Berlin Heidelberg, 2013. [7, 8, 198, 199, 200, 201, 202, 204, 216]
- [59] J. L. Griffin. The Cinderella story of metabolic profiling: does metabolomics get to go to the functional genomics ball? *Philosophical Transactions of the Royal Society B*, 361(1465):147–161, 2006. [3, 30]
- [60] B. Grünbaum. *Convex Polytopes*. Graduate Texts in Mathematics. Springer, 2 edition, 2003. [19]

- [61] Z. H. Gümiş and C. A. Floudas. Global optimization of mixed-integer bilevel programming problems. *Computational Management Science*, 2(3):181–212, 2005. [180]
- [62] V. Hatzimanikatis, C. Li, J. A. Ionita, C. S. Henry, M. D. Jankowski, and L. J. Broadbelt. Exploring the diversity of complex metabolic networks. *Bioinformatics*, (8):1603–1609, 2005. [13]
- [63] U.-U. Haus, S. Klamt, and T. Stephen. Computing knock-out strategies in metabolic networks. *Journal of Computational Biology*, 15(3):259–268, 2008. [7, 19, 198]
- [64] C. Henry, M. Jankowski, L. Broadbelt, and V. Hatzimanikatis. Genome-scale thermodynamic analysis of *Escherichia coli* metabolism. *Biophysical Journal*, 90(4):1453–1461, 2006. [218]
- [65] C. S. Henry, L. J. Broadbelt, and V. Hatzimanikatis. Thermodynamics-based metabolic flux analysis. *Biophysical Journal*, 92:1792–1805, 2007. [53]
- [66] W. J. Heuett and H. Qian. Combining flux and energy balance analysis to model large-scale biochemical networks. *Journal of Bioinformatics and Computational Biology*, 4(6):1227–1243, 2006. [53]
- [67] I. V. Hicks and S.-I. Oum. *Wiley Encyclopedia of Operations Research and Management Science*, chapter Branch-Width and Tangles. Wiley, 2011. [155]
- [68] A. Hoppe, S. Hoffmann, A. Gerasch, C. Gille, and H.-G. Holzhütter. FASIMU: flexible software for flux-balance computation series in large metabolic networks. *BMC Bioinformatics*, 12:28, 2011. [68]
- [69] A. Hoppe, S. Hoffmann, and H.-G. Holzhütter. Including metabolite concentrations into flux balance analysis: thermodynamic realizability as a constraint on flux distributions in metabolic networks. *BMC Systems Biology*, 1:23, 2007. [53, 68, 70, 205, 218]
- [70] M. D. Jankowski, C. S. Henry, L. J. Broadbelt, and V. Hatzimanikatis. Group contribution method for thermodynamic analysis of complex metabolic networks. *Biophysical Journal*, 95:1487–1499, 2008. [31]
- [71] S. J. Jol, A. Kümmel, M. Terzer, J. Stelling, and M. Heinemann. System-level insights into yeast metabolism by thermodynamic analysis of elementary flux modes. *PLoS Computational Biology*, 8:3, 2012. [218]
- [72] C. Kaleta, L. de Figueiredo, J. Behre, and S. Schuster. Efmevolver: Computing elementary flux modes in genome-scale metabolic networks. In I. Grosse, S. Neumann, S. Posch, F. Schreiber, and P. Stadler, editors, *Lecture Notes in Informatics - Proceedings*, volume P-157, pages 179–189. Gesellschaft für Informatik, 2009. [217]

-
- [73] 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*, 19:1872–1883, 2009. [8]
- [74] D. E. Kaufman and R. L. Smith. Direction choice for accelerated convergence in hit-and-run sampling. *Operations Research*, 46(1):84–95, 1998. [217]
- [75] S. M. Kelk, B. G. Olivier, L. Stougie, and F. J. Bruggeman. Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. *Scientific Reports*, 2:580, 2012. [105, 125, 126, 135, 136, 138, 139, 217]
- [76] L. Khachiyan, E. Boros, K. Borys, K. Elbassioni, and V. Gurvich. Generating all vertices of a polyhedron is hard. *Discrete Computational Geometry*, 39(1-3):174–190, 2008. [153]
- [77] C. Khannapho, H. Zhao, B. L. Bonde, A. M. Kierzek, C. A. Avignone-Rossa, and M. E. Bushell. Selection of objective function in genome scale flux balance analysis for process feed development in antibiotic production. *Metabolic Engineering*, 10(5):227–233, 2008. [7]
- [78] P. Kharchenko, D. Vitkup, and G. M. Church. Filling gaps in a metabolic network using expression information. *Bioinformatics*, 20:178–185, 2004. [1]
- [79] S. Klamt and E. D. Gilles. Minimal cut sets in biochemical reaction networks. *Bioinformatics*, 20(2):226–234, 2004. [8, 186]
- [80] E. Klipp, R. Herwig, A. Kowald, C. Wierling, and H. Lehrach. *Systems Biology in Practice: Concepts, Implementation and Application*. Wiley-Blackwell, 1 edition, 2005. [18]
- [81] S. Krogdahl. The dependence graph for bases in matroids. *Discrete Mathematics*, 19:47–59, 1977. [42, 114, 132]
- [82] A. Kümmel, S. Panke, and M. Heinemann. Putative regulatory sites unraveled by network-embedded thermodynamic analysis of metabolome data. *Molecular Systems Biology*, 2:2006.0034, 2006. [1, 7, 30, 69, 218]
- [83] A. Kümmel, S. Panke, and M. Heinemann. Systematic assignment of thermodynamic constraints in metabolic network models. *BMC Bioinformatics*, 7:512, 2006. [30, 218]
- [84] M. Kvasnica, P. Grieder, and M. Baotić. Multi-Parametric Toolbox (MPT), 2004. [189]
- [85] A. Larhlimi. *New Concepts and Tools in Constraint-based Analysis of Metabolic Networks*. PhD thesis, Freie Universität Berlin, 2008. http://www.diss.fu-berlin.de/diss/receive/FUDISS_thesis_000000009198. [198]

