

Institut für Mathematik
AG Mathematics in Life Sciences

Dissertation

zur Erlangung des Grades Dr. rer. nat.

Untersuchung metabolischer Netzwerke mit Verbandstheorie

Eingereicht von: Yaron Amir Benjamin Goldstein

Adresse: Binger Straße 75

14197 Berlin

mail@goldsteiny.de

Betreuer: Prof. Dr. Alexander Bockmayr

Berlin, Januar 2013

Erstgutachter: Prof. Dr. Alexander Bockmayr

Zweitgutachter: Prof. Dr. Bernhard Ganter

Tag der Disputation: 19. Dezember 2012

Die reine Formulierung eines Problems ist oftmals weit wichtiger als seine Lösung.

Neue Fragen aufzuwerfen, neue Möglichkeiten zu finden, alte Probleme aus einem neuen Blickwinkel zu betrachten, erfordert schöpferische Vorstellungskraft und macht die wirklichen Fortschritte in der Wissenschaft aus.

ALBERT EINSTEIN



Ziel dieser Arbeit ist es, metabolische Netzwerke und Zellstoffwechselfade durch eine algebraische Struktur zu beschreiben. Mit dieser sollen Untersuchungsmethoden entwickelt werden, die so formuliert sind, dass sie einfach für verschiedene qualitative Modelle genutzt werden können. In der Vergangenheit diente häufig ein Kegel als mathematische Grundlage, um die stationären Zustände des Stoffwechsels zu untersuchen. Konzepte wie elementare Flussmoden, Flusskopplungsuntersuchung (FCA) und Minimal Cut Sets wurden basierend auf diesem Flusskegel eingeführt. Diese Arbeit soll Modellierungskonzepte und Anwendungen identifizieren, die sich auf unsere abstrakte Form übertragen lassen, und so ein verallgemeinertes Framework vorstellen, mit dem der Zellstoffwechsel qualitativ untersucht werden kann.

Reaktionemengen, die gemeinsam im Stoffwechsel wirken können, werden durch Elemente eines Vereinigungsverbands repräsentiert. Elementarmoden werden qualitativ beschrieben und auf verbandstheoretische Konzepte übertragen. Basierend auf existierenden Algorithmen zur FCA wird ein Verfahren entwickelt, mit dem das maximale Element im Verband sehr schnell gefunden wird. Mit Hilfe dieses Maximums wird eine abstrakte, qualitative Flusskopplung definiert. Der zugehörige Algorithmus wurde implementiert und vollständige Reaktionen-Doppel-Knockouts (EFCA) wurden für verschiedene Zellmodelle erstmalig simuliert. Die Implementierung wirkt auf stationären Zuständen des Zellstoffwechsels, die um thermodynamische Nebenbedingungen ergänzt werden können. Weiterhin wird ein Konzept zur Target Prediction mittels Maximum-Berechnung vorgestellt und der Zusammenhang zu Cut Sets diskutiert.

Insgesamt stellt sich heraus, dass die bekanntesten Analysemethoden des Stoffwechsels zu den speziellsten Elementen im Vereinigungsverband korrespondieren. Es werden Einsatzmöglichkeiten der Verbandstheorie für das Standardmodell der stationären Zustände, sowie für Hyperpfade und ein logisches Modell als Grundkonzept gezeigt. Dabei stellt sich das logische Modell als verallgemeinerte Relaxation des Standardmodells heraus. Je nach Anwendungsfall entsprechen Elementarmoden den minimalen oder den unzerlegbaren Elementen eines Verbandes. Im Starbucks-Lemma wird gezeigt, dass sich mit diesen Elementen in vielen Modellkonzepten einfach arbeiten und sich Flusskopplung sogar für jedes qualitative Modell definieren lässt. Verschiedene Schreibweisen des Verbandsmaximums ergeben schließlich ein leicht verständliches Verfahren, das für alle Modelle gültig ist. Es kann gezeigt werden, dass die vorgestellte Implementierung für das Standardmodell ähnlich schnell wie die besten bekannten FCA-Algorithmen ist. Hierbei stellt sich sogar heraus, dass sich wesentliche Laufzeitverbesserungen nur noch durch effizientere Programmierung ergeben sollte.

In künftigen Arbeiten sollte untersucht werden, welche weiteren Modellierungsansätze sich in die Verbandstheorie übertragen lassen und in welchem Zusammenhang sie mit den anderen Modellen stehen. Eine Implementierung der Target Prediction steht noch aus. Deren Ergebnisse sowie die der EFCA müssen noch praktisch überprüft werden. Verbandstheoretisch formuliert lassen sich viele Fragen einfacher beantworten, da der Bezug zu bekannten Ergebnissen klarer wird. Damit ist das Framework der Verbandstheorie ausgezeichnet dafür geeignet, qualitative Fragen zu beantworten und Untersuchungsverfahren zu optimieren oder neu zu entwickeln.

Stichworte: Zellstoffwechsel; Modell; Verbandstheorie; allgemeines Framework; Stoffwechselfade; Flusskopplung; Reaktionen-Doppel-Knockout; Target Prediction; thermodynamische Nebenbedingungen; neue Modellierungskonzepte; qualitative Untersuchungsmethoden; schnelle Verfahren.



This thesis aims to describe metabolic networks and reaction pathways in the cell using an algebraic structure. On this basis we want to establish analytical methods that can be used for a variety of approaches in qualitative modeling. To analyze the metabolism, the steady state flux cone is presently the most commonly used mathematical model. Concepts such as elementary flux modes, flux coupling analysis (FCA) and minimal cut sets have been introduced based on this flux cone. This thesis identifies modeling approaches and their applications based on the developed abstract algebraic structure – thus introducing a general framework for qualitative analysis of the metabolism.

Reaction sets that can be simultaneously active in a cell are represented as the elements of a join (semi)lattice. A qualitative description of elementary flux modes is given and the concepts used are translated in lattice theory. Partially based on existing FCA algorithms, a method is presented that can rapidly identify the maximum of a lattice. This maximum is then used to define an abstract term of flux coupling that can be applied to a diversity of qualitative models. The implemented algorithm is applied to simulate a complete reaction double knockout (EFCA) in different cell models. These models are based on a steady state assumption that can be combined with further thermodynamic constraints. Additionally a concept of target prediction is introduced, utilizing the calculation of lattice maxima, and related to (minimal) cut sets.

Taken as a whole, it can be said that the most known analytical approaches for the metabolism correspond perfectly well to the most special elements of join lattices. We are able to apply lattice theory to both the standard model based on the flux cone, pathways through hyper graphs and a model using logic formulas. It becomes apparent that this logic model is a relaxation of the standard model. Depending on the application, the elementary flux modes are shown to be either the minimal or the irreducible elements of a lattice. The Starbucks lemma proves that both these sets can be easily worked with in many qualitative models. It further proves our FCA algorithm is correct for literally all kinds of qualitative models. We will show that our implementation, even though much easier to adapt to other models, has a running time comparable to well-established FCA algorithms. As a matter of fact we will see that further improvements of speed are most likely reachable only by more efficient ways of programming, not by more effective theoretical approaches.

Additional modeling approaches that can be translated into the terms of lattice theory should be identified in future. This translation into our general framework should further allow an easier comparison of those different approaches. Using lattice theory it is much easier to answer many open questions about the cell's metabolism. One reason for this is that it becomes easier to apply known results from certain models on other ones. Hence, the general framework of lattice theory is perfectly suited for qualitative modeling and to develop or optimize analytical methods.

Keywords: cell metabolism; models; lattice theory; general framework; reaction pathways; flux coupling; reaction double knockout; target prediction; thermodynamic constraints; new modeling concepts; qualitative analytical methods; fast algorithms.

Inhaltsverzeichnis

1. Einleitung – Ein Hauch von Magie	1
2. Biologische Modellierung	3
2.1 Mit den Augen eines Kindes	3
2.1.1 Blick auf die Zelle	3
2.1.2 Blick auf die Produktionswege	4
2.1.3 Formalisierung	6
2.2 Hyperpfade, sind doch logisch!	9
2.2.1 Über total logische Regeln	9
2.2.2 Mathematische Formulierung	11
2.2.3 Von Hyperpfaden und abgewandelten logischen Regeln	14
2.3 Klassisch stationär	18
2.3.1 Ursprung und Definition	18
2.3.2 Anwendungen	22
3. Algebraische Grundlagen	29
3.1 Endliche (Halb-)Verbände und Ordnung	30
3.1.1 Jeder endliche Halbverband mit 0 ist ein Verband	30
3.1.2 Ein Bisschen mehr Ordnung, bitte!	33
3.2 Minimale und unzerlegbare Elemente	34
3.2.1 Definitionen...	34
3.2.2 ...und Eigenschaften	37
3.3 Unterverbände	39
3.4 Homomorphismen	49
4. Vereinigungsverbände	53
4.1 Allgemeine Eigenschaften	54
4.1.1 Bedingte Unterhalbverbände	55
4.1.2 Projektionen	59
4.2 Überdeckungen	61
4.2.1 Exkurs: Warum keine Graphentheorie	61
4.2.2 Back to Metabolism: Ein wenig Graphentheorie	62
4.3 Kegelverbände	67
4.3.1 Polyedrische Verbände	67
4.3.2 Konische Verbände	70
5. Verbände in der Biologie	75
5.1 Flussmuster	76
5.1.1 Elementarmoden und Flussmuster	76
5.1.2 Vergleich von Untersystem-Modellen	79

5.2 Minimal Metabolic Behaviours	84
5.2.1 Wie spannt man einen Flusskegel auf?	84
5.2.2 Das Innerste nach außen stützen	85
6. Flusskopplung	91
6.1 Now and Then – Global ist besser als lokal	91
6.2 Von blockierten und gekoppelten Reaktionen	92
6.2.1 Blockiert VS. Essenziell	92
6.2.2 Gekoppelt	94
6.3 Traditionelle Verfahren zur FCA	98
6.3.1 Aufbau des Algorithmus	98
6.3.2 MaxCoupling: Schranken für $\mathcal{C}(L)$	102
6.3.3 Zeugen beschleunigen FCA	104
7. Anwendungen für das Maximum	107
7.1 Berechnung des Maximums im Vereinigungsverband	107
7.1.1 Eines für alle!	108
7.2 Erweiterte Flusskopplung	111
7.2.1 Wer 1 sagt...	111
7.2.2 ... kann auch 2 sagen!	112
7.3 Target Prediction – Minesweeper im Zellstoffwechsel	115
7.3.1 FBA und Derivate	115
7.3.2 Zwei Probleme, ähnlich definiert, gleich gelöst	116
7.3.3 Von Genen, Enzymen und Reaktionen	124
7.3.4 War da nicht was? – Minimal Cut Sets, 2. Akt	125
8. Das Starbucks-Lemma	129
8.1 Über Geschmack lässt sich nicht streiten	129
8.2 Guter Geschmack ist Ordnungsideal	133
8.3 Von Kreisen und Thermodynamik	138
A. So haben wir's gemacht	147
A.1 Qual der Wahl	147
A.2 Blaupausen	149
A.3 Verbesserungen und Feintuning	153
B. Graph & Zahl	155
B.1 Was wir gerechnet haben	155
B.2 FCA	158
B.3 EFCA	171
C. Was übrig bleibt	185
Glossar	i
Symbolverzeichnis	ix
Literaturverzeichnis	xv

KAPITEL 1

Einleitung – Ein Hauch von Magie

§1.1 Die Krankheiten der Gegenwart

Das Universum hatte 15 Milliarden Jahre Zeit, uns zu den komplexen Wesen zu machen, die wir heute sind. 4 Milliarden davon gehen auf die Evolution zurück, die per Trial and Error Einzeller weiterentwickelt hat.

Die medizinische Forschung steht vor immer größeren Problemen. Krebs, Altersdemenz und Bakterien, die gegen bekannte Antibiotika resistent sind, seien da nur als drei Beispiele von vielen genannt. Dass die Probleme immer schwerer werden, ist eigentlich Folge einer positiven Entwicklung: In den vergangenen 150 Jahren haben es Forscher weltweit geschafft, hunderte Krankheiten zu besiegen, die bis dahin für den Tod Millionen von Menschen jährlich verantwortlich waren.

Der technische Fortschritt in der Medizin ist immens. Doch sei es Medikamentenentwicklung, bildgebende Verfahren (Feeman 2010) oder eine teilweise Heilung von Blindheit (Nirenberg & Pandarinath 2012) – diesen raschen Fortschritt gäbe es nicht ohne Mathematik.

§1.2 Wie kann Mathematik helfen?

Mathematik zweigt die Welt. Wer in Schule und womöglich Universität mit ihr konfrontiert wird, der liebt sie oder der hasst sie. Dazwischen gibt es wenig.

Das hat vermutlich mit ihrer Kompromisslosigkeit zu tun. Mit Mathematik teilt man die Welt in wahr und falsch. Eine Aussage stimmt. Oder sie stimmt nicht. Tertium non datur.

Um diese Absolutheit zu erreichen, ist die Mathematik eine hoch präzise und rigide Sprache. Man einigt sich auf einen später indiskutablen Satz von Grundregeln (Axiomen) und logische

Regeln, wie für wahr befundene Aussagen miteinander kombiniert werden dürfen, um andere Aussagen zu prüfen.

Jeder Begriff, der hierbei verwendet wird, muss vorher definiert werden. Eine Definition muss eindeutig und in sich abgeschlossen sein. Wenn zwei die Sprache der Mathematik beherrschen und eine Definition lesen, müssen beide im Anschluss das gleiche Verständnis vom Definierten haben. Auch deshalb ist Mathematik oft sehr abstrakt. Doch gerade in dieser Abstraktion liegt ihre Stärke.

Mathematik hat viel mit Sprachen gemein: Bemüht man sich noch so sehr, keine deutsche Fassung von Catulls *Odi et amo* kommt in Bedeutung, Sprachmelodie, Metrik und Stil dem Original gleich. Die Kunst besteht darin, den Kompromiss zu finden, mit dem alle momentan wichtigen Aspekte erhalten bleiben. Oftmals lernt man das Objekt der Translation durch diese genauen Überlegungen zur Neuformulierung in der neuen Sprache viel besser kennen.

Mit der Mathematik ist es nicht anders. Übersetzt man eine Frage in die Mathematik, erstellt man ein mathematisches Modell von einem realexistierendem Objekt oder Prozess, so gehen Details verloren. Dafür wird die Sicht auf andere umso besser. Die Kunst besteht darin, das Modell zu finden, mit dem alle wesentlichen Eigenschaften erhalten bleiben, die zur Beantwortung der Frage wichtig sind. Oftmals lernt man die wesentlichen Probleme, die hinter der Frage stehen, durch die auf Exaktheit getrimmte Formulierung in die Mathematik viel besser kennen.

Die Abstraktion im Modell erlaubt es nun, Techniken aus der gesamten Mathematik anzuwenden, auch wenn sie nur scheinbar wenig mit dem ursprünglichen Problem zu tun haben. Manchmal entdeckt man im Abstrakten ganz wesentliche Eigenschaften des Objekts, die man zuvor nicht beachtet hatte. Die aber wichtig sind. Oder man ändert die Perspektive nur ganz leicht und merkt, wie trivial viele Probleme sind, an denen man zuvor hart gearbeitet hatte. Das ist die Magie der Mathematik: Der Zuschauer, der die Tricks nicht kennt, sieht, wie ein Kaninchen im Zylinder verschwindet und eine Taube herausfliegt.

In dieser Arbeit wollen wir den Stoffwechsel von Zellen untersuchen. Dafür vereinfachen wir sehr stark und untersuchen danach abstrakt. Wir untersuchen oftmals nur stationäre Stoffwechselzustände und berauben die Zelle somit ihrer Dynamik. Wir betrachten nur, welche Reaktionen stattfinden können und welche nicht. Welche gleichzeitig stattfinden können und welche redundant abgesichert sind. Schließlich werden wir so zeigen, dass man mit dem gleichen Verfahren herausfindet, wie man eine Zelle zur Biogasfabrik umprogrammiert, wie man auch nebenwirkungsarme Medikamentenkombinationen findet. Wir ändern zwei Vorzeichen und schon wird aus einem Verfahren, wie es wirtschaftlich hoch interessant ist, eines, dass medizinische Grundlagenforschung erlaubt.

Und so soll auch diese Arbeit von einem Hauch Magie durchzogen sein.

KAPITEL 2

Biologische Modellierung

Kapitelübersicht

2.1 Mit den Augen eines Kindes	3
2.1.1 Blick auf die Zelle	3
2.1.2 Blick auf die Produktionswege	4
2.1.3 Formalisierung	6
2.2 Hyperpfade, sind doch logisch!	9
2.2.1 Über total logische Regeln	9
2.2.2 Mathematische Formulierung	11
2.2.3 Von Hyperpfaden und abgewandelten logischen Regeln	14
2.3 Klassisch stationär	18
2.3.1 Ursprung und Definition	18
2.3.2 Anwendungen	22

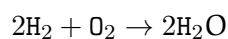
§2.1 Mit den Augen eines Kindes

§2.1.1 Blick auf die Zelle

Wenn man einem Kind erklären will, was der Stoffwechsel einer Zelle ist, könnte man es so versuchen:

Eine Zelle ist wie eine kleine Fabrik im Körper. Man beliefert sie mit einigen Ausgangsmaterialien, die sie dann über mehrere Verarbeitungsschritte in neue Produkte umwandelt. Dabei nennen wir alle Stoffe, die vorkommen – also die Ausgangsmaterialien sowie die Zwischen- und Endprodukte – **Metabolit**. Das bedeutet Stoffwechselprodukt.

Die Verarbeitungsschritte hingegen nennen wir (biochemische) **Reaktionen**. So eine Reaktion kann folgendermaßen aussehen:



Welche Reaktionen in einer Zelle möglich sind, hängt vor allem von dem Genom der Zelle ab. Das ist eine Art Bauplan der Zelle, mit dem die Zelle alles herstellen kann, was sie zum Funktionieren benötigt. Unter anderem weiß die Zelle so, wie sie Enzyme herstellt. Enzyme sind so etwas wie die Maschinen in einer Fabrik: Zwar schwimmen die Enzyme in der Zelle herum, haben keinen festen Platz, aber bei ihnen finden die chemischen Reaktionen, die Bearbeitungsprozesse statt. Ohne Enzym (meistens) keine Reaktion. Deshalb nennt man die Enzyme auch Katalysatoren der Reaktionen. Ein Katalysator ist ein Beschleuniger.

Metaboliten,
Reaktionen



Beispiel 2.1 (Metaboliten, Reaktionen).

In Abbildung 2.1 ist das sehr einfache Netzwerk *etc_purple* zu sehen (Klamt, Grammel, Straube, Ghosh & Gilles 2008).

In dieser Zelle sind die internen Metaboliten *ATP*, *c2-red*, *Hp*, *NADH*, *O2*, *QH2* und *Succ* enthalten. Zudem sind die Reaktionen *ATPdrain*, *ATPSynthase*, *bc1*, *c2-oxidase*, *NADH-DH*, *NADH.in*, *O2.in*, *Photo*, *Q-oxidase*, *Succ-DH* und *Succ.in* möglich.

Bei den Reaktionen *NADH.in*, *Succ.in* und *ATPdrain* handelt es sich um Transportreaktionen, die die entsprechenden Metaboliten aus einer externen Quelle in die Zelle hinein befördern oder aus ihr heraus transportieren können. Wir betrachten die Prozesse außerhalb und somit insbesondere externe Metaboliten nicht.

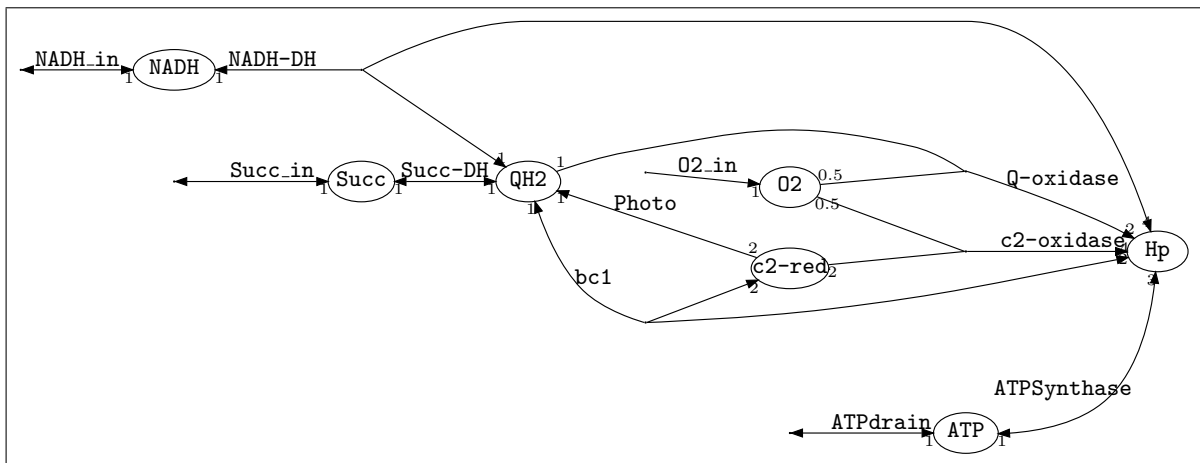


Abbildung 2.1.: Ein metabolisches Netzwerk (*etc_purple*).

In Wirklichkeit ist alles natürlich noch viel komplizierter. Wir werden aber in dieser Arbeit sehen, wie weit man mit dieser einfachen Beschreibung kommt.

§2.1.2 Blick auf die Produktionswege

Die wissenschaftliche Disziplin, in der man mittels Mathematik und (Bio-)Informatik biologische Prozesse untersucht, wird auch Systembiologie genannt. Die Fragestellungen und Anwendungsgebiete, die Techniken der Systembiologie sind mannigfaltig. Palsson (2009) gibt auf wenigen Seiten einen Überblick über die Entwicklung dieses Wissenschaftszweigs. Dem interessierten Leser stehen aber auch viele andere, ausführlichere Werke zur Verfügung (z. B. Klipp, Liebermeister, Wierling, Kowald, Lehrach & Herwig 2009).

Diese Arbeit bewegt sich in eng markierten Grenzen innerhalb der Systembiologie. Wir untersuchen mit diskreten mathematischen Modellen Stoffwechselprozesse innerhalb einzelner Zellen. Dafür stellen wir ein Framework vor – die Verbandstheorie – mit der sich unterschiedliche Modelle standardisieren lassen. Das werden wir für einige mögliche Modelle zeigen. Zudem werden wir Untersuchungswerkzeuge vorstellen und einführen, die sich in diesem Rahmen bewegen, in diesem Framework entstanden sind. Am Ende werden wir effiziente Verfahren haben, um beliebige Modelle in Verbandsform untersuchen zu können. Anstelle bloßer Werkzeuge, die auf spezielle Modelle eingetrimmt sind, wie bisher.

Diskrete Stoffwechselmodelle drehen sich meist um die Produktionswege bestimmter Metaboliten. Wenn man weiß, welche Reaktionen in einer Zelle stattfinden können, kann man versuchen, diese Produktionswege zu rekonstruieren. So ein Produktionsweg umfasst die verschiedenen Verarbeitungsschritte, die stattfinden, ist also eine Aufzählung der Reaktionen.

Nicht alle Reaktionen passen zueinander, nur bestimmte Kombinationen ergeben einen in sich abgeschlossenen Produktionsablauf. Biologen haben verschiedene Erklärungen und Regeln, die so ein Reaktionsweg durch die Zelle erfüllen sollte. Eine Forderung lautet beispielsweise, dass jedes Zwischenprodukt, das in der Zelle hergestellt wird, durch andere Reaktionen wieder im gleichen Maße abgebaut werden sollte. Weil hierbei innerhalb der Zelle quantitativ nichts Neues entsteht und auch nichts abgebaut wird, die Konzentrationen der Metaboliten unverändert bleiben, spricht man hierbei von **stationärer Zustand** oder in einigen Fällen stabiler Zustand. Die Menge aller stationären Zustände bilden einen Kegel, den wir in Definition 2.8 offiziell einführen werden. Da der Ansatz, die stationären Zustände zu untersuchen, so verbreitet ist, werden wir auch vom *Standardmodell* reden.

Standardmodell

Diese Forderung nach stationären Zuständen ist sehr stark, sie nimmt der Zelle viel Freiraum. Da die Konzentrationen der Metaboliten in einer Zelle in der Realität innerhalb eines gewissen Rahmens schwanken, versuchen einige Wissenschaftler mit anderen Regeln die Stoffwechselwege zu bestimmen. Einige Abwandlungen werden wir hier gleich schon kennenlernen.

Egal an welche Regeln man sich hält, wir wollen jedes Mal von einem **Stoffwechselfad** reden, wenn eine Reaktionenmenge als gültige Kombination betrachtet wird.

Stoffwechselfad

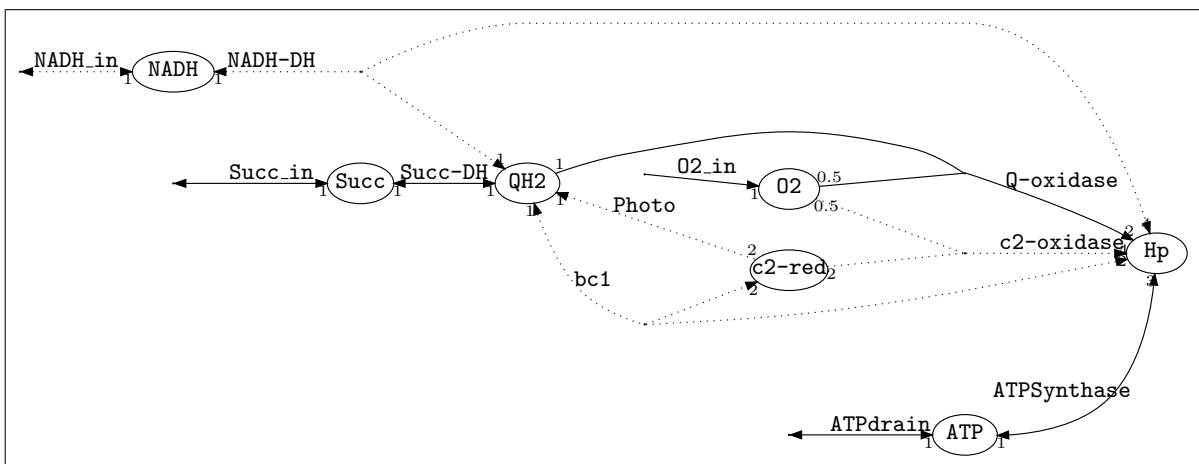


Abbildung 2.2.: Ein Stoffwechselfad (etc_purple).

**Beispiel 2.2 (Stoffwechselfad).**

In Abbildung 2.2 ist ein Stoffwechselfad im *etc.purple*-Netzwerk zu sehen.

Wählt man die Reaktionsgeschwindigkeiten $Succ_in = 1, Succ_DH = 1, O2_in = \frac{1}{2}, Q_oxidase = 1, ATPSynthase = \frac{2}{3}, ATPdrain = \frac{2}{3}$, so erhält man einen stationären Zustand.

§2.1.3 Formalisierung

Wir haben gerade gesagt, dass man den Zellstoffwechsel allein schon durch die Reaktionen und Metaboliten beschreiben kann, die in der Zelle ablaufen können bzw. vorkommen.

Alle chemischen Reaktionen haben einen Namen, der meistens ihre Funktion beschreibt. Wir wollen in dieser Arbeit jedoch nicht den Stoffwechsel einer bestimmten Zelle betrachten, sondern ganz allgemeine Aussagen treffen, die möglichst für alle Zellen gelten sollen. Statt der echten Namen werden wir deshalb unsere Reaktionen schlicht durchnummerieren. Der Stoffwechsel einer Zelle, in der 42 Reaktionen stattfinden können, umfasst also die Reaktionen $\{1, \dots, 42\}$. Ganz allgemein sagen wir, dass in einer Zelle die n Reaktionen $[n] = \{1, \dots, n\}$ ablaufen können.

Einige dieser Reaktionen haben eine besondere Eigenschaft: Sie können in beiden Richtungen wirken. So eine Reaktion nennen wir **reversible Reaktion**, die anderen **irreversible Reaktion**. Wir teilen somit unsere n -Menge an Reaktionen in zwei Teilmengen auf:

$$[n] = \text{Rev} \cup \text{Irr}.$$

**Beispiel 2.3 (Rev, Irr).**

Im Netzwerk *etc.purple* gilt $\text{Rev} = \{ATPdrain, ATPSynthase, bc1, NADH-DH, NADH_in, Succ-DH, Succ_in\}$ und $\text{Irr} = \{c2-oxidase, O2_in, Photo, Q-oxidase\}$.

Ein gültiger Stoffwechselfad ist ebenso wie Rev und Irr eine Teilmenge von $[n]$ – egal wie unsere Regeln lauten. Die Menge aller gültigen Stoffwechselfade L ist somit also eine Teilmenge von $2^{[n]}$, der Potenzmenge von $[n]$.

Da wir hier unabhängig von einem bestimmten Regelsatz arbeiten wollen, brauchen wir einige Grundregeln, die jeder sinnvolle Regelsatz erfüllen sollte. Dabei kommen wir auf zwei Forderungen an L .

- Es ist erlaubt, dass *nichts* passiert.


Wir sprechen dann vom leeren Stoffwechselfad. Wir fordern deshalb $\emptyset \in L$, da keine Reaktion stattfindet.

- Wir können zwei Stoffwechselfade miteinander kombinieren.

Wenn $\alpha_1, \alpha_2 \in L$ sinnvolle Produktionspfade darstellen, dann können die Prozesse auch gleichzeitig stattfinden. Also soll auch $\alpha_1 \cup \alpha_2 \in L$ sein.

Diese beiden Eigenschaften zusammen machen L aus mathematischer Sicht zu einem **Halbverband** mit neutralem Element $0 = \emptyset$. Da $L \subseteq 2^{[n]}$ gilt, sprechen wir von einem Vereinigungshalbverband. Im nächsten Kapitel werden wir sehen, dass L sogar ein mathematischer Vereinigungsverband ist, weil wir immer nur endliche Mengen betrachten.

Doch zunächst halten wir fest:



Definition 2.1 (Halbverband).
Sei $(L, +)$ abelsche Halbgruppe und idempotent, also $a + a = a$ für alle $a \in L$. Dann heißt $(L, +)$ Halbverband.

**Halb-
verband**

That's it. Unsere erste Definition steht nun auch formell und damit verfügen wir über einen mächtigen Apparat, wie wir noch oft sehen werden.

Die Halbverbandsstruktur ist alles andere als neu. Dennoch gibt es kaum Literatur über sie – Mathematikbücher stellen sie meist nur als Einleitung für allgemeine Verbandstheorie vor.

Das liegt auch daran, dass man aus Mathematikersicht nur wenig Arbeitsmaterial hat: Wir haben eine nicht-invertierbare Operation für zwei Elemente. Jede Gruppe bietet mehr. Bei einem richtigen Verband, den wir in Definition 3.1 vorstellen, verfügen wir zumindest über zwei Operationen. Dann können wir sehen, wie die beiden miteinander interagieren.

In der Biologie reicht uns aber die eine Operation. Wir wollen zwei Stoffwechselfade miteinander kombinieren. Das genügt.

Im Gegensatz zu den Standardwerken betrachten wir nicht allgemeine Verbände, sondern endliche Halbverbände. Unendliche Reaktionsmengen werden wir in dieser Arbeit nie begegnen – jedes Mal, wenn wir von Halbverband oder Verband reden werden, wird eine endliche Menge gemeint sein. Entsprechend treffen wir auch nur Aussagen für solche Verbände. Wir werden beispielsweise zeigen, dass jeder endliche Halbverband mit Nullelement auch ein Verband ist. Wir können also auch mögliche Eigenschaften von (endlichen) Verbänden nutzen und prüfen, ob sie zu einer interessanten Anwendung führen.

Klassische Verbandstheorie hatte ihre Hochzeit in den 1970er Jahren. Da uns Eigenschaften unendlicher Verbände gar nicht interessieren, reichen uns oftmals die reinen Grundlagen aus. Die meisten Ergebnisse, die wir in diesen beiden Kapiteln kennenlernen, kann man deshalb in den Einführungskapiteln beliebiger Bücher zur Verbandstheorie in mal kürzerer, mal ausführlicherer Form nachlesen. Hervorgehoben soll hierbei das Buch von Davey & Priestley (1990) sein, das Pate für viele Dinge steht, die wir in diesem Kapitel vorstellen, und unverzichtbares Hilfsmittel zur Einarbeitung in die Halbverbandstheorie war. Als Alternative sei auf die Einführung in die Verbandstheorie von Nation (2012) verwiesen. Es ist eines der wenigen Werke, das explizit auf den Fakt hinweist, dass beschränkte, endliche Halbverbände immer auch Verbände sind.

Wir werden Verbandstheorie verwenden, um Stoffwechselprozesse abstrahiert betrachten zu können. Doch dabei soll es nie zu Abstraktion um ihrer Selbst willen kommen. Viel wichtiger ist es uns, die Verbände *zur Anwendung* zu bringen. Damit stehen wir in der Tradition von Wille (2009), der seinen Artikel (Original 1982) mit den folgenden Worten einleitet:

Lattice theory today reflects the general status of current mathematics: there is a rich production of theoretical concepts, results, and developments, many of which are reached by elaborate mental gymnastics; on the other hand, the connections of the theory to its surroundings are getting weaker and weaker, with the result that the theory and even many of its parts become more isolated. Restructuring lattice theory is an attempt to reinvigorate connections with our general culture by interpreting the theory as concretely as possible, and in this way to promote better communication between lattice theorists and potential users of lattice theory.

Wille ist einer der Mitbegründer der Formalen Begriffsanalyse (Ganter, Stumme & Wille 2005), die vollständige Verbände und deren Eigenschaften untersucht. Verbandselemente werden immer zusammen mit ihren Eigenschaften betrachtet und zu Kontexten zusammengefasst. Diese Kontexte bilden dann ihrerseits wieder Verbände, an denen u. a. die Abhängigkeiten der verschiedenen Eigenschaften voneinander abgelesen werden können.

Formale Begriffsanalyse findet vor allem in Gebieten Anwendung, wo Informationen miteinander abgeglichen werden müssen. Etwa data mining, dem semantischen Web, aber auch in der Wirtschaft.

Wir werden in dieser Arbeit nicht auf Kontexte eingehen, aber wir versuchen etwas Ähnliches zu erreichen: Verbandstheorie als Framework, als Arbeitswerkzeug zu etablieren, mit dem sich Stoffwechselprozesse exzellent beschreiben und untersuchen lassen.

Die klassische Literatur verwendet als Halbverbandsoperation meist \wedge oder \vee , für Vereinigungshalbverbände manchmal auch \cup . Wir haben uns für das $+$ als Operationssymbol entschieden, um zu verdeutlichen, dass wir in unserer Anwendung Stoffwechselfpade miteinander verbinden wollen. Entsprechend nutzen wir das Summenzeichen \sum , um mehrere Pfade gleichzeitig zu kombinieren.

Damit kommen wir zurück zur Biologie. Hier nun definieren wir:

met.
System



Definition 2.2 (metabolisches System).

Sei L Vereinigungshalbverband der gültigen Reaktionsmengen und $Rev \subseteq [n]$ Menge der reversiblen Reaktionen. Dann nennen wir (L, Rev) *metabolisches System* über n Reaktionen.

Informationen über die Reaktionen und Metaboliten werden uns meist in Matrixform geliefert. Wir nennen sie *stöchiometrische Matrix*, da sie uns Informationen über die Stöchiometrie (Mengenverhältnisse der Reaktionen) verrät. So kommen wir zu unserer nächsten Grundlagendefinition:



Definition 2.3 (metabolische Bedingungen).

Sei $S \in \mathbb{R}^{m \times n}$ stöchiometrische Matrix, d. h. $|s_{ij}|$ gibt an, wie oft Metabolit i in der Reaktion j ab- ($s_{ij} < 0$) bzw. aufgebaut ($s_{ij} > 0$) wird.

Zusätzlich sei die Menge $\text{Irr} \subseteq [n]$ die Menge der irreversiblen Reaktionen.

Dann nennen wir (S, Irr) metabolische Bedingungen.



Beispiel 2.4 (stöchiometrische Matrix (etc_purple)).

Die metabolischen Bedingungen von *etc_purple* lauten^a:

$$S = \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & -2 & 0 & -2 & 0 \\ 0 & -3 & 4 & 4 & 0 & 0 & 0 & 2 & 0 & 0 & 2 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -0.5 & 1 & 0 & -0.5 \\ 0 & 0 & -1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$\text{Irr} = \{8, \dots, 11\}.$$

^aBeachte: Metaboliten, reversible und irreversible Reaktionen sind lexikographisch sortiert indiziert.

Stöchio-
metrie

§2.2 Hyperpfade, sind doch logisch!

§2.2.1 Über total logische Regeln

Unser erster Regelsatz, der uns zu einem (Halb-)verband führen wird, ist nicht der Klassiker in der Literatur. Die meisten wissenschaftlichen Veröffentlichungen, die ein diskretes Modell zur Beschreibung des Stoffwechsels verwenden, fordern wie erwähnt stationäre Zustände für die Metabolitkonzentrationen.

Zwar lässt sich experimentell belegen, dass Zellen schnell auf Stress reagieren können und dann in einen neuen stationären Zustand wechseln (z. B. Jozefczuk, Klie, Catchpole, Szymanski, Cuadros-Inostroza, Steinhäuser, Selbig & Willmitzer 2010), aber dies ist eben nicht die einzige Möglichkeit, wie ihr Stoffwechsel funktionieren kann.

Wir wollen hier einen alternativen Ansatz vorstellen, Stoffwechselbeziehungen innerhalb einer Zelle zu untersuchen. Dafür fordern wir nicht mehr einen stationären (Konzentrations-)zustand der Metaboliten, sondern lediglich, dass jeder Metabolit, der produziert wird, auch wieder konsumiert wird. Und umgekehrt.

Dass solche Regeln sinnvoll sind (und möglichst von allen Regelsätzen erfüllt werden sollten) sieht man schnell ein:

- Wenn ein Metabolit umgewandelt werden soll, muss er zunächst einmal vorhanden sein. Aus nichts kommt nichts.
- Zudem sollte die Zelle sich darum kümmern, dass alle Stoffwechselprodukte, die sie herstellt, auch benötigt werden. Schließlich würde sie nicht nur umsonst Energie (oder Nährstoffe) verschwenden, wenn sie ein Zwischenprodukt bedarfslos herstellt. Zusätzlich würde sie sich langfristig selbst vergiften oder platzen, wenn sie einen Stoff in immer größerer Menge produzierte, den sie nicht weiterverwendet, sondern nur in sich selbst lagert.

logisches
Modell



Beispiel 2.5 (logisches Modell).

In Abbildung 2.3 ist ein Stoffwechselfad in *etc_purple* zu sehen, der den logischen Regeln folgt.

Gewichtet man alle aktiven Reaktionen mit einer Flussgeschwindigkeit von 1, so werden genau die Metaboliten *QH2*, *c2-red*, *Hp* und *NADH* produziert und konsumiert. Es ist jedoch kein stationärer Zustand möglich, da *QH2* in größerer Menge verbraucht wird, als es durch die Reaktion *Photo* produziert wird.

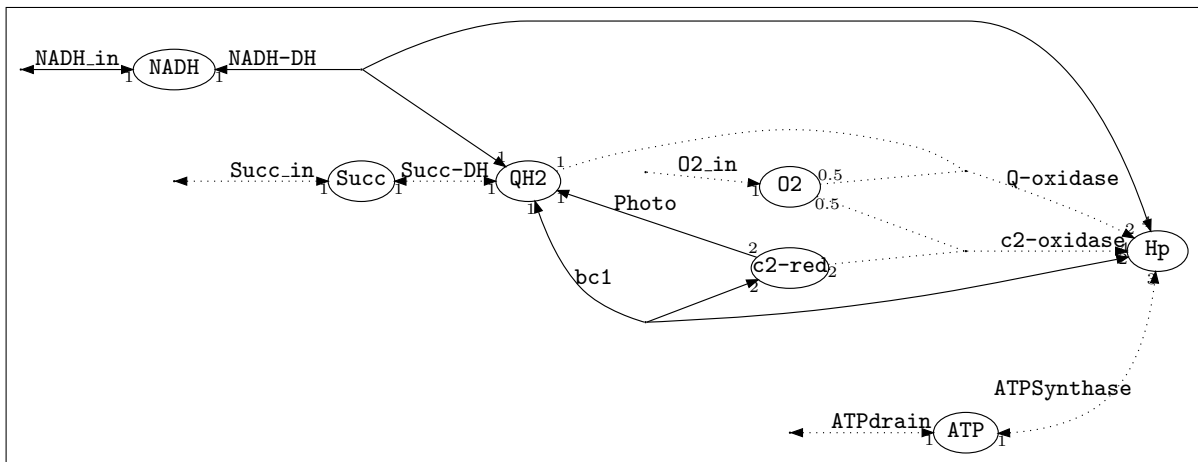


Abbildung 2.3.: Ein logischer Stoffwechselfad (*etc_purple*).

Unperfekte Logik

Im gewissen Rahmen können Zellen Metaboliten unverbraucht vorrätig halten und später von diesen Vorräten zehren. Sie können Konzentrationen nach oben und unten abpuffern.

Unser Minimalregelsatz greift darauf zurück, dass Metaboliten überschüssig produziert werden können oder eben dem Bedarf genügend. Eine quantitative Prüfung findet nicht statt. Insofern werden die physikalischen Gegebenheiten ebenso ignoriert wie beim Standardmodell mit der Forderung nach stationären Zuständen.

Auch andere Aspekte wie Genregulation (Covert 2002) sind bei beiden Ansätzen nicht integriert. Es ist wichtig, diese systematischen Fehler nicht zu vergessen, wenn man die Vor- und Nachteile der Verfahren abwägt. Perfekt sind sie beide nicht!

§2.2.2 Mathematische Formulierung

Unser System lässt sich gut als logische Formel definieren. Die einzige Schwierigkeit besteht darin, mit reversiblen Reaktionen richtig umzugehen.

Definition 2.4 (logisches Modell).

Seien (S, Irr) metabolische Bedingungen.

Seien C^{01} , $(M_i)_{i \in [m]}$ sowie $(R_j)_{j \in \text{Rev}}$ aussagenlogische Klauseln und wie folgt definiert:

$$M_i^{(+)} := \bigvee_{\substack{k \in \text{Irr} \\ S_{ik} > 0}} x_k \vee \bigvee_{\substack{k \in \text{Rev} \\ S_{ik} > 0}} x_k^{(+)} \vee \bigvee_{\substack{k \in \text{Rev} \\ S_{ik} < 0}} x_k^{(-)},$$

$$M_i^{(-)} := \bigvee_{\substack{k \in \text{Irr} \\ S_{ik} < 0}} x_k \vee \bigvee_{\substack{k \in \text{Rev} \\ S_{ik} < 0}} x_k^{(+)} \vee \bigvee_{\substack{k \in \text{Rev} \\ S_{ik} > 0}} x_k^{(-)},$$

$$M_i := M_i^{(+)} \Leftrightarrow M_i^{(-)},$$

$$R_j := \neg (x_j^{(+)} \wedge x_j^{(-)}) \wedge (x_j \Leftrightarrow x_j^{(+)} \vee x_j^{(-)}),$$

$$C^{01} := \bigwedge_{i \in [m]} M_i \wedge \bigwedge_{j \in \text{Rev}} R_j.$$

Dann heißt C^{01} *logisches Modell* zu (S, Irr) .