- [86] A. Larhlimi and A. Bockmayr. A new approach to flux coupling analysis of metabolic networks. *Computational Life Sciences II*, 4216:205–215, 2006. [198]
- [87] A. Larhlimi and A. Bockmayr. Minimal direction cuts in metabolic networks. *Proceedings of Computational Life Sciences III, CompLife'07*, 940:73–86, 2007. [8, 187]
- [88] A. Larhlimi and A. Bockmayr. On inner and outer descriptions of the steady-state flux cone of a metabolic network. In *Computational Methods in Systems Biology*, volume 5307/2008 of *Lecture Notes in Computer Science*, pages 308–327, 2008. [19]
- [89] 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. [8, 19]
- [90] 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, 2012. [8, 198]
- [91] W. Liebermeister and E. Klipp. Bringing metabolite networks to life: convenience rate law and thermodynamic constraints. *Theoretical Biology and Medical Modelling*, 3:41, 2006. [27]
- [92] A. Lodi, T. K. Ralphs, F. Rossi, and S. Smriglio. Interdiction branching, 2011. http://www.optimization-online.org/DB_FILE/2011/09/3189.pdf. [63]
- [93] Z.-Q. Luo, J.-S. Pang, and D. Ralph. *Mathematical Programs with Equilibrium Constraints*. Cambridge University Press, 1996. [179]
- [94] 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:i515–i521, 2012. [217]
- [95] R. Mahadevan and C. Schilling. The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic Engineering*, 5:264–276, 2003. [7, 125]
- [96] M. L. Mavrouniotis. Group contributions for estimating standard Gibbs energies of formation of biochemical compounds in aqueous solution. *Biotechnology and Bioengineering*, 36:1070–1082, 1990. [31]
- [97] M. L. Mavrouniotis. Duality theory for thermodynamic bottlenecks in bioreaction pathways. *Chemical Engineering Science*, 51(9):1495–1507, 1996. [32, 34]
- [98] D. M. Moyles and G. L. Thompson. An algorithm for finding a minimum equivalent graph of a digraph. *Journal of the ACM*, 16(3):455–460, 1969. [205, 206, 213]

-
- [99] A. C. Müller. Thermodynamic constraints in metabolic networks. Master's thesis, Freie Universität Berlin, Fachbereich Mathematik und Informatik, February 2012. <http://page.mi.fu-berlin.de/arnem/theses/master.pdf> (AC Müller is now called AC Reimers). [33, 34, 43, 51, 52, 53, 54, 64, 65, 135, 203]
- [100] A. C. Müller and A. Bockmayr. Flux modules in metabolic networks. *Journal of Mathematical Biology*. in press (AC Müller is now called AC Reimers). [105, 106, 115, 117, 120, 127, 135]
- [101] A. C. Müller and A. Bockmayr. Fast thermodynamically constrained flux variability analysis. *Bioinformatics*, 29(7):903–909, 2013. (AC Müller is now called AC Reimers). [28, 43, 64, 65, 128, 226]
- [102] A. C. Müller, F. J. Bruggeman, B. G. Olivier, and L. Stougie. Fast flux module detection using matroid theory. In *Research in Computational Molecular Biology*, volume 8394 of *Lecture Notes in Computer Science*, pages 192–206, 2014. (AC Müller is now called AC Reimers). [105, 106, 130, 139]
- [103] S. Müller, G. Regensburger, and R. Steuer. Enzyme allocation problems in kinetic metabolic networks: Optimal solutions are elementary flux modes. *Journal of Theoretical Biology*, 2014. in press. [7, 18]
- [104] I. E. Nikerel, W. A. van Winden, P. J. Verheijen, and J. J. Heijnen. Model reduction and a priori kinetic parameter identifiability analysis using metabolome time series for metabolic reaction networks with linlog kinetics. *Metabolic Engineering*, 11:20–30, 2009. [4]
- [105] R. P. Nolan, A. P. Fenley, and K. Lee. Identification of distributed metabolic objectives in the hypermetabolic liver by flux and energy balance analysis. *Metabolic Engineering*, 8:30–45, 2006. [36]
- [106] E. Noor, B.-E. Arren, A. Flamholz, E. Reznik, W. Liebermeister, and R. Milo. Pathway thermodynamics highlights kinetics obstacles in central metabolism. *PLoS Computational Biology*, 2014. [7, 69]
- [107] E. Noor, H. S. Haraldsdóttir, R. Milo, and R. M. T. Fleming. Consistent estimation of gibbs energy using component contributions. *PLoS Computational Biology*, 9(7):e1003098, 2013. [31, 99, 100]
- [108] E. Noor, N. E. Lewis, and R. Milo. A proof for loop-law constraints in stoichiometric metabolic networks. *BMC Systems Biology*, 6:140, 2012. [28]
- [109] R. A. Notebaart, B. Teusink, R. J. Siezen, and B. Papp. Co-regulation of metabolic genes is better explained by flux coupling than by network distance. *PLoS Computational Biology*, 4(1):e26, 2008. [8, 198]