Seien nun $x_i, x_j^{(\pm)} \in \{\text{true}, \text{false}\}$ ($i \in [n], j \in \text{Rev}$) so, dass $C^{01}(x, x^{(+)}, x^{(-)}) = \text{true}$.

Dann heißt $(x, x^{(+)}, x^{(-)})$ *gültige Belegung* von C^{01} .

Es ist $(C^{01})^{-1}(\text{true})$ die Menge der gültigen Belegungen des Systems.

 logisches
Modell

Die Klauseln $M_i^{(\pm)}$ repräsentieren die Aussagen *Metabolit i wird produziert* bzw. *i wird konsumiert*. Für reversible Reaktionen $j \in \text{Rev}$ stellen wir durch die R_j zudem sicher, dass nur eine Reaktionsrichtung gleichzeitig genutzt wird.

Aus einem solchen Modell ergibt sich der logische Verband zu (S, Irr) :

Definition 2.5 (logischer Verband).

Sei C^{01} logisches Modell zu (S, Irr) .

Sei nun $L^{01} \subseteq 2^{[n]}$ wie folgt definiert:

$$L^{01} := \left\{ a \subseteq [n] \mid \exists (x, x^{(+)}, x^{(-)}) \in (C^{01})^{-1}(\text{true}) : (x_i = \text{true} \Leftrightarrow i \in a) \right\}.$$

Dann heißt L^{01} *logischer Verband* zu den metabolischen Bedingungen (S, Irr) .

 logischer
Verband

Bemerkung 2.1 (Wohldefiniertheit).

L^{01} ist wirklich Vereinigungshalbverband und somit (L^{01}, Irr) metabolisches System.

Beweis. Es reicht zu zeigen, dass L^{01} Halbverband mit 0 ist; wir werden im nächsten Kapitel sehen, dass L^{01} dann auch Verband ist. Also ist zu zeigen, dass $0 = \emptyset \in L^{01}$ und $\mathbf{a}_1, \mathbf{a}_2 \in L^{01} \Rightarrow \mathbf{a}_1 + \mathbf{a}_2 \in L^{01}$.

Es ist $0 \in L^{01}$, da $C^{01}(\text{false}, \dots, \text{false}) = \text{true}$.

Seien nun $\mathbf{a}_1, \mathbf{a}_2 \in L^{01}$ und $(\mathbf{x}^{(1)}, \mathbf{x}^{(1,+)}, \mathbf{x}^{(1,-)}), (\mathbf{x}^{(2)}, \mathbf{x}^{(2,+)}, \mathbf{x}^{(2,-)}) \in (C^{01})^{-1}(\text{true})$ zugehörige gültige Belegungen. Es ist zu zeigen, dass es eine gültige Belegung $(\mathbf{x}, \mathbf{x}^{(+)}, \mathbf{x}^{(-)})$ gibt mit $\mathbf{x}_i = \text{true} \Leftrightarrow i \in \mathbf{a}_1 + \mathbf{a}_2 = \mathbf{a}_1 \cup \mathbf{a}_2$.

Seien nun $(\mathbf{x}, \mathbf{x}^{(+)}, \mathbf{x}^{(-)})$ wie folgt definiert:

$$\begin{aligned} \mathbf{x}_i &:= \mathbf{x}_i^{(1)} \vee \mathbf{x}_i^{(2)}, \\ \mathbf{x}_j^{(\pm)} &:= \begin{cases} \mathbf{x}_j^{(1,\pm)} & \text{für } \mathbf{x}_j^{(1)}, \\ \mathbf{x}_j^{(2,\pm)} & \text{für } \neg \mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(2)}, \\ \text{false} & \text{sonst.} \end{cases} \end{aligned}$$

Wir wollen nun zeigen, dass $(\mathbf{x}, \mathbf{x}^{(+)}, \mathbf{x}^{(-)})$ gültige Belegung ist.

Sei $j \in \text{Rev}$ beliebig, dann:

$$\begin{aligned} \mathbf{x}_j^{(+)} \vee \mathbf{x}_j^{(-)} &= \left(\mathbf{x}_j^{(1)} \wedge \left(\mathbf{x}_j^{(1,+)} \vee \mathbf{x}_j^{(1,-)} \right) \right) \vee \left(\neg \mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(2)} \wedge \left(\mathbf{x}_j^{(2,+)} \vee \mathbf{x}_j^{(2,-)} \right) \right) \vee (\neg \mathbf{x}_j \wedge \text{false}) \\ &= \left(\mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(1)} \right) \vee \left(\neg \mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(2)} \wedge \mathbf{x}_j^{(2)} \right) \vee \text{false} \\ &= \mathbf{x}_j^{(1)} \vee \left(\neg \mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(2)} \right), \\ &= \mathbf{x}_j^{(1)} \vee \mathbf{x}_j^{(2)}, \\ &= \mathbf{x}_j, \\ \mathbf{x}_j^{(+)} \wedge \mathbf{x}_j^{(-)} &= \left(\mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(1,+)} \wedge \mathbf{x}_j^{(1,-)} \right) \vee \left(\neg \mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(2)} \wedge \mathbf{x}_j^{(2,+)} \wedge \mathbf{x}_j^{(2,-)} \right) \vee (\neg \mathbf{x}_j \wedge \text{false}) \\ &= \left(\mathbf{x}_j^{(1)} \wedge \text{false} \right) \vee \left(\neg \mathbf{x}_j^{(1)} \wedge \mathbf{x}_j^{(2)} \wedge \text{false} \right) \vee \text{false} \\ &= \text{false}, \\ &\Rightarrow R_j(\mathbf{x}, \mathbf{x}^{(+)}, \mathbf{x}^{(-)}) = \text{true}. \end{aligned}$$

Dass auch $M_i^{(+)} \Leftrightarrow M_i^{(-)}$, folgt aus der Konstruktion von $\mathbf{x}^{(\pm)}$, bei der wir $\mathbf{x}^{(1,\pm)}$ direkt übernommen und durch Reaktionen von $\mathbf{x}^{(2,\pm)}$ ergänzt haben:

Sei $i \in [m]$ ein Metabolit, der in $(\mathbf{x}, \mathbf{x}^{(+)}, \mathbf{x}^{(-)})$ produziert wird. Also wird er auch in \mathbf{a}_1 oder \mathbf{a}_2 produziert.

Betrachte zuerst $i \in [m]$ Metabolit, der in \mathbf{a}_1 produziert wird. Wenn i in \mathbf{a}_1 produziert wird, so gibt es eine Reaktion $k \in \mathbf{a}_1$, die i konsumiert, also $\mathbf{x}_k = \text{true}$ und i wird auch in \mathbf{a} von k konsumiert, da alle Reaktionsrichtungen von \mathbf{a}_1 übernommen werden.

Betrachten wir jetzt den Fall, dass i nicht in \mathbf{a}_1 sondern nur in \mathbf{a}_2 produziert wird. Dann gibt es eine Reaktion $j \in \mathbf{a}_2$, die i konsumiert, also $x_j = \mathbf{true}$. Dann gilt insbesondere $x_j^{(1)} = \mathbf{false}$, da ansonsten i in \mathbf{a}_1 konsumiert wird und somit auch produziert werden muss oder (Reaktion j ist reversibel und wird entgegengesetzt zu $(x^{(2)}, x^{(2,+)}, x^{(2,-)})$ genutzt) direkt durch j produziert. Wegen $x_j^{(1)} = \mathbf{false}$ gilt $x_j^{(\pm)} = x_j^{(2,\pm)}$, also konsumiert j auch in $(x, x^{(+)}, x^{(-)})$ den Metaboliten i .

Sollte i in $(x, x^{(+)}, x^{(-)})$ nicht produziert werden, so wird er auch nicht konsumiert. Das folgt aus Kontraposition, die sich analog zu Obigem zeigen lässt: Wenn i konsumiert wird, dann wird i auch produziert.

Also wird in $(x, x^{(+)}, x^{(-)})$ ein Metabolit genau dann produziert, wenn er auch konsumiert wird und es gilt $M_i(x, x^{(+)}, x^{(-)})$. \square

Was ist ein Stoffwechselfad?

Wir haben nun gesehen, dass unser logisch hergeleiteter Regelsatz tatsächlich unsere gewünschten Halbverbandseigenschaften erfüllt. Schauen wir uns nun einmal genauer an, wie gut wir damit arbeiten können.

Eine Frage, die sich häufig stellen wird, ist, ob eine Menge an Reaktionen einen gültigen **Stoffwechselfad** widerspiegelt. Dafür müssen wir testen, ob es eine passende Belegung der Variablen $(x, x^{(+)}, x^{(-)})$ gibt, so dass wir die Formel C^{01} erfüllen. Für einen Kandidaten $\alpha \subseteq [n]$ gibt es theoretisch $2^{|\alpha \cap \text{Rev}|}$ Belegungen zum Testen. Also exponentiell viele.

Der Verzicht auf reversible Reaktionen birgt manchmal Vorteile. Um dieser kombinatorischen Explosion zu entkommen, kann man **reversible Reaktionen** aufsplitten, d. h. durch eine identische **irreversible Reaktion** und eine in Gegenrichtung verlaufende, irreversible Reaktion austauschen. Dieser Ansatz wird oft gewählt, um Berechnungen zu vereinfachen (z. B. im ersten Artikel zur Flusskopplung (Burgard, Nikolaev Evgeni V., Schilling & Maranas 2004)). Benutzt man diese **Aufspaltung**, so muss man seine Verfahren um ein Pre- und vor allem Postprocessing erweitern. Die Interpretation der Ergebnisse wird ansonsten schwerer.

Auch in unserer Modellierung C^{01} werden die reversiblen Reaktionen intern aufgesplittet ($x^{(\pm)}$). Aus Gründen der Vergleichbarkeit und vor allem der Weiternutzung der Ergebnisse wird jedoch direkt das Postprocessing durchgeführt (Vereinigung zu $x = x^{(+)} \vee x^{(-)}$) und die internen Informationen verworfen.

SAT-Solver sind heute sehr schnell. Nach diesem kleinen Ausflug in die Vor- und Nachteile der Aufspaltung zurück zu unserer eigentlichen Frage: Wie prüfen wir, ob eine Menge an Reaktionen einen gültigen (logischen) Stoffwechselfad darstellt? Für das bisher nur angedeutete, jedoch nicht vorgestellte Modell, das stationäre Zustände verlangt, werden wir später einen polynomiellen Algorithmus vorstellen. Wie schnell geht es für unser logisches Modell?

Wir fragen danach, ob ein bestimmter Kandidat gültig ist oder nicht – für gewöhnlich ein sehr einfaches Problem. Doch leider gibt es keinen offensichtlichen, trivialen Algorithmus. Die beste Antwort derzeit lautet, einen SAT-Solver einzusetzen. Das sind Programme, die versuchen, gültige Belegungen beliebiger logischer Formeln zu finden. Ihr Name stammt vom *satisfiability problem* SAT. Bei SAT handelt es sich um das vielleicht bekannteste NP-vollständige Problem (Papadimitriou 1994).

Möglicherweise schießen wir hierbei mit Kanonen auf Spatzen, eventuell geht es nicht einfacher, wenn wir die Informationen über die Orientierung der reversiblen Reaktionen nicht kennen; die Komplexität des Problems wurde bisher noch nicht genauer untersucht. Da wir mit einer sehr speziellen Untermenge an logischen Formeln arbeiten, ist das Problem möglicherweise auch in \mathcal{P} enthalten, also der Klasse an Problemen, die sich der Theorie nach effizient in polynomieller Laufzeit lösen lassen.

Die gute Nachricht ist, dass SAT-Solver in den vergangenen Jahren eine enorme Entwicklung durchgemacht haben. Auch motiviert durch diverse Wettbewerbe wurden viele sehr schnelle SAT-Solver entwickelt (z. B. Biere 2008, Een & Sörensson 2005), so dass Nutzer des logischen Modells von dieser jüngsten wissenschaftlichen Entwicklung stark profitieren können.

§2.2.3 Von Hyperpfaden und abgewandelten logischen Regeln

Wir haben unser logisches Modell mit einer logischen Erklärung hergeleitet: Von nichts kommt nichts und die Zelle will sich nicht selbst vergiften. Also sollen Metaboliten genau dann produziert werden, wenn sie auch verbraucht werden.

Wir haben auch schon darauf verwiesen, dass diese Regeln Streitbar sind. Normalerweise wird zusätzlich gefordert, dass sich das System in einem stationären Zustand befindet. Das betrachten wir gleich im nächsten Abschnitt. Einige wollen dem Modell ein noch engeres Korsett anlegen und fordern zudem, dass thermodynamische Bedingungen erfüllt werden sollen, auf die wir am Ende dieser Arbeit eingehen wollen. Zum Abschluss dieses Abschnitts wollen wir indes zeigen, dass auch weniger Regeln zu nutzbaren Verbänden führen, und damit gleich noch ein anderes Modell vorstellen.



Bemerkung 2.2 (logischer Notwendigkeitsverband).

Ersetze in Definition 2.4 die Bedingung $M_i = (M_i^{(+)} \Leftrightarrow M_i^{(-)})$ durch eine der beiden folgenden Bedingungen:

$$M_i^{01 \rightarrow} := (M_i^{(+)} \Rightarrow M_i^{(-)}),$$

$$M_i^{\rightarrow 01} := (M_i^{(+)} \Leftarrow M_i^{(-)}).$$

Dann ergeben sich zwei neue logische Modelle mit zugehörigen logischen Verbänden. Es ist jeweils $L^{01} \subseteq L^{01 \rightarrow}$ bzw. $L^{01} \subseteq L^{\rightarrow 01}$ Halbverband.

$L^{01 \rightarrow}$ entspricht der Hypothese, dass ein Metabolit, der aufgebaut wird, auf jeden Fall wieder abgebaut werden soll, damit sich die Zelle nicht daran vergiftet oder platzt. Die Zelle kann jedoch auch im gewissen Rahmen Reaktionen durchführen, die Metaboliten abbauen, welche

derzeit nicht neu hergestellt werden. Wie oben schon einmal angedeutet, verfügen Zellen meist über einen Puffer für Metabolitenkonzentrationen. Zuvor gespeicherte Metaboliten werden bei solchen Stoffwechselfaden wie denen von $L^{01\rightarrow}$ genutzt.


Bei $L^{\rightarrow 01}$ geht man hingegen davon aus, dass ein Stoff, der umgewandelt werden soll, zuvor produziert werden muss. Von nichts kommt nichts. Wenn genügend externe Nährstoffe vorhanden sind, können Metaboliten produziert und gespeichert werden, auch wenn sie nicht unmittelbar benötigt werden. Sie greifen wieder auf den Zellpuffer zurück.

Alle drei Verfahren setzen mehr oder weniger voraus, dass Zellen Metabolitenkonzentrationen puffern können. Sie ignorieren aber völlig, in welchem Rahmen die Zelle das tatsächlich kann! Das muss noch einmal so hart benannt werden.

Carbonell, Fichera, Pandit & Faulon (2012) stellen ein völlig anderes Konzept vor, um auf Elemente aus $L^{\rightarrow 01}$ zu kommen. Sie verwenden Hypergraphen und erstellen sich damit inkrementell Stoffwechselfade. Mehr noch, sie stellen einen polynomiellen Algorithmus vor, mit dem sie die Menge aller unblockierten Reaktionen in ihrem Modell bestimmen können. Dass werden wir später noch ganz nützlich finden, wenn wir Flusskopplung betrachten.

Ihr Ansatz geht von irreversiblen Reaktionen aus. Unser logisches Modell aus Definition 2.4 fügt wie diskutiert intern aufgesplittete Reaktionen wieder zusammen, um aus ihnen einen Verband zu erstellen. Hier nun wollen wir nachvollziehen, wie es Carbonell et al. (2012) mit einem geschickten Trick gelingt, reversible Reaktionen auch im aufgesplitteten Zustand nur auf sinnvolle Weise Pfade hinzuzufügen.

Betrachten wir hierfür eine leichte Variation ihrer Definition eines Hyperpfades, um danach zu prüfen, wie sie mit dem logischen Notwendigkeitsverband $L^{\rightarrow 01}$ zusammenhängen.



Definition 2.6 (Hyperpfad).
Seien (S, Irr) metabolische Bedingungen.
Betrachte nun den Hypergraphen $H = (V, A)$ mit $V = [m]$ und $A = [n] \cup \{-j \mid j \in \text{Rev}\}$.
Zudem gibt es zwei Funktionen $\alpha, \beta : A \rightarrow 2^V$, Start- und Zielfunktion, die wie folgt definiert sind:

$$\alpha : \quad a \mapsto \begin{cases} \{i \in [m] \mid S_{i,a} < 0\} & , a > 0, \\ \{i \in [m] \mid S_{i,-a} > 0\} & , a < 0, \end{cases}$$


$$\beta : \quad a \mapsto \begin{cases} \{i \in [m] \mid S_{i,a} > 0\} & , a > 0, \\ \{i \in [m] \mid S_{i,-a} < 0\} & , a < 0. \end{cases}$$

Offensichtlich gilt $\alpha(\pm j) = \beta(\mp j)$ für $j \in \text{Rev}$.
*Sei nun $p = (a_1, \dots, a_t) \subseteq A$ geordnete Teilmenge^a an Reaktionen. Wir nennen p **Hyperpfad**, wenn $|p| = 0$ oder $p_{t-1} = (a_1, \dots, a_{t-1})$ Hyperpfad und $\alpha(a_t) \subseteq \bigcup_{i=1}^{t-1} \beta(a_i)$.*
Es bezeichne \mathcal{H} die Menge aller Hyperpfade.

^aIm Allgemeinen anders geordnet als \mathbb{Z} !

Hyperpfad

$L^{\mathcal{H}}$


 **Definition 2.7 (Verband der Hyperpfade).**
 Seien (S, Irr) metabolische Bedingungen und \mathcal{H} die Menge aller Hyperpfade.
 Definiere $L^{\mathcal{H}}$ wie folgt:

$$L^{\mathcal{H}} = \{\{j \mid j \in p\} \mid p = (a_1, \dots, a_t) \in \mathcal{H}\}.$$

Dann nennen wir $L^{\mathcal{H}}$ den Verband der Hyperpfade, die durch (S, Irr) bestimmt sind.

Aus den Hyperpfaden lässt sich ein Verband herleiten, dessen Elemente aus $L^{\rightarrow 01}$ stammen. Dies zeigen wir in folgendem Satz:

$p \in L^{\rightarrow 01}$

 **Satz 2.1 (Hyperpfade folgen logischen Regeln).**
 Seien (S, Irr) metabolische Bedingungen und \mathcal{H} die Menge aller Hyperpfade.
 Dann ist $L^{\mathcal{H}}$ Unterhalbverband von $L^{\rightarrow 01}$. Im Allgemeinen gilt jedoch keine Gleichheit $L^{\mathcal{H}} \subseteq L^{\rightarrow 01}$.

Beweis. Wir werden zunächst zeigen, dass $L^{\mathcal{H}}$ wirklich Halbverband ist und danach, dass es Teilmenge von $L^{\rightarrow 01}$ ist.

Es ist $p = \emptyset \in \mathcal{H}$. Darum müssen wir jetzt zeigen, dass $L^{\mathcal{H}}$ vereinigungsstabil ist.

Seien $a, \tilde{a} \in L^{\mathcal{H}}$. Für $a = \emptyset$ oder $\tilde{a} = \emptyset$ ist nichts zu zeigen, also gehen wir davon aus, dass es zwei nicht-triviale Hyperpfade $p = (a_1, \dots, a_t), \tilde{p} = (\tilde{a}_1, \dots, \tilde{a}_{\tilde{t}}) \in \mathcal{H}$ gibt mit $t, \tilde{t} \geq 1$.

Definiere nun einen neuen Pfad $p^* = (a_1^*, \dots, a_{t^*}^*)$ mit $t^* = |a \cup \tilde{a}|$ und $a_i^* = a_i$ für $i \leq t$ und $a_i^* = \min_{\tilde{p}} (\tilde{p} \setminus \{p_{i-1}^*\})$ für $i > t$. Hierbei nutzen wir aus, dass der Hyperpfad \tilde{p} geordnet ist und jede nicht-leere endliche geordnete Menge ein eindeutiges Minimum besitzt.

Offensichtlich gilt $\{j \mid j \in p^*\} = a + \tilde{a}$. Daher müssen wir uns nur noch vergewissern, dass p^* wirklich Hyperpfad ist. Der Beweis erfolgt per Induktion über die Teilpfade $p_k^* = (a_1^*, \dots, a_k^*)$:

Sei $k \leq t \leq 1$. Dann ist $p_k^* = p_k$ Hyperpfad, da $p \in \mathcal{H}$ Hyperpfad nach Voraussetzung.

Sei nun also p_k^* Hyperpfad und $t < k + 1 \leq t^*$. Wir werden jetzt zeigen, dass dann auch p_{k+1}^* Hyperpfad ist. Nach Definition der Hyperpfade verbleibt zu zeigen, dass $\alpha(a_{k+1}^*) \subseteq \bigcup_{i=1}^k \beta(a_i^*)$.

Nach Konstruktion von p^* ist aber (wegen $k > t$) $a_{k+1}^* = \tilde{a}_l$ für ein $l \in [t]$. Da \tilde{p} Hyperpfad, gilt $\alpha(a_{k+1}^*) = \alpha(\tilde{a}_l) \subseteq \bigcup_{i=1}^{l-1} \beta(\tilde{a}_i)$. Nach Konstruktion von p^* ist jedoch $(\tilde{a}_1, \dots, \tilde{a}_{l-1}) = \tilde{p}_{l-1} \subseteq p_k^*$.

Damit ist p^* Hyperpfad und somit $a + \tilde{a} = a \cup \tilde{a} = p^* \in L^{\mathcal{H}}$. Also ist $L^{\mathcal{H}}$ wirklich Halbverband.

Werfen wir jetzt einen kurzen Blick auf $L^{\rightarrow 01}$. Wenn wir uns Definition 2.4 in Erinnerung rufen, dann sehen wir, dass die logischen Klauseln $M_i^{(+)}$ sowie $M_i^{(-)}$ eng mit den Hyperpfad-Funktionen β bzw. α zusammenhängen. Die Konstruktion von \mathcal{H} sorgt dafür, dass Hyperpfade $\mathbf{p} = (\mathbf{a}_1, \dots, \mathbf{a}_t)$ insbesondere $\bigcup_{i=1}^t \alpha(\mathbf{a}_i) \subseteq \bigcup_{i=1}^t \beta(\mathbf{a}_i)$ erfüllen, was den Klauseln $M_i^{\rightarrow 01}$ entspricht.

Kümmern wir uns daher jetzt um die Klauseln R_j bezüglich den reversiblen Reaktionen. Diese stellen sicher, dass im aufgesplitteten Zustand nur eine der beiden Richtungen einer reversiblen Reaktion verwendet wird.

Dies lässt sich nicht direkt auf Hyperpfade übertragen, da es hier durchaus möglich ist, dass $\pm j \in \mathbf{p} \in \mathcal{H}$ für $j \in \text{Rev}$ vorkommt.

Aber jetzt kommt das Aber. Hyperpfade sind geordnet. Sei jetzt also $\mathbf{p} = (\mathbf{a}_1, \dots, \mathbf{a}_t)$ Hyperpfad mit $\mathbf{a}_l = -\mathbf{a}_k$ und $k < l$. Dann ist auch $\tilde{\mathbf{p}} = \mathbf{p} \setminus \{\mathbf{a}_l\}$ Hyperpfad:

Betrachte hierfür wieder die Start- und Zielmetabolitenfunktionen $\alpha, \beta : A \rightarrow 2^V$. Wegen $\mathbf{a}_l = -\mathbf{a}_k$ wissen wir $\beta(\mathbf{a}_l) = \alpha(\mathbf{a}_k)$.

Ist \mathbf{p} Hyperpfad, dann ist auch \mathbf{p}_k Hyperpfad, also gilt $\beta(\mathbf{a}_l) = \alpha(\mathbf{a}_k) \subseteq \bigcup_{i=1}^k \beta(\mathbf{a}_i)$. Damit kann $\mathbf{a}_l \in \mathbf{p}$ keine neuen Metaboliten exklusiv produzieren, die von anderen folgenden Reaktionen benötigt werden. Es gilt stets $\bigcup_{i=1}^t \beta(\mathbf{a}_i) = \bigcup_{\substack{i=1 \\ i \neq l}}^t \beta(\mathbf{a}_i)$. Damit gelingt der rekursive

Gültigkeitsnachweis und $\tilde{\mathbf{p}}$ ist ein Hyperpfad, der genauso wie \mathbf{p} zu $\mathbf{a} \in L^{\mathcal{J}c}$ korrespondiert, aber nur eine Reaktionsrichtung von $j \in \text{Rev}$ benötigt. Iterativ lässt sich hiermit ein Hyperpfad $\mathbf{p}^* \in \mathcal{H}$ konstruieren, der immer noch zu \mathbf{a} korrespondiert, aber auch die Klauseln R_j erfüllt.

Da \mathbf{p}^* immer noch die Klauseln $M_i^{\rightarrow 01}$ erfüllt, ist $\mathbf{a} \in L^{\rightarrow 01}$, also wirklich $L^{\mathcal{J}c} \subseteq L^{\rightarrow 01}$.

Der in Beispiel 2.5 gezeigte Stoffwechselfad ist zwar Element des logischen Verbandes L^{01} , jedoch nicht im Verband der Hyperpfade, da sich die Reaktionen `photo` und `bc1` nicht in eine eindeutige Reihenfolge sortieren lassen. \square

Dass Hyperpfade eng mit logischen Regeln verbunden sind, haben Soh & Inoue (2012) erkannt. In ihrer Arbeit übersetzen sie das Prinzip der (geordneten) Hyperpfade in logische Klauseln. Eine ihrer Arbeiten war auch Inspiration für unser logisches Modell: Denn Soh & Inoue (2012) stellte ein rekursives Verfahren vor, um minimale gültige Belegungen logischer Formeln zu finden.

Beide Arbeiten (Carbonell et al. 2012, Soh & Inoue 2012) reden noch ganz allgemein von Start- und Zielmetaboliten bzw. Quell-/Zielmengen der Stoffwechselfade. Davon sind wir in unserer Hyperpfaddefinition insofern abgewichen, dass unsere Startmetaboliten zwangsläufig solche

sein müssen, die von *Transportreaktionen* in die Zelle befördert werden. Was es genau mit diesen Transportreaktionen auf sich hat, werden wir viel später noch einmal erläutern. Auch ob die Zielmetaboliten heraustransportiert (oder weiterverarbeitet) werden müssen, kann diskutiert werden.

Wir haben uns bei unserer Interpretation dagegen entschieden, dass die Gültigkeit eines Pfades von einer Zielmenge abhängt. Deshalb können wir die Eigenschaft $p \in \mathcal{H} \Rightarrow p_k \in \mathcal{H}$ nutzen. Bei Wunsch kann man jedoch einen Begriff wie *abgeschlossener Hyperpfad* einführen, bei dem die Menge aller produzierten Metaboliten gleich der Menge derer ist, die abgebaut werden. Diese Menge ist auch wieder Verband, was wir hier jedoch nicht mehr untersuchen werden.

Gehen wir nun zum Klassiker über: Dem Standardmodell, das Konzentrationen konstant lässt, was im Englischen *steady state* genannt wird.

§2.3 Klassisch stationär

§2.3.1 Ursprung und Definition

In der Biologie stellen sich Fragen ...

Die Zelle kann auf veränderte Umweltbedingungen sehr schnell reagieren, wie Partridge, Sanguinetti, Dibden, Roberts, Poole & Green (2007) am Beispiel *E. coli* erläutern. Sie wechselt von einem Stoffwechselzustand zu einem neuen, der nun von größerem Vorteil ist. Weil das so schnell geht und sich die Zelle anschließend über längere Zeit stabil im eingenommenen Zustand befindet, haben diskrete Modelle in der Vergangenheit vor allem stationären Zustände umfasst, die eine Obermenge der stabilen Zustände darstellen.

Dabei haben sich unterschiedliche Fragen ergeben:

- Welche Zustände kann eine Zelle annehmen? Was heißt *stabil*?
- Gibt es Zustände, die für die Zelle besonders vorteilhaft sind?
- Was sind die einfachsten Zustände, die eine Zelle einnehmen kann? Gibt es viele davon? Was bewirken sie? Und gibt es Redundanzen, so dass auch eine gestörte Zelle überleben kann?
- Was passiert bei konkreten Störungen? Wie hängen verschiedene Reaktionen miteinander zusammen?

... und die Mathematik findet Antworten

Das alles sind Fragen, die sich aus biologischer Sichtweise ergeben. Für alle von ihnen wurden mathematische Verfahren entwickelt, mit denen sie mal mehr, mal weniger gut beantwortet werden können.

Die meisten dieser Verfahren werden wir gleich noch kurz vorstellen und damit einen wichtigen Teil zum Verständnis dieser Arbeit liefern.

Wir haben bisher schon gesehen, dass sich der Stoffwechsel durch einen **Verband** darstellen lässt. In den kommenden Kapiteln werden wir sehen, dass die meisten der hier gestellten Fragen in Wirklichkeit Fragen nach Verbandseigenschaften sind. Wir werden einige der klassischen Verfahren nur leicht ändern müssen, andere werden wir stark optimieren können. Am Ende werden wir ein ansehliches Repertoire an Untersuchungsmethoden zur Verfügung haben, mit dem sich Stoffwechselverbände untersuchen lassen. Sei es basierend auf klassischen stationären Modellen oder auf anderen Regeln – die meisten unserer Verfahren werden universell, eben weil sie nicht auf einen speziellen Verband optimiert sind.

Eines wird jedoch unanzweifelbar sein: Die Verbandstheorie verbindet all die verschiedenen Verfahren, die bisher isoliert voneinander entwickelt wurden, und führt uns zu neuen interessanten Anwendungen.

Ein diskretes Modell basierend auf Differenzialgleichungen

Wir haben aus Abbildung 2.2 bereits eine stöchiometrische Matrix, metabolische Bedingungen für `etc_purple` abgeleitet. Doch wozu benötigen wir so präzise Zahlen, wenn doch auch das logische Modell vernünftig begründbar ist und mit weniger Informationen auskommt?

Weil wir mit Hilfe der stöchiometrischen Koeffizienten physikalische Gesetze ins Spiel bringen können. So können wir zeitliche Vorhersagen wagen. Dafür betrachten wir den Metaboliten `Hp` in `etc_purple` einmal genauer:

Er ist Bestandteil von fünf Reaktionen, genauer Produkt von drei irreversiblen und beteiligt an zwei reversiblen. Folgen wir den Hauptreaktionsrichtungen der metabolischen Bedingungen, so können wir folgende Differenzialgleichung aufstellen:

$$\frac{d}{dt} \text{Hp} = 4v^{(3)} + 4v^{(4)} + 2v^{(8)} + 2v^{(11)} - 3v^{(2)}.$$

Die zeitliche Änderung der Gesamtkonzentration von `Hp` entspricht der Summe der Änderungen induziert durch die Einzelreaktionen. So einfach kann's gehen.¹


Physikalische Systeme sind nach Gleichgewichten bemüht, in unserem Fall einem stationären

¹Bei weitergehendem Interesse sei auf Klipp et al. (2009) verwiesen, woher auch diese Herleitung stammt.

Zustand der Metabolitenkonzentrationen. Gleichgewicht heißt keine Konzentrationsänderung für irgendeinen Metaboliten. Im stationären Zustand gilt $\frac{d}{dt}Hp = 0$ genauso wie $\frac{d}{dt}O_2 = 0$.

Die Matrixschreibweise der stöchiometrischen Informationen erlaubt uns dafür folgende elegante Definition:

Flusskegel



Definition 2.8 (Flusskegel).
 Seien (S, Irr) metabolische Bedingungen.
 Sei nun $F = F(S, Irr)$ wie folgt definiert:


$$F = \{v \in \mathbb{R}^n \mid Sv = 0, v_i \geq 0 \forall i \in Irr\}.$$

Dann heißt F Flusskegel zu den metabolischen Bedingungen (S, Irr) . Die Elemente $v \in F$ heißen Flussvektoren.

It's done. Der Schritt zum Flusskegel ist getan. Die Stoffwechselprozesse, die sich so elegant als Graph darstellen lassen (oder wohl eher als eine Art gewichteter, gerichteter Hypergraph...), sind zum Polyeder geworden. Die Mathematiker können ihren Zauber wirken und Verfahren aus der kombinatorischen Optimierung einsetzen, um den Stoffwechsel zu untersuchen. Ein Vorgehen, auf das man nicht gleich kommen muss.

Wir werden sehen, dass die bisherigen Verfahren diese Möglichkeit auch fleißig nutzen. So gut wie alles wird mit Polyedertheorie berechnet, wie es auch Schilling, Letscher & Palsson (2000) ausführlich motivieren. Doch interessanterweise berufen sich die meisten Definitionen wiederum auf die Träger der Flussvektoren. Das ist unser Dreh- und Angelpunkt, der uns zu Verbänden befördert. Wir stellen gleich die klassischen Verfahren vor. Zuvor wollen wir aber noch dem Flussverband seinen angemessenen Platz gleich neben dem Flusskegel widmen:

Flussverband



Definition 2.9 (Flussverband).
 Sei F Flusskegel zu den metabolischen Bedingungen (S, Irr) .
 Sei außerdem $\text{supp} : F \rightarrow 2^{[n]}$ Trägerfunktion, also:

$$\text{supp} : v \mapsto \{i \in [n] \mid v_i \neq 0\}.$$

Definiere nun $L^F \subseteq 2^{[n]}$ wie folgt:

$$L^F := \text{supp}(F) = \{\text{supp}(v) \mid v \in F\}.$$

Dann heißt L^F Flussverband zu den metabolischen Bedingungen (S, Irr) . Die Elemente $\alpha \in L^F$ heißen Flussmuster.

Der Begriff der Flussmuster wurde von Kaleta, Figueiredo & Schuster (2009) geprägt. Kaleta et al. untersuchen in ihrer Arbeit Teilnetzwerke von Zellen und definieren dort *elementare Flussmuster*. Damit meinen sie die Indizes der Reaktionen von Flussvektoren eingeschränkt auf ihr Teilnetzwerk, die sich nicht als Summe kleinerer Elemente schreiben lassen (s. Definition 3.7, unzerlegbares Element).

Im Unterschied zu uns verlangen sie, dass alle Reaktionen irreversibel sind (und dafür notfalls aufgesplittet werden). Dies müssen wir nicht fordern, wie wir später zeigen werden.

Bemerkung 2.3.

L^F ist wirklich ein Verband. Das werden wir aber erst später als Korollar 4.5 zeigen. Bis dahin wollen wir uns zu Nutze machen, dass $\text{supp} : F \rightarrow L^F$ surjektiv ist.

Beispiel 2.6 (Flussverband, logischer Verband).

In Abbildung 2.4 ist ein weiteres Netzwerk, `adenin_compressed`, abgebildet. Daneben ist das Hasse-Diagramm des zugehörigen logischen Verbandes zu sehen. Hervorgehoben sind zudem die Elemente des Flussverbandes.

Verbände

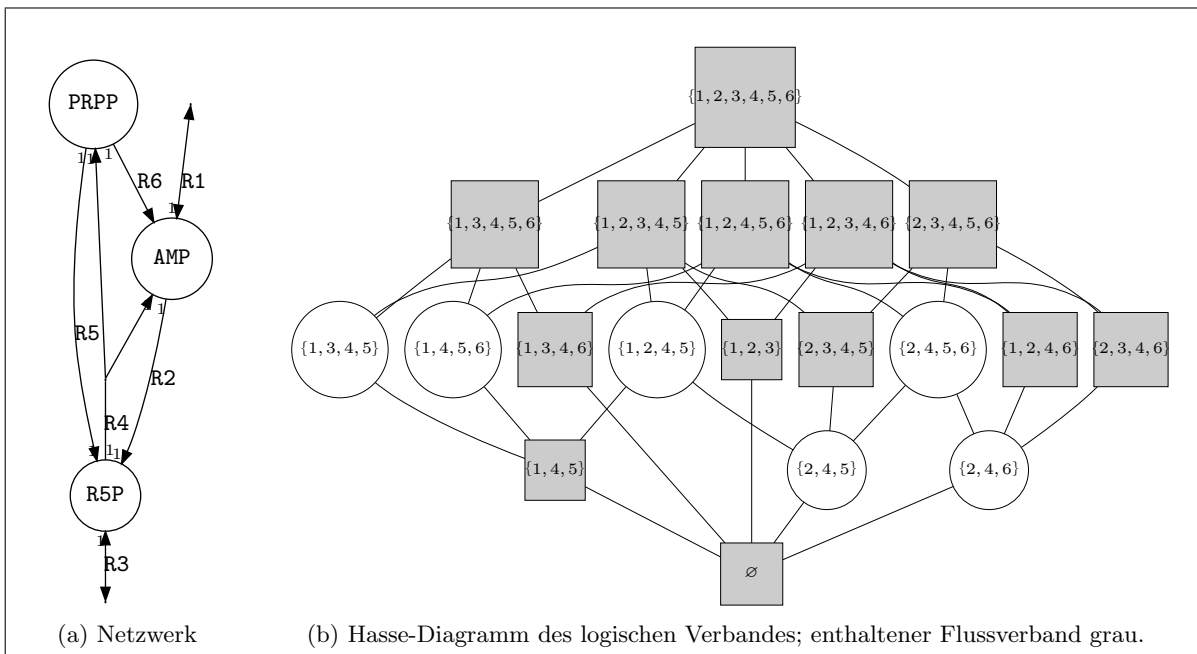


Abbildung 2.4.: Beispielnetzwerk `adenin_compressed`.

Diese Arbeit hat Verbände schon im Titel. Daher wird es niemanden überraschen, dass wir ständig über sie reden werden. Sie bieten viele Vorteile, Modelle einheitlich zu entwerfen – seien es logische Modelle, stationäre Zustände oder doch etwas ganz anderes. Sie sind ein tolles Framework, in dem man leicht arbeiten kann.

Aber es gibt auch ganz andere Herangehensweisen. Damit sind nicht einmal kontinuierliche (Differenzialgleichungs-)Modelle gemeint, sondern durchaus auch andere diskrete. Petri-Netze beispielsweise. Auch wenn wir sie in dieser Arbeit nicht diskutieren werden, wollen wir sie zumindest erwähnen. Zum Einstieg sei beispielsweise auf Voss, Heiner & Koch (2003) verwiesen. Ein weiterer interessanter Artikel über ihren Einsatz, bei dem auch genregulatorische Effekte einbezogen werden, stammt von Grunwald, Speer, Ackermann & Koch (2008).

Doch kommen wir jetzt zu den Anwendungen des Flusskegel-Ansatzes.

§2.3.2 Anwendungen

Flux Balance Analysis – oder: Was bringt am meisten?

Was ist der Sinn des Lebens?

Diese Frage werden wir heute nicht beantworten. Aber wir werden darüber sprechen, was Zellen versuchen. Möglicherweise ist das der Sinn des Lebens. Doch wie gesagt: Diese Frage werden wir heute nicht beantworten.

Leben versucht zu überleben. Alle Lebensprozesse drehen sich um eine einzige Sache: Das eigene Genom zu vervielfältigen und so viele Nachfolger wie möglich zu produzieren. Sei es durch Zellteilung, eingeschlechtliche oder zweigeschlechtliche Fortpflanzung. Der Stoffwechsel dient der Vermehrung. Alle Spezialisierungen, die die unterschiedlichen Zelltypen angenommen haben, dienen einzig allein dazu, in einer Nische zu überleben, in der sie die für sich persönlich optimalen Bedingungen zur Vermehrung finden.

Die Evolution hat dabei quasi unzählbar viele unterschiedliche Zelltypen hervorgebracht. So ähnlich sind die Zellen – man denke daran, dass das Genom aller Zellen stets nur aus den gleichen vier Buchstaben, die Proteine aus den gleichen 22 Aminosäuren bestehen –, so unterschiedlich sind die Aufgaben, die der Stoffwechsel im Speziellen erfüllen muss.

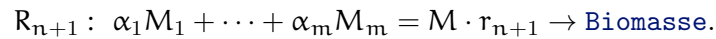
Nicht nur auf genetischer Ebene ähneln sich die meisten Zellen hochgradig. Viele Stoffwechselprozesse – wie der Zitronensäurezyklus – finden bei sehr vielen Zellen statt und laufen dort immer gleich ab. Viele Reaktionen gehören also zu (fast) jedem Stoffwechsel.

Doch hat die Evolution all die unterschiedlichen Zellen für ihren jeweiligen Lebensraum optimiert. Ein Einzeller in der Nähe eines schwarzen Rauchers in der Tiefsee ist mit anderen Umweltbedingungen konfrontiert als ein Mammutbaum, ein Fliegenpilz, ein Darmbakterium oder ein Nacktmull. Sie alle haben unterschiedliche Nährstoffressourcen, verschiedene Energiequellen, unterschiedlichen Bewegungsfreiraum. Einige bestehen aus nur einer Zelle, die allein für ihr Überleben und Vermehren verantwortlich ist, andere aus einer komplexen Struktur unzähliger, zu bestimmten Aufgaben optimierter Zellen, die sich in einem Individuum teilen und insgesamt versuchen, „Kinder“ des Lebewesens zu schaffen.

Ob Ein- oder Mehrzeller, Zellen sind hochgradig spezialisiert. Der Stoffwechsel dient der Vermehrung. Wie soll man einen so komplexen Prozess nur im Computer nachzubilden versuchen?

Nun, ein naiver Ansatz, eine einzelne Zelle zu simulieren, ist es, die Möglichkeiten des Stoffwechsels (in Form der Reaktionen) in das Modell zu packen, dieses anschließend bestimmten Umweltbedingungen auszusetzen (folgende Nährstoffe stehen zur Verfügung – nur sie können in die Zelle transportiert werden) und sie **Biomasse** zum Wachsen und Teilen produzieren zu lassen. Biomasse als Produkt einer künstlich erdachten Reaktion, die Metaboliten zu einem

Biomassemetabolit umwandeln.



Wir fügen diese Reaktion dem Stoffwechsel hinzu und suchen dann einen stationären Zustand, der diese Reaktion mit Maximalgeschwindigkeit ablaufen lässt.

Algorithm $FBA(S, Irr, lower_bound, upper_bound)$

1.

$$\begin{aligned} & \text{maximize} && v_{n+1} \\ & \text{s.t.} && (Sr) \cdot v = 0 \\ & && v_i \geq 0 && \forall i \in Irr \\ & && lower_bound \leq v_i \leq upper_bound && \forall i \in [n] \end{aligned}$$

So naiv die Herleitung des Verfahrens ist, so erstaunlich präzise Vorhersagen hat es in der Vergangenheit gemacht. In der Wissenschaft ist es als **Flux Balance Analysis**, kurz **FBA**, bekannt. Nach einiger Vorarbeit schlug Watson (1984) schließlich den Einsatz eines linearen Programms vor – dem Ansatz, der seitdem etliche Male genutzt wurde. Einen schönen Übersichtartikel liefern auch Orth, Thiele & Palsson (2010). Mögliche Anwendungen hat zudem Gianchandani, Chavali & Papin (2010) zusammengestellt.

In den letzten Jahren arbeiten viele Gruppen daran, das Verfahren zu verbessern. Essenziell für präzise Vorhersagen ist die Güte der Biomasse-Produktionsreaktion. Doch gibt es nur *so* viel, wie man mit Rumschrauben an einer Gleichung tun kann, ohne sie bloß an alte Messergebnisse anzupassen und sie zu überspezialisieren.

Wir haben schon mehrfach angesprochen, dass es eine sogenannte Zellregulation gibt. Regulation bedeutet, dass sich die Zelle veränderten (Umwelt-)Bedingungen anpassen kann. Sie produziert mehr oder weniger von bestimmten Enzymen oder schaltet sie zwischenzeitlich ab. Damit kann sie ganz gezielt den Stoffwechsel beeinflussen.

Es gibt eine Unterdisziplin der Systembiologie, die sich speziell mit diesen regulatorischen Prozessen beschäftigt. Diese Arbeit hingegen beschäftigt sich ausschließlich mit Stoffwechselprozessen und stellt keine direkten Verfahren vor, um regulatorische Prozesse einzubinden.

Auch die FBA werden wir nicht noch einmal diskutieren. Aber sie verdient Erwähnung in jeder Arbeit über metabolische Modellierung, weil sie das Verfahren ist, das bisher die größte Zahl an Anwendern gefunden hat.

Ebenso wie die klassische FBA wollen wir erwähnen, dass in den vergangenen Jahren mehrere Verfahren publiziert wurden, die regulatorische Prozesse in die Simulation integrieren. Exemplarisch seien hierfür rFBA (Covert 2002) und SR-FBA (Shlomi, Eisenberg, Sharan & Rupp 2007) genannt, die dem interessierten Leser als Einstieg in die Materie dienen können.

Elementare Flussmoden – oder: Das KISS-Prinzip

Keep it simple, stupid! Eine Devise, auf die man überall im Leben als gut gemeinter Rat stößt. Sei es beim Layout einer Präsentation, bei der zu viele unterschiedliche Designelemente nur verwirren, der Menge an (CG-)Animationen in Disney-Filmen (Wikipedia 2012b) oder in Form von *Ockhams Rasierklinge* in den Naturwissenschaften. Schneide alles weg, was zu kompliziert ist. Mache die Dinge so einfach wie möglich – aber nicht einfacher. (Albert Einstein, Wikipedia 2012d)

Eine Devise, auf die man überall stößt und an die sich auch die Natur oftmals hält. Unnötige Prozesse verschwenden Energie, was einen Nachteil im Kampf ums Überleben darstellen kann. Die Evolution bevorzugt Lebensformen, die so effizient und entsprechend meist auch so einfach wie möglich arbeiten.

Man kombiniere Reaktionen miteinander, die einen Stoffwechselfad bilden. Aber jede davon ist fundamental wichtig, damit es einen gültigen Pfad geben kann. Mache es noch einfacher, verzichte auf eine Reaktion, und die Funktion bricht zusammen. Was man in Händen hält, nennt die Literatur **elementarer Flussmodus (EFM)**. Die Träger dieser **Elementarmoden** sind die **minimalen Elemente** des Verbandes:

$$e = \text{supp } v \in \mathcal{M}(L) \Leftrightarrow \forall a \in L : (a \subsetneq e \Rightarrow a = \emptyset).$$

minimale
Elemente



Beispiel 2.7 ($\mathcal{M}(L)$).

Am Hasse-Diagramm 2.4 erkennt man:

logischer Verband $\mathcal{M}(L^{01}) = \{\{1, 4, 5\}, \{1, 3, 4, 6\}, \{1, 2, 3\}, \{2, 4, 5\}, \{2, 4, 6\}\}$,

Flussverband $\mathcal{M}(L^F) = \{\{1, 4, 5\}, \{1, 3, 4, 6\}, \{1, 2, 3\}, \{2, 3, 4, 5\}, \{1, 2, 4, 6\}, \{2, 3, 4, 6\}\}$.

Schuster (1994) führt das Konzept der EFM ein. Sie dienen seitdem als beinahe universelles Hilfsmittel zur Stoffwechselanalyse und werden für so unterschiedliche Fragen wie Effizienzvergleich verschiedener metabolischer Pfade (Carlson 2007) bis zur Vorhersage von Veränderungen im Stoffwechsel durch Mutationen (z. B. Zhao & Kurata 2009) genutzt.

Das klappt deshalb so gut, weil diese minimalen Stoffwechselfade – die wirklich die von der Evolution bevorzugten Energiesparer sind (Figueiredo, Podhorski, Rubio, Kaleta, Beasley, Schuster & Planes 2009) – im stationären Fall den gesamten Flusskegel aufspannen, wie wir später sehen werden.

Das Hauptproblem an den EFM ist ihre schiere Masse. Ihre Anzahl wächst rasant mit der Komplexität des Netzwerkes, obere Zählschranken sind bisher allesamt exponentiell zur Anzahl an Reaktionen (Klamt & Stelling 2002). Für viele Anwendungen müsste man alle EFM kennen. Das scheitert bei genomweiten Netzwerken an ihrer Berechnung – von der anschließenden Auswertung ganz zu schweigen.

Die gängigen Verfahren zur Berechnung der EFM im stationären Fall sind Abwandlungen der

Double Description Methode (Schuster, Hilgetag, Woods & Fell 2002). Sie versuchen in einem Durchgang alle Elementarmoden zu berechnen. Dafür wurden im Laufe der Jahre immer raffiniertere Verbesserungen des Algorithmus eingeführt, wie die sogenannten *Bit Pattern Trees* durch Terzer & Stelling (2008) als Fortführung des binären Ansatzes (Gagneur & Klamt 2004, Klamt, Gagneur & Kamp 2005). Terzer & Stelling stellen mit den *efmtools* ein sehr schnelles Programm zur Verfügung, mit dem sich die Elementarmoden berechnen lassen.

Wenn wir uns später die Elementarmoden noch einmal genauer anschauen, werden wir ein anderes Verfahren kennenlernen, das die Elementarmoden nacheinander berechnet. Das von Figueiredo et al. (2009) vorgestellte Verfahren ist zwar noch einmal viel langsamer, wenn man alle Elementarmoden berechnen will, bietet jedoch viel Spielraum, wenn man an Umweltbedingungen geknüpfte Systeme untersuchen will. Vor allem liefert es einem eine beständig wachsende Menge an Zwischenergebnissen, mit denen man bereits arbeiten kann.

Flux Coupling Analysis – oder: Der Dominoeffekt von Störungen

Jeder hat es schon einmal gespielt: Domino-Steine auf ihre kleine Seite stellen, dicht beieinander. Freude, wenn sie am Ende nacheinander wie gewünscht umfallen, nachdem man nur einen angetippt hat. Frust, wenn das unbeabsichtigt zu früh passiert. RTL machte daraus von 1998 bis 2009 ein großes Fernsehevent für ein Millionenpublikum.

Mathematik-Studenten werden in ihrer ersten Woche an der Universität in die Nützlichkeit des Dominoeffekts eingeweiht. Bei der vollständigen Induktion zeigt man, dass ein Stein wirklich seinen Nachfolger umwerfen wird (Induktionsschritt oder Induktionsschluss) und stößt den ersten Stein um (Induktionsanfang). Der Mathematiker muss gar nicht erst Steinen beim Umfallen zugucken. Sie haben die Folgen verstanden – felsenfest, so wie es Mathematiker immer wollen – und wissen, dass ihr Satz bewiesen ist.

In der medizinischen Forschung versucht man, die Entstehung von Krankheiten zu verstehen und Therapien zu entwickeln, die dem entgegenwirken. Gibt es genetische Defekte als Ursache, die eine bestimmte Krankheit als Folge haben? Was für Folgen auf den Körper hat der Einsatz eines bestimmten Medikaments? Was für Folgen hat es für unsere Zellkultur, wenn wir sie Extremsituationen aussetzen?

Eine der einfachsten Fragen, die man beim Stoffwechsel stellen kann, ist die nach den Folgen von Nährstoffmangel. Was passiert, wenn ein bestimmter Stoff nicht zur Verfügung steht? Kann die Zelle auch ohne Sauerstoff überleben?

Man entfernt eine Transportreaktion aus unserem Modell und guckt, was passiert. Und wenn man schon dabei ist, kann man auch gleich gucken, was passiert, wenn man eine beliebige Reaktion entfernt. Damit kann man einen genetischen Defekt simulieren.

Und: Tritt ein Dominoeffekt ein? Hat die Reaktion eventuell gar keine Funktion, war nur zur Redundanz verfügbar? Oder ist sie fundamental wichtig, weil andere Reaktionen an sie gekoppelt sind? Ist sie *too big to fail*?


Bei der **Flux Coupling Analysis (FCA)** (Burgard et al. 2004) stellt man sich diese Frage: Welche Reaktionen, fallen weg, wenn eine bestimmte blockiert ist oder fehlt. Wie dicht ist dieses Abhängigkeitsnetz unter den Reaktionen?

Im Verband suchen wir nach den Abhängigkeiten i, j mit:

$$i \xrightarrow{=0} j \Leftrightarrow \forall a \in L : (i \notin a \Rightarrow j \notin a).$$

Dann ist j an i gekoppelt und wir reden von **Flusskopplung**.

FCA

 **Beispiel 2.8 (FCA).**
 Im Netzwerk *adenin_compressed* gibt es lediglich die Flusskopplungen $4 \xrightarrow{=0} 5$ und $4 \xrightarrow{=0} 6$.
 Die Kopplungen im Beispiel *etc_purple* sind der Abbildung 2.5 zu entnehmen.

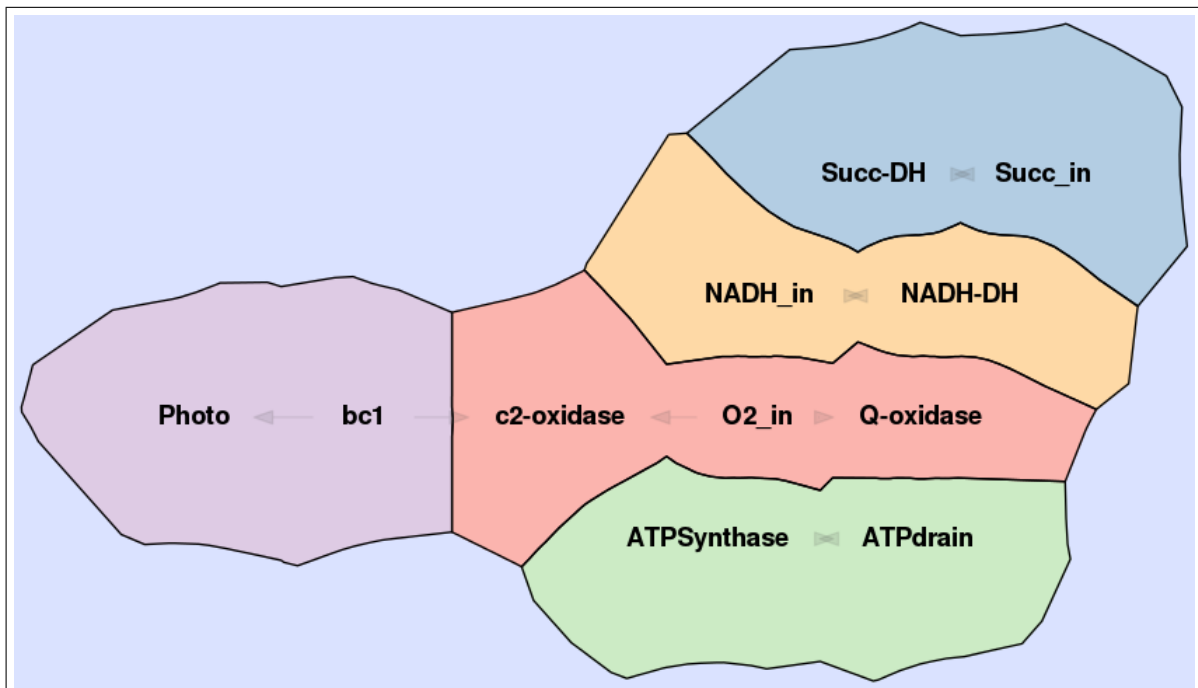


Abbildung 2.5.: Flusskopplung im System *etc_purple*.

Der Flusskopplung sieht man der Definition auf den ersten Blick an, dass Reaktionsgeschwindigkeiten hier nicht relevant sind. Wir fragen *Welche Reaktionen sind aneinander gekoppelt?* und nicht *Wie wird die Reaktionsgeschwindigkeit durch andere Reaktionen beeinflusst?*. Diesen Fragen gehen beispielsweise Edwards & Palsson (2000) nach, die eine Flussbalanceanalyse nach Löschung diverser Gene durchführen und interpretieren.

Doch genug der einleitenden Worte. Gehen wir nun ans mathematische Werk!



Kapitel 2 – Kurz und knapp

Verband

- ▷ Verbandstheorie *vereinheitlicht* Stoffwechselmodellierung.
- ▷ Operation + entspricht *Kombination* von Stoffwechselfaden.

logische Modelle

- ▷ Stationärer Zustand der Metabolitkonzentrationen wird *nicht* ausdrücklich gefordert.
- ▷ Modelle können auf logischen *Funktionen* oder Pfaden in *Hypergraphen* basieren.

Anwendungen

- ▷ FBA sucht stationäre Zustände mit maximaler *Wachstumsrate*.
- ▷ Elementarmoden sind *minimale* Stoffwechselfade; sie *erzeugen* den Flusskegel und erlauben so seine Untersuchung.
- ▷ FCA bestimmt die Folgen vereinzelter *Reaktionsausfalls*.



KAPITEL 3

Algebraische Grundlagen

Kapitelübersicht

3.1	Endliche (Halb-)Verbände und Ordnung	30
3.1.1	Jeder endliche Halbverband mit 0 ist ein Verband	30
3.1.2	Ein Bisschen mehr Ordnung, bitte!	33
3.2	Minimale und unzerlegbare Elemente	34
3.2.1	Definitionen...	34
3.2.2	... und Eigenschaften	37
3.3	Unterverbände	39
3.4	Homomorphismen	49

Wir haben gerade wichtige Untersuchungsmethoden für den Stoffwechsel kennengelernt und gesehen, dass sich viele Fragen auf Verbandsebene diskutieren lassen. Doch, was ist ein Verband? Was für Aussagen können wir mit dieser algebraischen Struktur treffen?

In diesem und dem nächsten Kapitel entfernen wir uns von der Biologie und machen das, was Mathematiker so lieben: Definition. Lemma. Satz. Beweis. Folgerung. Und von vorne!

Das ist trocken, eigenartig, lebensfremd. Mathematiker lieben das, weil es effizient ist und keinen Raum für Diskussionen lässt. Es geht gerade nicht darum, die Vor- und Nachteile diverser Modellierungsansätze abzuwägen. Es geht darum, die Vor- und Nachteile auszuarbeiten. Manchmal besonders akribisch: Gibt es ein unwahrscheinliches, realitätsfernes Szenario, in dem das Modell garantiert nicht funktioniert? Welche Aussagen können wir **IMMER** treffen? Welche Eigenschaften sind mathematisch interessant? Wenn nebenbei etwas herauskommt, dass man biologisch verwerten kann, dann ist das schön. Wenn nicht, ist auch gut. Mathematisch stimmt's ja.

Diese Ferne zum ursprünglichen Problem ist ein zweiseitiges Schwert. Als Mathematiker kann man wochenlang an einem Sonderfall arbeiten, auf den ein Anwender nie stößt. Der aber im Beweis stört. Andererseits kann man nebenbei auf Trivialitäten stoßen, die zu hocheffizienten Verfahren für Anwender führen, wie in dieser Arbeit bei der Flusskopplung passiert.


Mathematik ist die Sprache, mit der man alles vereinen kann. Gemeinsamkeiten ausmachen und eine strahlende Kathedrale aus ihnen errichten. Kümmern wir uns jetzt um das Fundament, damit unser Bauwerk später nicht einstürzt.

§3.1 Endliche (Halb-)Verbände und Ordnung

§3.1.1 Jeder endliche Halbverband mit 0 ist ein Verband

Wir haben bereits die Halbverbandsdefinition kennengelernt und auch schon für den Stoffwechsel nachgewiesen, dass die logischen Belegungen zu einem Halbverband führen. Doch ständig haben wir von Verbänden gesprochen, ohne genauer zu sagen, worum es sich dabei genau handelt. Hier sei nun die Definition nachgereicht:

Verband




Definition 3.1 (Verband).
Seien $(L, +, 0)$ und $(L, \circ, 1)$ Halbverbände über L mit neutralen Elementen 0 bzw. 1 . Es gelten zudem die beiden Absorptionsgesetze:

$$\forall a_1, a_2 \in L : a_1 + (a_1 \circ a_2) = a_1 \circ (a_1 + a_2) = a_1.$$

Dann heißt $(L, +, \circ, 0, 1)$ **Verband** mit Nullelement 0 und *Maximum* 1 .


Aus mathematischer Sicht ist keine der beiden Operationen wichtiger als die andere. Wir können sie vertauschen und erhalten wieder einen Verband. Einzig die **Ordnung** der Elemente dreht sich um (s. Definition 3.2). Manchmal sagen wir auch nur kurz *Sei L Verband*.

+ und \circ
sind dual



Bemerkung 3.1.
 $(L, \circ, +, 1, 0)$ ist *dual^a* zu $(L, +, \circ, 0, 1)$. Eigenschaften, die für die Addition $+$ nachgewiesen werden, lassen sich also durch Dualität auf \circ anpassen.

^aim Sinne der Kategorientheorie



Beispiel 3.1 $(+, \circ)$.
Im Netzwerk *adenin_compressed* gilt für logisches Modell und Flussverband:

$$\{1, 2, 4, 5, 6\} + \{2, 3, 4, 5, 6\} = \{1, 2, 3, 4, 5, 6\},$$
$$\{1, 2, 4, 5, 6\} \circ_{L01} \{2, 3, 4, 5, 6\} = \{2, 4, 5, 6\}$$
$$\neq \{1, 2, 4, 5, 6\} \circ_{LF} \{2, 3, 4, 5, 6\} = \emptyset.$$

Wir sind zur Verbandstheorie übergegangen, um Eigenschaften des Stoffwechsels aus einem anderen Blickwinkel finden zu können. Mit 0 und 1 haben wir schon zwei Elemente gefunden, die für besondere Zustände in der Zelle stehen:



Bemerkung 3.2.

Das Minimum 0 ist neutrales Element, 1 hingegen absorbierendes Element bzgl. der Addition +, d. h. $a + 1 = 1$ für alle Elemente $a \in L$.

neutral, absorbierend

Darauf, dass es das neutrale Elemente 0 gibt, haben wir bei unserer Vorstellung von logischem Modell und Hyperpfaden geachtet und hingewiesen. Neu ist hier nun das ebenfalls eindeutig bestimmte Maximum.



Beispiel 3.2 (Maximum).

Im Netzwerk *adenin_compressed* gilt:

$$0 = \emptyset, \quad 1_{L^{01}} = 1_{L^F} = \{1, 2, 3, 4, 5, 6\}.$$

Gerade war schon mal von Ordnung die Rede, da wir die 0 Minimum nennen. Das können wir machen, da jedem Halbverband eine partielle Ordnung zugrunde liegt.



Definition 3.2 (Ordnung).

Sei $(L, +)$ Halbverband. Dann ist (L, \leq) partiell geordnet mit

$$a_1 \leq a_2 \Leftrightarrow a_1 + a_2 = a_2.$$

Ordnung
 \leq

Jetzt können wir auch mathematisch fundiert festhalten, dass die 0 das eindeutig bestimmte Minimum, die 1 das eindeutige Maximum im endlichen Verband ist. Vor allem jedoch lernen wir eine alternative Schreibweise für das Maximum 1 kennen, auf die wir später in unseren Berechnungen zurückgreifen werden.



Lemma 3.1 (Minimum/Maximum).

Sei $(L, +, 0)$ Halbverband mit neutralem Element 0. Dann gibt es ein eindeutig bestimmtes Maximum 1_L :

$$\forall a \in L : 0 \leq a \leq 1_L = \sum_{a \in L} a.$$

Maximum

Die Addition in Halbverbänden ist ordnungserhaltend, so wie wir es auch von den reellen Zahlen her kennen.



Lemma 3.2 (Ordnungserhaltung).

Sei L Halbverband. Dann:

$$\forall a, a_1, a_2 \in L : a_1 \leq a_2 \Rightarrow a + a_1 \leq a + a_2.$$

+ ist ordnungserhaltend


Beweis.

$$\mathbf{a}_1 \leq \mathbf{a}_2 \Rightarrow \mathbf{a}_2 = \mathbf{a}_1 + \mathbf{a}_2 \Rightarrow (\mathbf{a} + \mathbf{a}_1) \leq (\mathbf{a} + \mathbf{a}_1) + \mathbf{a}_2 = \mathbf{a} + (\mathbf{a}_1 + \mathbf{a}_2) = \mathbf{a} + \mathbf{a}_2.$$

□

In Halbverbänden haben zwei Elemente stets ein eindeutig bestimmtes Supremum: Nämlich ihre Summe. Für die Zweitoperation \circ gilt deshalb aus Dualitätsgründen, dass sie das eindeutige Infimum bestimmt.


+ ist
Supremum



Lemma 3.3 (Supremumseigenschaft).
 Sei L Halbverband. Dann:

$$\forall \mathbf{a}, \mathbf{a}_1, \mathbf{a}_2 \in L : \mathbf{a}_1 \leq \mathbf{a}, \mathbf{a}_2 \leq \mathbf{a} \Rightarrow \mathbf{a}_1 + \mathbf{a}_2 \leq \mathbf{a}.$$


Halbverbände besitzen Suprema gemäß der induzierten Ordnung. Das ist notwendig und hinreichend dafür, dass es sich um einen Halbverband handelt. Darum werden Halbverbände manchmal auch andersrum definiert:



Bemerkung 3.3.
 Sei (L, \leq) eine partiell geordnete Menge, bei der je zwei Elemente ein eindeutig bestimmtes Supremum $\sup(\{\mathbf{a}_1, \mathbf{a}_2\})$ haben.
 Dann ist $(L, +_{\leq})$ mit $\mathbf{a}_1 +_{\leq} \mathbf{a}_2 := \sup(\{\mathbf{a}_1, \mathbf{a}_2\})$ Halbverband.

Mit Hilfe der Supremumseigenschaft haben wir alles zusammen, um zu zeigen, dass endliche Halbverbände wirklich Verbände sind.

induzierter
Verband



Satz 3.1 (induzierter Verband).
 Sei $(L, +, 0)$ endlicher Halbverband mit neutralem Element 0 . Definiere nun den Operator \circ wie folgt:

$$\begin{aligned} \circ : L \times L &\rightarrow L \\ \mathbf{a}_1 \circ \mathbf{a}_2 &\mapsto \sum_{\substack{\mathbf{a} \leq \mathbf{a}_1 \\ \mathbf{a} \leq \mathbf{a}_2}} \mathbf{a}. \end{aligned}$$

Dann bildet $(L, +, \circ, 0, 1)$ den eindeutig durch $(L, +, 0)$ bestimmten Verband.

$\mathbf{a}_1 \circ \mathbf{a}_2$ ist also das Supremum aller Elemente, die kleiner als \mathbf{a}_1 und \mathbf{a}_2 sind. Oder mit anderen Worten: $\inf(\{\mathbf{a}_1, \mathbf{a}_2\})$ (wie es ja auch gewünscht war).

Beweis. Es ist klar, dass $(L, +, \circ, 0, 1)$ Verband ist.

Sei $(L, +, *)$ nun ein Verband. Es ist für $\mathbf{a}_1, \mathbf{a}_2 \in L$ stets $\mathbf{a}_1 * \mathbf{a}_2 \leq \mathbf{a}_1$ und auch $\mathbf{a}_1 * \mathbf{a}_2 \leq \mathbf{a}_2$,

somit insbesondere:

$$a_1 * a_2 \leq \sum_{\substack{a \leq a_1 \\ a \leq a_2}} a = a_1 \circ a_2.$$


Weiterhin ist $a_1 \circ a_2 \leq a_1$ und auch $a_1 \circ a_2 \leq a_2$. Ist $(L, +, *)$ Verband, so greift Lemma 3.3 und aus Dualität folgt $a_1 \circ a_2 \leq a_1 * a_2$.

Also ist $a_1 \circ a_2 = a_1 * a_2$. □

Es reicht also wirklich aus, endliche Halbverbände mit 0 zu betrachten, um Aussagen über endliche Verbände zu stellen. Wir schreiben verkürzt $(L, +, 0)$ und meinen damit den eindeutig bestimmten Verband. Eigenschaften von \circ ergeben sich durch Dualität aus Eigenschaften von $+$. Bei Bedarf kann jederzeit auf den zweiten Operator \circ zugegriffen werden.

§3.1.2 Ein Bisschen mehr Ordnung, bitte!

Wie können Stoffwechselfade geordnet sein? Wagen wir kurz den Schritt zurück zum Stoffwechsel und seinen Reaktionen: $a_1 \leq a_2 \Leftrightarrow a_1 \subseteq a_2$. Ein Stoffwechselfad ist also „kleiner“ als ein anderer, wenn er weniger Reaktionen und keine anderen enthält. Schauen wir nun, wie uns \leq hilft, den Stoffwechsel besser zu verstehen:




Definition 3.3 (*Antikette*).

Sei L Verband, $S \subseteq L$.

$$S \text{ Antikette} :\Leftrightarrow \forall a_1, a_2 \in S : a_1 \leq a_2 \Rightarrow a_1 = a_2.$$

Antikette

Mit Antiketten können wir jetzt auch formal von Pfaden reden, die sich nicht ordnen lassen. Wagen wir jetzt das Gegenteil: Wir betrachten Elemente, die sich nur marginal voneinander unterscheiden.



Definition 3.4 (unmittelbare Vorgänger/Nachfolger).

Sei L Verband, $a_1, a_2 \in L$.

$$a_1 \prec a_2 :\Leftrightarrow \forall a \in L : a_1 \leq a \leq a_2 \Rightarrow a \in \{a_1, a_2\}.$$

Wir nennen a_1 (unmittelbaren) Vorgänger von a_2 , a_2 (unmittelbaren) Nachfolger von a_1 und sagen a_2 überdeckt a_1 .

\prec


Partielle Ordnungen lassen sich sehr schön graphisch darstellen. Hilfsmittel dazu ist das sogenannte Hasse-Diagramm: Ein Graph, bei dem unmittelbare Vorgänger und Nachfolger durch eine Kante verbunden sind und $a < b$ durch eine höhere Positionierung von b symbolisiert wird. Am Hasse-Diagramm sieht man auch noch mal deutlich, dass je zwei Elemente nicht nur Supremum, sondern auch Infimum besitzen. Ein Beispiel für ein Hasse-Diagramm haben wir bereits in Abbildung 2.4 gesehen.

Wir können den gesamten Apparat über endliche Verbände anwenden, um den Stoffwechsel zu untersuchen. Aber geht da noch mehr?

Je mehr spezielle Eigenschaften eine algebraische Struktur hat, desto mehr Aussagen lassen sich über sie treffen – mehr Arbeitsmaterial. Auch bei Verbänden wird vergleichsweise viel über einen speziellen Vertreter geforscht: Den geometrischen Verband, der eine wichtige Rolle in der algebraischen Geometrie spielt.

Eine Vorstufe zum geometrischen Verband ist der semi-modulare Verband:

semi-
modular



Definition 3.5 (semi-modularer Verband).
Sei L Verband, der für $a_1, a_2 \in L$ folgende Bedingung erfüllt:

$$a_1 \circ a_2 \prec a_1 \Rightarrow a_2 \prec a_1 + a_2.$$

Dann heißt L semi-modularer Verband.

Die Verbände, die uns interessieren, sind im Allgemeinen nicht semi-modular. Wieder können wir uns auf Abbildung 2.4, die Verbandsdarstellung des Systems `adenin_compressed` beziehen: Sowohl im logischen Verband L^{01} , als auch im Flussverband gilt $0 = \{1, 4, 5\} \circ \{2, 3, 4, 6\} \prec \{1, 4, 5\}$, es ist aber $\{2, 3, 4, 6\} \not\prec \{1, 2, 3, 4, 5, 6\} = \{1, 4, 5\} + \{2, 3, 4, 6\}$. Es ist nämlich $\{2, 3, 4, 6\} < \{1, 2, 3, 4, 6\} < \{1, 2, 3, 4, 5, 6\}$.

Dies reicht uns schon als Gegenbeispiel dafür, dass konische Verbände (die wir später definieren wollen und Grundlage des Flussverbandes sind) im Allgemeinen nicht semi-modular sind.

Interessanterweise benötigen wir diese starke Eigenschaft gar nicht, um den Stoffwechsel zu untersuchen. Uns reichen einige Grundaussagen, die für alle (endlichen) Verbände gelten, und können damit bereits schnelle Algorithmen entwickeln und ein besseres Verständnis des Stoffwechsels erlangen. Das hat damit zu tun, dass viele Ergebnisse über geometrische Verbände Aussagen über das Zusammenspiel der beiden Operationen $+$ und \circ machen. Uns hingegen reicht immer $+$.

§3.2 Minimale und unzerlegbare Elemente


§3.2.1 Definitionen...

Davey & Priestley (1990) diskutieren verschiedene Darstellungsmöglichkeiten für alle möglichen Spielarten von Verbänden. Auch das Konzept von minimalen und unzerlegbaren Elementen wird hierbei vorgestellt.

Hierfür reicht die $+$ Operation des Halbverbandes, ohne auch nur einmal die induzierte Zweitoperation \circ zu verwenden. Im endlichen Fall können wir die Ideen und Aussagen ganz unbekümmert übernehmen.

Betrachtet man das Hasse-Diagramm eines endlichen Verbandes, so steht es auf einem einzigen Element und läuft in eine Spitze zusammen – 0 bzw. 1. Gerade das Maximum 1_L verrät uns wichtige Eigenschaften des Verbandes, doch es gibt noch andere interessante Elemente im Verband. In diesem Abschnitt wollen wir eben solche Elemente kennenlernen, die uns den Verband besser beschreiben.

Anfangen wollen wir hierbei mit der 1. Etage über dem Minimum:




Definition 3.6 (minimales Element/Atom).
 Sei L Verband, $e \in L \setminus \{0\}$.

$$e \text{ minimales Element von } L \Leftrightarrow \forall a \in L : a < e \Rightarrow a = 0.$$

Es bezeichne $\mathcal{M}(L)$ die Menge aller minimalen Elemente bzw. aller Atome von L .

minimal
 $\mathcal{M}(L)$



Beispiel 3.3 ($\mathcal{M}(L)$).
 Der in Abbildung 2.2 abgebildete Stoffwechselfad ist ein minimaler Pfad im logischen Verband wie auch im Flussverband von *etc.purple*.
 Im stationären Flussverband L^F vom Netzwerk *adenin_compressed* gibt es sechs minimale Elemente; der logische Verband umfasst hingegen nur fünf:

$$\mathcal{M}(L^{01}) = \{\{1, 4, 5\}, \{1, 3, 4, 6\}, \{1, 2, 3\}, \{2, 4, 5\}, \{2, 4, 6\}\},$$


$$\mathcal{M}(L^F) = \{\{1, 4, 5\}, \{1, 3, 4, 6\}, \{1, 2, 3\}, \{2, 3, 4, 5\}, \{1, 2, 4, 6\}, \{2, 3, 4, 6\}\}.$$

$\mathcal{M}(L)$

Man sieht am Hasse-Diagramm, kann aber auch leicht durch Einsetzen der Definition zeigen:



Bemerkung 3.4 (Antikette aller minimalen Elemente).
 $\mathcal{M}(L)$ ist Antikette in L .




Bemerkung 3.5.

$$\forall a \in L \setminus \{0\} \exists e \in \mathcal{M}(L) : e \leq a.$$

Auch das ist mit Blick aufs Hasse-Diagramm trivial. Wir werden es ausnutzen können, wenn wir später Überdeckungen des Verbandes und die Flusskopplung in sogenannten atomaren Verbänden betrachten.


Nachdem wir nun die 1. Etage betrachtet haben, geht unsere Suche nach speziellen Elementen weiter. Als nächstes stoßen wir hierbei auf unzerlegbare Elemente:

unzerlegbar
 $\mathcal{J}(L)$

 **Definition 3.7 (unzerlegbares Element).**
 Sei L Verband, $b \in L \setminus \{0\}$.

b *unzerlegbares Element* von $L \Leftrightarrow \forall a_1, a_2 \in L : a_1 + a_2 = b \Rightarrow b \in \{a_1, a_2\}$.

Es sei $\mathcal{J}(L)$ die Menge aller unzerlegbaren Elemente / aller irreduziblen Elemente von L .

 **Bemerkung 3.6.**
 Äquivalente Definitionen sind:

- $b \in L \setminus \{0\}$ unzerlegbar in $L \Leftrightarrow \forall a_1, a_2 \in L : a_1 < b, a_2 < b \Rightarrow a_1 + a_2 < b$,
- $b \in L \setminus \{0\}$ unzerlegbar in $L \Leftrightarrow \forall a_1, a_2 \in L : a_1 < b, a_2 < b \Rightarrow a_1 = a_2$.


Im Hasse-Diagramm sind die unzerlegbaren Elemente also genau die Punkte mit nur einem einzigen Vorgänger.

Minimale und unzerlegbare Elemente stehen im engen Zusammenhang:

minimal \Rightarrow
 unzerlegbar

 **Bemerkung 3.7.**
 Es gilt $\mathcal{M}(L) \subseteq \mathcal{J}(L)$.

$\mathcal{J}(L)$

 **Beispiel 3.4 ($\mathcal{J}(L)$).**
 Der in Abbildung 2.2 abgebildete minimale Stoffwechselfad von *etc_purple* ist damit auch unzerlegbar.
 Im stationären Flussverband L^F von Netzwerk *adenin_compressed* entsprechen die minimalen den unzerlegbaren Elementen; der logische Verband umfasst hingegen sechs zusätzliche Elemente:


$$\mathcal{J}(L^{01}) = \{\{1, 4, 5\}, \{1, 3, 4, 6\}, \{1, 2, 3\}, \{2, 4, 5\}, \{2, 4, 6\},$$

$$\quad \{1, 3, 4, 5\}, \{1, 4, 5, 6\}, \{1, 3, 4, 6\}, \{2, 3, 4, 5\}, \{1, 2, 4, 6\}, \{2, 3, 4, 6\}\},$$

$$\mathcal{J}(L^F) = \{\{1, 4, 5\}, \{1, 3, 4, 6\}, \{1, 2, 3\}, \{2, 3, 4, 5\}, \{1, 2, 4, 6\}, \{2, 3, 4, 6\}\}.$$


Aus der Ringtheorie in der Algebra kennen wir die Begriffe Primelemente und unzerlegbare Elemente. Wie im Fall des Verbandes ist jedes Primelement unzerlegbares Element, jedoch nicht notwendig umgekehrt.

So wie es viele wichtige Ringe gibt, bei denen die Primelemente gleich den unzerlegbaren Elementen sind, so gibt es Verbände, wo nur die minimalen Elemente unzerlegbar sind. Dann reden wir von den bereits erwähnten atomaren Verbänden:

 **Definition 3.8** (atomarer Verband).
 Sei L Verband mit $\mathcal{M}(L) = \mathcal{J}(L)$. Dann heißt L *atomarer Verband*.


atomar

Viele der Verbände, mit denen wir beim Stoffwechsel zu tun haben – insbesondere der stationäre Flussverband L^F – sind atomar. Nun haben wir alles zusammen für die Definition des wichtigen Verbandstyps, den wir gar nicht brauchen:

 **Definition 3.9** (geometrischer Verband).
 Sei L endlicher Verband. L heißt *geometrischer Verband genau dann, wenn L atomar und semi-modular ist*.

geometrisch

Dass wir nicht mit geometrischen Verbänden arbeiten, ist uns kein sehr großes Hindernis. Teilweise auch, weil es im Flussverband L^F ergänzende Summanden gibt, die wir hier vorstellen:


 **Lemma 3.4** (ergänzender Summand).
 Sei L Verband.

$$\forall \mathbf{a}, \mathbf{a}_1, \mathbf{a}_2 \in L : (\mathbf{a}_1 \prec \mathbf{a}, \mathbf{a}_2 \prec \mathbf{a}, \mathbf{a}_1 \neq \mathbf{a}_2 \Rightarrow \mathbf{a}_1 + \mathbf{a}_2 = \mathbf{a}),$$

$$(\forall \mathbf{a} \in L \forall \mathbf{a}_1 \in L \setminus \{0\} \exists \mathbf{a} > \mathbf{a}_2 \in L : \mathbf{a}_1 < \mathbf{a} \Rightarrow \mathbf{a} = \mathbf{a}_1 + \mathbf{a}_2) \Rightarrow \mathcal{M}(L) = \mathcal{J}(L).$$
 Verbände, für die die zweite Zeile gilt, nennen wir *Verband mit ergänzenden Summanden*.
 Die Rückrichtung der Aussagen gilt im Allgemeinen nicht.

ergänzende Summanden

Die Gültigkeit der ersten Gleichung erkennt man wieder leicht am Hasse-Diagramm; auch diesmal ist auf die Supremumseigenschaft (Lemma 3.3) verwiesen. Für die zweite betrachte man die Definition von $\mathcal{J}(L)$.


 **Beispiel 3.5** (ergänzender Summand).
 Im Netzwerk *adenin_compressed* (Abbildung 2.4) ist $\{1, 4, 5\} < \{1, 2, 4, 5, 6\}$. Wähle als ergänzenden Summanden $\{1, 2, 4, 6\}$.
 Im zugehörigen logischen Verband gibt es jedoch keinen ergänzenden Summanden für $\{1, 4, 5\} \leq \{1, 3, 4, 5\}$.

ergänzender Summand

§3.2.2 ... und Eigenschaften

Wo wir jetzt schon langsam anfangen, Verbandselemente als Summe kleinerer Elemente zu schreiben, wollen wir das nun formalisieren. Hierfür definieren wir:


Erzeugnis

 **Definition 3.10** (Erzeugnis, Erzeuger).
 Sei L Verband und $A, B \subseteq L$.

$$\langle A \rangle := \left\{ \sum_{a \in \tilde{A}} a \mid \tilde{A} \subseteq A \right\}.$$

Dann heißt $\langle A \rangle$ *Erzeugnis* von A . Weiterhin heißt B *Erzeuger* von L , wenn $L = \langle B \rangle$ gilt.


Zerlegungen

 **Bemerkung 3.8** (Zerlegbarkeit).
 Sei $L = \langle B \rangle$. Dann:

$$\forall a \in L \setminus B \exists b_1, \dots, b_t \in B \text{ Antikette: } a = b_1 + \dots + b_t.$$

Davey & Priestley (1990) formulieren ein schönes Lemma, wie man geschickt zerlegt:

Zerlegung mit $\mathcal{J}(L)$

 **Lemma 3.5** (Zerlegung in $\mathcal{J}(L)$).
 Sei L Verband, dann:


$$\forall a_1, a_2 \in L \setminus \{0\} \exists b \in \mathcal{J}(L) : a_1 \not\leq a_2 \Rightarrow b \leq a_1, b \not\leq a_2.$$

$$\forall a \in L : a = \sum_{\substack{b \in \mathcal{J}(L) \\ b \leq a}} b.$$

Unzerlegbare Elemente zerlegen also alle anderen Verbandselemente (außer der 0 natürlich).

Daraus ergibt sich ein einfacher Algorithmus, um \circ zu berechnen, wenn man die unzerlegbaren Elemente des Verbandes kennt:


+ und \circ mit $\mathcal{J}(L)$

 **Bemerkung 3.9.**
 Für $L = \langle B \rangle$ gilt:

$$\forall a_1, a_2 \in L : B_{i=1,2} = \{b \in B \mid b \leq a_i\} \Rightarrow a_1 + a_2 = \sum_{b \in B_1 \cup B_2} b,$$

$$a_1 \circ a_2 = \sum_{b \in B_1 \cap B_2} b.$$

Damit wissen wir, dass sie nicht nur hübsch im Hasse-Diagramm aussehen, sondern als Menge eine besondere Bedeutung haben. Wir stellen fest:

 **Satz 3.2** (minimaler Erzeuger).
 Sei L endlicher Verband. Dann ist $\mathcal{J}(L)$ eindeutig bestimmter minimaler Erzeuger von L , d. h.


$$L = \langle \mathcal{J}(L) \rangle,$$

$$\forall B \subseteq L: \quad L = \langle B \rangle \Rightarrow \mathcal{J}(L) \subseteq B.$$

minimaler, eindeutiger Erzeuger

Und das gilt für jeden endlichen Verband! D. h. insbesondere, dass wir jeden Stoffwechselverband durch seine unzerlegbaren Elemente repräsentieren und generieren können.


Wir fassen unsere bisherigen Ergebnisse zusammen und erhalten:

 **Korollar 3.1.**
 Ist B Erzeuger, also $L = \langle B \rangle$, dann:

$$\mathcal{M}(L) \subseteq \mathcal{J}(L) \subseteq B \subseteq L.$$

Erzeuger enthalten $\mathcal{J}(L)$

Wir schließen den Abschnitt mit einem kleinen Trick, wie man feststellen kann, ob ein Verband atomar ist:

 **Lemma 3.6.**
 Sei $B \subseteq L$ Antikette und Erzeuger von $L = \langle B \rangle$. Dann gilt:

$$\mathcal{M}(L) = \mathcal{J}(L) = B.$$

Beweis. Sei $L = \langle B \rangle$, dann:

$$\tilde{b} \in B \setminus \mathcal{M}(L) \Rightarrow \exists e \in \mathcal{M}(L) : e < \tilde{b} \Rightarrow \exists e \in B : e < \tilde{b} \Rightarrow B \text{ keine Antikette.}$$

□

§3.3 Unterverbände

Viele Fragestellungen in der Biologie lassen sich so zusammenfassen: Ich verändere die Umweltbedingungen meiner Zellkultur. Was für Auswirkungen hat das auf den Stoffwechsel?

Viele dieser veränderten Bedingungen lassen sich 1-zu-1 in Verbandssprache übersetzen. Entzieht man der Nährlösung Zucker, so blockiert man im Modell die Reaktion, die den Zucker in die Zelle transportiert. An den Veränderungen des Verbandes kann man dann die Auswirkungen auf den Stoffwechsel ablesen.

Je mehr solcher Bedingungen – sogenannte *Constraints* – man an das System stellt, desto kleiner fällt für gewöhnlich der Verband aus, den man erhält. Trotzdem steht der neue Verband in engem Zusammenhang zu unserem ursprünglichen, unveränderten System: Er enthält alle Stoffwechselffade, die den Constraints genügen, stellt also eine Teilmenge von L dar.

Sind die Bedingungen so gewählt, dass auch unsere eingeschränkte Menge den Verbandseigenschaften erfüllt, so reden wir vom Unterhalbverband:

Unterhalbverband



Definition 3.11 (Unterhalbverband).

Sei L Verband, $U \subseteq L$ mit $u_1, u_2 \in U \Rightarrow u_1 + u_2 \in U$.
Dann heißt $U \subseteq L$ *Unterhalbverband*.

Wir haben in Satz 3.1 gesagt, dass jeder endliche Halbverband mit 0 eindeutig zum Verband erweitert werden kann. Die zugehörige Operation \circ ist eindeutig durch $+$ beschrieben.

Auch Unterhalbverbände können entsprechend mit einer zweiten Operation ausgestattet werden. Da diese jedoch oftmals von der Zweitoperation des Oberverbandes abweicht, unterscheiden wir zwischen Unterhalbverbänden und Unterverbänden:

Unterverband



Definition 3.12 (Unterverband).

Sei L Verband und $U \subseteq L$ mit $0 \in U$.

U *Unterverband* von L $:\Leftrightarrow u_1, u_2 \in U \Rightarrow u_1 + u_2, u_1 \circ u_2 \in U$.

Unterverband



Beispiel 3.6 (Unterverband).

Es ist $L^F \subseteq L^{01}$ Unterhalbverband.

Wie man jedoch am Netzwerk *adenin_compressed* sieht, handelt es sich im Allgemeinen nicht um eine Unterverbandsbeziehung (Vergleich Beispiel 3.1).