- [110] R. A. Notebaart, F. H. van Enckevort, C. Francke, R. J. Siezen, and B. Teusink. Accelerating the reconstruction of genome-scale metabolic networks. *BMC Bioinformatics*, 7:296, 2006. [1]
- [111] J. S. Oliveira, C. G. Bailey, J. B. Jones-Oliveira, and D. A. Dixon. An algebraic-combinatorial model for the identification and mapping of biochemical pathways. *Bulletin of Mathematical Biology*, 63:1163–1196, 2001. [23]
- [112] M. A. Orman, I. P. Androulakis, F. Berthiaume, and M. G. Ierapetritou. Metabolic network analysis of perfused livers under fed and fasted states: Incorporating thermodynamic and futile-cycle-associated regulatory constraints. *Journal of Theoretical Biology*, 293:101–110, 2012. [19]
- [113] J. Orth, T. Conrad, J. Na, J. Lerman, H. Nam, A. Feist, and B. Palsson. A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism. *Molecular Systems Biology*, 7:535, 2011. [223]
- [114] J. D. Orth, I. Thiele, and B. O. Palsson. What is flux balance analysis. *Nature Biotechnology*, 28:245–248, 2010. [6, 51]
- [115] G. Oster, A. Perelson, and A. Katchalsky. Network thermodynamics. *Nature*, 234(5329):393–399, 1971. [5]
- [116] S.-i. Oum and P. Seymour. Approximating clique-width and branch-width. *Journal of Combinatorial Theory, Series B* 96:514–528, 2006. [114]
- [117] S.-i. Oum and P. Seymour. Testing branch-width. *Journal of Combinatorial Theory, Series B* 97:385–393, 2007. [114]
- [118] J. Oxley. *Matroid Theory*. Oxford Graduate Texts in Mathematics. Oxford University Press, New York, second edition edition, 2011. [9, 21, 23, 25, 26, 40, 112, 120, 131]
- [119] C. H. Papadimitrou. *Computational Complexity*. Addison-Wesley, 1994. [38, 218, 219]
- [120] A. J. Papin, J. Stelling, N. D. Price, S. Klamt, S. Schuster, and B. O. Palsson. Comparison of network-based pathway analysis methods. *TRENDS in Biotechnology*, 22(8):400–405, 2004. [19]
- [121] J. L. Papin, Jason A. Reed and B. O. Palsson. Hierarchical thinking in network biology: the unbiased modularization of biochemical networks. *TRENDS in Biochemical Sciences*, 29(12):641–647, 2004. [217]
- [122] 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. [4, 16]

-
- [123] N. D. Price, I. Thiele, and B. O. Palsson. Candidate states of *Helicobacter pylori*'s genome-scale metabolic network upon application of "loop law" thermodynamic constraints. *Biophysical Journal*, 90:3919–3928, 2006. [6, 30, 218]
- [124] H. Qian and D. A. Beard. Thermodynamics of stoichiometric biochemical networks in living systems far from equilibrium. *Biophysical Chemistry*, 114:213–220, 2005. [5, 26]
- [125] H. Qian, D. A. Beard, and S.-d. Liang. Stoichiometric network theory for nonequilibrium biochemical systems. *European Journal of Biochemistry*, 270:415–421, 2003. [5]
- [126] V. N. Reddy, M. L. Mavrovouniotis, and M. N. Liebman. Petri net representations in metabolic pathways. In *Proceedings International Conference on Intelligent Systems for Molecular Biology*, volume 1, pages 328–336, 1993. [8]
- [127] A. Reimers, Y. Goldstein, and A. Bockmayr. Qualitative and thermodynamic flux coupling analysis. Matheon Preprint #1054, 2014. [197]
- [128] A. Reimers, A. Reimers, and Y. Goldstein. Minimal equivalent subgraphs containing a given set of arcs. Matheon Preprint #1053, 2014. [205]
- [129] A. C. Reimers and L. Stougie. A decomposition theory for vertex enumeration of convex polyhedra. submitted to Discrete and Computational Geometry. Preprint: <http://arxiv.org/abs/1404.5584>. [105, 106]
- [130] N. Robertson and P. Seymour. Graph minors. x. obstructions to tree-decomposition. *Journal of Combinatorial Theory, Series B*, 52(2):153–190, 1991. [155]
- [131] N. Robertson and P. Seymour. Graph minors. xiii. the disjoint paths problem. *Journal of Combinatorial Theory, Series B*, 63(1):65–110, 1995. [42]
- [132] B. Sarıyar, S. Perk, U. Akman, and A. Hortaçsu. Monte carlo sampling and principal component analysis of flux distributions yield topological and modular information on metabolic networks. *Journal of Theoretical Biology*, 242:389–400, 2006. [217]
- [133] U. Sauer. Metabolic networks in motion: ^{13}C -based flux analysis. *Molecular Systems Biology*, 2:62, 2006. [3]
- [134] J. Schellenberger, N. E. Lewis, and B. Ø. Palsson. Elimination of thermodynamically infeasible loops in steady-state metabolic models. *Biophysical Journal*, 100:544–553, 2011. [6, 28, 53, 137, 217, 218, 223, 226]
- [135] J. Schellenberger and B. O. Palsson. Use of randomized sampling for analysis of metabolic networks. *The Journal of Biological Chemistry*, 284(9):5457–5461, 2009. [217]

- [136] J. Schellenberger, J. O. Park, T. M. Conrad, and B. O. Palsson. BiGG: a biochemical genetic and genomic knowledgebase of large scale metabolic reconstructions. *BMC Bioinformatics*, 11:213, 2010. [213, 223]
- [137] 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, J. Kang, D. R. Hyduke, and B. O. Palsson. Quantitative prediction of cellular metabolism with constraint-based models: the COBRA toolbox v2.0. *Nature Protocols*, 6(9):1290–1307, 2011. [4, 70, 223]
- [138] C. H. Schilling, D. Letscher, and B. O. 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*, 203:229–248, 2000. [8, 217]
- [139] A. Schrijver. *Theory of linear and integer programming*. Wiley-Interscience series in discrete mathematics and optimization. Wiley, Chichester ; New York, 1986. [160]
- [140] 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*, 18:326–332, 2000. [19]
- [141] S. Schuster and C. Hilgetag. On elementary flux modes in biochemical systems at steady state. *J. Biol. Systems*, 2:165–182, 1994. [8, 19, 123, 198, 217]
- [142] S. Schuster and R. Schuster. Detecting strictly detailed balanced subnetworks in open chemical reaction networks. *Journal of Mathematical Chemistry*, 6:17–40, 1991. [132]
- [143] J.-M. Schwartz and M. Kanehisa. Quantitative elementary mode analysis of metabolic pathways: the example of yeast glycolysis. *BMC Bioinformatics*, 7:186, 2006. [19]
- [144] D. Segre, D. Vitkup, and G. Church. Analysis of optimality in natural and perturbed metabolic networks. *Proceedings of the National Academy of Sciences of the United States of America*, 99:15112–15117, 2001. [1, 8]
- [145] Y. Shiloach. A polynomial solution to the undirected two paths problem. *Journal of the Association for Computing Machinery*, 27(3):445–456, 1980. [42]
- [146] T. Shlomi, T. Benyamini, E. Gottlieb, S. Roded, 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, 7]
- [147] T. Shlomi, O. Berkman, and E. Ruppin. Regulatory on/off minimization of metabolic flux changes after genetic perturbations. *PNAS*, 102(21):7695–7700, 2005. [8]

-
- [148] A. Singh, K. C. Soh, V. Hatzimanikatis, and R. T. Gill. Manipulating redox and atp balancing for improved production of succinate in *E. coli*. *Metabolic Engineering*, 13:76–81, 2011. [218]
- [149] E. J. Smid, D. Molenaar, J. Hugenholtz, W. M. de Vos, and B. Teusink. Functional ingredient production: application of global metabolic models. *Current Opinion in Biotechnology*, 16(2):190–197, 2005. [1]
- [150] T. Soh and K. Inoue. Identifying necessary reactions in metabolic pathways by minimal model generation. In *Frontiers in Artificial Intelligence and Applications*, volume 215 of *European Conference on Artificial Intelligence*, 2010. [198]
- [151] J. Stelling, T. Klamt, K. Bettenbrock, S. Schuster, and E. D. Gilles. Metabolic network structure determines key aspects of functionality and regulation. *Nature*, 420:190–193, 2002. [1]
- [152] R. Steuer, T. Gross, J. Selbig, and B. Blasius. Structural kinetic modeling of metabolic networks. *Proceedings of the National Academy of Sciences of the United States of America*, 103(32):11868–11873, 2006. [4]
- [153] T. W. Tee, A. Chowdhury, C. D. Maranas, and J. V. Shanks. Systems metabolic engineering design: Fatty acid production as an emerging case study. *Biotechnology and Bioengineering*, 2014. in press. [1]
- [154] N. Tepper, E. Noor, D. Amador-Noguez, Haraldsdóttir, R. Milo, J. Rabinowitz, W. Liebermeister, and T. Shlomi. Steady-state metabolite concentrations reflect a balance between maximizing enzyme efficiency and minimizing total metabolite load. *PLoS ONE*, 8(9):e75370, 2013. [30, 97, 99]
- [155] M. Terzer. *Large Scale Methods to Enumerate Extreme Rays and Elementary Modes*. PhD thesis, Swiss Federal Institute of Technology, Zurich, 2009. [19, 204]
- [156] 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. [1, 4, 16]
- [157] B. Teusink, A. Wiersma, L. Jacobs, R. Notebaart, and E. Smid. Understanding the adaptive growth strategy of *Lactobacillus plantarum* by in silico optimisation. *PLoS Computational Biology*, 5(6):e1000410, 2009. [7]
- [158] I. Thiele, N. D. Price, T. D. Vo, and B. O. Palsson. Candidate metabolic network states in human mitochondria. *The Journal of Biological Chemistry*, 280:11683–11695, 2005. [217]
- [159] G. H. Thomas, J. Zucker, S. Macdonald, A. Sorokin, I. Goryanin, and A. E. Douglas. A fragile metabolic network adapted for cooperation in the symbiotic bacterium *Buchnera aphidicola*. *BMC Systems Biology*, 3:24, 2009. [1]