Auf einen förmlichen Beweis verzichten wir an dieser Stelle. Als wesentliche Beweisidee erinnern wir nur an die Motivation von L^{01} : Wir wollen die Bedingung M produziert in genau dem Maße wie M konsumiert vereinfachen auf M produziert genau dann, wenn M konsumiert. Offensichtlich folgt das Zweite aus dem Ersten.

Wir haben im vergangenen Abschnitt bereits das Maximum eines Verbandes vorgestellt. Da wir im Laufe dieser Arbeit viele, viele Anwendungen für dieses Maximum kennenlernen werden, betonen wir hier nun ausdrücklich, dass auch Unterhalbverbände Maxima besitzen:



Bemerkung 3.10 (Minimum/Maximum im Unter(halb)verband).

Sei U Unter(halb)verband von L , dann verfügt U über ein Maximum 1_U .

Von besonderer Bedeutung ist:



Lemma 3.7.

* Es gilt $0 \leq 1_U \leq 1_L$.

Die Hauptaussage dieses Lemmas ist, dass Elemente im Unterhalbverband im Allgemeinen nicht größer sind als Elemente im Oberverband. Das gilt sowohl für die Maxima, als auch für die Zweitoperation:



Bemerkung 3.11.

Sei L Verband und U Unterhalbverband von L . Dann ist $(U, +, \circ_U, 0, 1_U)$ Verband und:

$$\forall u_1, u_2 \in U : u_1 \circ_U u_2 \leq u_1 \circ u_2.$$

Jedoch gilt im Allgemeinen $u_1 \circ_U u_2 \neq u_1 \circ u_2$, also U nicht Unterverband von L .

Das Maximum werden wir jetzt auch gleich zum Einsatz bringen, um Unterverbände zu identifizieren. Dafür definieren wir zunächst:



Definition 3.13 (Hauptideal).

Sei L Verband und $a \in L$ beliebig. Definiere nun $a \circ L$ wie folgt:

$$a \circ L := \{a \circ a_1 \mid a_1 \in L\}.$$

Dann heißt $a \circ L$ Hauptideal von L .

Hauptideal



Satz 3.3 (Hauptideale sind Unterverbände).

Sei $(L, +, \circ, 0)$ Verband und $U \subseteq L$ Unterhalbverband mit maximalem Element 1_U . Dann:

$$\begin{aligned} U &\subseteq 1_U \circ L, \\ U = 1_U \circ L &\Rightarrow U \text{ Unterverband} \end{aligned}$$

Außerdem wissen wir:

$$U = 1_U \circ L \Rightarrow U = \{a \in L \mid a \leq 1_U\}$$

$a \circ L \leq L$

Beweis. Sei $a \in U$ beliebig. Da U die gleiche Operation $+$ wie L hat, gilt $a \leq 1_U$ und somit $a = 1_U \circ a$. Demnach ist auf jeden Fall $U \subseteq 1_U \circ L$.

Angenommen $U \subseteq L$ ist kein Unterverband. Dann gibt es $a_1, a_2 \in U$ mit $a_1 \circ_U a_2 \neq a := a_1 \circ a_2 \in L \setminus U$.

Es ist $a_1, a_2 \in U$, damit $a_1, a_2 \leq 1_U$ und insbesondere $a = a_1 \circ a_2 \leq 1_U$. Darum gilt $a = 1_U \circ a \in 1_U \circ L$. Also ist wirklich $U \subseteq 1_U \circ L$.

Widmen wir uns der letzten Aussage.

$$a \in \mathcal{U} \Leftrightarrow a \in 1_{\mathcal{U}} \circ L \Leftrightarrow a = 1_{\mathcal{U}} \circ a \Leftrightarrow a \leq 1_{\mathcal{U}}.$$

Hierbei nutzen wir die Idempotenz von \circ . □

Unterstrukturen



Beispiel 3.7 (verschiedene Unterstrukturen).

In Abbildung 3.1 sind ein Unterhalbverband \mathcal{U}_1 , ein Unterverband \mathcal{U}_2 und eine Hauptideal \mathcal{U}_3 von Flussverband L^F (3.1a) des Netzwerkes `adenin_compressed` zu sehen.

Der Unterhalbverband \mathcal{U}_1 ist kein Unterverband, da $\{1, 2, 3\} = \{1, 2, 3, 4, 5\} \circ \{1, 2, 3, 4, 6\} \notin \mathcal{U}_1$ gilt (3.1b).

Bei \mathcal{U}_2 ist kein Hauptideal, da $\mathcal{U}_2 \ni \{1, 2, 3, 4, 6\} \geq \{1, 2, 3\} \notin \mathcal{U}_2$ gilt (3.1c).

Es ist $\mathcal{U}_3 = \{1, 2, 3, 4, 6\} \circ L^F$ Hauptideal (3.1d).

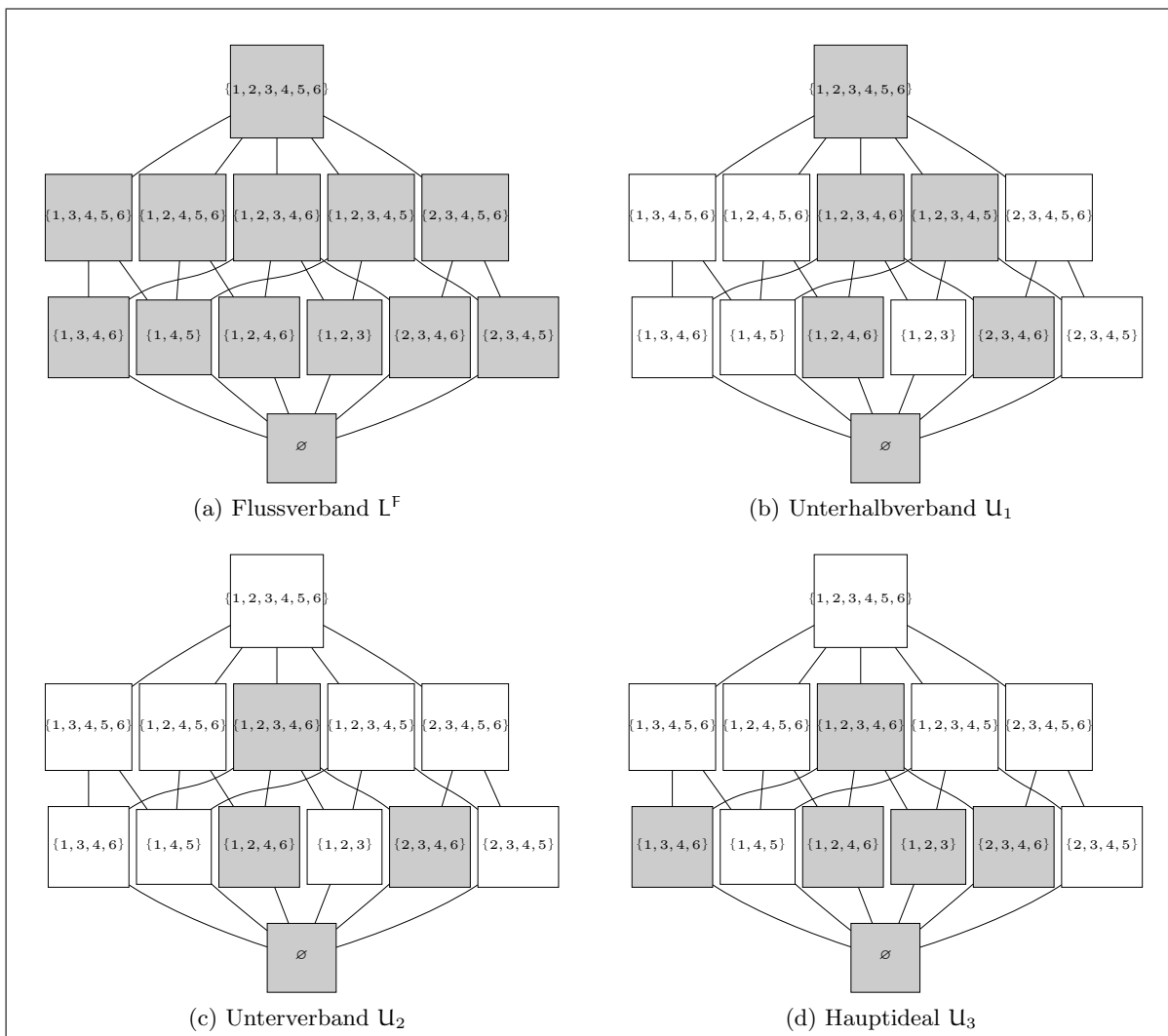


Abbildung 3.1.: Einige Unterstrukturen des Flussverbands zu `adenin_compressed`.

Widmen wir uns nun Schnitten:



Lemma 3.8 (Schnitt von Unterverbänden).

Sei L endlicher Verband und $U_1, U_2 \subseteq U$ Unter(halb)verbände von L .
 Dann ist auch $U_1 \cap U_2 \subseteq L$ Unter(halb)verband von L und es gilt:

$$\mathcal{M}(U_1) \cap \mathcal{M}(U_2) \subseteq \mathcal{M}(U_1 \cap U_2),$$

$$\mathcal{J}(U_1) \cap \mathcal{J}(U_2) \subseteq \mathcal{J}(U_1 \cap U_2).$$

Im Allgemeinen herrscht keine Gleichheit.

$U_1 \cap U_2$

Beweis. Seien $a_1, a_2 \in U_1 \cap U_2$. Dann ist $a_1 + a_2 \in U_1$ und auch $a_1 + a_2 \in U_2$. Wegen $0 \in U_1 \cap U_2$ ist $U_1 \cap U_2$ also wirklich Unterhalbverband.

Analog argumentieren wir für U_1, U_2 Unterverbände mit $a_1 \circ a_2 \in U_1 \cap U_2$.

Sei nun $a \in U_1 \cap U_2 \setminus \mathcal{J}(U_1 \cap U_2)$. Dann gibt es für a wieder eine echte Zerlegung in Elemente aus $U_1 \cap U_2$ und somit auch in U_1 und in U_2 , also $a \notin \mathcal{J}(U_1) \cup \mathcal{J}(U_2)$.

Sei schließlich $a \in U_1 \cap U_2 \setminus \mathcal{M}(U_1 \cap U_2)$. Dann gibt es $0 < e < a$ mit $e \in \mathcal{M}(U_1 \cap U_2)$. Aber es ist auch $e \in U_1$ und $a \in U_2$, also $a \notin \mathcal{M}(U_1) \cup \mathcal{M}(U_2)$. \square



Beispiel 3.8 (Schnitt von Unterverbänden).

In Abbildung 3.2a ist der Schnitt zweier Unterverbände zu sehen.
 Es ist $\{1, 2, 4, 5\} \in \mathcal{M}(U_1 \cap U_2) \setminus (\mathcal{M}(U_1) \cap \mathcal{M}(U_2))$ sowie beispielhaft $\{1, 2, 3, 4, 5\} \in \mathcal{J}(U_1 \cap U_2) \setminus (\mathcal{J}(U_1) \cap \mathcal{J}(U_2))$.

Schnitte

Für Hauptideale halten wir fest:



Bemerkung 3.12 (Schnitt von Hauptidealen).

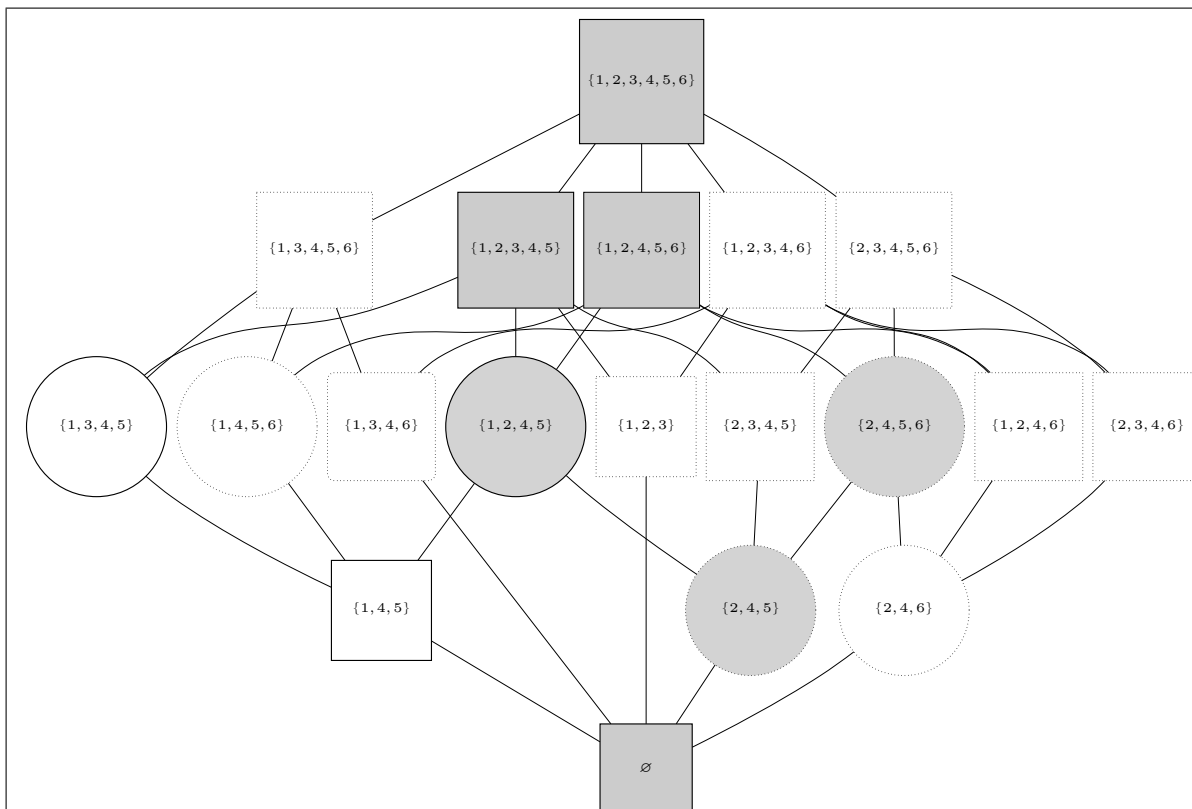
Sei L endlicher Verband und $U_1 = u_1 \circ L, U_2 = u_2 \circ L$ Hauptideale von L .
 Dann ist $U_1 \cap U_2 = (u_1 \circ u_2) \circ L$ ebenfalls *Hauptideal* von L .

$(a_1 \circ a_2) \circ L$

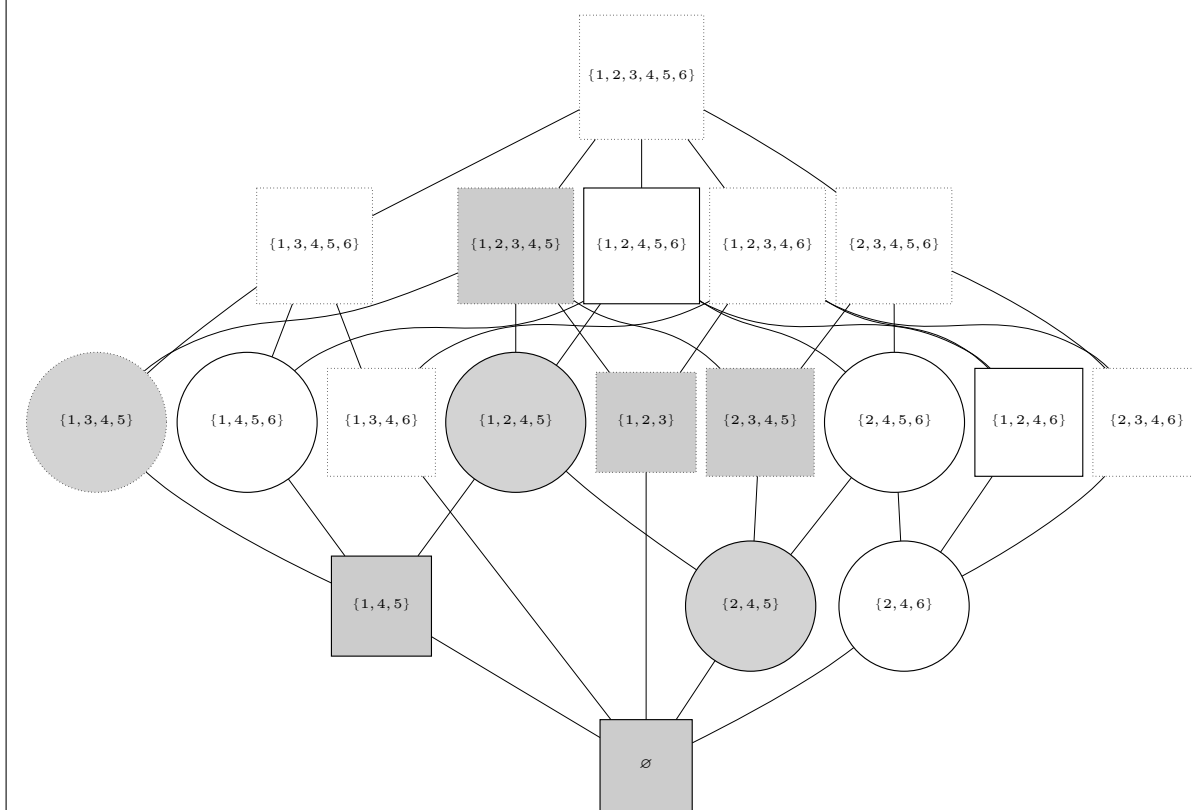
Beweis. Sei $a \in U_1 \cap U_2$. Dann gilt $a \leq u_1$ und $a \leq u_2$, also insgesamt $a \leq u_1 \circ u_2$ und daher $a \circ (u_1 \circ u_2) = a$. Somit ist $U_1 \cap U_2 \subseteq (u_1 \circ u_2) \circ L$.

Nach Definition des Hauptideals gilt jedoch auch $a = (u_1 \circ u_2) \circ a = u_1 \circ (u_2 \circ a)$. Damit ist $(u_1 \circ u_2) \circ L \subseteq u_1 \circ L$. Analog zeigen wir $(u_1 \circ u_2) \circ L \subseteq u_2 \circ L$. Also gilt wirklich Gleichheit. \square

Wir werden gleich sogar noch sehen, dass man starke Aussagen für die minimalen/unzerlegbaren Elemente treffen kann.



(a) Schnitter zweier Unterverbände: U_1 gerahmt, U_2 grau.



(b) Schnitt zweier Hauptideale

Abbildung 3.2.: Schnitte von Unterstrukturen von L^{01} zu adenin_compressed.



Beispiel 3.9 (Schnitt von **Hauptidealen**).

In *Abbildung 3.2b* ist der *Durchschnitt* von zwei *Hauptidealen* zu sehen.
 Man erkennt sofort $(\{1, 2, 3, 4, 5\} \circ L^{01}) \cap (\{1, 2, 4, 5, 6\} \circ L^{01}) = (\{1, 2, 4, 5\} \circ L^{01})$, wobei
 $\{1, 2, 4, 5\} = \{1, 2, 3, 4, 5\} \circ \{1, 2, 4, 5, 6\}$ gilt.

Nachdem wir jetzt einige generelle Strukturaussagen zu Unterverbänden treffen können, wollen wir nun die Unterverbände betrachten, die uns besonders interessieren. Was passiert, wenn wir der Zelle bestimmte Umwelteigenschaften aufzwingen, unsere Stoffwechselfade also diesen Eigenschaften genügen müssen?

Erhalten wir wirklich einen vollwertigen Unterhalbverband, so gilt für seine minimalen und unzerlegbaren Elemente:



Satz 3.4 (Eigenschafts-Unterhalbverbände).

Sei L Verband, $P \subseteq L$. Dann ist das *Erzeugnis* $\langle P \rangle$ Unterhalbverband von L und es gilt:

$$\begin{aligned} \mathcal{M}(L) \cap P &\subseteq \mathcal{M}(\langle P \rangle) \subseteq P, \\ \mathcal{J}(L) \cap P &\subseteq \mathcal{J}(\langle P \rangle) \subseteq P. \end{aligned}$$

Eigen-
schaften

Beweis. Wir verfahren im Wesentlichen wie bei [Lemma 3.8](#).

$\mathcal{M}(\langle P \rangle) \subseteq P$ ist [Korollar 3.1](#).

Sei $a \in \langle P \rangle \setminus \mathcal{J}(\langle P \rangle)$. Dann gibt es für a wieder eine echte Zerlegung in Elemente aus P und somit auch in L , also $a \notin \mathcal{J}$.

Sei nun $a \in \langle P \rangle \setminus \mathcal{M}(\langle P \rangle)$. Dann gibt es $0 < e < a$ mit $e \in \mathcal{M}(\langle P \rangle)$. Aber es ist auch $e \in L \setminus \{0\}$, also $a \notin \mathcal{M}(L)$. □

Die meisten Unterhalbverbände, mit denen wir arbeiten werden, basieren auf diesen Eigenschaften. Aber in der Biologie können auch andere Nebenbedingungen relevant sein, so dass die gültige Menge an Stoffwechselfaden doch keinen Verband mehr bildet.



Satz 3.5 (**Verbandsstümpfe**).

Sei L Verband und $\varphi : L \rightarrow [\alpha, \beta]$ ($\alpha, \beta \in \mathbb{R}$) *monoton steigende Funktion*, also $a_1 \leq a_2 \Rightarrow \varphi(a_1) \leq \varphi(a_2)$, mit $\varphi(0) = \alpha$ und $\varphi(1) = \beta$.

Sei weiterhin $\tau \in [\alpha, \beta]$. Sei nun $L_{\varphi \leq \tau} := \langle P_{\varphi \leq \tau} \rangle$ mit $P_{\varphi \leq \tau} := \{a \in L \mid \varphi(a) \leq \tau\}$.

Dann ist $L_{\varphi \leq \tau}$ Unterhalbverband von L und es gilt:

$$\begin{aligned} \mathcal{M}(L_{\varphi \leq \tau}) &= \mathcal{M}(L) \cap P_{\varphi \leq \tau}, \\ \mathcal{J}(L_{\varphi \leq \tau}) &= \mathcal{J}(L) \cap P_{\varphi \leq \tau}. \end{aligned}$$

Insbesondere ist $L_{\varphi \leq \tau}$ *atomar*, wenn L *atomar* ist. $P_{\varphi \leq \tau}$ heißt *Verbandsstumpf* von L .

$L_{\varphi \leq \tau}$

Beweis. Es ist $L_{\varphi \leq \tau}$ Erzeugnis einer Teilmenge $P_{\varphi \leq \tau} \subseteq L$ und somit Unterhalbverband. Also greift Satz 3.4 und wir müssen nur noch einen kleinen Teil zeigen.


Angenommen es ist $\mathfrak{a} \in \mathcal{J}(L_{\varphi \leq \tau}) \setminus \mathcal{J}(L)$. Dann gibt es $\mathfrak{b}_1, \dots, \mathfrak{b}_t \in \mathcal{J}(L)$ mit $\mathfrak{a} = \mathfrak{b}_1 + \dots + \mathfrak{b}_t$ echte Zerlegung.

Wegen Satz 3.4 können wir sagen, dass $\varphi(\mathfrak{a}) \leq \tau$, da $\mathfrak{a} \in \mathcal{J}_{\varphi \leq \tau} \subseteq P_{\varphi \leq \tau}$. Da nun aber $\mathfrak{b}_i \leq \mathfrak{a}$ und φ monoton steigend, gilt $\varphi(\mathfrak{b}_i) \leq \varphi(\mathfrak{a}) \leq \tau$, also $\mathfrak{b}_i \in P_{\varphi \leq \tau} \subseteq L_{\varphi \leq \tau}$. Widerspruch dazu, dass \mathfrak{a} unzerlegbar in $L_{\varphi \leq \tau}$.

Also gilt wirklich $\mathcal{J}(L_{\varphi \leq \tau}) \subseteq \mathcal{J}(L)$ und mit Satz 3.4 insgesamt $\mathcal{J}(L_{\varphi \leq \tau}) = \mathcal{J}(L) \cap P_{\varphi \leq \tau}$.

$\mathcal{M}(L_{\varphi \leq \tau}) = \mathcal{M}(L) \cap P_{\varphi \leq \tau}$ lässt sich analog zeigen. □

Verbands-
stumpf

 **Beispiel 3.10 (Verbandsstumpf).**
 In Abbildung 3.3 ist ein Verbandsstumpf $P_{\varphi \leq 3}$ von L^{01} zu *adenin_compressed* zu sehen.
 Hierbei ist $\varphi(\mathfrak{a}) = \sum_{i \in \mathfrak{a}} w_i$ mit $w_3 = 2$ und $w_i = 1$ für $i \neq 3$.
 Es ist $P_{\varphi \leq 3} \subsetneq L_{\varphi \leq 3}$ echte Teilmenge.

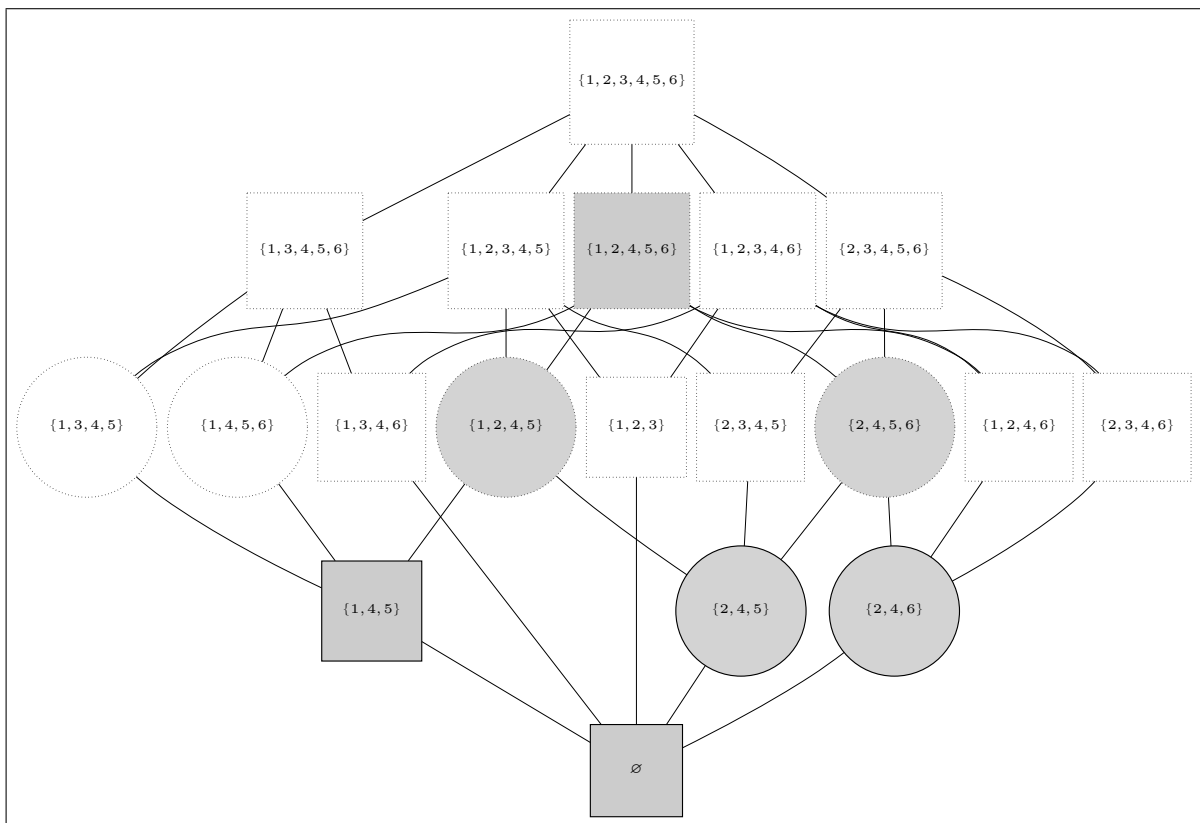



Abbildung 3.3.: Ein Verbandsstumpf von $L = L^{01}$ zu *adenin_compressed*. ($P_{\varphi \leq \tau}$ gerahmt)

Wenn wir später (Minimal) Cut Sets und Flusskopplung definieren, werden wir sehen, dass die minimalen bzw. unzerlegbaren Elemente ausreichen, um Aussagen über den gesamten

Verband zu fällen. Insbesondere somit also für $P_{\varphi \leq \tau} \subseteq L_{\varphi \leq \tau}$. Da alle minimalen und unzerlegbaren Elemente von $L_{\varphi \leq \tau}$ auch in $P_{\varphi \leq \tau}$ enthalten sind, werden wir also auch diese beiden Eigenschaften bei den Verbandsstümpfen betrachten können, auch wenn $P_{\varphi \leq \tau}$ selbst gar kein Verband ist.

Im engen Zusammenhang mit den Cut Sets steht das sogenannte Innere des Verbandes. Wir stellen es hier vor, da auch das Innere Unterhalbverband ist:



Definition 3.14 (Innere von L).
 Sei L Verband und $\mathcal{M}(L)$ Antikette der minimalen Elemente.


$$L^\circ := \langle \mathcal{M}(L) \rangle.$$

L° heißt das Innere von L .

L°




Bemerkung 3.13.
 L° ist Unterhalbverband von L aber im Allgemeinen kein Unterverband.



Beispiel 3.11 (Innere).
 In Abbildung 3.4 ist das Innere des (nicht-atomaren) Verbandes $L = L^{01}$ zu `adenin_compressed` zu sehen.
 Es ist $\{1, 2, 3, 4, 5\} \in L \setminus L^\circ$ eines der insgesamt sechs Elemente des logischen Verbandes, die nicht von $\mathcal{M}(L^{01})$ erzeugt werden.

Zum Abschluss dieses Abschnitts betrachten wir noch einmal kurz die Hauptideale:



Bemerkung 3.14 (Hauptideale sind Verbandsstümpfe).
 Sei L Verband und $\mathfrak{a} \circ L$ Hauptideal mit $\mathfrak{a} \in L$ beliebig. Dann ist $\mathfrak{a} \circ L$ Verbandsstumpf.

Insbesondere gilt:

$$\begin{aligned} \mathcal{M}(\mathfrak{a} \circ L) &= \{e \in \mathcal{M}(L) \mid e \leq \mathfrak{a}\}, \\ \mathcal{J}(\mathfrak{a} \circ L) &= \{b \in \mathcal{J}(L) \mid b \leq \mathfrak{a}\}. \end{aligned}$$

Zudem gilt für zwei verschiedene Hauptideale $\mathfrak{U}_1 = \mathfrak{a}_1 \circ L$, $\mathfrak{U}_2 = \mathfrak{a}_2 \circ L$:

$$\begin{aligned} \mathcal{M}(\mathfrak{U}_1) \cap \mathcal{M}(\mathfrak{U}_2) &= \mathcal{M}(\mathfrak{U}_1 \cap \mathfrak{U}_2) = \{e \in \mathcal{M}(L) \mid e \leq \mathfrak{a}_1 \circ \mathfrak{a}_2\}, \\ \mathcal{J}(\mathfrak{U}_1) \cap \mathcal{J}(\mathfrak{U}_2) &= \mathcal{J}(\mathfrak{U}_1 \cap \mathfrak{U}_2) = \{b \in \mathcal{J}(L) \mid b \leq \mathfrak{a}_1 \circ \mathfrak{a}_2\}. \end{aligned}$$

Beweis. Nach Satz 3.3 wissen wir, $\mathfrak{a} \circ L = \{\mathfrak{a}_1 \in L \mid \mathfrak{a}_1 \leq \mathfrak{a}\}$.

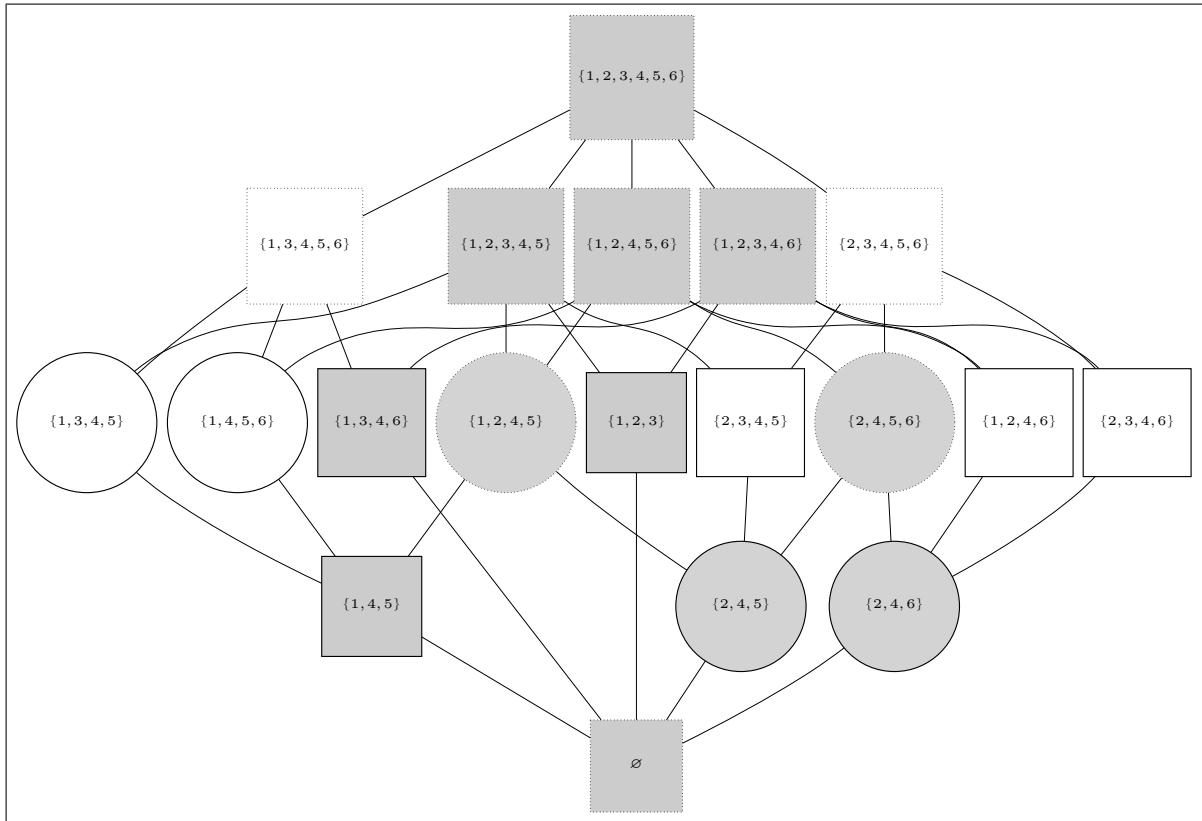


Abbildung 3.4.: Das Innere von $L = L^{01}$ zu `adenin_compressed`.

Definiere nun $\varphi_{\leq a}$ wie folgt:

$$\varphi_{\leq a} : L \rightarrow \{0, 1\},$$

$$a_1 \mapsto \begin{cases} 0 & a_1 \leq a, \\ 1 & \text{sonst.} \end{cases}$$

Man sieht durch die Idempotenz von \circ schnell, dass $\varphi_{\leq a}$ tatsächlich ordnungserhaltend ist. Zudem gilt:

$$a \circ L = P_{\varphi_{\leq a} \leq 0}.$$

Also ist $a \circ L$ Verbandsstumpf.

Damit greift Satz 3.5 und es folgt sofort:

$$e \in \mathcal{M}(a \circ L) \Leftrightarrow e \in \mathcal{M}(L) \cap a \circ L \Leftrightarrow e \in \mathcal{M}(L) \wedge e \leq a,$$

$$b \in \mathcal{J}(a \circ L) \Leftrightarrow b \in \mathcal{J}(L) \cap a \circ L \Leftrightarrow b \in \mathcal{J}(L) \wedge b \leq a.$$

Damit sind wir so gut wie fertig.

Aus Bemerkung 3.12 wissen wir $U_1 \cap U_2 = (a_1 \circ a_2) \circ L$. Durch die Infimumseigenschaft ergibt

sich abschließend:

$$\begin{aligned} e \in \mathcal{M}(\mathbf{U}_1 \cap \mathbf{U}_2) &\Leftrightarrow e \in \mathcal{M}(\mathbf{L}) \wedge e \leq \mathbf{a}_1 \circ \mathbf{a}_2 \\ &\Leftrightarrow e \in \mathcal{M}(\mathbf{L}) \wedge e \leq \mathbf{a}_1 \wedge e \leq \mathbf{a}_2 \\ &\Leftrightarrow e \in \mathcal{M}(\mathbf{U}_1) \cap \mathcal{M}(\mathbf{U}_2), \end{aligned}$$

$$\begin{aligned} \mathbf{b} \in \mathcal{J}(\mathbf{U}_1 \cap \mathbf{U}_2) &\Leftrightarrow \mathbf{b} \in \mathcal{J}(\mathbf{L}) \wedge \mathbf{b} \leq \mathbf{a}_1 \circ \mathbf{a}_2 \\ &\Leftrightarrow \mathbf{b} \in \mathcal{J}(\mathbf{L}) \wedge \mathbf{b} \leq \mathbf{a}_1 \wedge \mathbf{b} \leq \mathbf{a}_2 \\ &\Leftrightarrow \mathbf{b} \in \mathcal{J}(\mathbf{U}_1) \cap \mathcal{J}(\mathbf{U}_2). \end{aligned}$$



□

§3.4 Homomorphismen

Wie für die meisten algebraischen Strukturen gibt es auch für Verbände eine Homomorphismentheorie. Viele Aussagen lassen sich analog zu anderen Strukturen übernehmen – wie der Homomorphiesatz über Gruppen oder Vektorräume. Wir wollen in diesem Abschnitt nicht ins Detail gehen, sondern verweisen noch einmal auf die eingängige Literatur (z. B. Davey & Priestley 1990, Nation 2012). Stattdessen wollen wir weiter bloß auf Aussagen eingehen, die wir biologisch anwenden können.

Im vergangenen Abschnitt haben wir gesehen, dass es durchaus noch Fälle gibt, wo wir zwischen Halbverband und Verband unterscheiden müssen. So konnte eine Teilmenge $\mathbf{U} \subseteq \mathbf{L}$ durchaus die Operation $+$ von \mathbf{L} erben, also einen Unterhalbverband stellen, ohne über die gleiche Operation \circ wie \mathbf{L} zu verfügen.

Biologisch anwendbar ist vor allem die $+$ Operation, so dass es uns nicht weiter stört, wenn die zweite Operation sich ändert. Deshalb starten wir nun zunächst mit der Definition eines Halbverbandshomomorphismus – der uns in den meisten Fällen genügen wird.

Definition 3.15 (Halbverbandshomomorphismus).
Seien L_1, L_2 endliche Verbände und $\varphi : L_1 \rightarrow L_2$. Es gelte weiterhin:

$$\forall \mathbf{a}_1, \mathbf{a}_2 \in L_1 : \varphi(\mathbf{a}_1 + \mathbf{a}_2) = \varphi(\mathbf{a}_1) + \varphi(\mathbf{a}_2).$$

Dann heißt φ Halbverbandshomomorphismus.

φ ist Verbandshomomorphismus, wenn zudem gilt:

$$\forall \mathbf{a}_1, \mathbf{a}_2 \in L_1 : \varphi(\mathbf{a}_1 \circ \mathbf{a}_2) = \varphi(\mathbf{a}_1) \circ \varphi(\mathbf{a}_2).$$

**Homomor-
phismus**

ordnungs-
erhaltend



Bemerkung 3.15 (Ordnungserhaltung).

Jeder Verbandshomomorphismus ist auch Halbverbandshomomorphismus und jeder Halbverbandshomomorphismus ist ordnungserhaltend.

Da die Ordnungserhaltung wichtig ist, sei sie hier noch einmal formalisiert. Man beachte, dass sie von der Schreibweise her den monoton steigenden Funktionen $\varphi : L \rightarrow \mathbb{R}$ gleichen, die wir zur Definition des Kegelstumpfes genutzt haben.



Bemerkung 3.16.

Sei φ Halbverbandshomomorphismus. Dann ist φ ordnungserhaltend, denn:

$$a_1 \leq a_2 \Rightarrow \varphi(a_1) \leq \varphi(a_2).$$

ordnungs-
erhaltend



Beispiel 3.12 (Hauptideale verhalten sich nur ordnungserhaltend).

Sei L Verband und a in L beliebig. Es sei $\psi_a : L \rightarrow a \circ L, a_1 \mapsto a \circ a_1$.

Dann ist ψ_a ordnungserhaltend. Dies ergibt sich durch Dualität aus Lemma 3.2.

Als Gegenbeispiel zur Homomorphieeigenschaft verweisen wir wieder auf den logischen Verband zum Netzwerk *adenin.compressed* in Abbildung 2.4. Denn es gilt:

$$\begin{aligned} \{2, 3, 4, 6\} \circ (\{2, 4, 6\} + \{1, 2, 3\}) &= \{2, 3, 4, 6\} \circ \{1, 2, 3, 4, 6\} = \{2, 3, 4, 6\}, \\ (\{2, 3, 4, 6\} \circ \{2, 4, 6\}) + (\{2, 3, 4, 6\} \circ \{1, 2, 3\}) &= \{2, 4, 6\} + 0 = \{2, 4, 6\}. \end{aligned}$$

Ähnlich dem Vektorraumhomomorphismus, der eindeutig durch die Bilder einer Basis definiert ist, reicht es beim Verband, sich die Bilder der unzerlegbaren Elemente anzuschauen:



Bemerkung 3.17.

φ ist eindeutig bestimmt durch die Bilder von $b \in \mathcal{J}(L)$.



Beispiel 3.13 (Halbverbandshomomorphismus).

In Abbildung 3.5 ist ein Halbverbandshomomorphismus φ zu sehen.

Es handelt sich jedoch nicht um einen Verbandshomomorphismus, da:

$$\begin{aligned} \varphi(\{1, 3, 4, 6\} \circ \{1, 2, 4, 6\}) &= \varphi(0) = 0, \\ \varphi(\{1, 3, 4, 6\}) \circ \varphi(\{1, 2, 4, 6\}) &= \{1, 3\} \circ \{1, 2\} = \{1\}. \end{aligned}$$

Surjektive Homomorphismen erkennt man sehr einfach, denn:



Lemma 3.9.

Sei $\varphi : L_1 \rightarrow L_2$ Halbverbandshomomorphismus.

$$\varphi \text{ surjektiv} \Leftrightarrow \mathcal{J}(L_2) \subseteq \varphi(\mathcal{J}(L_1)).$$

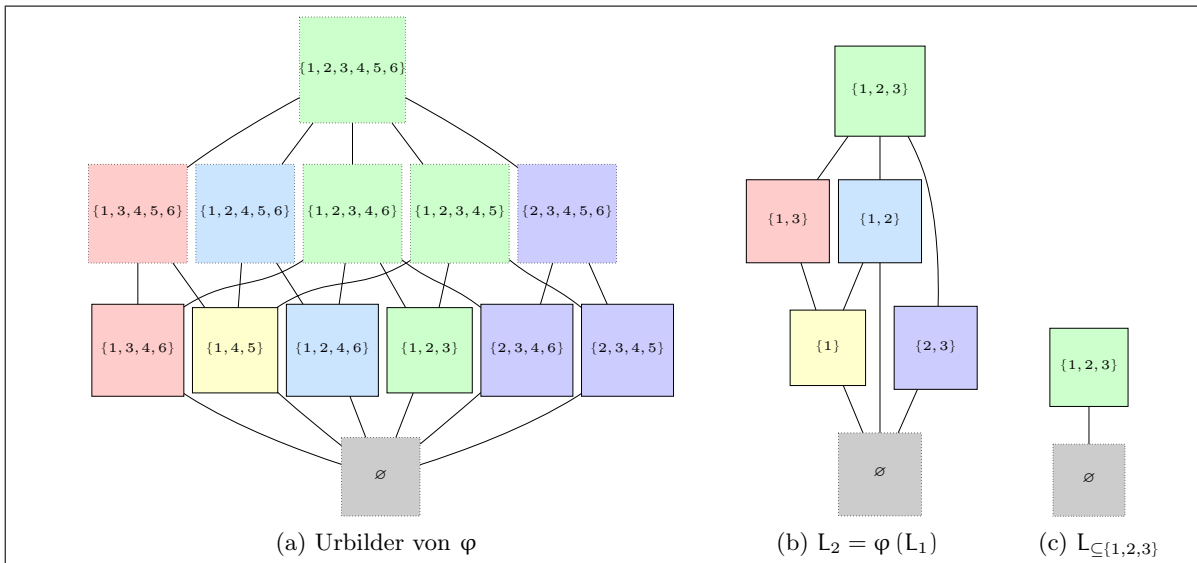


Abbildung 3.5.: Ein Verbandshomomorphismus $\varphi = P_{\{1,2,3\}}$ zu `adenin_compressed`.