- [160] S. G. Thorleifsson and I. Thiele. rBioNet: A COBRA toolbox extension for reconstructing high-quality biochemical networks. *Bioinformatics*, 27(14):2009–2010, 2011. [1]
- [161] K. Truemper. Partial matroid representations. *European Journal of Combinatorics*, 1984. [131]
- [162] R. J. van Berlo, D. de Ridder, J.-M. Daran, P. A. Daran-Lapujade, B. Teusink, and M. J. Reinders. Predicting metabolic fluxes using gene expression differences as constraints. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 8(1):206–216, 2011. [4]
- [163] A. Varma and B. O. Palsson. Metabolic flux balancing: Basic concepts, scientific and practical use. *Nature Biotechnology*, 12:994–998, 1994. [6, 51]
- [164] K. Voss, M. Heiner, and I. Koch. Steady state analysis of metabolic pathways using Petri nets. *In Silico Biology*, 3:31, 2003. [8]
- [165] U. Wen and Y. Yang. Algorithms for solving the mixed integer two-level linear programming problem. *Computers & Operations Research*, 17(2):133–142, 1990. [180]
- [166] W. Wiechert. ^{13}C metabolic flux analysis. *Metabolic Engineering*, 3:195–206, 2001. [3]
- [167] M. T. Wortel, H. Peters, J. Hulshof, B. Teusink, and F. J. Bruggeman. Metabolic states with maximal specific flux rate carry flux through an elementary mode. *FEBS Journal*, 2014. in press. [7, 18]
- [168] J. Wright and A. Wagner. Exhaustive identification of steady state cycles in large stoichiometric networks. *BMC Systems Biology*, 2:61, 2008. [204]
- [169] Y. Xi, Y.-P. P. Chen, Q. Chen, and F. Wang. Comparative study of computational methods to detect the correlated reaction sets in biochemical networks. *Briefings in Bioinformatics*, 12(2):132–150, 2011. [217, 226]
- [170] M. Xu, R. Smith, and J. Sadhukhan. Optimization of productivity and thermodynamic performance of metabolic pathways. *Industrial & Engineering Chemistry Research*, 47(15):5669–5679, 2008. [36]
- [171] F. Yang and D. A. Beard. Thermodynamically based profiling of drug metabolism and drug-drug metabolic interactions: A case study of acetaminophen and ethanol toxic interaction. *Biophysical Chemistry*, 120:121–134, 2006. [7, 30]
- [172] G. M. Ziegler. *Lectures on Polytopes*, chapter 6.3, pages 160–162. Springer, 2007. [21]

Index

- 0-module, 159
- μ -bound, 84
- 0-module, 115
- 1-module, 97, 145
- 13C labeling experiments, 143
- 2-connected, 150
- 2-connected component, 131
- 2-separator, 150
- 3-connectivity, 149

- A. platensis*, 166, 168
- ACHR, 217
- aff, 12
- affine hull, 12, 108
- Alexander Bockmayr, 197
- Alexandra Grigore, 205
- alternative pathway, 136
- anomaly, 142
- application, 97
- approximation, 54
- APX-hard, 54, 187
- artifacts, 218
- artificial centering hit and run, 217
- average flux, 16, 17

- benders cut, 73
- bilevel optimization, 165, 173, 180, 189
- binary rooted, 155
- biofuel, 165
- biomass reaction, 2
- bioreactor, 165
- block flux, 55
- block-diagonalization, 132
- blocked reaction, 98

- blocking graph, 129, 136
- blocking set, 186
- bottleneck, 32
- branch-decomposition, 155
- branch-width, 155
- branching, 54, 93

- C. reinhardtii*, 165
- Carathéodory's theorem, 160
- cell concentration, 167
- cell density, 165
- channeling, 30
- Chlamydomonas reinhardtii*, 165
- circuit, 21, 29, 32, 131
- circuit axiom, 22
- closed, 52
- closure, 70, 201
- closure (lattice theory), 201
- cocircuit, 25
- cocycle, 25
- combinatorial Benders cut, 73
- complementarity constraint, 179
- complementary slackness, 188
- complete digraph, 210
- component contribution, 31, 100
- composition, 24
- compressed network, 139, 153
- computation, 136, 149, 206
- computational complexity, 37
- computing modules, 114, 125
- concentration, 17, 30, 100
- cone, 12
- conical hull, 12
- connected component, 26, 129, 131, 155

connectivity, 149
 connectivity function, 155
 constant interface, 106, 157
 constraint integer programming, 53
 constraint programming, 55
 constraint-based analysis, 6
 constraint-based method, 4
 conv, 12
 convex, 52
 convex hull, 12
 coparallel element, 149
 coparallel reaction, 149
 coupling data, 205
 covector, 25, 29
 critical region, 180, 181
 cut set, 8, 186
 cutting plane, 54, 71
 cyanobacteria, 165
 cycle, 21, 24
 cycle subtraction, 64

DAG, 209
 decision problem, 37
 decision variables, 73
 decomposition, 117, 119, 121, 126, 145, 157
 decomposition existence, 120
 decomposition uniqueness, 119
 default concentration bounds, 30
 deletion, 25
 density, 165
 dependent set, 23
 directed acyclic graph, 209
 directed flux coupling, 15, 59, 88
 directed reaction, 13
 directional flux coupling, 98
 directionally coupled, 198
 diverting reaction, 173
 diverting set, 174, 175
 diverting set of type 1, 174
 diverting set of type 2, 175
 duality, 25, 34, 67, 74, 96, 150, 187

EFM, 8, 19, 90, 121, 126, 147, 154

elementary flux mode, 8, 19, 29, 90, 121, 126, 147, 154
 elementary mode, 19, 29
 empirical results, 136
 empirical study, 97
 energetic constraint, 190
 environment, 136
 equilibrium constant, 5, 30, 66, 100
 equilibrium constant of formation, 31
 equilibrium constant of reaction, 31, 100
 equilibrium constraint, 179
 equilibrium potential, 5
 estimation error, 31
 eukariotic green algae, 165
 exchange reaction, 2, 12, 26
 experiment, 97
 extension size, 208
 extreme ray, 19

F, 115
 face, 157
 FBA, 6, 18, 51, 165
 FCA, 8, 59, 98, 150, 197
 feasibility check, 73
 feasible A-face, 157
 feasible face, 157
 feasible pathway, 11
 feasible sampling algorithm, 219
 finding modules, 114, 125
 finite model, 13
 flow condition, 82
 flux, 3, 16
 flux balance analysis, 6, 18, 51, 165
 flux bounds, 14
 flux cone, 19
 flux coupling, 15, 59, 88, 98
 flux coupling analysis, 8, 150, 197
 flux forcing, 127, 173
 flux forcing coefficient, 178
 flux mode matroid, 24
 flux module, 97, 105, 106, 115
 flux optimization, 51
 flux polyhedron, 19

flux space, 63
 flux variability analysis, 7, 28, 98, 127, 222
 flux-forcing, 64
 flux-forcing reaction, 170
 full extension, 210
 fully coupled reaction, 149
 fundamental circuit, 131
 FVA, 7, 98, 127, 222

gas constant, 5
 generalized FCA, 197
 generalized infeasible set, 59
 generating vector, 19
 generator, 201
 green algae, 165
 group contribution, 31, 100
 growth condition, 136
 growth rate, 168
 growth-rate, 166
 Guillaume Cogne, 165

Hamiltonian cycle, 213
 heuristics, 63
 homogenization, 110, 150
 hypergraph, 8

implementation, 189
 implicit constraint, 55
 in silico, 1
 in vitro, 1
 in vivo, 1
 independent, 22
 independent set, 22
 infeasible reaction subset, 32
 infeasible set, 32, 33, 59, 75
 injective, 157, 162
 inner description, 19
 integer programming, 53
 interdiction branching, 63
 interface flux, 106, 139
 internal circuit, 25
 internal cycle, 28, 64, 179, 186, 222
 internal cycle matroid, 24

internal reaction, 2, 12, 26
 interplay, 139
 irreducible, 201
 irreducible element, 201
 irreversible, 19
 irreversible 1-module, 145
 irreversible module, 145
 irreversible reaction, 12, 14
 IS, 33, 75
 isCoupled, 202, 204

k-module, 106, 153
 k-separator, 112, 155
 Karush-Kuhn-Tucker conditions, 179
 ker, 12
 kernel, 12
 kinetic model, 4
 kinetics, 3
 KKT, 179
 KKT-conditions, 179
 knock-out, 8

Lattice, 8
 lattice theory, 197, 200
 Leen Stougie, 105
 linear matroid, 23
 linear module, 106, 145
 linear program, 162
 linear programming, 9
 linear span, 12
 linear vector space, 107
 local concentration, 30
 local feasible, 79
 locality, 209
 loop-law, 29
 Lotka-Volterra, 17
 lower-bound feasible, 78
 LP relaxation, 54

Markov chain Monte Carlo, 217
 mass conservation, 2
 matroid, 21, 29, 112, 130, 149, 155
 matroid connectivity, 25, 114, 149
 matroid deletion, 25

matroid duality, 25
 matroid intersection, 114
 matroid restriction, 25
 matroid theory, 112, 130, 149
 maximal concentration, 30
 maximal element, 204
 MCMC, 217
 MCS, 186
 metabolic network, 2, 11
 metabolite, 2, 11
 metabolite concentration, 3, 30, 69
 metabolite in path, 83
 metabolite potential, 28
 MILP, 9, 53, 180
 min-max problem, 189
 minimal A-face, 157
 minimal concentration, 30
 minimal cut set, 186
 minimal equivalent subgraph, 205
 minimal extension, 206
 minimal face, 157
 minimal generator, 201
 minimal infeasible set, 56, 75
 minimal metabolic behavior, 19
 minimal module, 119
 minimal representation, 205
 minimally infeasible set, 32
 MinimumCut, 186
 MIS, 32, 56, 75
 mixed integer linear programming, 9, 53
 mixed integer programming, 180
 module, 97, 105, 106
 module candidate, 129
 module union, 115
 MOMA, 8
 MPEC, 179
 mpLP, 180
 mpMILP, 180
 mpTFBA, 180
 mu-bound, 84
 multi-parametric LP, 180
 multi-parametric MILP, 180
 multi-parametric TFBA, 180
 NADH-GOGAT, 190
 near-optimal solution, 70
 negative growth rate, 170
 network compression, 139, 145
 non-trivial sampling algorithm, 219
 notation, 11
 NP (complexity class), 38, 43, 220
 NP-hard, 43, 157, 218
 null space, 12
 null-space matrix, 132
 numerical instability, 70
 numerically unstable, 135
 nutrients, 136
 objective, 136
 objective reaction, 64
 obstructions to sampling, 222
 ODE, 4
 OptKnock, 8
 ordinary differential equation, 4
 oriented matroid, 8, 21
 oscillation, 170
 outer description, 19
 P (complexity class), 38, 42, 64, 159
 P-k-module, 106
 P-module, 106, 115
 P/O-ratio, 190
 parallel element, 149
 parallel reaction, 149
 parallel reactions, 202
 parametrized complexity, 163
 parametrized solution, 181
 partial representation, 132
 partially coupled, 198
 partition, 63
 path, 79, 83
 Petri-net, 8
 phosphate-oxygen ratio, 190
 photo-bioreactor, 165
 photo-inhibition, 167
 polyhedral cone, 19
 polyhedron, 8, 19, 155
 polynomial time, 159