Beweis. Sei $\tilde{a} = \tilde{b}_1 + \dots + \tilde{b}_t \in L_2$ mit $\tilde{b}_i \in \mathcal{J}(L_2) \subseteq \varphi(\mathcal{J}(L_1))$. Sei $\tilde{b}_i = \varphi(b_i)$ für $i = 1, \dots, t$. Somit:

$$\tilde{a} = \varphi(b_1) + \dots + \varphi(b_t) = \varphi(b_1 + \dots + b_t) \Rightarrow \tilde{a} \in \varphi(L_1).$$

Sei nun φ surjektiver Homomorphismus. Angenommen $\tilde{b} \in \mathcal{J}(L_2) \setminus \varphi(\mathcal{J}(L_1))$.

Es ist $\tilde{b} \in \mathcal{J}(L_2)$, somit gilt für jede Zerlegung $\tilde{b} = \tilde{b}_1 + \dots + \tilde{b}_t$ o.B.d.A. $\tilde{b}_1 = \tilde{b}$.

Da φ surjektiv ist, lässt sich schreiben $\tilde{b} = \varphi(a)$ mit $a = b_1 + \dots + b_t$, wobei $b_1, \dots, b_t \in \mathcal{J}(L_1)$. Nun gilt jedoch:

$$\tilde{b} = \varphi(a) = \varphi(b_1 + \dots + b_t) = \varphi(b_1) + \dots + \varphi(b_t).$$

Wegen $\tilde{b} \notin \varphi(\mathcal{J}(L_1))$ gilt dann aber $\tilde{b} \neq \varphi(b_i)$ für alle $i \in \{1, \dots, t\}$. Widerspruch. □

➡ Korollar 3.2.

φ Isomorphismus $\Rightarrow \mathcal{J}(L_2) = \varphi(\mathcal{J}(L_1))$.

Doch nun genug der theoretischen Grundlagen im allgemeinen Verband. Im nächsten Kapitel wollen wir uns Verbände über Mengensysteme konzentrieren. Schließlich geht es ja in dieser Arbeit darum, mit dieser Struktur den Stoffwechsel zu modellieren.



Kapitel 3 – Kurz und knapp

Ordnung

- ▷ $+$ ist Supremum, \circ Infimum.
- ▷ Minimale/unzerlegbare Elemente *beschreiben* den Verband.
- ▷ Maximum 1_L und \circ lassen sich *als Summen* schreiben.

Unterstrukturen

- ▷ Element*eigenschaften* definieren oft Unterstrukturen.
- ▷ Verbandstümpfe und *Hauptideale* lassen sich gut beschreiben.

Homomorphismen

- ▷ Homomorphismen sind *ordnungserhaltend*.
- ▷ $\varphi(\mathcal{J}(L))$ definiert Homomorphismen eindeutig.

KAPITEL 4

Vereinigungsverbände

Kapitelübersicht

4.1 Allgemeine Eigenschaften	54
4.1.1 Bedingte Unterhalbverbände	55
4.1.2 Projektionen	59
4.2 Überdeckungen	61
4.2.1 Exkurs: Warum keine Graphentheorie	61
4.2.2 Back to Metabolism: Ein wenig Graphentheorie	62
4.3 Kegelverbände	67
4.3.1 Polyedrische Verbände	67
4.3.2 Konische Verbände	70

Wir beschäftigen uns mit Verbandstheorie, weil es das Mittel der Wahl zur Beschreibung des Stoffwechsels ist. Zwar gibt es auch andere mathematische Theorien, die man bemühen kann¹, doch keine andere ist so intuitiv. Ein Stoffwechselfad ist eine Teilmenge der Reaktionen. Zwei Stoffwechselfade lassen sich zu einem neuen kombinieren. Das bildet einen Verband *and Bob's your uncle*.

Im vorangegangenen Kapitel haben wir allgemeine Verbandstheorie betrachtet, wie man sie für jeden endlichen Verband betreiben kann. Beim Stoffwechsel wissen wir aber, dass es sich um eine bestimmte Art von Verband handelt: Die Elemente sind immer Teilmenge von $[n]$, die Hauptoperation $+$ entspricht der Mengenvereinigung.


Solche Verbände heißen Vereinigungsverbände und haben spezielle Eigenschaften. Die Beispiele, die wir bisher kennengelernt haben, gehören alle in diese Kategorie. Schauen wir jetzt an, was uns die Theorie zu bieten hat.

¹Etwas orientierte Matroide, s. Oliveira, Bailey, Jones-Oliveira & Dixon (2001).

§4.1 Allgemeine Eigenschaften


Wir beginnen mit der formalen Definition des Vereinigungsverbandes, der uns erlaubt, auf Reaktionen basierende Stoffwechselfade abstrakt zu betrachten.

Vereinigungs-
verband

 **Definition 4.1 (Vereinigungsverband).**
 Sei $L \subseteq 2^{[n]}$ vereinigungsstabiles Mengensystem, das die leere Menge enthält. Dann heißt $(L, +, 0)$ *Vereinigungsverband* mit $0 = \emptyset$ als neutralem Element und


$$a_1 + a_2 := a_1 \cup a_2.$$

Mit dieser ordentlichen Definition ist es nicht mehr weit zu folgender Beobachtung:

 **Bemerkung 4.1.**
 Sei L Vereinigungsverband. Dann:


$$a_1 \leq a_2 \Leftrightarrow a_1 \subseteq a_2.$$

Jetzt können wir Bemerkung 3.9 noch weiter vereinfachen:

 **Bemerkung 4.2.**
 Seien $a_1, a_2 \in L$, dann:

$$a_1 \circ a_2 = \sum_{a \subseteq a_1 \cap a_2} a.$$

Wir haben für allgemeine Verbände schon die Unterhalbverbandsbeziehung für $U \subseteq L$ vorgestellt. Bei Vereinigungsverbänden gilt in Abwandlung von Lemma 3.7 Folgendes:²

 **Lemma 4.1.**
 Seien $U \subseteq L$ Vereinigungsverbände über $[n]$, U Unterhalbverband von L . Dann:

$$\forall e \in \mathcal{M}(U) \exists \tilde{e} \in \mathcal{M}(L) : \tilde{e} \subseteq e,$$

und außerdem:


$$1_U \subseteq 1_L.$$

²Auch wenn wir den ersten Teil bisher nicht offiziell vorgestellt haben, so gilt er natürlich auch für allgemeine Verbände. Man substituiere in diesem Falle lediglich das \subseteq durch die Verbandsordnung \leq .

§4.1.1 Bedingte Unterhalbverbände

Vereinigungshalbverbände haben sehr unterschiedliche motivierte Unterhalbverbänden. So gut wie alles, was mit der An- oder Abwesenheit bestimmter Zahlen $R \subseteq [n]$ zu tun hat, ist möglich und kann miteinander kombiniert werden. Hier wollen wir nun die wichtigsten Beispiele kennenlernen.

Wir starten mit einem Verwandten des Inneren L° , dem Abschluss.




Definition 4.2 (Abschluss).
 Sei L Vereinigungsverband über $[n]$.

$$\bar{L} := \{ a \in 2^{[n]} \mid \exists e \in L \setminus \{0\} : e \subseteq a \} \cup \{0\}.$$

\bar{L} heißt *Abschluss* von L .

\bar{L}

Ebenso wie L° ist \bar{L} eindeutig durch die minimalen Elemente definiert:




Bemerkung 4.3.
 \bar{L} ist Vereinigungsverband und es ist L Unterhalbverband von \bar{L} . Ferner ist:

$$\mathcal{J}(\bar{L}) = \{ e \cup \{r\} \mid e \in \mathcal{M}(L), r \in [n] \}.$$

 $\mathcal{J}(\bar{L})$

Nach dem generellen Abschluss, der unabhängig von speziellen Zahlen ist, lernen wir nun einige dieser Eigenschaftsverbände kennen, die sich aus speziellen Mengen $R \subseteq [n]$ ergeben.




Definition 4.3 (L entlang R).
 Sei L Vereinigungsverband über $[n]$ und $R \subseteq [n]$.

$$L_{\parallel R} := \{ a \in L \mid a \cap R \neq \emptyset \} \cup \{0\}$$

$L_{\parallel R}$ heißt *L entlang R*.

 $L_{\parallel R}$

Zu $L_{\parallel R}$ gibt es noch einen kleinen Bruder:




Definition 4.4 (L über R).
 Sei L Vereinigungsverband über $[n]$ und $R \subseteq [n]$.

$$L_{\supseteq R} := \{ a \in L \mid a \cap R = R \} \cup \{0\}$$

$L_{\supseteq R}$ heißt *L über R*.


 $L_{\supseteq R}$

Folgende Beobachtung hilft uns, Eigenschaften von $L_{\parallel R}$ auf $L_{\supseteq R}$ zu übertragen:

 **Bemerkung 4.4.**
 Sei L Vereinigungsverband, $R \subseteq [n]$. Dann:

$$L_{\supseteq R} = \bigcap_{r \in R} L_{\parallel \{r\}}.$$


Wir werden später vor allem auf diesen Unterverband stoßen:

$L_{\perp R}$  **Definition 4.5 (L ohne R).**
 Sei L Vereinigungsverband über $[n]$ und $R \subseteq [n]$.


$$L_{\perp R} := \{a \in L \mid a \cap R = \emptyset\}.$$

$L_{\perp R}$ heißt L ohne R .

Bei genauer Betrachtung bemerkt man:


 **Bemerkung 4.5 ($L_{\perp R}$ ist Hauptideal).**
 Es ist $L_{\perp R} = 1_{L_{\perp R}} \circ L$.

Zwilling von L ohne R ist L in R :

$L_{\subseteq R}$  **Definition 4.6 (L in R).**
 Sei L Vereinigungsverband über $[n]$ und $R \subseteq [n]$.

$$L_{\subseteq R} := L_{\perp [n] \setminus R} = \{a \in L \mid a \subseteq R\}.$$

$L_{\subseteq R}$ heißt L in R .

 **Beispiel 4.1.**
 Abbildung 4.1 zeigt Unterstrukturen des Flussverbandes zu *adenin_compressed*.
 Wir sehen $L_{\parallel \{6\}}$ (4.1a), gefolgt von $L_{\supseteq \{1,6\}}$ (4.1b).
 Der Verband $L_{\perp \{4\}}$ besteht aus nur einem nicht-trivialen Element (4.1c). Wir werden
 damit später sehen, dass es die Kopplungen $4 \xrightarrow{=0} 5$ und $4 \xrightarrow{=0} 6$ gibt.
 Abschließend sehen wir $L_{\subseteq \{1,2,3,4,5\}} = L_{\perp \{6\}}$ (4.1d).

Wo $L_{\parallel R}$ bloß verlangt, dass irgendeine Zahl aus R vertreten sein muss, verbietet $L_{\subseteq R}$ den ganzen Rest. Was lässt sich noch über die Beziehungen dieser Verbände sagen?

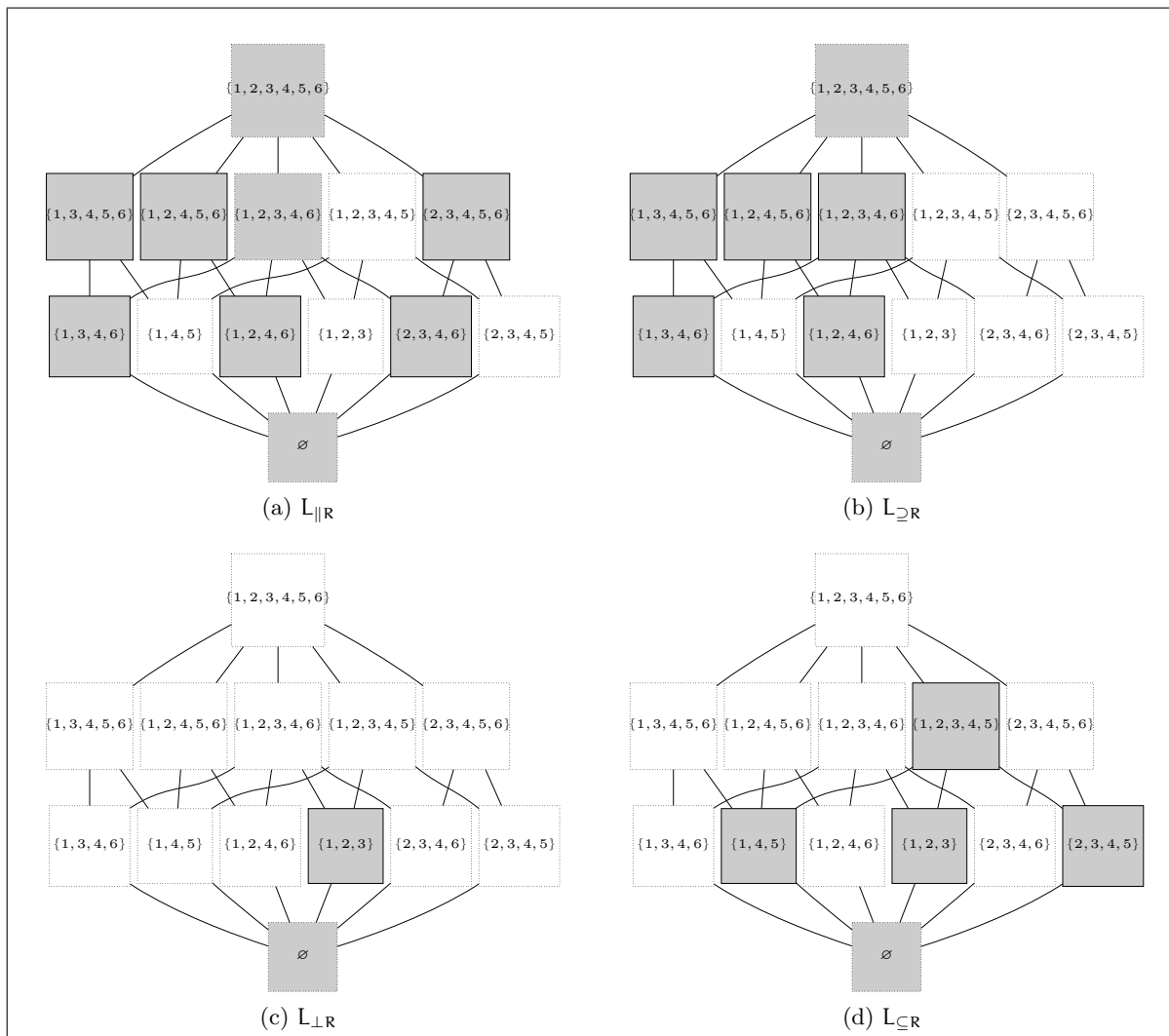


Abbildung 4.1.: Einige Unterstrukturen des Flussverbands zu `adenin.compressed`.



Bemerkung 4.6.

$L_{||R}$ ist Unterhalbverband, $L_{\perp R}$, $L_{\subseteq R}$ sind Unterverbände von L . Zudem ist $L_{\subseteq R} \subseteq L_{||R}$ Unterverband.

Äußerst hilfreich ist das folgende Lemma, das uns Konkatination erlaubt:



Lemma 4.2.

★ Sei $U \subseteq L$ Unter(halb)verband mit $\emptyset \in U$. Dann sind auch $U_{||R} \subseteq L_{||R}$, $U_{\perp R} \subseteq L_{\perp R}$ und $U_{\subseteq R} \subseteq L_{\subseteq R}$ Unter(halb)verbände.

Die Konkatinationen dieser Eigenschaften lassen sich vereinfacht ausdrücken:

**Bemerkung 4.7.**

Sei L Vereinigungsverband über $[n]$ und $R_1, R_2, R_3 \subseteq [n]$. Dann:

$$\begin{aligned} (L_{\parallel R_1})_{\parallel R_2} &= L_{\parallel (R_1 \cap R_2)}, \\ (L_{\perp R_1})_{\perp R_2} &= L_{\perp (R_1 \cup R_2)}, \\ (L_{\subseteq R_1})_{\subseteq R_2} &= L_{\subseteq (R_1 \cap R_2)}. \end{aligned}$$

Außerdem gilt:

$$L_{\subseteq R_1 \perp R_2 \parallel R_3} := \left((L_{\subseteq R_1})_{\perp R_2} \right)_{\parallel R_3} = \left((L_{\subseteq R_1})_{\parallel R_3} \right)_{\perp R_2} = \cdots = \left((L_{\parallel R_3})_{\perp R_2} \right)_{\subseteq R_1}$$

Stellen wir uns nun den Eigenschaften. Wir starten mit $L_{\subseteq R}$, da sich hiermit die Zweitoperation im Oberverband \circ für Vereinigungsverbände besser beschreiben lässt.

**Lemma 4.3** (\circ im Vereinigungsverband).

Sei $L \subseteq [n]$ Vereinigungsverband und $\mathfrak{a}, \mathfrak{a}_1, \mathfrak{a}_2 \in L$. Dann:

$$\begin{aligned} \mathfrak{a}_1 \circ \mathfrak{a}_2 &= 1_{L_{\subseteq (\mathfrak{a}_1 \cap \mathfrak{a}_2)}}, \\ \mathfrak{a} &= 1_{L_{\subseteq \mathfrak{a}}}. \end{aligned}$$

Beweis. Das Zweite folgt aus dem Ersten, da $\mathfrak{a} = \mathfrak{a} \circ \mathfrak{a}$ und $\mathfrak{a} = \mathfrak{a} \cap \mathfrak{a}$.

Um den Rest zu zeigen übernehmen wir die Vereinfachung von \circ aus Bemerkung 4.2 und erhalten durch Kombination mit Lemma 3.1:

$$\mathfrak{a}_1 \circ \mathfrak{a}_2 = \sum_{\mathfrak{a} \subseteq (\mathfrak{a}_1 \cap \mathfrak{a}_2)} \mathfrak{a} = \sum_{\mathfrak{a} \in L_{\subseteq (\mathfrak{a}_1 \cap \mathfrak{a}_2)}} \mathfrak{a} = 1_{L_{\subseteq (\mathfrak{a}_1 \cap \mathfrak{a}_2)}}.$$

Alternativ kann man auch über Hauptideale argumentieren. □

Gehen wir nun zu den minimalen und den unzerlegbaren Elementen über:

**Lemma 4.4.**

Es gilt:

$$\begin{aligned} \mathcal{M}(L_{\parallel R}) &\supseteq \mathcal{M}(L) \cap L_{\parallel R}, & \mathcal{J}(L_{\parallel R}) &\supseteq \mathcal{J}(L) \cap L_{\parallel R}, \\ \mathcal{M}(L_{\perp R}) &= \mathcal{M}(L) \cap L_{\perp R}, & \mathcal{J}(L_{\perp R}) &= \mathcal{J}(L) \cap L_{\perp R}, \\ \mathcal{M}(L_{\subseteq R}) &= \mathcal{M}(L) \cap L_{\subseteq R}, & \mathcal{J}(L_{\subseteq R}) &= \mathcal{J}(L) \cap L_{\subseteq R}. \end{aligned}$$

\mathcal{M}, \mathcal{J}

Beweis. Die Aussagen für $L_{\subseteq R}$ folgen aus $L_{\subseteq R} := L_{\perp [n]} \setminus R$.

Der Rest folgt aus Satz 3.4 über Unterverbände mit Eigenschaften bzw. Bemerkung 3.14 über Hauptideale. \square



Beispiel 4.2.

Es ist $\{1, 3, 4, 5, 6\} \in \mathcal{J}(L_{\parallel\{6\}}) \setminus \mathcal{J}(L)$ für den Flussverband von *adenin_compressed* (s. Abbildung 4.1).

Zum Abschluss wollen wir uns noch einmal mit $L_{\supseteq R}$ beschäftigen. Bisherige Aussagen lassen sich dank Bemerkung 4.4 und Lemma 3.8 leicht anpassen. Da jedoch im Allgemeinen $\mathcal{M}(L_{\supseteq R}) \neq \mathcal{M}(L) \cap \mathcal{P}_{\supseteq R}$ gilt, ist folgender Satz interessant:



Satz 4.1 (kleinstes Element in $L_{\supseteq R}$).

Sei L Vereinigungsverband über $[n]$ und $R \subseteq [n]$, $|R| = k$.

Sei weiterhin $\mathbf{a}^* \in L_{\supseteq R} \setminus \{0\}$, mit $|\mathbf{a}^*| \leq |\mathbf{a}|$ für alle $\mathbf{a} \in L_{\supseteq R} \setminus \{0\}$.

Dann gibt es $\mathbf{b}_1, \dots, \mathbf{b}_k \in \mathcal{J}(L)$ mit $\mathbf{a}^* = \mathbf{b}_1 + \dots + \mathbf{b}_k$.

Beweis. Betrachte die Zerlegung $\mathbf{a}^* = \mathbf{b}_1 + \dots + \mathbf{b}_t$ mit $\mathbf{b}_i \in \mathcal{J}(L)$. Das geht, da $\mathbf{a}^* \neq 0$.

Da $\mathbf{a}^* \in L_{\supseteq R}$, gibt es $i_r \in [t]$ mit $r \in \mathbf{b}_{i_r}$ für $r \in R$ beliebig.

Betrachte nun $\tilde{\mathbf{a}} = \mathbf{b}_{i_1} + \dots + \mathbf{b}_{i_k}$. Da $\tilde{\mathbf{a}}$ Summe von unzerlegbaren Elementen ist, die gemeinsam \mathbf{a}^* ergeben, gilt $0 \neq \tilde{\mathbf{a}} \leq \mathbf{a}^*$, also $\tilde{\mathbf{a}} \subseteq \mathbf{a}^*$ und $|\tilde{\mathbf{a}}| \leq |\mathbf{a}^*|$. Da nun aber $\tilde{\mathbf{a}} \in L_{\supseteq R} \setminus \{0\}$ nach Konstruktion, gilt $|\mathbf{a}^*| \leq |\tilde{\mathbf{a}}|$. Also $\mathbf{a}^* = \tilde{\mathbf{a}}$. \square

§4.1.2 Projektionen

Schließlich wollen wir uns noch mit Projektionen von Vereinigungsverbänden auf Teilmengen R von $[n]$ beschäftigen:



Definition 4.7 (Projektion auf R).


Sei L Vereinigungshalbverband über $[n]$ und $R \subseteq [n]$. Sei nun P_R wie folgt:


$$P_R : \quad L \rightarrow 2^R, \\ \mathbf{a} \mapsto \mathbf{a} \cap R.$$

P_R heißt *Projektion* von L auf R und ist Halbverbandshomomorphismus. Ferner sei L_R der von L auf R projizierte Halbverband:

$$L_R := P_R(L)$$

Projektion

 **Beispiel 4.3** (Projektion).
 ⚡ Der in Beispiel 3.13 vorgestellte Homomorphismus ist eine Projektion. (Abbildung 3.5)

 **Bemerkung 4.8.**
 Es ist $\mathcal{M}(L_R) \subseteq P_R(\mathcal{M}(L))$ und $\mathcal{J}(L_R) \subseteq P_R(\mathcal{J}(L))$ aber im Allgemeinen weder $\mathcal{J}(L_R) \supseteq P_R(\mathcal{J}(L))$ noch $\mathcal{M}(L_R) \supseteq P_R(\mathcal{M}(L))$. Zudem ist L_R im Allgemeinen nicht atomar, auch wenn L atomar war.

Beweis. Die Positivaussagen sind trivial.


Bei der in Abbildung 3.5 abgebildeten Projektion erkennt man ganz deutlich, dass $P_R(\mathcal{J}(L)) = L_R \setminus \{0\}$ gilt. Insbesondere ist $\{1, 2, 3\} = P_R(\{1, 2, 3\})$ zerlegbar in L_R aber nicht in L .

Wir sehen gleich, dass L_R nicht atomar ist. Wir halten fest:

$$\mathcal{M}(L_R) = \{\{2, 3\}\} \subsetneq \mathcal{J}(L_R) = \{\{1\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\} \subsetneq P_R(\mathcal{M}(L)) = P_R(\mathcal{J}(L)).$$

Daraus folgt die ganze Aussage. □

Schauen wir uns an, was passiert, wenn wir den neuen Homomorphismus Projektion mit den bereits vorgestellten Eigenschaftsunterstrukturen verbinden.

 **Bemerkung 4.9.**

$$\begin{aligned} L_R &= (L_{\parallel R})_R = (L_R)_{\parallel R}, \\ L_{\subseteq R} &= (L_{\subseteq R})_R \subseteq L_R = (L_R)_{\subseteq R}. \end{aligned}$$

 **Lemma 4.5.**
 ✦ Sei $\mathcal{U} \subseteq L$ Unter(halb)verband mit $\emptyset \in \mathcal{U}$. Dann ist auch $\mathcal{U}_R \subseteq L_R$ Unter(halb)verband.

 **Korollar 4.1.**
 Es ist $L_{\subseteq R} \subseteq L_R$ Unterverband.

Beweis. Nach Bemerkung 4.6 ist $L_{\subseteq R} \subseteq L_{\parallel R}$ Unterverband.

Somit ist nach Lemma 4.5 auch $(L_{\subseteq R})_R \subseteq (L_{\parallel R})_R$ Unterverband.

$$L_{\subseteq R} \stackrel{\text{Bem. 4.9}}{=} (L_{\subseteq R})_R \stackrel{\text{UV}}{\subseteq} (L_{\parallel R})_R \stackrel{\text{Bem. 4.9}}{=} L_R$$

□



Bemerkung 4.10.

Es ist $\mathcal{M}(L_{\subseteq R}) \subseteq P_R(\mathcal{M}(L))$, da $L_{\subseteq R} = (L_{\subseteq R})_R$.

Im Allgemeinen gilt jedoch weder $\mathcal{M}(L_{\subseteq R}) \subseteq \mathcal{J}(L_R)$ noch $\mathcal{M}(L_R) \subseteq \mathcal{J}(L_{\subseteq R})$ (s. Abbildung 3.5b im Vergleich zu 4.1a).

Fassen wir alle Ergebnisse zusammen, so erhalten wir die Übersicht 4.2.

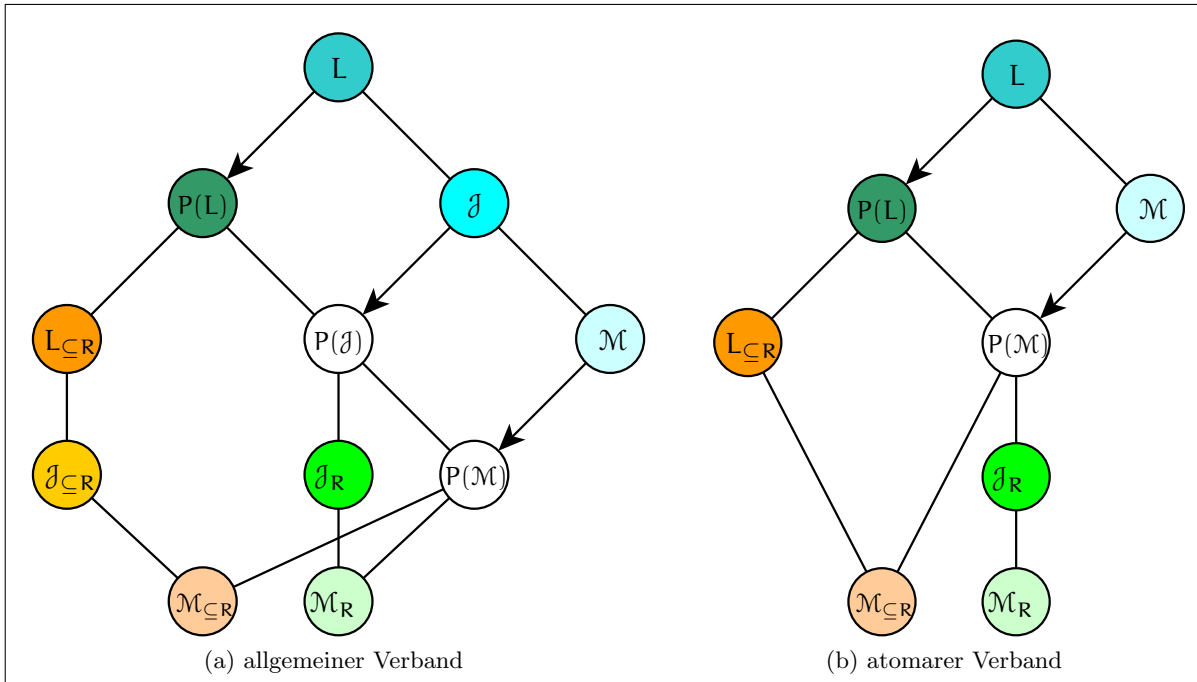


Abbildung 4.2.: Beziehungen zwischen Verband L , Projektion $P_R(L)$ und Untersystem $L_{\subseteq R}$.

Führen wir uns das Bild vor Augen, so sehen wir, dass die minimalen/unzerlegbaren Elemente von $L_{\subseteq R}$ wenig mit den entsprechenden Mengen von L_R zu tun haben müssen. Das ist brisant, da wir meistens nicht alle Reaktionen einer Zelle kennen, sondern nur $R \subseteq [n^*]$, mit n^* der echten Anzahl Reaktionen. Wir wollen so gut es geht L_R verstehen, beschreiben aber in Wirklichkeit das theoretische Konstrukt $L_{\subseteq R}$. Darauf kommen wir später noch einmal zurück, dürfen es aber nie vergessen.

§4.2 Überdeckungen

§4.2.1 Exkurs: Warum keine Graphentheorie

Erklärt man einem Biologen das Thema dieser Arbeit, so spricht man am besten vom Stoffwechsel von Zellen, den man mathematisch modellieren will. Bald fällt das Stichwort „Stoffwechselfad“ und kurz darauf werden die vielfältigen Anwendungsprobleme erklärt, die man mittels Verbandsmodell lösen kann.

Anders einem Mathematiker: Auch hier mag man erwähnen, dass es um den Zellstoffwechsel geht, den man per Verbandstheorie untersucht. Aber im gleichen Atemzug fällt eine Erklärung, was das sein soll, Zellstoffwechsel. Statt von Reaktionen und Metaboliten spricht man wohl von einem Hypergraph, vielleicht sogar einem *gewichteten, gerichteten Hypergraph*. Als nächstes zeichnet man ein Beispielnetzwerk und jeder Mathematiker sieht sofort: Ja! Das hat definitiv etwas mit Graphen zu tun.

Stoffwechselnetzwerke müssen irgendwas mit Graphen zu tun haben. Das sieht man sofort, wenn man eine Illustration hat. Trotzdem fällt es sehr schwer, sie mit klassischer Graphentheorie zu beschreiben. Das liegt an den vielen Erweiterungen, die man wählen muss. Ein klassischer Graph kennt nur ungerichtete Verbindungen zwischen zwei Knoten, gerichtete Graphen kennen zumindest das Konzept von „Vorgänger“ und „Nachfolger“.

Reaktionen wandeln aber nicht bloß einen Metaboliten in einen anderen um, sondern tun das manchmal für mehrere auf einmal. Hier könnten Hypergraphen helfen. Und tun sie auch. Das haben Carbonell et al. (2012) getan, als sie ihre [Hyperpfade](#) untersuchten, die wir hier auch schon als Verband $L^{\mathcal{H}}$ vorgestellt haben.

Aber ganz genau genommen gibt es da noch Stöchiometrie, auf die wir zwar auch beim L^{01} verzichten, aber nicht schon in den Grundannahmen ignoriert werden sollte.

Seitdem Euler die Graphentheorie im Jahre 1736 formalisiert hat, ist enorm viel in der Mathematik passiert. Die Vernetzung der Welt durch das Internet, moderne Logistik und viele, viele weitere Bequemlichkeiten unserer heutigen Zivilisation wären nicht denkbar, könnten wir sie nicht vernünftig planen und steuern. Das schafft man oftmals mit Graphentheorie und den rasanten Algorithmen, die Mathematiker und Theoretische Informatiker in den vergangenen Jahrzehnten hervorgebracht haben.

Wäre es nicht schön, nun auch den Stoffwechsel mittels klassischer Graphentheorie beschreiben zu können? Wenn es nicht einen einfachen Algorithmus gäbe, mit dem man ein metabolisches Netzwerk in einen klassischen Graphen umwandeln und dann analysieren könnte?

Es gibt einige Ansätze, die genau dies versuchen. So definieren etwa Planes & Beasley (2009) ihre metabolischen Pfade geschickt so, dass sie Pfade von Ursprungs-(Nähr-)stoffen zu Zielmetaboliten in gerichteten Graphen suchen können.

Hier wollen wir direkt mit Verbandstheorie als Modellierungsgrundsatz der Wahl fortfahren. Das machen wir im nächsten Schritt, indem wir eine weitere Anwendung aus der Hypergraphentheorie übernehmen.

§4.2.2 Back to Metabolism: Ein wenig Graphentheorie


Wie stoppt man den Stoffwechsel einer Zelle? Welche Reaktionen gemeinsam sind so fundamental wichtig, dass ohne sie gar nichts mehr klappt? Wie kann man auf eine Zelle einwirken, um bestimmte Prozesse zu verhindern?

Klamt & Gilles (2004) haben das Konzept der Minimal Cut Sets vorgestellt, haben viel zu

diesen Fragen geforscht und veröffentlicht. Anfangs ging es darum, Wege zu finden, Reaktionen in allen stationären Zuständen zu verhindern oder überhaupt keinen stationären Zustand übrig zu lassen.

Klamt & Gilles's (2004) jahrelange Arbeit hat sich gelohnt, so dass sie mittlerweile raffinierte Verfahren vorstellen konnten, um Minimal Cut Sets zu berechnen – etwa mittels eines dualen Netzwerks (Ballerstein, Kamp, Klamt & Haus 2011). Doch auch Acuña, Chierichetti, Lacroix, Marchetti-Spaccamela, Sagot & Stougie (2009) haben sich mit ihnen beschäftigt.

Auch wir wollen sie ein wenig betrachten und widmen ihnen dafür zwei Abschnitte dieser Arbeit. Folgend stellen wir einige Grundlagen zusammen, die wir in die Sprache der Verbandstheorie übersetzt haben. Dabei weisen wir auch die von Klamt (2006) erwähnte Dualität nach, die wir unter einem neuen Licht betrachten. Viel später – wenn wir Verbandsmaxima rasant berechnen können – werden wir dann ein neues Verfahren vorstellen, wie man das Problem und Verallgemeinerungen angehen kann. Doch beginnen wir nun mit den Grundlagen, die Spezialisten der Hypergraphentheorie leicht anders formuliert wohlbekannt sind.




Definition 4.8 (Überdeckung).
 Sei L Vereinigungsverband über $[n]$ und $c \subseteq [n]$.

$$c \text{ Überdeckung von } L \Leftrightarrow L = L_{\parallel c}.$$

c heißt *minimale Überdeckung*, wenn jede echte Teilmenge von c keine Überdeckung ist.


Über-
deckung

Sowohl die Überdeckungen wie auch die gleich vorgestellten unabhängigen Mengen sind Begriffe aus der Hypergraphentheorie, wie sie Füredi (1988) vorstellt.



Bemerkung 4.11.
 Sei L Vereinigungsverband über $[n]$. So ist $H = ([n], L \setminus \{0\})$ Hypergraph.
 Es ist $c \subseteq [n]$ Überdeckung des Verbandes L genau dann, wenn c Überdeckung des Hypergraphen H ist.

Alle Überdeckungen gemeinsam erzeugen wieder einen Halbverband. Entsprechend ergibt sich ein Hypergraph der Überdeckungen.



Definition 4.9 (Überdeckungshalbverband).
 Sei L Vereinigungsverband über $[n]$. Sei $\bar{C}(L)$ wie folgt definiert:

$$\bar{C}(L) := \{c \subseteq [n] \mid c \text{ Überdeckung von } L\} \cup \{0\}.$$

Dann heißt $\bar{C}(L)$ *Überdeckungsverband* von L .

Will man in ein System so wenig wie möglich eingreifen, so sollte man die minimalen Elemente der Überdeckungen verwenden. Diese erzeugen wieder einen Halbverband. Im Fall der

Überdeckungen steht dieser in einem ganz besonderen Verhältnis zum ursprünglichen Verband – er ist dual.³ Das wollen wir jetzt zeigen.

dualer
Verband



Definition 4.10 (dualer Verband).

Sei $\mathcal{M}^\perp(L) := \mathcal{M}(\overline{C}(L))$ die Menge der minimalen Überdeckungen von L und

$$L^\perp := C(L) := \langle \mathcal{M}^\perp(L) \rangle.$$

Dann heißt L^\perp zu L dualer Verband.

Der duale Verband L^\perp ist durch die minimalen Überdeckungen bestimmt. Zu der Dualität kommt es vor allem deswegen, weil die Menge der Überdeckungen selbst eindeutig durch die minimalen Elemente von L bestimmt werden kann:



Satz 4.2.

Sei L Vereinigungsverband über $[n]$ und $c \subseteq [n]$. Dann:

$$c \text{ Überdeckung von } L \Leftrightarrow \forall e \in \mathcal{M}(L) : e \cap c \neq \emptyset.$$

Beweis. Die Hinrichtung ist klar. Für die Rückrichtung gilt:

$$a \in L \setminus \{0\} \Rightarrow \exists e \in \mathcal{M}(L) : e \leq a \Rightarrow a \cap c = (e \cup a) \cap c \supseteq e \cap c \neq \emptyset.$$

□

Kennen wir die minimalen Elemente eines Verbandes, so können wir alle seine Überdeckungen bestimmen – egal, um welche Art Vereinigungsverband es sich handelt. Insbesondere gilt das für nicht-atomare Verbände, die wir allein aus $\mathcal{M}(L)$ noch nicht rekonstruieren können. Oder sogar für Verbandsstümpfe.



Korollar 4.2.

L^\perp und $\overline{C}(L)$ sind eindeutig durch $\mathcal{M}(L)$ bestimmt.

Eng mit den Überdeckungen eines Hypergraphen verwandt ist das Konzept der unabhängigen Mengen. Auf den Stoffwechsel bezogen sind das die Reaktionsmengen, die keinen gültigen Stoffwechselfad enthalten. Formal für alle Vereinigungsverbände lautet das:

³Nicht aus Sicht von Kategorien, wo man unter dem dualen Verband eine Vertauschung von $+$ und \circ versteht.



Definition 4.11 (Unabhängigkeit).

Sei L Vereinigungsverband über $[n]$ und $i \subseteq [n]$.

$$i \text{ unabhängig von } L \Leftrightarrow i = \emptyset \vee i \notin \bar{L}.$$

i heißt *maximale unabhängige Menge*, wenn \bar{L} jede echte Obermenge von i enthält.

Aus der Hypergraphentheorie übernehmen wir folgendes praktische Lemma:



Lemma 4.6 (Unabhängigkeit und Überdeckung).

Sei L Vereinigungsverband und $c \subseteq [n]$. Sei außerdem $i = [n] \setminus c$. Dann:

$$c \text{ (minimale) Überdeckung von } L \Leftrightarrow i \text{ (maximal) unabhängig von } L.$$

Auch folgendes Lemma ist für Hypergraphen wohlbekannt. Seine Gültigkeit lässt sich durch kurze Überlegung bestätigen. Damit ausgestattet verfügen wir schließlich über alle Hilfsmittel, die wir zum Beweis der Dualität benötigen.



Lemma 4.7.

Sei L Vereinigungsverband und $c \subseteq [n]$ Überdeckung von L . Dann:

$$c \text{ minimale Überdeckung} \Leftrightarrow \forall k \in c \exists a \in L : a \cap c = \{k\}.$$

Nun sind wir in der Lage, die Dualität von den minimalen Elementen zu den minimalen Überdeckungen zu zeigen, wie sie Klant (2006) im Falle des stationären Flussverbandes L^F für elementare Flussmoden und Minimal Cut Sets zeigt.



Satz 4.3 (Dualität von L und L^\perp).

Sei L Vereinigungshalbverband über $[n]$. Dann:

$$\mathcal{M}(L) = \mathcal{M}\left(\left(L^\perp\right)^\perp\right).$$

Dualität

Beweis. Es ist offensichtlich $L \subseteq C(L^\perp)$:

Sei $a \in L, \tilde{a} \in L^\perp$ beliebig. Wegen $\tilde{a} \in L^\perp$ Überdeckung gilt $a \cap \tilde{a} \neq \emptyset$. Da aber $\tilde{a} \in L^\perp$ beliebig gewählt war, sind also alle $a \in L$ Überdeckungen von L^\perp .

Zeigen wir nun $\mathcal{M}(L) \subseteq \mathcal{M}(C(C(L)))$.

Sei $e \in \mathcal{M}(L)$ und $k \in e$ beliebig. Betrachte $i = e \setminus \{k\}$. i ist unabhängig in L , da $\mathcal{M}(L)$

die Antikette der minimalen Elemente ist. Ergänze nun i zu einer maximalen unabhängigen Menge i_{\max} . Beachte, dass $k \notin i_{\max}$ gilt. Somit ist $c = [n] \setminus i_{\max}$ minimale Überdeckung von L und somit $c \in \mathcal{M}^\perp(L)$. Da andererseits aber $e \cap c = \{k\}$ (mit k anfangs beliebig gewählt) und außerdem noch e Überdeckung von L^\perp , ist e eine minimale Überdeckung von $C(L)$ und $e \in \mathcal{M}(C(C(L)))$.

Sei nun $e \in C(C(L))$ minimal, also $c = [n] \setminus e$ (maximale) unabhängige Menge in $C(L)$. Somit ist c insbesondere keine Überdeckung von L . Demnach gibt es $\tilde{a} \in L$ mit $\tilde{a} \cap c = \emptyset$ und damit ein $\tilde{e} \in \mathcal{M}(L)$ mit $\tilde{e} \cap c = \emptyset$. Somit ist $\tilde{e} \subseteq e$. Nun haben wir aber schon gezeigt, dass $\tilde{e} \in \mathcal{M}(C(C(L)))$ und damit auch $\tilde{e} \leq e$ in $C(C(L))$. Da e minimal angenommen war, folgt $\tilde{e} = e$ und somit $e \in \mathcal{M}(L)$.⁴ □

Daraus folgt sofort die Dualität der Verbände:

→ Korollar 4.3.

$$L^\circ = (L^\perp)^\perp = C(C(L)) \quad \wedge \quad \bar{L} = \overline{C(L)}.$$

Als weiteres Korollar ergibt sich eine neue Äquivalenz zur Definition atomarer Vereinigungsverbände:

→ Korollar 4.4.