polynomial time sampling algorithm, 219
 polytope, 19, 155, 159
 potential, 5, 26, 28, 69
 potential bounds, 66
 potential propagation, 77
 potential space, 28, 32, 55
 practice, 150
 probability space, 126
 propagation, 77, 87
 pseudo-reaction, 13, 21, 83, 172

 qFCA, 197
 qualitative models, 197

 rank, 155
 reaction, 2, 11
 reaction equivalence, 13
 reaction in internal cycle, 64
 reaction with variability, 107
 realizable matroid, 24
 reduced costs, 187
 reduction, 39
 regulation, 4
 relative interior, 109
 relaxed thermodynamically feasible, 27
 representative reaction, 147
 restrict potential space, 55
 restriction, 25
 reverse reaction, 13
 reversible, 87
 reversible reaction, 87, 97
 ROOM, 8
 RP (complexity class), 219
 run time, 135, 153

 sample space, 218
 sampling, 217
 sampling obstruction, 222
 second law of thermodynamics, 5
 sensitivity, 136
 separator, 112, 130, 150
 sign function, 23
 signed set, 21, 32
 signed subset, 21, 23
 signed support, 20, 23
 simple matroid, 149
 solution space, 63
 span, 12
 split reaction, 147
 splitted reaction, 82
 steady-state, 4, 16, 83
 steady-state flux space, 115
 steady-state path, 83
 stoichiometric matrix, 11
 stoichiometry, 2
 strict inequality, 52, 69
 strong elimination, 22
 strong thermodynamic constraint, 52
 strongly connected component, 206
 strongly thermodynamically feasible, 26
 sublinear growth, 165, 170
 subnetwork, 32, 90
 suboptimal, 145
 suboptimal growth, 170
 substitution, 147

 T, 116
 temperature, 5
 thermo. constrained flux space, 116
 thermo. feasible, 116
 thermodynamic constraint, 5, 26
 thermodynamically constrained flux space, 115
 thermodynamically feasible, 26–28
 ThermoFlux, 218
 Thermoflux, 37, 38
 topological closure, 70
 total growth rate, 165
 transitive closure, 205
 transitive reduction, 205

 unbounded face, 155
 uncertainty, 100
 uncoupled, 198
 undirected flux coupling, 15
 unknown complexity, 49
 update potential bounds, 87

upper-bound feasible, 78

variability, 107

variable flux rate, 107

variable interface, 106, 107, 145, 157

variable reaction, 107

vector, 21, 24

vertex, 159

vertex correlation, 125

vertex enumeration, 126, 154

vertex feasible A-face, 157

vertex feasible face, 157

visualization, 139

visualization method, 139

visualization results, 143

weak thermodynamic constraint, 52

weakly thermodynamically feasible, 27

witness for infeasibility, 36

Yaron Goldstein, 197, 205

zero column, 13

Kurzzusammenfassung

Biologische Experimente sind zeitraubend und teuer. Deshalb werden Computer immer häufiger benutzt um solche Experimente zu bestimmen, die am ehesten erfolgreich sind und zu neuen Erkenntnissen führen.

In dieser Arbeit betrachte ich Varianten der Constraint-Basierten Methode *Flussbalance-Analyse*. Mittels Nebenbedingungen werden biologisch unrealistische Verhaltensweisen ausgeschlossen. Viele der Nebenbedingungen (z.B. Flusserhaltung) lassen sich mit linearen Ungleichungen beschreiben, was effiziente Analyse mittels linearer Programmierung ermöglicht. Thermodynamische Nebenbedingungen sorgen dafür, dass auch energetische Aspekte berücksichtigt werden. Allerdings sind diese Nebenbedingung nicht linear und induzieren zudem oft einen nicht abgeschlossenen Lösungsraum.

In dieser Arbeit zeige ich, dass diese nicht-Linearität der thermodynamischen Nebenbedingungen häufig zu NP-schweren Problemen führt, was sich z.B. auf die Verlässlichkeit von Samplingmethoden auswirkt. Aber ich diskutiere auch Lösungsansätze, wie dennoch in der Praxis Optimierungsprobleme und qualitative Analysemethoden, wie Flusskopplungsanalyse, mit thermodynamischen Nebenbedingungen effizient gelöst werden können. Anwendung finden diese Erkenntnisse in der Analyse des Wachstum der Grünalge *C. reinhardtii*, wo ich eine Methode zum Lösen von bilevel-Optimierungsproblemen mit thermodynamischen Nebenbedingungen entwickle.

Ein weiteres Gebiet meiner Arbeit umfassen Flussmodule. Inspiriert von einer Arbeit von Kelk et al. gebe ich eine mathematische Definition von Flussmodul und zeige dass die Definition die erwünschten Eigenschaften erfüllt. Die Definition erlaubt mir auch mehrere Zerlegungssätze zu zeigen, die die Analyse metabolischer Netzwerke vereinfachen, und Module effizient mittels Matroidtheorie zu finden. Mit der Definition von k -Modul zeige ich auch einen Zerlegungssatz für die Eckenenumeration allgemeiner Polyeder und nutze hierfür das Konzept der Branchweite in Matroiden.

Mittels Flussmodulen und algorithmischen Ansätzen um auch komplizierte Nebenbedingungen zu integrieren, zeige ich in dieser Arbeit Methoden auf, die die Analyse metabolischer Netzwerke vereinfachen und beschleunigen. Dadurch können biologische Erkenntnisse schneller gewonnen werden und bessere Methoden in der Biotechnologie zur Herstellung von Biotreibstoffen und in der Medizin für Krebstherapien entwickelt werden.

Curriculum Vitae

Education

| | |
|---------|---|
| 02/2012 | MSc in Mathematics Freie Universität Berlin Thesis: Thermodynamic constraints for metabolic networks Supervisor: Prof. Dr. Alexander Bockmayr |
| 09/2010 | BSc in Mathematics Freie Universität Berlin Thesis: Selection of cut-lines to generate zigzag-paths for efficient computation of pseudotriangulations Supervisor: Prof. Dr. Günter Rote |
| 06/2010 | BSc in Computer Science Freie Universität Berlin Thesis: Selection of cut-lines to generate zigzag-paths for efficient computation of pseudotriangulations Supervisor: Prof. Dr. Günter Rote |

Teaching

| | |
|-----------------|--|
| 03/2014-05/2014 | Lecturer & Supervisor at Freie Universität Berlin Department of Mathematics and Computer Science Block course “project management in software development” |
| 10/2009-09/2011 | Teaching Assistant at Freie Universität Berlin Department of Mathematics and Computer Science Tutored courses: Algorithms and Datastructures, Foundations of Theoretical Computer Science |

Publications

- [Arne C. Müller](#)¹, Frank J. Bruggeman, Brett G. Olivier, Leen Stougie. Fast flux-module detection using matroid theory, *Research in Computational Molecular Biology*, Lecture Notes in Computer Science 8394:192-206, 2014 (original research,

published)

http://link.springer.com/chapter/10.1007%2F978-3-319-05269-4_16

- [Arne C. Müller](#)¹, Alexander Bockmayr. Flux modules in metabolic networks, *Journal of Mathematical Biology* (original research, in press)
<http://link.springer.com/article/10.1007%2Fs00285-013-0731-1>
- [Arne C. Müller](#)¹, Alexander Bockmayr. Fast thermodynamically constrained flux variability analysis, *Bioinformatics*, 29:903-909, 2013 (original research, published)
<http://bioinformatics.oxfordjournals.org/content/29/7/903>
- Christina Gunkel, Alexander Stepper, [Arne C. Müller](#)¹, Christine H. Müller. Micro crack detection with Dijkstra's shortest path algorithm, *Machine Vision and Applications* 23 (3): 589-601, 2012 (original research, published)
<http://www.springerlink.com/content/m1812u6gv6147623/>

Conference Talks

- *Fast flux module detection using matroid theory*, RECOMB 2014, Pittsburgh, USA
- *Flux modules in metabolic networks*, Workshop on Integrative Omics 2013, Pucon, Chile
- *Flux modules in metabolic networks*, Mini-Symposium on Stoichiometric Network Analysis, Berlin, Germany, 2013
- *Modules in metabolic networks*, Metabolic Pathway Analysis 2013, Oxford, UK, 2013
- *Fast thermodynamically constrained flux variability analysis*, Otto Warburg International Summer School And research Symposium on Genes, Metabolism, and Systems Modeling, Shanghai, China, September 2012
- *Fast thermodynamically constrained flux variability analysis*, International Symposium on Mathematical Programming 2012, Berlin, Germany, August 2012

Conference Posters

- *Propagation of Bounds on Gibbs Free Energy of Formation*, Metabolic Pathway Analysis 2013, Oxford, UK, 2013
- *Fast thermodynamically constrained flux variability analysis*, COBRA 2012 - 2nd Conference on Constraint Based Reconstruction and Analysis, Helsingør, Denmark, October 2012

¹Publications published before my name change are listed with my old name.

Awards, Prizes & Fellowships

| | |
|------------|--|
| since 2012 | Member of the International Max Planck Research School for Computational Biology and Scientific Computing |
| since 2011 | PhD-Stipend from Berlin Mathematical School (BMS) |
| since 2010 | Member of the BMS |
| 2007 | 2nd place, final round of robotics competition RoboKing 2007 (Team competition, Germany wide) |
| 2007 | Prize winner, final round of computer science competition Bundeswettbewerb Informatik 2007 (Germany wide) |
| 2006 | Prize winner, final round of computer science competition Bundeswettbewerb Informatik 2006 (Germany wide) |

Political Work Experience

| | |
|---------------|---|
| since 10/2011 | Member of the communal parliament of Steglitz-Zehlendorf in the parliamentary group of Piratenpartei Deutschland |
|---------------|---|

Declaration

I assert to have written this PhD-thesis on my own, that I cited all the sources I used, and that I did not use any illegitimate tools.

Ich versichere diese Arbeit selbständig, ohne unerlaubte Hilfsmittel verfasst und keine außer den angeführten Quellen verwendet zu haben.

Berlin, den 19. August 2014

Arne Reimers