Sei L Vereinigungsverband über $[n]$, $\mathcal{J}(L)$ und $\mathcal{M}(L)$ die unzerlegbaren bzw. minimalen Elemente von L sowie $B \subseteq L$ Antikette. Dann sind äquivalent:

$$\begin{aligned} \mathcal{M}(L) = \mathcal{J}(L), & \Leftrightarrow L = L^\circ, \\ \Leftrightarrow L = \langle B \rangle, & \Leftrightarrow L = (L^\perp)^\perp. \end{aligned}$$

Hiermit soll der erste Ausflug in die Welt der Überdeckungen enden, bis wir ihnen am Ende dieser Arbeit als Anwendung der Verbandstheorie wiederbegegnen. Dem interessierten Leser, der mehr über Überdeckungen im Hypergraph erfahren will, hilft beispielsweise Füredi (1988), der einen sehr ausführlichen Überblick zu Matchings und Überdeckungen gewährt.

Abschluss unserer Grundlagen zur Verbandstheorie soll ein kleines Juwel sein, das wir schon oft angedeutet haben und nun endlich heben wollen: Der Beweis, dass der Flusskegel aller stationären Zustände im Stoffwechsel tatsächlich einen Verband bildet, und warum das ein ganz besonderer Vereinigungsverband ist.

⁴Ich danke Roman Glebov und der Arbeitsgruppe von Professor Tibor Szabo für diesen eleganten Beweis.


§4.3 Kegelverbände

Wer in der modernen Mathematik nach der Lösung für ein graphenbasiertes Problem sucht, versucht es häufig mit Optimierung über Polyeder. Schrijver (2003) stellt zig Beispiele vor, die sich mittels LPs und MILPs lösen lassen.

Viele der gängigen Untersuchungsmethoden zum Zellstoffwechsel tun genau das. Bei FBA und FCA werden LPs gelöst und die Lösungsvektoren interpretiert. Die elementaren Flussmoden berechnet man mit Variationen der Double Description Method (Motzkin, Raiffa, Thompson & Thrall 1953).

Ein Grund dafür ist die Beschreibung des Stoffwechsels durch stöchiometrische Matrix S und irreversible Reaktionen Irr , die die Arbeit mit einem Kegel, dem Flusskegel F , nahe legen. Doch Polyeder im Allgemeinen und Kegel im Speziellen eignen sich auch exzellent, um mit ihnen Verbandstheorie zu betreiben. Warum, wollen wir in diesem Abschnitt näher beleuchten.

§4.3.1 Polyedrische Verbände



Definition 4.12 (polyedrischer Verband).
Sei $P = \{x \in \mathbb{R}^n \mid Sx = c, Tx \leq d\}$ Polyeder und $L^P \subseteq 2^{[n]}$ wie folgt definiert:

$$L^P := \{\text{supp}(x) \mid x \in P\} \cup \{0\} = \{\emptyset\} \cup \{\text{supp}(x) \mid Sx = c, Tx \leq d\}.$$

Dann ist L^P Vereinigungsverband und heißt polyedrischer Verband zum Polyeder P .

poly-
edrischer
Verband

Es folgt der längst versprochene Beweis:



Satz 4.4.
Sei $P = \{x \in \mathbb{R}^n \mid Sx = c, Tx \leq d\}$. Dann ist L^P definiert wie in 4.12 wirklich Vereinigungsverband.

Beweis. Nach Definition 4.1 ist zu zeigen, dass L^P vereinigungsstabil ist. Wegen $\emptyset \cup a = a$ reicht es, $a_1 \cup a_2 \in L^P$ für $a_1, a_2 \in L^P \setminus \{0\}$ zu zeigen.

Seien also $a_1, a_2 \in L^P \setminus \{0\}$. Dann gibt es $x^{(1)}, x^{(2)} \in P$ mit $a_i = \text{supp}(x^{(i)})$ ($i = 1, 2$). Betrachte nun die Konvexkombinationen $x^{(\lambda)} = \lambda x^{(1)} + (1 - \lambda)x^{(2)}$ für $\lambda \in [0, 1]$. Da $x^{(\lambda)}$ Konvexkombination, gilt $x^{(\lambda)} \in P$. Offensichtlich gilt $\text{supp}(x^{(\lambda)}) \subseteq a_1 \cup a_2$.

Betrachte nun die Komponenten von $x^{(\lambda)}$. Es ist $x_i^{(\lambda)} = 0 \Leftrightarrow \lambda x_i^{(1)} + (1 - \lambda)x_i^{(2)} = 0$.

Sei nun $i \in a_1 \cup a_2$, dann gibt es höchstens ein $\lambda_i \in [0, 1]$, so dass $x_i^{(\lambda_i)} = 0$.

Es gibt also nur maximal n verschiedene Werte für λ , so dass $\text{supp}(x^{(\lambda)}) \subsetneq \mathbf{a}_1 \cup \mathbf{a}_2$. Das bedeutet, dass $\text{supp}(x^{(\lambda)}) = \mathbf{a}_1 \cup \mathbf{a}_2$ für fast alle $\lambda \in [0, 1]$ und somit $\mathbf{a}_1 \cup \mathbf{a}_2 \in L^P$, also L^P wirklich vereinigungsstabil. Da $\emptyset \in L^P$ nach Konstruktion ist L^P Vereinigungsverband. \square

 **Korollar 4.5.**

$L^F = \{\text{supp}(v) \mid v \in F\}$ ist Vereinigungsverband über $[n]$.

Soweit kann man auch für beliebige konvexe Mengen $A \subseteq \mathbb{R}^n$ arbeiten. Uns interessieren aber letztlich Kegel. Die Double Description Method berechnet alle Elementarmoden des Flusskegels in einem einzigen, sehr langsamen Schritt. Figueiredo et al. (2009) berechnen hingegen mittels MILPs schrittweise die sogenannten *Shortest Elementary Flux Modes*. Wir stellen hier nun eine Verallgemeinerung für beliebige Polyederverbände vor, mit der sich die minimalen bzw. die unzerlegbaren Elemente nacheinander berechnen lassen. Figueiredo et al. schreiben den kurzen Elementarmoden eine besonders hohe biologische Bedeutung zu.

Algorithm *CalculateMinimalElements*(S, T, c, d)

Output: $\mathcal{M}(L^P)$.

1. $\mathcal{M}(L) \leftarrow \emptyset$
2. $k \leftarrow 0$
3. **while** is feasible
4. **do**

$$\begin{aligned} &\text{minimize} && \sum_{i=1}^n a_i \\ &\text{s.t.} && S \cdot (x^{(+)} - x^{(-)}) = c \\ &&& T \cdot (x^{(+)} - x^{(-)}) \leq d \end{aligned}$$

$$\begin{aligned} a_i^{(+)} + a_i^{(-)} &= a_i && \forall i \in [n] \\ a_i^{(\pm)} &\leq x_i^{(\pm)} \leq M \cdot a_i^{(\pm)} && \forall i \in [n] \\ a_i^{(\pm)}, a_i &\in \{0, 1\} && \forall i \in [n] \end{aligned}$$

$$\begin{aligned} \sum_{i \in e^{(j)}} a_i &\leq |e^{(j)}| - 1 && \forall j \in [k] \\ \sum_{i=1}^n a_i &\geq 1 \end{aligned}$$

5. $e^{(k+1)} \leftarrow \{i \in [n] \mid a_i = 1\}$
6. $\mathcal{M}(L) \leftarrow \{e^{(k+1)}\} \cup \mathcal{M}(L)$
7. $k \leftarrow k + 1$
8. **return** $\mathcal{M}(L)$

Algorithm *CalculateIrreducibleElements*(S, T, c, d)

Output: $\mathcal{J}(L^P)$.

1. $\mathcal{J}(L) \leftarrow \emptyset$
2. $k \leftarrow 0$
3. **while** is feasible
4. **do**

$$\begin{aligned}
 & \text{minimize} && \sum_{i=1}^n a_i \\
 & \text{s.t.} && S \cdot (x^{(+)} - x^{(-)}) = c \\
 & && T \cdot (x^{(+)} - x^{(-)}) \leq d \\
 & && a_i^{(+)} + a_i^{(-)} = a_i \quad \forall i \in [n] \\
 & && a_i^{(\pm)} \leq x_i^{(\pm)} \leq M \cdot a_i^{(\pm)} \quad \forall i \in [n] \\
 & && a_i^{(\pm)}, a_i \in \{0, 1\} \quad \forall i \in [n] \\
 & && h_i \leq a_i, h_i \in \{0, 1\} \quad \forall j \in [n] \\
 & && \sum_{i \in b^{(j)}} (a_i + h_i) \leq |b^{(j)}| \quad \forall j \in [k] \\
 & && \sum_{i=1}^n h_i \geq 1
 \end{aligned}$$

5. $b^{(k+1)} \leftarrow \{i \in [n] \mid a_i = 1\}$
6. $\mathcal{J}(L) \leftarrow \{b^{(k+1)}\} \cup \mathcal{J}(L)$
7. $k \leftarrow k + 1$
8. **return** $\mathcal{J}(L)$

Beweis. Es sei nur darauf hingewiesen, dass h dafür sorgt, dass man a nicht in schon bekannte Elemente zerlegen kann:

$$\begin{aligned}
 b^{(j)} \subseteq a & \Rightarrow h \cap b^{(j)} = \emptyset, \\
 |b^{(j)} \setminus a| = l & \Rightarrow |h \cap b^{(j)}| \leq l.
 \end{aligned}$$

Hieraus folgt:

$$h \cap \bigcup_{b^j \subsetneq a} b^j = \emptyset.$$


$|h| \geq 1$ fordert also, dass es ein $i \in a$ gibt, dass noch nicht von echt kleineren Verbandselementen getroffen wurde, also ist a unzerlegbar, weil es von minimaler Kardinalität ist. \square

Das Lösen eines MILP ist im Allgemeinen \mathcal{NP} -schwer, so dass MILP-Solver häufig viel längere Laufzeiten als LP-Solver haben (Schrijver 2003). Mit modernen Solvern wie Gurobi (Gurobi

Optimization Inc 2012) lassen sich einzelne Elemente aus $\mathcal{M}(L)$ oder $\mathcal{J}(L)$ meist auch für genomweite Netzwerke mit 1000 und mehr Reaktionen in wenigen Sekunden oder noch schneller berechnen.


§4.3.2 Konische Verbände

Im Falle des klassischen Modells, dem stationären Flussverband, arbeiten wir nicht mit einem Verband, der durch ein beliebiges Polyeder induziert ist. Stattdessen handelt es sich um einen Kegel. Und der produziert ganz besondere Verbände.

 **Definition 4.13** (konischer Verband).
 Sei L^C polyedrischer Verband zum Polyeder C und $C = \{x \in \mathbb{R}^n \mid Ax = 0, x_i \geq 0 \forall i \in I \subseteq [n]\}$ Kegel. Dann heißt L^C konischer Verband oder Kegolverband.

Wir werden gleich zeigen, dass alle Kegolverbände atomar sind. Dafür brauchen wir vor allem folgendes Lemma:

ergänzende
Summanden

 **Lemma 4.8** (ergänzende Summanden in konischen Verbänden).
 * Sei L^C konischer Verband, dann gibt es die *ergänzenden Summanden* aus Lemma 3.4.
 * Zudem lässt sich jeder Vektor $x \in C$ mit $\text{supp}(x) \notin \mathcal{M}(L^C)$ als Summe zweier Vektoren
 * kleineren Trägers schreiben.

Beweis. Die erste Aussage folgt aus dem Beweis der zweiten.

Sei $\emptyset \subsetneq \alpha_1 = \text{supp}(x^{(1)}) \subsetneq \alpha = \text{supp}(x)$.

Für den Fall $\alpha_1 \cap I = \emptyset$ wähle $i_0 \in \alpha_1$ beliebig und $\lambda = x_{i_0}/x_{i_0}^{(1)}$. Ansonsten sei $\lambda = \min\left(\left\{x_i/x_i^{(1)} \mid i \in \alpha_1 \cap I\right\}\right)$ und $i_0 \in \alpha_1 \cap I$ beliebig gewählt mit $x_{i_0}/x_{i_0}^{(1)} = \lambda > 0$.

Definiere $\tilde{x}^{(1)} = \lambda x^{(1)}$ und $x^{(2)} = x - \tilde{x}^{(1)}$. Insbesondere $x = \tilde{x}^{(1)} + x^{(2)}$.

Es gilt $\text{supp}(\tilde{x}^{(1)}) = \alpha_1 \subsetneq \alpha$ und somit $\alpha_2 = \text{supp}(x^{(2)}) \subseteq \alpha$. Es gilt jedoch $\alpha \ni i_0 \notin \alpha_2$ wegen:

$$x_{i_0}^{(2)} = x_{i_0} - \tilde{x}_{i_0}^{(1)} = x_{i_0} - \lambda x_{i_0}^{(1)} = x_{i_0} - \frac{x_{i_0}}{x_{i_0}^{(1)}} \cdot x_{i_0}^{(1)} = x_{i_0} - x_{i_0} = 0.$$

Also $\alpha_2 \subsetneq \alpha$ und $\alpha_1 \cup \alpha_2 = \alpha$. Es reicht nun zu zeigen, dass $x^{(2)} \in C$:


Da C ein Kegel ist, ist (wegen der Wahl $\lambda > 0$ für $\alpha_1 \cap I \neq \emptyset$) auch $\tilde{x}^{(1)} = \lambda x^{(1)} \in C$.

Sei nun $i \in \alpha \cap I$ beliebig. Dann:

$$\begin{aligned}
 A \cdot x^{(2)} &= A \cdot (x - \lambda x^{(1)}) = A \cdot x - \lambda A \cdot x^{(1)} = 0 - \lambda \cdot 0 = 0, \\
 x_i^{(2)} &= x_i - \lambda x_i^{(1)} = x_i - \underbrace{\frac{x_{i_0}}{x_i^{(1)}}}_{\leq x_i/x_i^{(1)}} \cdot \underbrace{x_i^{(1)}}_{\geq 0} \geq x_i - \frac{x_i}{x_i} \cdot x_i^{(1)} = x_i - x_i = 0 \\
 &\Rightarrow x^{(2)} \in C.
 \end{aligned}$$


□

Mit Hilfe dieser sehr starken Eigenschaft konischer Verbände lässt sich zeigen:

 **Satz 4.5.**
Konische Verbände sind atomar aber im Allgemeinen nicht semi-modular, also nicht geometrisch.

Beweis. Dass Kegelverbände atomar sind, folgt aus Lemma 3.4 über ergänzende Summanden. Ein Beispiel für L^P nicht semi-modular ist in Abbildung 2.4 gegeben. □

Folgendes Lemma zeigt nun, dass wir sehr interessant zerlegen können.

 **Lemma 4.9** (Zerlegung ohne Kombination).
 * Sei $C = \{x \in \mathbb{R}^n \mid Ax = 0, x_i \geq 0 \forall i \in I\}$ Kegel und $x, y \in C$ mit $e = \text{supp } y \subsetneq \alpha = \text{supp } x$. Seien weiterhin $i^{(+)}, i^{(-)} \in [n]$ mit $x_{i^{(+)}} \cdot y_{i^{(+)}} > 0$ bzw. $x_{i^{(-)}} \cdot y_{i^{(-)}} < 0$.
 * Für den induzierten Kegel L^C gilt dann:
 * $\forall j \in e \exists a_j \in L^C : j \in a_j < \alpha, a_j \setminus e = \alpha \setminus e,$
 * $\exists B \subseteq \mathcal{J}(L^C) : \alpha = \sum_{b \in B} b, e \not\subseteq b \quad \forall b \in B.$

Zerlegung

Beweis. Die zweite Aussage folgt aus der ersten: Es ist $\alpha = \sum_{j \in e} a_j$ und jede Zerlegung der a_j in unzerlegbare Elemente $B_j \subseteq \mathcal{J}(L^C)$ enthält niemals e als Teilmenge eines Elements. Zeigen wir nun also die erste Aussage.

Sei $j \in e$ beliebig. Dann unterscheiden wir zwischen zwei Fällen:

Sei $x_j \cdot y_j > 0$, also besitzen beide Vektoren in der Komponente j das gleiche Vorzeichen. Dies gilt dann auch insbesondere für $y^{(\lambda)} = x + \lambda y \in C$ mit $\lambda > 0$. Wähle nun λ^* so, dass $y_{i^{(-)}}^{(\lambda^*)} = 0$.

Das geht, da $x_{i(-)} \cdot y_{i(-)} < 0$. Dann ist $a_j = \text{supp}(y^{(\lambda^*)})$ wie gefordert.

Seien nun die Vorzeichen an der Stelle j verschieden, also $x_j \cdot y_j < 0$. Wir konstruieren ähnlich dem ersten Fall ein $y^{(\lambda)} = x - \lambda y$ mit $\lambda > 0$. Da wir diesmal subtrahieren, gilt jedoch $y^{(\lambda)} \in C$ nur für λ genügend klein. Wegen $x_j \cdot y_j < 0$ wissen wir, dass auf jeden Fall $y_j^{(\lambda)} \neq 0$ gilt. Weiter folgt aus $x_{i(+)} \cdot y_{i(+)} > 0$, dass es mindestens ein λ gibt mit $\text{supp } y^{(\lambda)} \subsetneq a$. Sei λ^* das Minimum dieser Menge, so ist $a_j = \text{supp } y^{(\lambda^*)} \subsetneq a$ gültiges Element. Nach Konstruktion ist auch diesmal $e \notin a_j$. und wir sind fertig. \square

Für den Rest des Kapitels benötigen wir den Äquivalenzbegriff, der – bezogen auf Stoffwechselfade – keinen Unterschied zwischen Gesamtgeschwindigkeiten von Reaktionen macht, sondern nur auf Geschwindigkeitsverhältnisse zwischen verschiedenen Reaktionen beachtet.



Definition 4.14 (Äquivalenz).

Sei C Kegel, $x^{(1)}, x^{(2)} \in C \setminus \{0\}$.

$$x^{(1)} \sim x^{(2)} :\Leftrightarrow \exists \lambda \in \mathbb{R} : x^{(1)} = \lambda x^{(2)}$$

Wir sagen, dass $x^{(1)}$ äquivalent ist zu $x^{(2)}$.

Und jetzt kommt die Überraschung, die uns zu einem mächtigen Satz führt.



Lemma 4.10.

Sei L^C konischer Verband, $x^{(1)}, x^{(2)} \in C$ mit $a_1 = \text{supp}(x^{(1)}) = \text{supp}(x^{(2)})$ aber $x^{(1)} \not\sim x^{(2)}$. Dann gibt es $a \in L^C$ mit $0 < a < a_1$.

Beweis. Die Konstruktion von $a = \text{supp}(x)$ aus x_1 und x_2 erfolgt analog zur Konstruktion von $x^{(2)}$ aus x und x_1 im Beweis von Lemma 4.8. \square



Satz 4.6.

Sei L^C konischer Verband, $a = \text{supp}(x)$ mit $x \in C$. Dann sind äquivalent:

- $a \notin \mathcal{M}(L^C),$
- $a \notin \mathcal{J}(L^C),$
- $\Leftrightarrow \exists x^{(1)} \in C : \quad 0 < a_1 = \text{supp}(x^{(1)}) < a,$
- $\Leftrightarrow \exists x^{(1)}, x^{(2)} \in C : \quad 0 < a_i = \text{supp}(x^{(i)}) < a, \quad x = x^{(1)} + x^{(2)},$
- $\Leftrightarrow \forall i \in \mathbb{N} \exists x^{(i)} \in C \forall j < i : \quad a = \text{supp}(x^{(i)}), \quad x^{(i)} \not\sim x^{(j)},$
- $\Leftrightarrow \exists \tilde{x} \in C : \quad a = \text{supp}(\tilde{x}), \quad x \not\sim \tilde{x}.$

Beweis. Konische Verbände sind atomar, also $\mathfrak{a} \in \mathcal{M}(L) \Leftrightarrow \mathfrak{a} \in \mathcal{J}(L)$.

Sei $\mathfrak{a} = \text{supp}(x) \notin \mathcal{M}(L^C)$, dann gibt es $x^{(1)} \in C$ mit $0 < \mathfrak{a}_1 = \text{supp}(x^{(1)}) < \mathfrak{a}$.

Sei nun $0 < \mathfrak{a}_1 < \mathfrak{a}$. Dann finden wir mit Lemma 4.8 eine Zerlegung $x = x^{(1)} + x^{(2)}$.

Betrachte nun $x^{(\lambda)} = \lambda x^{(1)} + (1 - \lambda)x^{(2)} \in C$ mit $\lambda \in [0, 1]$. Dann gilt $\mathfrak{a}_\lambda = \text{supp}(x^{(\lambda)}) = \mathfrak{a} = \mathfrak{a}_1 + \mathfrak{a}_2$ für fast alle λ , jedoch $x^{(\lambda_1)} \not\sim x^{(\lambda_2)}$ für $\lambda_1 \neq \lambda_2$. Somit haben wir unendliche viele paarweise nicht äquivalente $x^{(\lambda)}$ mit $\text{supp}(x^{(\lambda)}) = \mathfrak{a}$.

Wenn wir unendlich viele solcher Paare haben, gibt es insbesondere einen zweiten Vektor \tilde{x} mit $x \not\sim \tilde{x}$ aber $\mathfrak{a} = \text{supp}(\tilde{x})$.

Gibt es nun so ein \tilde{x} , so folgt aus Lemma 4.10, dass $\mathfrak{a} \notin \mathcal{M}(L^C)$. □



Bemerkung 4.12 (Kegelstümpfe erhalten Verband).

Beschränkt man den Kegel, indem man die Beträge seiner Komponenten nach oben beschränkt, so bleibt der Trägerverband gleich; insbesondere bleibt er atomar.

Beweis. Man beachte, dass man im Beweis von Lemma 4.8 (zur Zerlegbarkeit) die Elemente am Ende noch skalieren kann, damit sie die Beschränkungsbedingung erfüllen. □

Nun können wir die Atome des Kegelverbandes mit dem einfacheren Polyederalgorithmus *CalculateMinimalElements* berechnen; dafür müssen wir $M > 0$ ausreichend groß wählen.

Eine kleine Nettigkeit

In Satz 4.1 haben wir uns schon einmal mit den kleinsten Elementen aus $L_{\supseteq R}$ beschäftigt. Laszlo David untersucht derzeit, ob es einen Weg gibt, die Elemente aus $\mathcal{M}(L) \cap L_{\supseteq R}$ zu berechnen und hat ein Verfahren entwickelt, mit dem das möglich ist. Das Problem ist nicht trivial: Acuña, Marchetti-Spaccamela, Sagot & Stougie (2010) zeigen, dass es bereits \mathcal{NP} -schwer ist, einen elementaren Flussmodus für nur zwei vorgegebene Reaktionen zu finden.

Auf dem Weg zu diesem Verfahren hatte er folgende hübsche Beobachtung vermutet, die wir gemeinsam beweisen konnten:



Satz 4.7.

Seien (S, Irr) metabolische Bedingungen mit $\text{Irr} = [n]$. Seien zudem $R = \{r_1, r_2\} \subseteq [n]$. Betrachte nun $L_{\supseteq R}^F$. Ist \mathfrak{a}^* nach Kardinalität kleinstes Element in $L_{\supseteq R}^F \setminus \{0\}$, dann ist $\mathfrak{a}^* \in \mathcal{M}(L^F)$ oder es gibt $e^{(1)}, e^{(2)} \in F$ mit $\mathfrak{a}^* = \mathfrak{a}_1 + \mathfrak{a}_2$ und $\mathfrak{a}_i = \text{supp}(e^{(i)}) \in \mathcal{M}(L^F)$ für $i \in \{1, 2\}$ und $\mathcal{M}(L_{\subseteq \mathfrak{a}^*}) = \{\mathfrak{a}_1, \mathfrak{a}_2\} = \mathcal{M}(L) \cap L_{\subseteq \mathfrak{a}^*}$.

Beweis. Wir werden eine stärkere Aussage zeigen; der Beweis verläuft ähnlich zu Lemma 4.8.

Sei $v \in F$ mit $\mathfrak{a} = \text{supp}(v)$ und $i, j \in \mathfrak{a}$ mit $i := r_1, j := r_2$. Seien weiter $\mathfrak{a}_1 = \text{supp}(v^{(1)})$, $\mathfrak{a}_2 = \text{supp}(v^{(2)})$ mit $v^{(1)}, v^{(2)} \in F$, $i \in \mathfrak{a}_1 \cap \mathfrak{a}_2$ und $\mathfrak{a}_1, \mathfrak{a}_2 \subsetneq \mathfrak{a}$. Weiterhin sei $\mathfrak{a}_1 \neq \mathfrak{a}_2$ und $k \in \mathfrak{a}_1 \setminus \mathfrak{a}_2$. Wegen $\mathfrak{a}_1 \leq \mathfrak{a}$ ist also auch $v_k > 0$. Wir werden zeigen, dass dann $\mathfrak{a} \notin \mathcal{M}(L_{\supseteq\{i,j\}})$ gilt und daraus folgt die Behauptung.

Sei $\lambda > 0$ beliebig. Dann definiere $v^{(\lambda)} := v + \lambda \cdot v^{(2)} \in F$. Dann gibt es ein eindeutiges $\mu_\lambda > 0$, so dass $w^{(\lambda)} := v^{(\lambda)} - \mu_\lambda \cdot v^{(1)} \in F$ mit $\mathfrak{a}_\lambda := \text{supp}(w^{(\lambda)}) \subsetneq \text{supp}(v^{(\lambda)}) = \mathfrak{a} = \mathfrak{a}_1 + \mathfrak{a}_2$. Nach Konstruktion wissen wir:

$$\begin{aligned} 0 \leq w_k^{(\lambda)} &= v_k + \lambda \cdot v_k^{(2)} - \mu_\lambda \cdot v_k^{(1)} \\ &= v_k - \mu_\lambda \cdot v_k^{(1)} \\ \Rightarrow \mu_\lambda &\leq \frac{v_k}{v_k^{(1)}} =: M_k. \end{aligned}$$

Angenommen es wäre $\mathfrak{a} \in \mathcal{M}(L_{\supseteq\{i,j\}})$. Wegen $i \in \mathfrak{a}_1 \cap \mathfrak{a}_2$ wäre dann $j \notin \mathfrak{a}_1 \cup \mathfrak{a}_2$ und somit $j \in \mathfrak{a}_\lambda$. Wir zeigen jetzt noch, dass $i \in \mathfrak{a}_\lambda$ für λ groß genug, was \mathfrak{a} minimal widerspricht:

$$\begin{aligned} w_i^{(\lambda)} &= v_i + \lambda \cdot v_i^{(2)} - \mu_\lambda \cdot v_i^{(1)} \\ &\geq v_i + \lambda \cdot v_i^{(2)} - M_k \cdot v_i^{(1)} \\ &\geq \lambda \cdot v_i^{(2)} - M_k \cdot v_i^{(1)} \geq v_i^{(2)} - \text{const}. \end{aligned}$$

□



Kapitel 4 – Kurz und knapp

Elementare Unterstrukturen

- ▷ Viele Mengenrelationen definieren wichtige Unterstrukturen.
- ▷ Viele dieser Unterstrukturen sind Hauptideale.

Überdeckungen

- ▷ Überdeckungen sind Reaktionsmengen, ohne die es keinen gültigen Stoffwechselfad gibt.
- ▷ Sie bilden zusammen einen dualen Verband, der eindeutig durch die minimalen Elemente definiert ist.

Kegelverbände

- ▷ Kegelverbände haben ergänzende Summanden und sind atomar.
- ▷ Lineare Ungleichungen definieren den Kegelverband. Der lässt sich deshalb mit Polyederalgorithmen untersuchen.

KAPITEL 5

Verbände in der Biologie

Kapitelübersicht

5.1 Flussmuster	76
5.1.1 Elementarmoden und Flussmuster	76
5.1.2 Vergleich von Untersystem-Modellen	79
5.2 Minimal Metabolic Behaviours	84
5.2.1 Wie spannt man einen Flusskegel auf?	84
5.2.2 Das Innerste nach außen stülpen	85

Die Motivation ist bekannt: Verständnis des Stoffwechsels für Fortschritt in der Medizin.

Das Ziel ist klar: Wir wollen Verbandstheorie als einheitliche Sprache benutzen, um möglichst viele Fragen und Lösungen zum Stoffwechsel diskret zu beschreiben. Das große Bild überblickbar machen, ohne sich dafür auf ein einziges Modell festlegen zu müssen. Wir kennen die bekanntesten Anwendungen und wollen sie in die Sprache der Verbände übersetzen. Wenn es geht auch neue finden.

Wir haben uns in Verbandstheorie eingearbeitet, ihre Grundlagen kennengelernt. Ausführlich. Sehr ausführlich.

Immer wieder sind Beispiele oder Motivationen aus der Biologie, aus dem Stoffwechsel aufge-
taucht. Aber was gab's da noch mal alles?

Das große Bild steht noch nicht. Was kennen wir schon, was können wir schon? Welche der zig
Möglichkeiten der Verbandstheorie löst uns welches Stoffwechselproblem? Und wie verbinden
sich dadurch die verschiedenen etablierten Modelle und ihre Anwendungen?

Spannen wir die Leinwand auf und fangen an zu malen!

§5.1 Flussmuster

§5.1.1 Elementarmoden und Flussmuster

Die meisten etablierten Verfahren nehmen das gleiche Modelle zur Grundlage, den Flusskegel (Definition 2.8), der die konkreten Reaktionsgeschwindigkeiten aller stationären Zustände enthält.

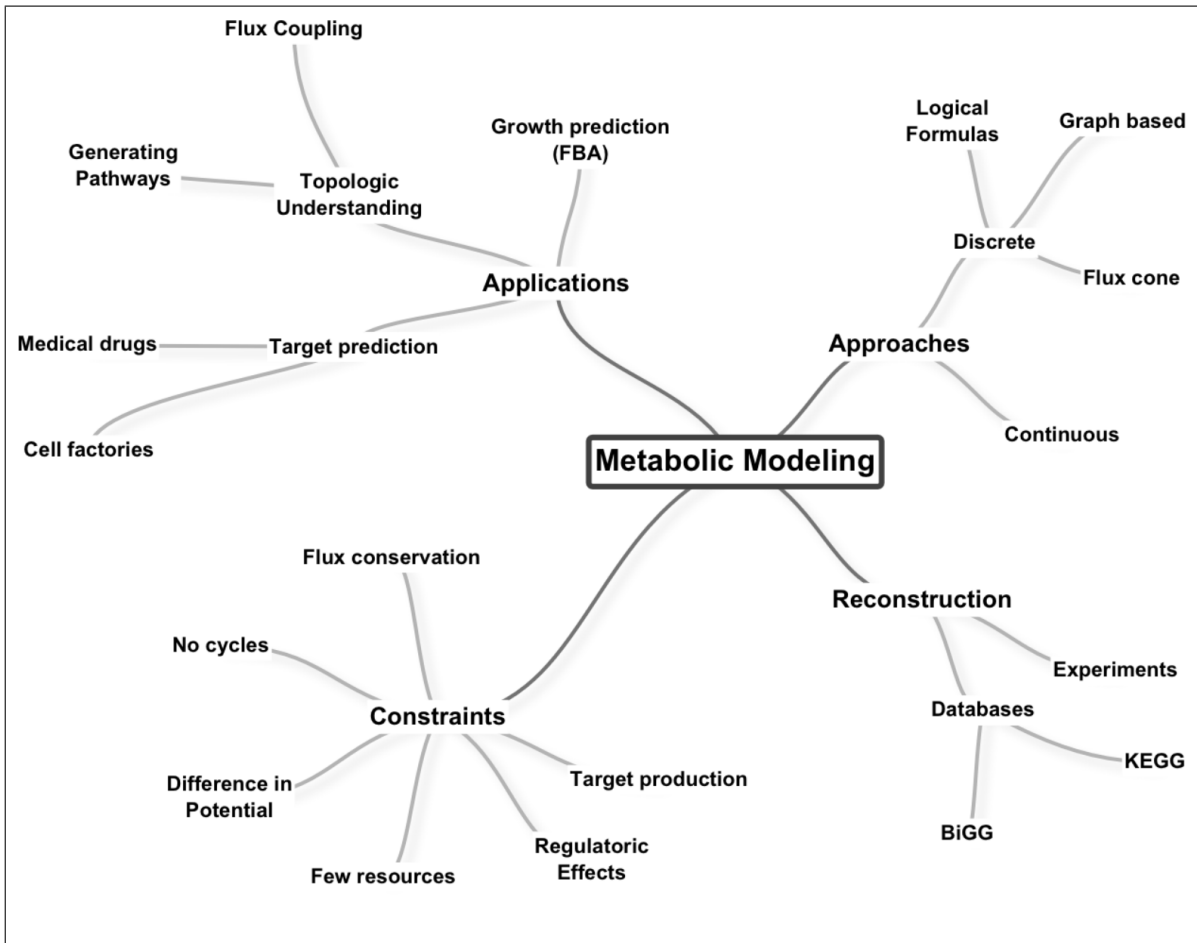
Der Flusskegel ist Dreh- und Angelpunkt für so unterschiedliche Verfahren wie Elementarmoden (Schuster 1994) und Extreme Pfade (Schilling et al. 2000), die ihn beide zu beschreiben versuchen (Savinell & Palsson 1992, Llaneras & Picó 2010), Wachstumsprognosen per Flussbalanceanalyse (FBA) (Orth, Thiele & Palsson 2010), oder überhaupt Fluxpotenzialuntersuchung (FVA) (Mahadevan & Schilling 2003), mit der die komponentenweisen Extremwerte eines beschränkten Flusskegels bestimmt werden. Auch die klassische Flusskopplungsuntersuchung (FCA), der wir uns im kommenden Kapitel ausführlich widmen werden, wurde bisher ausschließlich für den Flusskegel formuliert, wobei eine Verbandsdefinition so nahe liegt.

Neuere Verfahren wie rFBA (Covert 2002) oder SR-FBA (Shlomi et al. 2007) versuchen eine realistischere Beschreibung, indem sie regulatorische Prozesse integrieren. Bei der SR-FBA wird über ein mehrstufiges System logischer Formeln reguliert, welche Reaktionen stattfinden können. Reaktionen haben Nullfluss, wenn nicht passende Proteinkomplexe verfügbar sind. Diese wiederum werden in der Zelle nur erstellt, wenn es zu bestimmter Genexpression kommt. Doch ob dies klappt oder wichtige zugehörige Gene deaktiviert sind, ist auch reguliert – teilweise dadurch, welche Reaktionen stattfinden. Die SR-FBA verlangt nun einen stationären Zustand, der nicht nur auf die Metabolitenkonzentrationen, sondern auch auf genregulatorisches Gleichgewicht ausgelegt ist.

Jede dieser Anwendungen nimmt sich den Flusskegel mit seiner Beschreibung $Sv = 0$, $v_{\text{irr}} \geq 0$ vor und ergänzt ihn um einige anwendungsbezogene Nebenbedingungen.


Das Kapitel über Vereinigungshalbverbände hat uns damit vertraut gemacht, was für besondere Eigenschaften Verbände haben, die von Kegeln induziert wurden. Im Falle des Flusskegels kommen wir damit zum Flussverband L^F , den wir in diesen Kapitel genau betrachten wollen. Wir übersetzen die gängigen Anwendungen und Algorithmen wo möglich in Verbandssprache und überlegen, was man für allgemeine (metabolische) Verbände übernehmen kann.

Dadurch, dass wir uns mit Vereinigungsverbänden beschäftigen wollen, gehen die Informationen über die konkreten Reaktionsgeschwindigkeiten verloren. Wir unterscheiden nicht mehr, wie „wichtig“ eine bestimmte Reaktion in einem konkreten stationären Zustand ist. Damit gehen wir genau den entgegengesetzten Schritt zu Zhang, Borodovsky, Ozsoyoglu, Mikler, Ay & Kahveci (2010), wo die Wichtigkeit einzelner oder einiger Reaktionen in einem Stoffwechselfad mittels Polyedern bestimmt wird. Das ist interessant und man kann viele wichtige Fragen in diesem Bereich beantworten. Hier in dieser Arbeit werden wir aber konsequent auf die Geschwindigkeiten verzichten. Trotzdem werden wir viele Ergebnisse erhalten, deren zugrunde



liegende Fragestellungen unabhängig von den Reaktionsgeschwindigkeiten sind.

Starten wir jetzt mit den Elementarmoden:

 **Definition 5.1** (elementarer Flussmodus, elementare Flussmuster).
 Sei F Flusskegel zu den metabolischen Bedingungen (S, Irr) und $L = L^F$ der induzierte Flussverband. Sei ferner $e \in F$. Dann:

$$e \text{ elementarer Flussmodus} \Leftrightarrow \text{supp } e \in \mathcal{M}(L).$$

Es ist \mathcal{E} die Menge aller elementaren Flussmoden und $\mathcal{EP} = \mathcal{J}(L)$ die Menge der elementaren Flussmuster.

elementarer
Fluss-
modus

Der Begriff der Flussmuster wurde zuerst von Kaleta et al. (2009) vorgestellt, dort jedoch nur bezogen auf projizierte Netzwerke.

L^F ist Kegelverband, wie wir gezeigt haben. Da die Elementarmoden von F über die minimalen Elemente in L^F definiert sind, ergeben sich viele verschiedene Beschreibungen für sie:

**Bemerkung 5.1.**

Satz 4.6 liefert uns eine Vielzahl äquivalenter Definitionen für Elementarmoden. e ist elementarer Flussmodus, wenn sein Träger minimal ist oder keine Zerlegung durch Flussvektoren kleineren Trägers möglich ist.

Diese Vielfalt an äquivalenten Eigenschaften hat dazu geführt, dass die meisten Wissenschaftler unterschiedliche Bilder im Kopf haben, wenn sie an Elementarmoden denken. Die eine Fraktion hat zuerst die Minimalitätseigenschaft vor Augen: Ein Elementarmodus ist ein stationärer Zustand, bei dem alle Reaktionen für die Stationarität nötig sind. Er ist minimal, da auf keine Reaktion verzichtet werden kann.

Andere denken zuerst an die Erzeugereigenschaft: Die Elementarmoden sind der eindeutig bestimmte Erzeuger des Flusskegels, der aus allen Flussvektoren besteht, die selbst nicht zerlegbar sind. Anhänger dieser Denkschule sind sich wohl bewusst, dass die Elementarmoden in ihrer Art zwar eindeutig bestimmt sind, aber nicht der einzige denkbare Erzeuger für F sind. Auch die Extremen Pfade (Llaneras & Picó 2010) sind eindeutiger Erzeuger und sogar weniger als E . Wenn man nach einem Erzeuger sucht und sich bewusst für die Elementarmoden entschieden hat, will man im Diskurs mit anderen Wissenschaftlern nicht überrascht werden, dass deren Elementarmoden gar nicht die gesamte Struktur erzeugen.

Was also tun im allgemeinen metabolischen Verband? Am besten gar nichts. Lassen wir die Elementarmoden und die elementaren Flussmuster Alleinstellungsmerkmal des Flusskegelverbands und nennen im Allgemeinfall das Kind beim Namen: *minimale Elemente* oder *unzerlegbare Elemente*. Da weiß jeder, woran er ist.

Auch wenn sich das Konzept der Elementarmoden nicht auf andere Verbände übertragen lässt, bleibt es nach wie vor faszinierend. Warum, sehen wir jetzt.

Wir haben durch die mathematische Untersuchung von Verbänden schon viel Vorarbeit geleistet. Aus Satz 4.6 folgt sofort:

**Satz 5.1.**

Sei F Flusskegel zu (S, Irr) und $L = L^F$ der induzierte Verband. Sei ferner $e \in E$. Dann ist e bis auf Vielfache eindeutig durch seinen Träger bestimmt.

Für die Trägerfunktion eingeschränkt auf E gilt:

$$\begin{aligned} \text{supp}|_E : E &\rightarrow \mathcal{EP} && \text{surjektiv,} \\ \text{supp}|_E(e^{(1)}) = \text{supp}|_E(e^{(2)}) &\Leftrightarrow e^{(1)} \sim e^{(2)}. \end{aligned}$$

Da elementare Flussmoden bis auf Äquivalenz eindeutig durch ihren Träger bestimmt sind, können wir unseren Algorithmus zur Berechnung der minimalen Elemente nutzen, und bekommen so die Elementarmoden nach Trägerkardinalität sortiert, ohne uns Gedanken über die Double Description Method machen zu müssen.

→ Korollar 5.1.

Algorithmus *CalculateMinimalElements* berechnet nicht nur $\mathcal{M}(L^F)$, sondern als Neben-
 ergebnis gleich noch $E = \{x^{(+),(j)} - x^{(-),(j)} \mid j \in [|\mathcal{M}|]\}$.

Dass man überhaupt alle Elemente berechnen kann und es nicht unendlich viele gibt, ist auch dieser Zuordnung zum Träger geschuldet. Es gilt:

→ Korollar 5.2 (nur endlich viele Elementarmoden).

$$|E/\sim| = |\mathcal{M}(L^F)| \leq 2^n.$$

Für die bessere Abschätzung nach oben $\binom{n}{m+1}$ sei auf Klamt & Stelling (2002) verwiesen.

Doch unabhängig davon, wie viele elementare Flussmoden nun tatsächlich in unserem Netzwerk vorhanden sind, so wissen wir von Anfang an, dass es schwer ist, sie alle zu berechnen. *Schwer* hier durchaus im Sinne der Komplexitätstheorie: Acuña et al. (2009) zeigen, dass es sich tatsächlich um ein P#-schweres Problem handelt.

§5.1.2 Vergleich von Untersystem-Modellen

Unsere Sätze und Aussagen starten meist mit *Sei L Verband* und wir suchen dann eine hübsche neue Eigenschaft unseres Systems. Wo möglich legen wir uns nicht auf ein spezielles Modell fest, das unserem Verband zugrunde liegen soll, sondern überlassen es dem werten Anwender, sich selbst für ein Modell zu entscheiden.

Damit erhalten wir nicht nur elegante Verfahren, die von möglichst vielen genutzt werden können, sondern navigieren uns auch an einer anderen Frage vorbei: Wie kommen wir überhaupt an die Daten, die wir zum Aufbau unseres Systems brauchen?

Sowohl das klassische Verfahren über den stationären Flusskegel als auch der neue logische Verband benötigen eine Übersicht über alle biochemischen Reaktionen, die zum Stoffwechsel beitragen können. Ob mit oder ohne exakte stöchiometrische Werte sei hier nicht einmal diskutiert. Woher kommen die Zellmodelle?

Für viele verschiedene Zelltypen gibt es fertige Modelle im Internet, beispielsweise in der BiGG Datenbank (Schellenberger, Park, Conrad & Palsson 2010). Die haben es dorthin geschafft, weil sich Wissenschaftler zuvor hingesezt und vorhandene Daten zusammengetragen haben. Verschiedene Untersysteme wie der Zitronensäurezyklus, die aus anderen Zelltypen bekannt sind, werden übernommen, experimentell gefundene andere Reaktionen werden hinzugefügt. Häufig werden die Netzwerke auch aus Genomdaten rekonstruiert (*genom-scale reconstruction*).

Immer wieder gibt es Publikationen, in denen ein neues, großes Modell vorgestellt wird, das einen Kompromiss aus allen bekannten Informationen darstellt. Eine solche Arbeit – eine Netzwerksrekonstruktion für *Saccharomyces cerevisiae*, stammt von Herrgård, Swainston, Dobson,

Dunn, Arga, Arvas, Blüthgen, Borger, Costenoble, Heinemann, Hucka, Le Novère, Li, Liebermeister, Mo, Oliveira, Petranovic, Pettifer, Simeonidis, Smallbone, Spasić, Weichart, Brent, Broomhead, Westerhoff, Kürdar, Penttilä, Klipp, Palsson, Sauer, Oliver, Mendes, Nielsen & Kell (2008). Diese beeindruckende Autorenliste veranschaulicht, wie mannigfaltig die Informationen sind, die zusammengetragen werden können. Doch auch dieses langdiskutierte, mühevoll erarbeitete Modell stellt nur eine Approximation an die Wirklichkeit dar.

Dessen müssen wir uns immer bewusst sein: Egal wie viel wir über unser Modell nachdenken, die Grundlagendaten sind immer nur ein (großer, wichtiger) Teil von dem, was im lebenden Organismus tatsächlich passiert. Darum ist es oberstes Ziel des Modellierers, Simulationen zu produzieren, deren Resultate eine gute Annäherung an beobachtbare Daten liefern. Zu viel Komplexität überspezialisiert das Modell bloß an alte Experimente. Es gilt wieder einmal das Paradigma vom minimal adäquaten Modell.

Manchmal sind es aber auch ganz pragmatische Gründe, die Forscher dazu bewegen, unwichtigere Reaktionen aus ihrem System zu werfen: Es gibt zu viele Elementarmoden beim „echten“ genomweiten Netzwerk, als das wir sie berechnen können. Gut, dann konzentrieren wir uns halt auf einen wichtigen Kern des Systems und untersuchen den dafür umso genauer. Populäres Beispiel dafür ist das Netzwerk *E. coli Core* (Orth, Fleming & Palsson 2010).

Ob dieser Ansatz wirklich das Gelbe vom Ei ist, was für Alternativen uns die Verbandstheorie anbietet und was zu tun ist, wenn Experimente unter bestimmten Extrabedingungen simuliert werden sollen, wollen wir in diesem Abschnitt diskutieren.

Bedingte Untermodelle

Nach dieser ausführlichen Motivation können wir schnell zur Sache kommen: Für die allermeisten interessanten Szenarios haben wir bereits einen Unter(halb)verband kennengelernt. Verschaffen wir uns einen kurzen Überblick in Tabelle 5.1.

Im Fall von L^F können wir unseren Standardalgorithmus zur Berechnung der minimalen und der unzerlegbaren Elemente verwenden. Dank Bemerkung 4.7 können wir problemlos mehrere Bedingungen kombinieren und dann nach Vereinfachung auf Algorithmus *MinimalConstrainedElements* zurückgreifen. Hier sei auf die Arbeit von Kaleta et al. (2009) verwiesen, die dieses MILP zur Berechnung der unzerlegbaren Elemente im projizierten Flussverband vorstellen – auch wenn sie noch von *elementaren Flussmustern* sprechen und auf vollständig irreversible Systeme angewiesen sind.

Algorithm *MinimalConstrainedElements*($S, Irr, R_1, R_2, R_3, R_4$)

Output: $\mathcal{M} \left(\left(L_{\subseteq R_1 \perp R_2 \parallel R_3}^F \right)_{R_4} \right)$.

1. $\mathcal{M} \leftarrow \emptyset$
2. $k \leftarrow 0$
3. **while** is feasible

Anwendung		
Beschreibung	Beispielszenario	Verband
Nur R_1 kommt vor.	Wir arbeiten mit einer neuen Zelle und kennen nur einige Reaktionen.	$L_{\subseteq R_1}$
R_2 darf nicht vorkommen.	Die Enzyme, die die Reaktionen in R_2 katalysieren, wurden ausgeknockt.	$L_{\perp R_2}$
R_3 muss vorkommen.	Wir wollen die Produktion einiger Metaboliten fördern und untersuchen deshalb, welche Stoffwechselfade sie erzeugen.	$L_{\parallel R_3}$
Betrachte nur R_4 .	Wir interessieren uns nur für die Prozesse in einem bestimmten Zellabschnitt.	L_{R_4}
Endliche Ressourcen.	Wir wollen verstehen, wie der Stoffwechsel mit endlichen Ressourcen umgeht – nicht alle Reaktionen sind gleichzeitig möglich.	$L_{\varphi \leq \tau}$

Tabelle 5.1.: Biologische Constraints lassen sich gut durch Unterhalbverbände ausdrücken

4. **do**

$$\begin{aligned}
 & \text{minimize} && \sum_{i \in R_4} a_i \\
 & \text{s.t.} && S \cdot (v^{(+)} - v^{(-)}) = 0 \\
 & && v_i^{(-)} = 0 && \forall i \in \text{Irr} \\
 & && a_i^{(+)} + a_i^{(-)} = a_i && \forall i \in [n] \\
 & && a_i^{(\pm)} \leq v_i^{(\pm)} \leq M \cdot a_i^{(\pm)} && \forall i \in [n] \\
 & && a_i^{(\pm)}, a_i \in \{0, 1\} && \forall i \in [n] \\
 & && \sum_{i \in e^{(j)}} a_i \leq |e^{(j)}| - 1 && \forall j \in [k] \\
 & && \sum_{i=1}^n a_i \geq 1 \\
 & && \sum_{i \in R_2 \cup ([n] \setminus R_1)} a_i = 0 \\
 & && \sum_{i \in R_3} a_i = 1
 \end{aligned}$$

5. $e^{(k+1)} \leftarrow \{i \in [n] \mid a_i = 1\}$
6. $\mathcal{M} \leftarrow \{e^{(k+1)}\} \cup \mathcal{M}$
7. $k \leftarrow k + 1$
8. **return** \mathcal{M}

Man beachte, dass $L_{\parallel R}$ und L_R im Allgemeinen nicht atomar sind. Für $R_3 \cap R_4 \neq [n]$ müssen also gegebenenfalls noch die unzerlegbaren Elemente berechnet werden. Hierfür benötigt man wieder die Puffervariable h aus Algorithmus *CalculateIrreducibleElements*.

Im Satz 3.5 über Kegelstümpfe haben wir gezeigt, dass $\mathcal{M}(L_{\varphi \leq \tau}) = \mathcal{M}(L) \cap L_{\varphi \leq \tau}$ gilt (und entsprechend für die unzerlegbaren Elemente). Handelt es sich bei φ um eine lineare Funktion, lässt sich also schreiben als $\varphi(a) = \sum_{k=1}^n p^{(h)} \cdot ik \cdot a_k$, dann ergibt sich eine elegante und einfache Erweiterung des Algorithmus, um auch solche Bedingungen wie Ressourcenbeschränkung zu integrieren.

Warum sollte man mit den Konventionen brechen? Wir interessieren uns nur für eine kleine Teilmenge $R \subseteq [n]$ der Reaktionen. Wir wollen wissen, wie sie miteinander kombiniert werden können oder was mit ihnen passiert, wenn wir das System bestimmten Umweltbedingungen aussetzen.

Wie bereits erwähnt, ist es gängige Konvention, ein Untersystem zu konstruieren, das diese Reaktionen enthält, vielleicht noch alle anderen, die man für unabdingbar befindet, und dann bloß auf diesem System zu arbeiten. Weniger Reaktionen = weniger Arbeitsaufwand, weniger Rechenzeit. Populäres Beispiel ist das bereits erwähnte *E. coli Core* Modell.

Wir landen beim Unterverband $L_{\subseteq R}$, der sogar atomar ist, wenn L es ist. Er lässt sich einfach und nett mit den Standardalgorithmen untersuchen, die man nicht einmal anpassen muss. Wir füttern sie einfach mit einem kleineren System.

Wenn wir sagen, wir interessieren uns nur für die Reaktionen in R , dann wollen wir in Wirklichkeit die Eigenschaften von L_R bestimmen.¹

Dass das einen wesentlichen Unterschied macht, haben wir in Übersicht 4.2 gesehen. Es ist zwar $L_{\subseteq R} \subseteq L_R$ Unterverband, aber wir berechnen nur eine Teilmenge der Projektion von $\mathcal{M}(L)$, nicht jedoch die echten minimalen Elemente unserer Struktur. Im Allgemeinen ist nicht mal eine Teilmengenrelation zu $\mathcal{M}(L_R)$ möglich; wir kennen die Beziehung der minimalen Elemente im allgemeinen Fall nicht.

Für viele Leser mag nur der Fall des atomaren Verbandes L^F interessant sein, wie er in Abbildung 4.2b zusammengefasst ist. Noch schlimmer sieht es aber im nicht-atomaren Fall aus. Hier sind die unzerlegbaren Elemente als minimaler Erzeuger von besonderer Bedeutung. Doch hier können wir nicht einmal feststellen, dass wir mit $\mathcal{J}(L_{\subseteq R})$ wenigstens eine Untermenge von $P_R(\mathcal{J}(L))$ berechnen.

Da im Allgemeinen auch noch $1_{L_{\subseteq R}} < 1_{L_R}$ gilt, werden wir noch dazu falsche Aussagen zur Flusskopplung machen, wie wir im folgenden Kapitel sehen werden.

¹Ganz genau genommen wollen wir $(L_{\text{opt}})_R$ beschreiben, wobei L_{opt} der Verband ist, der durch ALLE Reaktionen bestimmt ist – also auch denen, die es gibt, die wir jedoch nicht kennen. Wir arbeiten stets mit einem vereinfachten Unterverband $L = (L_{\text{opt}})_{\subseteq [n]}$, der mit den gleichen Makeln versehen ist, wie wir sie hier für $L_{\subseteq R}$ im Vergleich zu L_R ankreiden.

Im Fall von L^F muss der Standardalgorithmus nur minimal abgewandelt werden, um Elemente aus L_R statt $L_{\subseteq R}$ zu berechnen. Das haben Kaleta et al. (2009) erkannt und dafür die elementaren Flussmuster definiert.

Doch am Ende zählt wieder das KISS-Prinzip. Was ist der einfachste Ansatz an Resultate zu kommen, die immer noch ausreichend gut sind?

Eine mögliche Antwort darauf liefern Marashi, David & Bockmayr (2012a). Wie auch schon Kaleta et al. haben sie erkannt, dass es nicht von Interesse sein kann, zu einer gegebenen Teilmenge an Reaktionen ausschließlich $\mathcal{M}(L_{\subseteq R}^F) = \mathcal{J}(L_{\subseteq R}^F)$ zu betrachten. Wo Kaleta et al. den Algorithmus vorstellen, mit dem man stückweise die elementaren Flussmuster $\mathcal{J}(L_R^F)$ berechnet, stellen sie die Projektion der Elementarmoden $P_R(\mathcal{M}(L^F))$ in den Vordergrund. Sie wandeln die Double Description Method ab und ergänzen den Algorithmus um die Projektion des Flusskegels auf Unterräume. Damit schaffen sie es in einem Schritt, die von ihnen *ProCEMs* genannten Elemente zu berechnen. Sie landen in der Mitte, $\mathcal{J}(L_R^F) \subseteq \text{ProCEMs} \subseteq P_R(\mathcal{M}(L^F))$.

Wo der stückweise Ansatz von Kaleta et al. in seiner Laufzeit sehr abhängig von der Größe des untersuchten Untersystems ist, sinkt die Laufzeit bei Marashi et al. sogar leicht mit wachsendem $|R|$. Im untersuchten Netzwerk zieht ihr Algorithmus für $|R| = 30$ nach und schlägt von da an das MILP-basierte Verfahren um Größenordnungen. Ihre Aussagen beziehen sich auf ein Modell von *Arabidopsis thaliana* (Oliveira Dal’Molin, Quek, Palfreyman, Brumbley & Nielsen 2010), für das sie die ProCEMs in Sekunden bis Minuten berechnen können, wo eine direkte Projektion der Elementarmoden des Gesamtnetzwerkes nicht möglich war – das verwendete *efmtool* (Terzer & Stelling 2008) quittierte wegen Speichermangel den Dienst.

Ob der größere Rechenaufwand durch Integration unwichtiger(er) Reaktionen zur Berechnung des vorher nur schwer bestimmbar genaueren Verbandserzeugers gerechtfertigt ist, sei dem Leser genauso überlassen wie die Entscheidung, welches Modell Grundlage des Verbandes sein soll. Wir betonen das hier so sehr, weil es eine wichtige Frage ist. Eine, deren Antwort man sich selbst gegenüber gut begründen können sollte. Wissenschaft besteht schließlich daraus, Thesen aufzustellen, die man verteidigen kann.

Diese Arbeit schneidet das Problem von Untersystemen immer mal wieder kurz an. Für eine ausführlichere Diskussion sei auf die Dissertation von Marashi (2011) verwiesen.

Kern der Matrix

Nach dieser Diskussion biologisch motivierter Bedingungen widmen wir uns einem anderen Untermodell, das alles andere als biologisch vertretbar scheint. Den Mathematikern unter uns brennt es aber vielleicht dennoch unter den Fingernägeln, es einmal genauer zu betrachten.

Der Flusskegel ist Teilmenge des Kerns der stöchiometrischen Matrix. Ignorieren wir mal gänzlich alle thermodynamischen Nebenbedingungen und betrachten nur den Kern, den wir in unserer Lineare Algebra I Vorlesung so lieb gewonnen haben.

**Definition 5.2 (Null-Verband).**

Sei F Flusskegel zu den metabolischen Bedingungen (S, Irr) . Definiere nun $L^{F,0}$ wie folgt:

$$L^{F,0} := \{a = \text{supp}(x) \mid x \in \text{Kern } S\}.$$

Dann heißt $L^{F,0}$ Null-Verband zu (S, Irr) .

**Bemerkung 5.2.**

$L^{F,0}$ ist konischer Verband und somit atomar.

**Bemerkung 5.3.**

Es ist $L^F \subseteq L^{F,0}$ Unterhalbverband.

Was das Ganze soll? Warten wir kurz ab. Das kleine unlogische Kerlchen $L^{F,0}$ wird uns später noch mit einem tollen Geschenk überraschen.

§5.2 Minimal Metabolic Behaviours

§5.2.1 Wie spannt man einen Flusskegel auf?

Einen n -dimensionalen Vektorraum kann man mit n Basisvektoren aufspannen. Eine zyklische Gruppe lässt sich sogar aus einem einzigen Element erzeugen.

Wie viele Elementarmoden zum Erzeugen des Flusskegels benötigt werden, lässt sich nicht so einfach sagen; man erinnere sich daran, dass $F \subseteq \mathbb{R}^n$ gilt. Es gibt keine einfache Formel, aber wenigstens wissen wir, dass es nicht mehr als $\binom{n}{m+1}$ viele sind (Klamt & Stelling 2002).

$\binom{n}{m+1}$, das bedeutet exponentiell in Anzahl der Metaboliten im System. Viel zu viel. Doch selbst wenn diese Schranke maßlos die Wahrheit übersteigt, so sind die echten Zahlen nicht viel besser – über 70 Millionen erzeugende Elemente für ein Untersystem von *S. cerevisiae*, das aus 230 (von 1149) Reaktionen besteht (Jol, Kümmel, Terzer, Stelling & Heinemann 2012), sind einfach zu viel, um mit aktuellen Computern noch wesentlich genauere Stoffwechselsysteme untersuchen zu können. Ganz zu schweigen von dem Aufwand, all diese Daten anschließend auswertbar zu machen. Kann man den Flusskegel nicht einfacher beschreiben und trotzdem alle seine Details beibehalten?

Palsson und andere setzen seit Jahren auf die Extremen Pfade, die eine Teilmenge der Elementarmoden darstellen. Die Anzahl schrumpft dabei aber nicht wesentlich und im Fall, dass alle Reaktionen irreversibel sind, gibt es sogar gar keinen Unterschied zwischen beiden Beschreibungen.

Einen etwas anderen Pfad gehen Larhlimi & Bockmayr (2009). In diesem Abschnitt wollen wir ihre Minimal Metabolic Behaviours kennenlernen und anschließend zeigen, in was für ein besonderes Licht die Verbandstheorie sie stellt.

§5.2.2 Das Innerste nach außen stülpen

Einen Kegel kann man auf zwei Weisen beschreiben: Von innen oder von außen.

innere Beschreibung Ein Polyeder ist konvex. Konvex bedeutet für einen Mathematiker, dass $\lambda v^{(1)} + (1 - \lambda) v^{(2)} \in P$ mit $\lambda \in [0, 1]$ für zwei beliebige Punkte/Vektoren aus dem Polyeder gilt.

Die konvexe Hülle $\text{Conv } A$ einer beliebigen Punktmenge $A \subseteq \mathbb{R}^n$ ist die Menge aller Punkte, die man so erzeugen kann:

$$\text{Conv}(A) = \left\{ \sum_{i=1}^t \lambda_i v^{(i)} \mid v^{(1)}, \dots, v^{(t)} \in A, \sum_{i=1}^t \lambda_i = 1, \lambda_i \geq 0 \right\}.$$

Ein Polyeder ist schon konvex, also gilt insbesondere $P = \text{Conv}(P)$.

Ein **Erzeuger** B von P ist eine Teilmenge $B \subseteq P$, die ausreicht, um P zu erzeugen. Oder mit anderen Worten $P = \text{Conv}(B)$.

Das Konzept des Erzeugers kennen wir vom Verband: Es ist $L = \langle \mathcal{J}(L) \rangle$.

Wir haben schon betont, dass die Elementarmoden Erzeuger des Flusskegels sind. Ein Kegel ist im Gegensatz zum Polyeder unbeschränkt, kann also nicht direkt durch die gewöhnliche Hülle endlich vieler Punkte beschrieben werden. Stattdessen ist eine Punktmenge $B \subseteq C$ Erzeuger des Kegels C , wenn $\left\{ \sum_{i=1}^t \lambda_i v^{(i)} \mid v^{(1)}, \dots, v^{(t)} \in B, \lambda_i \geq 0 \right\}$.

Das erfüllen die Elementarmoden oder auch die Extremen Pfade.

Ein Erzeuger enthält auf jeden Fall die Eckpunkte bzw. die äußersten Kanten. Da diese Beschreibung aus kegeleigenen Elementen besteht, heißt sie *innere Beschreibung*.

Bildlich gesprochen ist ein Erzeuger eine Punktmenge in P , wo man den gesamten Polyeder P erhält, wenn man die Luftlinien zwischen zwei beliebigen bereits erzeugten Punkten vereint. Im Falle eines Kegels C darf man dann ein wenig zusammendrücken oder auseinanderziehen.

äußere Beschreibung Man greife zum großen Messer, zerschneide den \mathbb{R}^n mit einem geraden Schnitt in zwei Teile und entsorge einen davon. Den zweiten behandelt man wieder so: Teile ihn mit einem sauberen Schnitt in zwei Teile und behalte nur den interessanten. Wiederhole das ein paar mal und du erhältst ein beliebiges Polyeder.

Mathematisch erreicht man das mittels linearer Ungleichungen, die Halbräume des \mathbb{R}^n abtrennen. Am Ende landet man auf der altbekannten Form $P = \{v \in \mathbb{R}^n \mid Av \geq b\}$ mit $A \in \mathbb{R}^{m \times n}, b \in \mathbb{R}^m$.

Wir beschreiben dadurch die Oberfläche des Polyeders, ohne irgendwelche Informationen über enthaltene Punkte direkt preiszugeben. Deshalb ist das die *äußere Beschreibung*.


Diese sehr knappe Einleitung sollte nur mit den wesentlichen Grundbegriffen der inneren und äußeren Beschreibung von Polyedern vertraut machen. Zur gründlichen Recherche sei etwa auf (Brøndsted 1983) oder (Grünbaum 1967) verwiesen.

Den Flusskegel können wir sehr einfach von außen beschreiben. Uns reichen m Gleichungen für die Reaktionen, $|Irr|$ Ungleichungen um die Reaktionsrichtungen vorzugeben. That's it. Weniger als $m + n$ Informationen insgesamt.

Aber auch innere Beschreibungen kennen wir schon. Elementarmoden oder Extreme Pfade. Leider gab's da die kombinatorische Explosion. Minimal Metabolic Behaviours (Larhlmi & Bockmayr 2009) gehen anders vor als diese beiden: Sie zerteilen den Kegel in den maximalen linearen Untervektorraum, den man durch seine Basisvektoren beschreiben kann, und die daran „festgeklebten“ Seiten. Wir erhalten eine im Allgemeinen viel kleinere Beschreibung.

Um das Konzept zu verstehen, brauchen wir ein wenig linearen Algebra:


MMB



Definition 5.3 (Minimal Metabolic Behaviours).
 Seien (S, Irr) metabolische Bedingungen und F zugehöriger Flusskegel. Sei außerdem $\text{lin. space}(F) = \text{span}(h^{(1)}, \dots, h^{(u)})$ maximaler linearer Unterraum, der in F enthalten ist mit $h^{(1)}, \dots, h^{(u)}$ als Basis.
 Dann kann man $h^{(0)}$ so wählen, dass $\text{lin. space}(F) = \{\lambda h \mid h \in \text{Conv}(h^{(0)}, \dots, h^{(u)}), \lambda \geq 0\}$.

Kommen wir nun zum Seiten-Begriff.

minimal
echte Seite



Definition 5.4 (minimale echte Seite).
 Seien (S, Irr) metabolische Bedingungen und F zugehöriger Flusskegel.
 Sei weiterhin $x \in \mathbb{R}^n$ beliebiger Vektor. Wir sagen, dass x eine gültige Ungleichung für F repräsentiert, wenn $\{v \in F \mid x^T v \geq 0\} = F$.
 Repräsentiert x eine gültige Ungleichung, so nennen wir $F_x = \{v \in F \mid x^T v = 0\}$ die durch x bestimmte Seite von F . Offensichtlich ist $\text{lin. space}(F) \subseteq F_x$.
 Schließlich nennen wir F_x minimale echte Seite von F , wenn $\dim F_x = \dim \text{lin. space}(F) + 1$. Also genau dann, wenn es ein $g \in F$ gibt mit $F_x = \{\lambda g + h \mid h \in \text{lin. space}(F), \lambda \geq 0\}$.
 Es ist offensichtlich $\text{supp } g \cap Irr \neq \emptyset$.

 **Bemerkung 5.4** (Träger minimal echter Seiten).

Sei $G \subseteq F$ minimal echte Seite. Für $g^{(1)}, g^{(2)} \in G \setminus \text{lin. space}(F)$ gilt dann:

$$\text{supp } g^{(1)} \cap \text{Irr} = \text{supp } g^{(2)} \cap \text{Irr}.$$

Das folgt direkt daraus, dass G konvexe Hülle von Flussvektoren ist, wo nur einer nicht vollständig reversibel ist.

 **Bemerkung 5.5** (minimale echte Seiten erzeugen F).

Es gibt nur endliche viele minimale Seiten G_1, \dots, G_t .

Sei nun $g^{(i)} \in G_i$ beliebig gewählt für $i \in [t]$. Dann erzeugen diese Vektoren zusammen mit dem linearen Unterraum den Flusskegel:

$$F = \left\{ \lambda v \mid v \in \text{Conv} \left(h^{(0)}, \dots, h^{(u)}, g^{(1)}, \dots, g^{(t)} \right), \lambda \geq 0 \right\}.$$

Damit kennen wir alle Strukturen, die wir brauchen, um die Minimal Metabolic Behaviours zu definieren.

 **Definition 5.5** (Minimal Metabolic Behaviours).

Seien (S, Irr) metabolische Bedingungen und F zugehöriger Flusskegel.

Es sei $\text{lin. space}(F)$ maximaler linearer Unterraum von F und G_1, \dots, G_t die minimalen echten Seiten mit Repräsentanten $g^{(1)}, \dots, g^{(t)}$.

Dann ist $D_i = \text{supp}(g^{(i)}) \cap \text{Irr}$ wohldefiniert. Die Elemente aus $\{D_1, \dots, D_t\}$ heißen *Minimal Metabolic Behaviours*.

\mathcal{MMB}

 **Satz 5.2** (Minimal Metabolic Behaviours).

Seien (S, Irr) metabolische Bedingungen mit Flusskegel F , minimalen echten Facetten G_1, \dots, G_t , Repräsentanten $g^{(i)} \in G_i$ und Minimal Metabolic Behaviours D_i .

Dann ist $G_i \setminus \text{lin. space}(F) = \{v \in F \mid \text{supp } v \cap \text{Irr} = D_i\}$.

Weiterhin ist $\mathcal{M}(L_{\text{Irr}}^F) = \mathcal{J}(L_{\text{Irr}}^F) = \mathcal{MMB} = \{D_1, \dots, D_t\}$.

$\mathcal{MMB} = \mathcal{M}(L_{\text{Irr}}^F)$

Beweis. Wir haben in Bemerkung 5.4 schon gesagt, dass die in Irr projizierten Träger von nicht-reversiblen Vektoren minimaler echten Seiten gleich sind. Das folgt daraus, dass G_i nur eine Dimension mehr besitzt als $\text{lin. space}(F)$. Damit kann die erste Gleichung gelten.

Wenden wir uns nun erst einmal der zweiten Aussage zu. Aus Bemerkung 5.5 wissen wir, dass

sich alle Flussvektoren $v \in F$ konisch zerlegen lassen in $v = h + \sum_{i=1}^t \lambda_i g^{(i)}$ mit $h \in \text{lin. space}(F)$

und $\lambda_i \geq 0$. Projizieren wir die Träger in Irr , so erhalten wir $\text{supp } v \cap \text{Irr} = \sum_{\substack{i \in [t] \\ \lambda_i > 0}} D_i$. Das

stimmt, da $v_j, g_j^{(i)} \geq 0$ für $j \in \text{Irr}$. Damit wissen wir, dass $L_{\text{Irr}}^F = \langle D_1, \dots, D_t \rangle$.

Die Minimal Metabolic Behaviours sind Erzeuger und damit gilt insbesondere $\mathcal{J}(L_{\text{Irr}}^F) \subseteq$

\mathcal{MMB} . Wegen $\mathcal{M}(L) \subseteq \mathcal{J}(L)$ reicht es nun zu zeigen, dass die Elemente aus \mathcal{MMB} minimal sind, und wir sind mit der zweiten Aussage fertig.

Seien nun also $0 < \alpha_1 < \alpha_2$ mit $\alpha_1, \alpha_2 \in L_{\text{Irr}}^F$ und $\alpha_i = \text{supp } v^{(i)} \cap \text{Irr}$, $v^{(i)} \in F$ für $1 \leq i \leq 2$. Betrachte $v^{(\lambda)} = \lambda v^{(1)} + v^{(2)}$. Es ist $v^{(\lambda)} \in F$ für $\lambda \geq 0$. Ähnlich wie im Beweis zu Lemma 4.8 lässt sich zeigen, dass es auch ein $\lambda^* < 0$ gibt mit $v^{(\lambda^*)} \in F$.

Angenommen es sei $\alpha_2 = D_i$ Minimal Metabolic Behaviour, also $v^{(2)} \in G_i$. Damit ist $v^{(2)}$ insbesondere Element einer Seite, die durch ein $x \in \mathbb{R}^n$ definiert ist und $x^\top v^{(2)} = 0$. Wenn x die Seite definiert, so gilt $x^\top v \geq 0$ für alle $v \in F$, also insbesondere für $v^{(\lambda)} \in F$. Nun gilt:

$$\begin{aligned} x^\top v^{(\lambda)} &= x^\top (v^{(2)} + \lambda v^{(1)}) \\ &= x^\top v_2 + \lambda x^\top v^{(1)} \\ &= \lambda x^\top v^{(1)} \geq 0. \end{aligned}$$

Wir wissen schon $x^\top v^{(1)} \geq 0$. Wir zeigen gleich, dass sogar $x^\top v^{(1)} > 0$ gilt, und damit muss auch $\lambda \geq 0$ gelten. Widerspruch zum zulässigen $\lambda^* < 0$.

$x^\top v^{(1)} > 0$ folgt daraus, dass G_i eine minimal echte Seite ist und somit nur um eine Dimension größer als $\text{lin. space } F$. Wegen $0 < \alpha_1 < \alpha_2$ wissen wir aber, dass $v^{(1)} \notin \langle \{v^{(2)}\} \cup \text{lin. space } F \rangle$ gilt. Demnach kann keine Gleichheit gelten.

Damit wissen wir, dass $\alpha_2 \neq D_i$ für $\alpha_2 \notin \mathcal{M}(L_{\text{Irr}}^F)$ gilt. Oder mit anderen Worten $\mathcal{MMB} \subseteq \mathcal{M}(L_{\text{Irr}}^F)$ und wir sind fertig mit der zweiten Aussage.

Seien $v^{(1)}, v^{(2)} \in F$ mit $\alpha = \text{supp } v^{(1)} \cap \text{Irr} = \text{supp } v^{(2)} \cap \text{Irr} \neq \emptyset$ und $v_j^{(1)} \neq \lambda v_j^{(2)}$ für alle $\lambda \geq 0$ und je mindestens ein $j \in \alpha$. Wieder können wir uns einen Spielverderber konstruieren, $v \in F$ mit $0 < \text{supp } v \cap \text{Irr} < \alpha$ (wieder wie im Beweis von Lemma 4.8).

Damit ist also $\alpha \notin \mathcal{M}(L_{\text{Irr}}^F)$ nicht minimal und somit $\alpha \neq D_i$ für alle $D_i \in \mathcal{MMB}$. Insbesondere gilt $D_i = D_j \Rightarrow G_i = G_j$, da innerhalb einer minimalen echten Seite die irreversiblen Reaktionen stets in einem konstanten Verhältnis zueinander stehen. \square

Es gibt also eine schöne verbandstheoretische Interpretation der Minimal Metabolic Behaviours von Larhlimi & Bockmayr.



Kapitel 5 – Kurz und knapp

Beschreibung des Stoffwechsels

- ▷ Elementarmoden sind die *minimalen* und zugleich die *unzerlegbaren* Elemente.
- ▷ Die MMBs erlauben eine Zerlegung mit *weniger Erzeugern*.
- ▷ Es gilt $\mathcal{M}(L_{\text{irr}}) = \mathcal{J}(L_{\text{irr}}) = \text{MMB}$.

Anwendungen

- ▷ Viele Umweltbedingungen an die Zelle entsprechen einfachen *Verbandsunterstrukturen*.
- ▷ Projektionen *beschreiben Untersysteme genauer* als das bloße Weglassen der restlichen Reaktionen – dies führt jedoch zu rechenintensiveren und daher langsameren Algorithmen.



KAPITEL 6

Flusskopplung

Kapitelübersicht

6.1	Now and Then – Global ist besser als lokal	91
6.2	Von blockierten und gekoppelten Reaktionen	92
6.2.1	Blockiert VS. Essenziell	92
6.2.2	Gekoppelt	94
6.3	Traditionelle Verfahren zur FCA	98
6.3.1	Aufbau des Algorithmus	98
6.3.2	MaxCoupling: Schranken für $\mathcal{C}(L)$	102
6.3.3	Zeugen beschleunigen FCA	104

§6.1 Now and Then – Global ist besser als lokal

Zu Beginn dieser Arbeit haben wir drei Untersuchungsmethoden des Stoffwechsels kennengelernt: Flux Balance Analysis, elementare Flussmoden und Flux Coupling Analysis.

Auf FBA gehen wir in dieser Arbeit nicht näher ein. Reaktionsgeschwindigkeiten betrachten wir nicht, sondern wir nutzen Verbände, um qualitative Eigenschaften zu beschreiben.

Dafür haben wir minimale und unzerlegbare Elemente kennengelernt, die Verbandsanaloge zu den Elementarmoden. Für den stationären Flusskegel L^F haben wir Algorithmen gesehen, mit denen sie sich sukzessive berechnen lassen – auch für Untermodelle. Zum Abschluss gab es einen Ausflug zu den MMBs, die auch wieder nur für L^F definiert sind.

In diesem und dem nächsten Kapitel wollen wir uns mit der letzten genannten Anwendung, der Kopplung von Reaktionen beschäftigen.


Bei den Grundlagen zu Verbänden haben wir ein drittes besonderes Element im Verband kennengelernt: Das **Maximum 1_L**. Gleich werden wir sehen, es hat eine besonders große Bedeutung für die **FCA** und noch einiges mehr.

§6.2 Von blockierten und gekoppelten Reaktionen

§6.2.1 Blockiert VS. Essenziell

Wir kennen das Maximum des Verbandes. Hier kommen nun biologische Motivationen:

blockierte
Reaktion


 **Definition 6.1 (blockierte Reaktion).**
Sei $L = L^F$ Flussverband zu den metabolischen Bedingungen (S, Irr) . Sei ferner $i \in [n]$.
Dann:
$$i \text{ blockierte Reaktion} :\Leftrightarrow \forall a \in L : i \notin a.$$

Es ist $\mathcal{B}(L)$ die Menge aller blockierten Reaktionen.

Eine blockierte Reaktion liegt wie erwähnt beispielsweise dann vor, wenn eine Reaktion einen Metaboliten als Edukt braucht, der von keinerlei Reaktion produziert wird.

Ein nicht-triviales Beispiel wollen wir uns hier anschauen:

blockierte
Reaktionen

 **Beispiel 6.1 (blockierte Reaktion).**
In Abbildung 6.1 vier Reaktionen zu sehen, bei denen alle stöchiometrischen Koeffizienten 1 sind. Betrachten wir nun die gestrichelte Reaktion:
Im stationären Flussverband L^F ist sie blockiert. Sie konsumiert A und B, die durch die Eingangsreaktion aufgebaut werden müssen. Zudem produziert sie C, dass durch die Ausgangsreaktion abgebaut werden muss. Dafür benötigen wir zusätzlich B. Also muss die Geschwindigkeit der Eingangsreaktion größer sein als die der gestrichelten. Wir produzieren nun einen Überschuss an A, den wir nicht abbauen können, $A \rightarrow C$ ins Spiel zu bringen. Nun können wir aber C nicht abbauen, ohne in einem argumentativen Kreislauf zu landen.
Anders im Fall des logischen Verbandes: Hier ist auch ein gleichzeitiges Stattfinden aller Reaktionen oder eines von allen außer $A \rightarrow C$ denkbar.

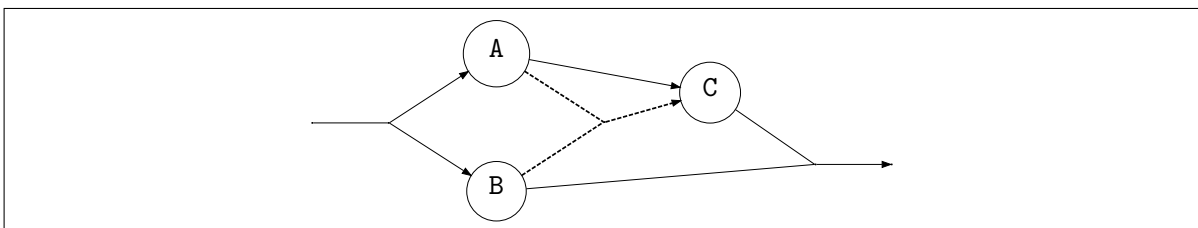


Abbildung 6.1.: Die gestrichelte Reaktion ist blockiert in L^F , aber nicht in L^{01} .



Definition 6.2 (essenzielle Reaktion).

Sei L Flussverband zu den metabolischen Bedingungen (S, Irr) . Sei ferner $k_1 \in [n]$. Dann:

$$k_1 \text{ essenzielle Reaktion} :\Leftrightarrow \forall \mathbf{a} \in L \setminus \{0\} : k_1 \in \mathbf{a}.$$

Es ist $\mathcal{ER}(L)$ die Menge aller essenziellen Reaktionen.

Essenzielle Reaktionen spielen eine wichtige Rolle für den Stoffwechsel von Zellen: Sollte nur eine einzige von ihnen nicht stattfinden können, so kann die gesamte Zelle in keinen stationären Zustand übergehen. Der Wegfall einer Reaktion kann beispielsweise durch einen genetischen Defekt oder auch das bewusste Einsetzen eines Hemmstoffes passieren.



Bemerkung 6.1.

Die Mengen $\mathcal{B}(L)$ und $\mathcal{ER}(L)$ sind identisch für allgemeine metabolische Systeme (L, Rev) mit beliebigem metabolischen Vereinigungsverband definiert.

Ob es blockierte oder essenzielle Reaktionen gibt, kann man an den unzerlegbaren bzw. den minimalen Elementen des Verbandes ablesen. Offensichtlich gilt:



Satz 6.1.

Sei (L, Rev) metabolisches System. Seien ferner $i, k_1 \in [n]$. Dann:

$$\begin{aligned} i \in \mathcal{B}(L) &\Leftrightarrow \forall \mathbf{b} \in \mathcal{J}(L) : i \notin \mathbf{b} \Leftrightarrow L_{\parallel\{i\}} = \{0\} \Leftrightarrow L_{\perp\{i\}} = L, \\ k_1 \in \mathcal{ER}(L) &\Leftrightarrow \forall \mathbf{e} \in \mathcal{M}(L) : k_1 \in \mathbf{e} \Leftrightarrow L_{\parallel\{k_1\}} = L \Leftrightarrow L_{\perp\{k_1\}} = \{0\}. \end{aligned}$$

Dem aufmerksamen Leser ist vermutlich nicht entgangen, dass blockierte Reaktionen und Verbandsmaximum zwei Seiten derselben Medaille sind. Wir halten fest:



Bemerkung 6.2.

Sei (L, Rev) metabolisches System mit Maximum 1_L . Dann:

$$[n] = 1_L \dot{\cup} \mathcal{B}(L).$$

Diese Feststellung kombiniert mit Lemma 3.7 führt uns zu:



Bemerkung 6.3.

$$\mathcal{B}(L^{01}) \subseteq \mathcal{B}(L^F).$$

1_L ist nicht nur nach Verbandsordnung maximal, es ist auch das eindeutig bestimmbare Element maximaler Kardinalität im Vereinigungsverband. Deshalb lässt sich Algorithmus *CalculateBlock* zu seiner Berechnung im klassischen Flussverband L^F nutzen.

Algorithm *CalculateBlock*(S, Irr)

Output: $\mathcal{B}(L^F)$.

(* solves 1 MILP *)

1.

$$\begin{aligned} & \text{maximize} && \sum_{j=1}^n a_j \\ & \text{s.t.} && S \cdot (v^{(+)} - v^{(-)}) = 0 \\ & && v_i^{(-)} = 0 && \forall i \in \text{Irr} \\ & && a_i^{(+)} + a_i^{(-)} = a_i && \forall i \in [n] \\ & && a_i^{(\pm)} \leq v_i^{(\pm)} \leq M \cdot a_i^{(\pm)} && \forall i \in [n] \\ & && a_i^{(\pm)}, a_i \in \{0, 1\} && \forall i \in [n] \end{aligned}$$


2. $1_L \leftarrow \{j \in [n] \mid a_j = 1\}$
3. $\mathcal{B}(L) \leftarrow \{j \in [n] \mid a_j = 0\}$
4. **return** $\mathcal{B}(L)$

§6.2.2 Gekoppelt

Im Stoffwechsel von Zellen kommt es manchmal vor, dass Reaktionen nur gemeinsam vorkommen. Eine Ursache hierfür kann sein, dass es einen Metaboliten gibt, der nur von einer einzigen Reaktion aufgebaut und auch nur von einer abgebaut wird. Ein stationärer Zustand ist nur möglich, wenn beide Reaktionen fest aneinander gekoppelt auftreten.

Wir werden uns hier nur mit der Kopplung von Reaktionen aneinander beschäftigen. Diese Flusskopplung wurde ursprünglich für das Standardmodell definiert (Burgard et al. 2004). Leser, die sich für die Kopplung von Metaboliten interessieren, können beispielsweise bei Becker, Price & Palsson (2006) nachlesen.

Fluss-
kopplung



Definition 6.3 (gekoppelte Reaktion).

Sei $L = L^F$ Flussverband zu den metabolischen Bedingungen (S, Irr) mit Flusskegel F . Seien ferner $i, j \in 1_L = [n] \setminus \mathcal{B}$. Dann:

$$\begin{aligned} i \xrightarrow{=0} j & \Leftrightarrow \forall a \in L : i \notin a \Rightarrow j \notin a, \\ i \longleftrightarrow j & \Leftrightarrow \forall a \in L : i \notin a \Leftrightarrow j \notin a, \\ i \longleftrightarrow_{\lambda} j & \Leftrightarrow \exists \lambda_{i,j} \neq 0 \forall v \in F : v_i = \lambda_{i,j} v_j. \end{aligned}$$

Gilt $i \xrightarrow{=0} j$, so sagen wir, dass j an i gekoppelte Reaktion ist. Es ist $\mathcal{C} = \{(i, j) \in [n]^2 \mid i \xrightarrow{=0} j\}$ die Menge aller Flusskopplungen.

Ignorieren wir die Kopplung $\longleftrightarrow_\lambda$, so können wir in jedem Vereinigungsverband arbeiten:

 **Bemerkung 6.4.**

$i \xrightarrow{=0} j$ und $i \longleftrightarrow j$ sind in allgemeinen metabolischen Systemen (L, Rev) ebenso definiert.


 **Bemerkung 6.5.**

\longleftrightarrow und $\longleftrightarrow_\lambda$ sind Äquivalenzrelationen auf L .

Äquivalenz-
relation

Das ist nicht nur für den Mathematiker interessant, sondern kann auch praktisch zur Kompression des Systems genutzt werden. Kennt man die Flusskopplungsbeziehungen innerhalb des Verbandes, kann man von nun an alle Elemente abspeichern, in dem man nicht mehr die genauen Zahlen, sondern stattdessen die Äquivalenzklassen speichert. Wir werden gleich sehen, wie man das im Flussverband L^F schon vor Berechnung der einzelnen Elemente durch Kompression der Eingabe schafft.

Die Zusammenfassung von Äquivalenzklassen vereinfacht auch die visuelle Darstellung der Ergebnisse:

 **Beispiel 6.2** (Flusskopplung).

Im Netzwerk *etc.purple* werden die Metaboliten *NADH*, *Succ* und *ATP* nur durch je ein Reaktionenpaar produziert/konsumiert (s. Abbildung 2.1). Entsprechend sind die Reaktionen *NADH_in* und *NADH-DH*, *Succ_in* und *Succ-DH* sowie *ATPdrain* und *ATPSynthase* fest gekoppelt.

Zudem ist die *Photo-Reaktion* an *bc1* und *Q-oxidase* an einkommenden Sauerstoff *O2_in* gekoppelt. *c2-oxidase* hängt sogar von beiden, also *bc1* und *O2_in* ab.

Die Kopplungen sind in der Landkartendarstellung 2.5 zusammengefasst.

Fluss-
kopplung

Nun noch eine Beobachtung, die schon Kaleta et al. (2009, Anhang) gemacht haben.

 **Bemerkung 6.6.**

Sei L^F Flussverband zu den Bedingungen (S, Irr) . Dann gilt für unblockierte Reaktionen $i, j \in 1_L$ und die Projektion auf $\{i, j\}$:

$$\begin{aligned} i \xrightarrow{=0} j &\Leftrightarrow L_{\{i,j\}} \subseteq \{\emptyset, \{i\}, \{i, j\}\} \Leftrightarrow \mathcal{M}(L_{\{i,j\}}) \in \{\{\{i\}\}, \{\{i, j\}\}\}, \\ i \longleftrightarrow j &\Leftrightarrow L_{\{i,j\}} = \{\emptyset, \{i, j\}\} \Leftrightarrow \mathcal{M}(L_{\{i,j\}}) = \{\{i, j\}\}, \\ i \longleftrightarrow_\lambda j &\Leftrightarrow \exists \lambda_{i,j} \forall e \in E : e_i = \lambda_{i,j} e_j. \end{aligned}$$

 **Korollar 6.1.**

Aus Bemerkung 6.6 kombiniert mit dem Algorithmus *MinimalConstrainedElements* folgt sofort ein Verfahren, mit dem man $\mathcal{C}(L^F)$ durch das Lösen von maximal $n(n-1)$ MILPs bestimmen kann.

Jetzt ist es an der Zeit, ein Versprechen einzulösen. Wir hatten den ominösen Null-Verband $L^{F,0}$ eingeführt, ohne eine überzeugende Erklärung seiner Modellierungseigenschaften zu nennen. Schauen wir deshalb, was wir über Flusskopplung im Null-Verband sagen können. Hierbei nutzen wir die *Kernelmatrix*, deren Spalten einer Basis des Kerns von S entsprechen.



Satz 6.2.

Sei $L^{F,0}$ Null-Verband zu den metabolischen Bedingungen (S, Irr) . Sei K Kernelmatrix von S und $i, j \in [n] \setminus \mathcal{B}(L^{F,0})$. Dann sind äquivalent:

$$i \longleftrightarrow_{\lambda} j \text{ (in } L^{F,0}) \Leftrightarrow i \xrightarrow{=0} j \text{ (in } L^{F,0}) \Leftrightarrow \exists \lambda_{i,j} : r_i = \lambda_{i,j} r_j,$$

wobei r_1, \dots, r_n die Zeilenvektoren von K sind.

Beweis. Sei $\neg(i \longleftrightarrow_{\lambda} j)$. Dann gibt es $x^{(1)}, x^{(2)} \in \text{Kern}(S)$ mit $x_i^{(1)} \cdot x_j^{(2)} \neq x_j^{(1)} \cdot x_i^{(2)}$.

Betrachte $\tilde{x} = x_i^{(1)} \cdot x^{(2)} - x^{(1)} \cdot x_i^{(2)}$. Es ist $\tilde{x}_i = 0 \neq \tilde{x}_j$. Also ist $(i, j) \notin \mathcal{C}(L^{F,0})$.

Der Rest ist trivial: Sei $r_i = \lambda_{i,j} r_j$ mit $\lambda_{i,j} \neq 0$ (da i, j nicht blockiert). Dann gilt für $x = K \cdot y \in \text{Kern}(S)$:

$$x_i = 0 \Rightarrow r_i \cdot y = 0 \Rightarrow 0 = \frac{1}{\lambda_{i,j}} r_i \cdot y = r_j \cdot y = x_j.$$

Zum Abschluss noch mal Kontraposition: Sei $r_i \neq \lambda_{i,j} r_j$ für $\lambda_{i,j} \in \mathbb{R}^*$ und o. B. d. A. $K_{i,1} \cdot K_{j,2} \neq K_{i,2} \cdot K_{j,1}$. Dann sind die beiden Spaltenvektoren $K_{\cdot,1}, K_{\cdot,2} \in \text{Kern}(S)$ Zeugen für $\neg(i \longleftrightarrow_{\lambda} j)$. \square

Kennt man die Kernelmatrix der stöchiometrischen Matrix, so kann man an ihr direkt die Flusskopplung im Null-Verband ablesen. Das greifen wir später auf.

Im nächsten Abschnitt führen wir alles zusammen und beschleunigen so das klassische Vorgehen zur FCA, die einzelnen Reaktionspaare auf Kopplung zu prüfen. Vorher lernen wir jedoch noch einen anderen Algorithmus für L^F kennen, der sich aus folgendem Satz ergibt:



Satz 6.3.

Sei (L, Rev) metabolisches System und $i, j \in 1_L = [n] \setminus \mathcal{B}(L)$. Dann:

$$i \xrightarrow{=0} j \Leftrightarrow \forall b \in \mathcal{J}(L) : j \in b \Rightarrow i \in b \Leftrightarrow i \in \mathcal{ER}(L_{\parallel\{j\}}) \Leftrightarrow j \in \mathcal{B}(L_{\perp\{i\}}).$$

Den recht einfachen Beweis, dass Flusskopplung durch die unzerlegbaren Elemente bestimmt ist, findet man in (Goldstein 2009), wo er für den Flusskegel und entsprechend die elementaren Flussmoden geführt ist.

Dies führt uns zu einem Algorithmus, mit dem man $\mathcal{C}(L^F)$ durch das Lösen von $n + 1$ MILPs bestimmen kann. Denn es gilt:



Bemerkung 6.7 (Zerlegung von $[n]$).

Sei $L = L^F$ Flussverband, $i \in 1_L = [n] \setminus \mathcal{B}$. Dann:

$$[n] = 1_{\perp\{i\}} \dot{\cup} \mathcal{C}_i \dot{\cup} \mathcal{B}(L).$$

Wobei:

$$\begin{aligned} \mathcal{C}(L^F) &= \bigcup_{i \in 1_L} \{(i, j) \mid j \in \mathcal{C}_i\}, \\ \mathcal{C}_i &= \{j \in 1_L \mid i \stackrel{=0}{\rightarrow} j\} \supseteq \{i\}. \end{aligned}$$

Wieder ist $1_{\perp\{i\}}$ Maximum nach Kardinalität in $L_{\perp\{i\}}$ und somit greift Algorithmus *FewStepsFCA*.

Algorithm *FewStepsFCA*($S, \text{Irr}, \mathcal{B}$)

Output: $\mathcal{C}(L^F)$.

(* solves n MILPs *)

1. $\mathcal{C}(L^F) \leftarrow \emptyset$
2. **for** $i \in [n] \setminus \mathcal{B}(L^F)$
3. **do**

$$\begin{aligned} &\text{maximize} && \sum_{j=1}^n a_j \\ &\text{s.t.} && S \cdot (v^{(+)} - v^{(-)}) = 0 \\ &&& v_i^{(-)} = 0 && \forall i \in \text{Irr} \\ &&& a_i^{(+)} + a_i^{(-)} = a_i && \forall i \in [n] \\ &&& a_i^{(\pm)} \leq v_i^{(\pm)} \leq M \cdot a_i^{(\pm)} && \forall i \in [n] \\ &&& a_i^{(\pm)}, a_i \in \{0, 1\} && \forall i \in [n] \\ &&& a_i = 0 \end{aligned}$$

4. $1_{\perp\{i\}} \leftarrow \{(i, j) \mid a_j = 1, j \in [n]\}$
5. $\mathcal{C}_i \leftarrow \{(i, j) \mid a_j = 0, j \in [n] \setminus \mathcal{B}(L^F)\}$
6. $\mathcal{C}(L^F) \leftarrow \mathcal{C}(L^F) \cup \mathcal{C}_i$
7. **return** $\mathcal{C}(L^F)$

§6.3 Traditionelle Verfahren zur FCA

FCA Die Berechnung von $\mathcal{C}(L^F)$ und die Bestimmung des Kopplungstyps fasst man gewöhnlich als **Flusskopplungsanalyse** (Flux Coupling Analysis, **FCA**) zusammen. Der Einfachheit halber werden wir diese Abkürzung auch beim Verband verwenden, auch wenn wir dort keine feste Kopplung identifizieren können. Der Begriff stammt ebenso wie die Definition der Flusskopplung von Burgard et al. (2004).

Wir haben oben gezeigt, wie man $\mathcal{C}(L^F)$ durch das Lösen von $n + 1$ MILPs ermitteln kann. In diesem Abschnitt wollen wir zeigen, wie man das auch durch das Lösen von $2n^2$ LPs erreichen kann – oder n^2 geeignete Tests für einen beliebigen Verband, der als Modellgrundlage dient. Später zeigen wir Möglichkeiten, die Anzahl an Tests stark einzuschränken. Dabei gehen wir zur Vorbereitung u. a. auf Arbeiten von Larhlimi sowie Marashi und David ein.

Neben ihren Laufzeitverbesserungen haben sich Marashi, David & Bockmayr auch mit dem Fakt beschäftigt, dass unsere Modelle nicht vollständig sind (Marashi & Bockmayr 2011, Marashi, David & Bockmayr 2012b). Was für einen Einfluss haben fehlende Reaktionen auf die Flusskopplung? Sehr anschaulich diskutieren sie dort Aussagen, die sich direkt aus Korollar 4.1 ergeben: Es ist $L_{\subseteq R} \subseteq L_R$ Unterverband und somit $1_{L_{\subseteq R}} \leq 1_{L_R}$. Kennen wir nur die Reaktionen in R , so bleibt uns nichts anderes übrig als $L_{\subseteq R}^F$ zu betrachten. Jetzt ist es aber möglich, dass $j \notin 1_{(L_{\subseteq R})_{\perp i}}$ gilt, aber $j \in 1_{(L_R)_{\perp i}}$. Dann sagen wir jedoch $i \xrightarrow{=0} j$ (in L_R), obwohl dies in der Realität nicht der Fall ist.

Marashi & Bockmayr weisen darauf hin, dass es einen Zusammenhang zwischen Genexpression und Flusskopplungstyp geben kann. Sie motivieren schließlich, Diskrepanzen zwischen Genexpression und FCA-Vorhersage zu nutzen, um fehlende Reaktionen als Lücken im Modell zu finden (Marashi & Bockmayr 2011). Notebaart, Teusink, Siezen & Papp (2008) befinden zusätzlich, dass es einen starken Zusammenhang zwischen dem Kopplungsverhältnis zweier Reaktionen mit ihrer (Co-)Regulierung gibt. Dafür vergleichen sie Experimentalergebnisse von *Escherichia coli* mit FCA-Ergebnissen eines genomweiten Netzwerkes.

Damit im Hinterkopf betrachten wir nun die klassischen FCA-Algorithmen verbandstheoretisch. Schnell kommen wir zu zwei Einzelproblemen, die sich viel einfacher lösen lassen. Am Ende dieses Kapitels stellen wir ein Algorithmenskelett vor, an dem wir alle möglichen Optimierungen aufhängen.

§6.3.1 Aufbau des Algorithmus

Unser Algorithmus ist mehrstufig aufgebaut. In einem ersten Schritt bestimmen wir die vollständig gekoppelten Reaktionen.


Satz 6.4.

Sei L Flussverband zu den metabolischen Bedingungen (S, Irr) . Seien ferner $\mathcal{B}(L^F) = \emptyset$ und K Kernmatrix von S . Dann:

$$i \longleftrightarrow_{\lambda} j \text{ (in } L^F) \Leftrightarrow i \longleftrightarrow_{\lambda} j \text{ (in } L^{F,0}).$$

Beweis. Wir behalten die Äquivalenzen aus Satz 6.2 im Hinterkopf (das geht, da $1_{L^F} \subseteq 1_{L^{F,0}}$) und zeigen:

Sei $\neg(i \longleftrightarrow_{\lambda} j \text{ (in } L^{F,0}))$, also $(i, j) \notin \mathcal{C}(L^{F,0})$. Dann gibt es $x \in \text{null } S$ mit $x_i = 0$ und $x_j = 1$.

Sei nun $I = \{k \in \text{Irr} \mid x_k < 0\}$. Es ist L^F Verband, also gibt es ein Element $v^{(\max)} \in F$ mit $\text{supp}(v^{(\max)}) = 1_{L^F} = [n]$. Offensichtlich ist $v_k^{(\max)} > 0$ für alle $k \in I \subseteq \text{Irr}$. Für $\lambda > 0$ groß genug gilt somit $v^{(\lambda)} = x + \lambda v^{(\max)} \in F$.

Angenommen $i \longleftrightarrow_{\lambda} j \text{ (in } L^F)$ zum Koeffizienten $\lambda_{i,j} \neq 0$. Dann gilt auch $v_i^{(\max)} = \lambda_{i,j} \cdot v_j^{(\max)}$ und $v_i^{(\lambda)} = \lambda_{i,j} v_j^{(\lambda)}$. Daraus folgt aber:

$$\lambda_{i,j} \cdot (\lambda v_j^{(\max)}) = \lambda v_i^{(\max)} = v_i^{(\lambda)} = \lambda_{i,j} \cdot v_j^{(\lambda)} = \lambda_{i,j} (1 + \lambda v_j^{(\max)}).$$

Widerspruch zu $\lambda_{i,j} \neq 0$. Somit wissen wir $i \longleftrightarrow_{\lambda} j \text{ (in } L^F) \Rightarrow i \longleftrightarrow_{\lambda} j \text{ (in } L^{F,0})$

Die Rückrichtung ist klar. □

Dass man vollständige Kopplung an der Kernel-Matrix ablesen kann, wurde schon lange vermutet und schließlich von Larhlimi, David, Selbig & Bockmayr (2012) gezeigt. Unser Beweis hier orientiert sich an ihrem Vorgehen, weist aber zwei wesentliche Unterschiede auf:

Null-Verband Wir haben die Chance ergriffen, einen Null-Verband zu definieren (auch wenn der zum Beweis des Satzes nicht notwendig ist).

Maximum Im Originalbeweis werden für alle Reaktionen $k \in I$ einzelne Flussvektoren zu Hilfe gezogen, um durch Summation ein spezielles \tilde{x} zu konstruieren, das die gleiche Funktion wie unser $v^{(\lambda)}$ hat. Die Verbandstheorie gewährleistet uns jedoch die Existenz von $v^{(\max)}$ und erspart uns somit das Hantieren mit größeren Summen im Beweis.

Larhlimi et al. (2012) nutzen Satz 6.4 mit der Darstellung aus Satz 6.2 als Grundlage für einen Algorithmus. Leicht abgewandelt kommen wir damit zu Algorithmus *FindFullyCouples*.

Algorithm *FindFullyCouples*(S, Irr)

Output: $\mathcal{C}_\lambda = \{(i, j, \lambda_{i,j}) \mid i \longleftrightarrow_\lambda j\}$.

(* $\mathcal{O}(n^2)$ point operations + Kernel calculation *)

1. $\mathcal{C}_\lambda \leftarrow \emptyset$
2. $K \leftarrow \text{CalculateKernel}(S)$
3. **for** $1 \leq i < j \leq n$
4. **do if** $i, j \notin \mathcal{B}$
5. **then** $\lambda_{i,j} \leftarrow \frac{K[j, \cdot] \cdot K[i, \cdot]^T}{K[j, \cdot] \cdot K[j, \cdot]^T}$
6. **if** $|K(i, \cdot) - \lambda_{i,j} \cdot K(j, \cdot)| < \epsilon$
(* $\epsilon > 0$ tolerance *)
7. **then** $\mathcal{C}_\lambda \leftarrow \mathcal{C}_\lambda \cup \{(i, j, \lambda_{i,j})\}$
8. **return** \mathcal{C}_λ

In meiner Diplomarbeit (Goldstein 2009) habe ich gezeigt, wie man das Eingangssystem komprimieren kann, wenn man \mathcal{C}_λ kennt. Hierfür wandeln wir Algorithmus *FindFullyCouples* leicht ab und kommen so zu Algorithmus *CompressNetwork*.

Algorithm *CompressNetwork*($S, \text{Irr}, \mathcal{B}(L^F), K$)

Input: K kernel matrix.

(* $\mathcal{O}(n^2)$ point operations *)

1. $\text{classes} \leftarrow \emptyset$
2. **for** $1 \leq i \leq n$
3. **do if** $i \notin \mathcal{B}(L^F)$
4. **then found** $\leftarrow \text{false}$
5. **for** $1 \leq j \leq |\text{classes}|$
6. **do if** $\neg \text{found}$
7. **then** $\mu[i] \leftarrow \frac{K[\text{classes}[j][1], \cdot] \cdot K[i, \cdot]^T}{K[\text{classes}[j][1], \cdot] \cdot K[\text{classes}[j][1], \cdot]^T}$
8. **if** $\|K[i, \cdot] - \mu[i] \cdot K[\text{classes}[j][1], \cdot]\| < \epsilon$
9. **then** $\text{classes}[j][|\text{classes}[j]| + 1] \leftarrow i$
10. **found** $\leftarrow \text{true}$
11. **if** $\neg \text{found}$
12. **then** $\mu[i] \leftarrow 1$
13. $\text{classes}[|\text{classes}| + 1][1] \leftarrow i$
14. **for** $\text{class} \in \text{classes}$
15. **do switch** $\leftarrow \text{false}$
16. **for** $j \in \text{class}$
17. **do if** $\neg \text{switch}$ and $j \in \text{Irr}$ and $\mu[j] < 0$
18. **then switch** $\leftarrow \text{true}$
19. **for** k in class
20. **do** $\mu[k] \leftarrow -\mu[k]$
21. $\tilde{S} \leftarrow 0 \in \mathbb{R}^{m \times |\text{classes}|}, \tilde{\text{Irr}} \leftarrow \emptyset$
22. **for** $1 \leq i \leq |\text{classes}|$
23. **do if** $\text{classes}[i] \cap \text{Irr} \neq \emptyset$
24. **then** $\tilde{\text{Irr}} \leftarrow \{i\} \cup \tilde{\text{Irr}}$
25. **for** $j \in \text{classes}[i]$
26. **do** $\tilde{S}[:, i] \leftarrow \tilde{S}[:, i] + \mu[j] \cdot S[:, j]$
27. **return** $(\tilde{S}, \tilde{\text{Irr}}, \text{classes}, \mu)$

Es kann passieren, dass der Kompressionsalgorithmus Nullzeilen in der Matrix erzeugt. Diese können getrost verworfen werden.



Beispiel 6.3 (komprimiertes Netzwerk).

Das Netzwerk *etc_purple* verfügt über drei fest gekoppelte Reaktionenpaare. Fügen wir diese zusammen, wie in Algorithmus *CompressNetwork* vorgestellt, so erhalten wir Abbildung 6.2.

Kompression

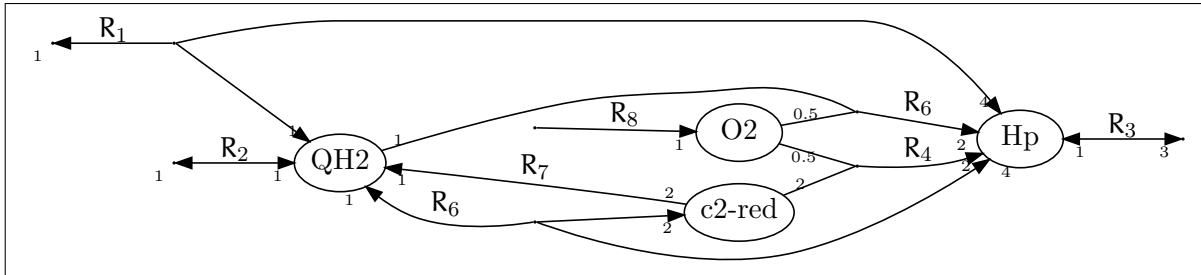


Abbildung 6.2.: Das komprimierte Netzwerk zu *etc_purple*.

Das komprimierte Netzwerk enthält nach Satz 6.4 keine fest gekoppelten Reaktionen; es reicht also, Flusskopplung in verbandstheoretischer Sicht festzustellen. Algorithmus *ClassicFCA* stellt damit ein Grundgerüst zur FCA dar.

Algorithm *ClassicFCA*(S, Irr)

Output: $\mathcal{C}(L^F)$.

(* $\mathcal{O}(n^4)$ point operations + $\mathcal{O}(n^2)$ LPs + Kernel calculation *)

1. $\mathcal{C}(L^F) \leftarrow \emptyset, \mathcal{C}_\lambda \leftarrow \emptyset$
2. $\mathcal{B} \leftarrow \text{CalculateBlock}(S, Irr)$
3. $K \leftarrow \text{CalculateKernel}(S)$
4. **if** $K \notin \mathbb{R}^{n \times 0}$
5. **then** $(\tilde{S}, \tilde{Irr}, \text{classes}, \mu) \leftarrow \text{CompressNetwork}(S, Irr, \mathcal{B})$
6. $\text{couplequeue} \leftarrow \text{MaxCoupling}(\tilde{S}, \tilde{Irr})$
7. $\tilde{\mathcal{C}} \leftarrow \emptyset$
8. **while** $\text{couplequeue} \neq \emptyset$
9. $(\text{couples}, \text{nocouples}) \leftarrow \text{ValidateCouples}(\text{couplequeue}, \tilde{S}, \tilde{Irr})$
10. $\text{couplequeue} \leftarrow \text{couplequeue} \setminus (\text{couples} \cup \text{nocouples})$
11. $\tilde{\mathcal{C}} \leftarrow \tilde{\mathcal{C}} \cup \text{couples}$
12. **for** $(\tilde{i}, \tilde{j}) \in \tilde{\mathcal{C}}$
13. **do** $\mathcal{C}(L^F) \leftarrow \mathcal{C}(L^F) \cup \{(i, j) \mid i \in \text{classes}[\tilde{i}], j \in \text{classes}[\tilde{j}]\}$
14. **for class** $\in \text{classes}$
15. **do** $\mathcal{C}(L^F) \leftarrow \mathcal{C}(L^F) \cup \{(i, j) \mid i \neq j, i \in \text{class}, j \in \text{class}\}$
16. $\mathcal{C}_\lambda \leftarrow \mathcal{C}_\lambda \cup \{(i, j, \frac{\mu[i]}{\mu[j]}) \mid i \neq j, i \in \text{class}, j \in \text{class}\}$
17. **return** $(\mathcal{C}(L^F), \mathcal{C}_\lambda)$

An diesem Gerüst können wir uns für die FCA im L^F orientieren, es kann aber auch als

Grundlage für jeden anderen Verband genutzt werden – notfalls durch Streichung der Kompression.

Je nachdem, wie man die Funktionen `MaxCoupling` und `ValidateCouples` wählt, landet man bei verschiedenen Fassungen des klassischen Verfahrens. Oder bei viel, viel schnelleren.¹ Diese beiden Funktionen sind auch das Einstiegstor für andere Verbände. Oder anders formuliert: Nur bei diesen beiden Funktionen kommt es auf das Modell an, das wir nutzen – das Gerüst bleibt identisch.

Eine besonders wichtige Frage ist, welche Datenstruktur wir nutzen wollen. `couplequeue` kann aus bis zu n^2 Elementen bestehen, die wir in bis zu n^2 Schleifenaufrufen filtern.

Häufig wird dazu eine Matrix genutzt. Das liegt nahe, da wir Reaktionenpaare betrachten. Ohne die Vor- und Nachteile dieses Vorgehens zu diskutieren, schließen wir nun die letzten Lücken im Gerüst und stellen damit einen funktionierenden und schnellen Algorithmus zur FCA im L^F zur Verfügung. Viel übersichtlicher und dank ähnlicher Techniken gleich schnell geht es dann im Folgekapitel zu.

§6.3.2 MaxCoupling: Schranken für $\mathcal{C}(L)$

Larhlimi & Bockmayr (2006) diskutieren, was es für Unterschiede bei reversiblen Reaktionen geben kann – aus mathematischer Sicht. Dafür wird eine disjunkte Zerlegung von $[n]$ eingeführt. Die nutzen sie als Grundlage für genauere Untersuchungen, welche Kopplungen untereinander möglich sind.

Wir wollen an dieser Stelle eine schwächere Aussage formulieren, die sich aus ihrer ableitet, und dann in allgemeinen Verbandsmodellen gilt:

Frev, NFrev



Satz 6.5 (Frev, NFrev).

Sei (L, Rev) metabolisches System. Definiere $\text{Frev}, \text{NFrev} \subseteq [n]$ wie folgt:

$$\begin{aligned} \text{Frev} &:= 1_{\perp \text{Irr}}, \\ \text{NFrev} &:= 1_L \setminus \text{Frev}. \end{aligned}$$

Dann gilt:

1. $\mathcal{C}(L) \subseteq \text{Frev}^2 \dot{\cup} \text{NFrev}^2$,
2. $(i, j) \in \mathcal{C}(L^F) \cap \text{Frev}^2 \Rightarrow i \longleftrightarrow_{\lambda} j$ (in L^F).

Beweis. Die erste Aussage zeigen wir in einer viel stärkeren Form im nächsten Kapitel.

Die zweite ergibt sich wieder aus unseren Beobachtungen des Null-Verbandes: $\text{Frev} = 1_{\perp \text{Irr}}$


¹In der Realität viel schneller als von der theoretischen Laufzeit. Da unsere später vorgestellte Implementierung `ValidateCouples` eine Laufzeit von $\mathcal{O}(n^2)$ Punktoperationen hat und dies jeweils nach bis zu $\mathcal{O}(n^2)$ LP-Solver-Aufrufen geschieht, ergibt sich eine theoretische Laufzeit von $\mathcal{O}(n^4)$. Wie wir im Anhang sehen werden, sind solche Grenzen jedoch fern von der Realität.

sind die unblockierten Reaktionen im Null-Verband, der sich aus den reversiblen Spalten S_{Rev} ergibt und somit $L^{\text{FRev}} = L^{\text{FRev},0}$. \square

Da in unserem komprimierten System nach Beispiel 3.4 keine festen Kopplungen mehr vorkommen, können wir $\text{MaxCoupling} = \text{NFrev}^2$ wählen.

Untere Schranken für $\mathcal{C}(L)$ stellen Paare dar, von denen wir mit Sicherheit wissen, dass sie in $\mathcal{C}(L)$ enthalten sind.

Um solche Mengen zu bestimmen, ist folgendes Lemma praktisch:



Lemma 6.1.

Seien $\mathbf{U} \subseteq L$ Vereinigungsverbände über $[n]$ und \mathbf{U} Unterhalbverband von L . Dann:

$$\mathcal{B}(L) \subseteq \mathcal{B}(\mathbf{U}),$$

$$\mathcal{C}(L) \setminus \mathcal{B}(\mathbf{U})^2 \subseteq \mathcal{C}(\mathbf{U}).$$


Beweis. Wir besinnen uns auf Lemma 4.1 und erhalten sofort die erste Aussage.

Für die zweite Aussage betrachten wir noch einmal die Gleichung 6.7 und erhalten für $i \in 1(\mathbf{U})$:

$$\begin{aligned} [n] &= 1_{L \setminus \{i\}} \dot{\cup} \mathcal{C}_i(L) \dot{\cup} \mathcal{B}(L) = 1_{\mathbf{U} \setminus \{i\}} \dot{\cup} \mathcal{C}_i(\mathbf{U}) \dot{\cup} \mathcal{B}(\mathbf{U}) \\ &\stackrel{4.1}{\Rightarrow} \mathcal{C}_i(L) \dot{\cup} \mathcal{B}(L) \subseteq \mathcal{C}_i(\mathbf{U}) \dot{\cup} \mathcal{B}(\mathbf{U}) \\ &\Rightarrow \mathcal{C}_i(L) \subseteq \mathcal{C}_i(\mathbf{U}) \dot{\cup} \mathcal{B}(\mathbf{U}). \end{aligned}$$

\square

Nach Beispiel 3.6 ist $L^{\text{F}} \subseteq L^{01}$ Unterhalbverband. Somit greift Lemma 6.1 und wir wissen:



Korollar 6.2 (L^{01} liefert untere Schranke).

$$\mathcal{C}(L^{01}) \setminus (\mathcal{B}(L^{\text{F}}))^2 \subseteq \mathcal{C}(L^{\text{F}}).$$

Kennt man also die Flusskopplungsergebnisse vom logischen Verband, so kann man daraus auf Kopplungen im Flusskegel schließen.

Da das Lösen eines SAT-Problems schon ohne zusätzliche Maximalitätsbedingung an die Lösung NP-schwer ist, ist eine tatsächliche Verbesserung der Laufzeit fraglich. Jedoch sollte dieser Ansatz nicht unerwähnt bleiben, um zu zeigen, wie Oberhalbverbände zur Untersuchung von L^{F} genutzt werden können.

§6.3.3 Zeugen beschleunigen FCA

Die üblichen Algorithmen zur FCA validieren die Paare aus `couplequeue` einzeln und prüfen für jedes einzelne geordnete Paar, ob es sich dabei um eine Reaktionskopplung handelt. Hierfür wird meistens das LP *TestCouple* gelöst.

Algorithm *TestCouple*(i, j)

(* solves 1 or 2 LPs *)

1. `notacouple` \leftarrow `false`
2. $\mathbf{a}_{\pm 1} \leftarrow 0$
3. $\sigma \leftarrow 1$
4. **while** $\sigma \geq -1$ **and** \neg `notacouple`
5. **do**

$$\begin{aligned} & \text{maximize} && \sigma \cdot v_j \\ & \text{s.t.} && S \cdot v = 0 \\ & && v_k \geq 0 \quad \forall k \in \text{Irr} \\ & && v_i = 0 \end{aligned}$$

6. $\mathbf{a}_\sigma \leftarrow \{k \in [n] \mid v_k \neq 0\}$
7. `notacouple` $\leftarrow j \in \mathbf{a}_\sigma$
8. $\sigma \leftarrow \sigma - 2$
9. **return** \neg `notacouple`

David, Marashi, Larhlmi, Mieth & Bockmayr (2011) beschreiben, wie man dieses LP leicht umformulieren kann und somit das Problem mathematisch-ideell anders angeht. Anstatt den Wert der unblockierten Reaktion zu maximieren, um herauszufinden, ob er überhaupt von 0 verschieden sein kann, prüfen sie, ob es einen Vektor mit $v_j = 0$ und $|v_i| \geq 1$ gibt, egal wie dieser lautet. Dadurch erreichen sie ein Laufzeitverbesserung, da der gewählte LP-Solver nicht mehr die maximale Reaktionsgeschwindigkeit bestimmen muss, sondern nur, ob es einen positiven Fluss durch die entsprechende Reaktion geben kann.

Als beiläufigen Abschluss dieses Kapitels wollen wir eine letzte Verbesserung vorstellen, die die Laufzeit enorm verkürzt.

Zeugen



Definition 6.4 (Zeuge für Nicht-Blockiertheit/Nicht-Kopplung).

Sei L Vereinigungsverband über $[n]$ und $\mathbf{a} \in L$ beliebig.

\mathbf{a} heißt *Zeuge* für die Nicht-Blockiertheit in L von $i \in \mathbf{a}$. Weiterhin heißt \mathbf{a} *Zeuge* für die Nicht-Kopplung in L von $(i, j) \in [n]^2$ mit $i \notin \mathbf{a}, j \in \mathbf{a}$:

$$\begin{aligned} i \in \mathbf{a} & \Rightarrow i \notin \mathcal{B}, \\ i \in [n] \setminus \mathbf{a}, j \in \mathbf{a} & \Rightarrow (i, j) \notin \mathcal{C}. \end{aligned}$$

Unter Verwendung von $\mathbf{a}_{\pm 1}$ aus Algorithmus *TestCouple* ergibt sich Kopplungsvalidierung

ValidateCouples, die unseren Algorithmus sehr stark beschleunigt.

Algorithm *ValidateCouples*(i, j)

(* needs access to both solutions $\mathbf{a}_{\pm 1}$ of *TestCouple* *)

(* $\mathcal{O}(n^2)$ point operations *)

1. $(i, j) \leftarrow \text{first}(\text{couplequeue})$
2. $\text{isacouple} \leftarrow \text{TestCouple}(i, j)$
3. $\text{nocouples} \leftarrow \left\{ (r, s) \in [n]^2 \mid r \notin \mathbf{a}_\sigma, s \in \mathbf{a}_\sigma, \sigma \in \{-1, 1\} \right\}$
4. $\text{couples} \leftarrow \{(i, j)\} \setminus \text{nocouples}$
5. **return** ($\text{couples}, \text{nocouples}$)

Der Einsatz von Zeugen wurde erst acht Jahre nach Vorstellung der FCA von Larhlimi et al. (2012) vorgeschlagen, trotz ihres enormen Nutzens. Sie weisen darauf hin, dass man für einen gültigen Flussvektor $\mathbf{v} \in F$ folgern kann, dass die Paare (i, j) mit $v^{(i)} = 0, v^{(j)} \neq 0$ nicht gekoppelt sein können. So wird wieder eine Sicht genutzt, die hochgradig zentriert ist auf das kleine untersuchte Teilproblem – die Elimination unmöglich gekoppelter Paare.

Verbandstheorie stellt sie hingegen von Anfang an als Mittel der Wahl vor, wie wir gleich sehen werden. Im Verband ist nichts natürlicher, als Elemente zu addieren. Tut man das, erklären sich viele Eigenschaften von alleine. Weil sie einem den direkten Blick auf das Herz der Probleme ermöglicht, sollte sie so oft wie möglich genutzt werden.

Finale Das ging Kreuz und quer und irgendwie haben wir es dennoch geschafft, ein fixes Verfahren zur produzieren. Doch wie implementieren? Was sind die besten Datenstrukturen? Können wir alles vergessen! Gehen wir es von vorne an und lernen einen tollen Algorithmus kennen.



Kapitel 6 – Kurz und knapp

Blockierte Reaktionen

- ▷ Das *Maximum* 1_L umfasst die nicht-blockierten Reaktionen.
- ▷ Flusskopplung sind Bedingungen, die Reaktionen blockieren.

FCA

- ▷ Flusskopplung ist für den Flussverband L^F und *auch andere Verbände* definiert.
- ▷ Vollständige Kopplung ist eine Eigenschaft des *Null-Verbands*. Den Null-Verband definiert man über eine Relaxation des Flusskegels.
- ▷ Viele Einzelveröffentlichungen haben FCA sehr *schnell*, den Algorithmus jedoch unnötig *unübersichtlich* gemacht.



KAPITEL 7

Anwendungen für das Maximum

Kapitelübersicht

7.1	Berechnung des Maximums im Vereinigungsverband	107
7.1.1	Eines für alle!	108
7.2	Erweiterte Flusskopplung	111
7.2.1	Wer 1 sagt...	111
7.2.2	... kann auch 2 sagen!	112
7.3	Target Prediction – Minesweeper im Zellstoffwechsel	115
7.3.1	FBA und Derivate	115
7.3.2	Zwei Probleme, ähnlich definiert, gleich gelöst	116
7.3.3	Von Genen, Enzymen und Reaktionen	124
7.3.4	War da nicht was? – Minimal Cut Sets, 2. Akt	125

§7.1 Berechnung des Maximums im Vereinigungsverband

Wir haben mit Algorithmus *CalculateBlock* ein Verfahren kennengelernt, mit dem sich das Maximum im Kegelverband durch das Lösen eines einzigen MILP bestimmen lässt. Das ist einfach, das ist schnell implementiert, das ist noch schneller verstanden. Für allgemeine Vereinigungsverbände lässt es sich aber nicht nutzen. Noch dazu kommt, dass das Lösen eines MILP \mathcal{NP} -schwer ist (Schrijver 2003), wir aber durch den klassischen Algorithmus wissen, dass das Problem im stationären Flussverband L^F in \mathcal{P} ist, da wir höchstens n LPs lösen müssen.

Verbandstheorie soll als generelles Framework dienen, mit denen sich allerlei Arten an Modellen einheitlich untersuchen lassen. Wir haben hier Flusskopplung verbandstheoretisch definiert als Interpretation der Unterverbände $L_{\perp\{i\}}$. So formuliert lässt sich der von Carbonell et al. (2012) vorgestellte polynomielle Algorithmus direkt nutzen, um Flusskopplung in Hyperpfaden zu untersuchen – ein bis dato nicht publiziertes Vorgehen. Die Verbandstheorie liefert es uns wie auf dem Silbertablett.

Doch sie hilft uns auch, die Algorithmen von Anfang an so zu formulieren, dass sie in möglichst vielen Modellen genutzt werden können, ohne groß angepasst werden zu müssen. So einen universellen Algorithmus und seine Anwendungen wollen wir in diesem Kapitel kennenlernen.

Natürlich wurden schon vor dieser Arbeit Algorithmen vorgestellt, mit denen Eigenschaften im Verband geprüft werden können. Hier sei auf Freese, Ježek & Nation (1995) verwiesen, die verbandstheoretischen Algorithmen ein eigenes Kapitel in ihrer Arbeit widmen.


Freese et al. gehen von anderen Voraussetzungen aus. Sie betrachten endliche Verbände, von denen alle Elemente bekannt sind und stellen Verfahren vor, diverse Datenstrukturen ineinander umzuwandeln oder besondere Elemente zu bestimmen. Wir hingegen verfügen nur über eine Testfunktion, um zu prüfen, ob eine Reaktionenmenge Verbandselement ist. Freese et al. geben Laufzeiten an, die polynomiell in der Anzahl an Verbandselementen $|L|$ sind. Das können wir nicht übernehmen, da unser $[n]$ die Reaktionenmenge widerspiegelt.

§7.1.1 Eines für alle!

In diesem Abschnitt stellen wir ein Verfahren vor, mit dem sich schnell das Maximum für beliebige Vereinigungsverbände berechnen lässt. Sobald wir in unserem Modell einen Pfad durch eine vorgegebene Reaktion finden können, sind wir im Rennen.

Dieses Verfahren wollen wir gleich in einer allgemeineren Form vorstellen, bei dem für ein metabolisches System beliebige $R \subseteq [n]$ blockiert werden dürfen.

Maximum

 **Satz 7.1** (Berechnung des Maximums $1_{L \perp R}$).
 Sei (L, Rev) metabolisches System und $R \subseteq [n]$.
 Sei $\mathcal{W} \subseteq L$ Liste möglicher Zeugen (beliebige, bereits bekannte Verbandselemente).
 Sei schließlich $\text{Test}(i, R) : [n] \rightarrow L$ Abbildung, die erfüllt:

$$i \in 1_{L \perp R} \Rightarrow i \in \text{Test}(i, R).$$

Dann berechnet Algorithmus *SimpleMax* wirklich $1_{L \perp R}$.

Algorithm *SimpleMax*(R, \mathcal{W})

Input: The set of blocked reactions R and a list of known elements $\mathcal{W} \subseteq L$.

(* $\mathcal{O}(|\mathcal{W}| \cdot n)$ point operations + $\mathcal{O}(n)$ Test *)

1. $\mathcal{W}_{\perp R} \leftarrow \mathcal{W} \cap L_{\perp R}$
2. $\text{lower_bound} \leftarrow \sum_{a \in \mathcal{W}_{\perp R}} a$
3. $\text{upper_bound} \leftarrow [n] \setminus R$
4. **for** $i \in [n]$
5. **do if** $i \in \text{upper_bound} \setminus \text{lower_bound}$
6. **then** $a \leftarrow \text{Test}(i, R)$
7. $\text{lower_bound} \leftarrow \text{lower_bound} + a$
8. **if** $i \notin a$
9. **then** $\text{upper_bound} \leftarrow \text{upper_bound} \setminus \{i\}$
10. **return** $1_{L \perp R} \leftarrow \text{lower_bound}$

Die Laufzeit ist wesentlich von drei Dingen bestimmt: Wie viele Zeugen haben wir? Wie gehen wir mit den Zeugen um? Wie viele Test-Aufrufe brauchen wir wirklich lösen? Diese Faktoren sollten gut ausbalanciert werden: Je mehr Zeugen wir kennen, desto mehr Zeit kostet unser Preprocessing, desto weniger kostspielige LP-Solver-Aufrufe müssen wir jedoch tätigen.

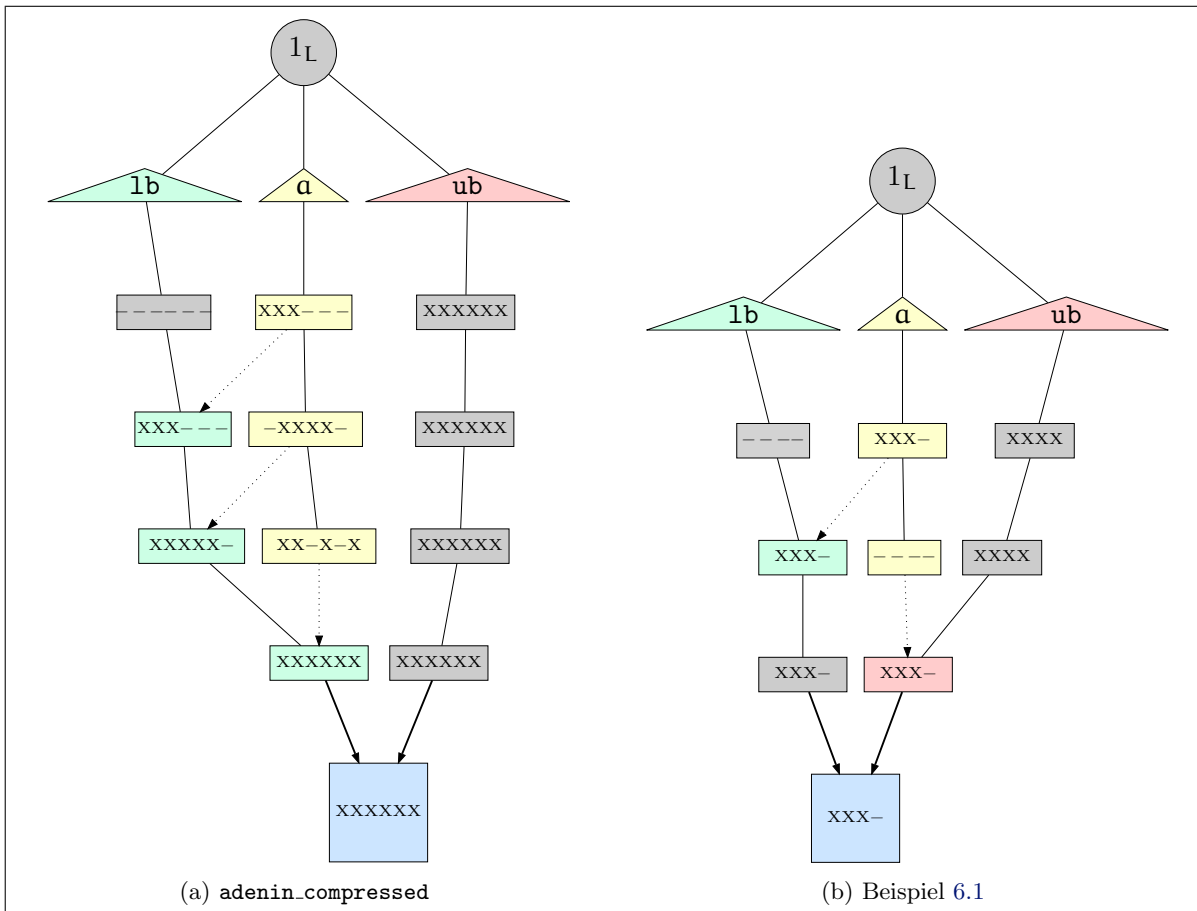


Abbildung 7.1.: Bestimmung des Verbandsmaximums 1_L in wenigen Schritten.



Beispiel 7.1 (Maximumsbestimmung).

In Abbildung 7.1 sind zwei Durchläufe von Algorithmus *SimpleMax* zu sehen.


In 7.1a ist die Bestimmung des Verbandsmaximums 1_L des bekannten Beispiels *adenin_compressed* zu sehen. Das Maximum $1_L = \{1, \dots, 6\}$ wird durch das Lösen von drei LPs bestimmt.

Es handelt sich um einen Worst-Case-Verlauf, bei dem jedes gelöste LP nur eine minimale Anzahl an Reaktionen als unblockiert aufgedeckt hat. Nach klassischer Flux Variability Analysis wären bis zu acht LPs (vier für irreversible, zwei mal je zwei für reversible Reaktionen) zu lösen gewesen.

In 7.1b kommt man mit zwei (statt vier) LPs zu dem Ergebnis, welche Reaktion im Beispiel aus Abbildung 6.1 blockiert ist.

Maximum

Dieser Algorithmus ist bereits sehr schnell, lässt sich aber noch einmal beschleunigen, wenn man einige vorangegangene Untersuchungsergebnisse integriert. Der Kern bleibt erhalten, optimiert ergibt sich Satz 7.2

 **Satz 7.2** (Optimierte Berechnung des Maximums $1_{L \perp R}$).
 Sei (L, Rev) metabolisches System und $R \subseteq [n]$.
 Sei $\mathcal{W} \subseteq L$ Liste möglicher Zeugen (beliebige, bereits bekannte Verbandselemente).
 Seien $1_L, \text{Frev}, \text{NFrev}, \mathcal{C}_i$ wie oben definiert (sofern bekannt) oder jeweils alternativ:

$$\begin{aligned} 1_L &= [n], & \text{Frev} &= \emptyset, \\ \mathcal{C}_i &= \{i\}, & \text{NFrev} &= \emptyset. \end{aligned}$$

Sei schließlich Test $(i, R) : [n] \rightarrow L$ Abbildung, die erfüllt:

$$i \in 1_{L \perp R} \Rightarrow i \in \text{Test}(i, R).$$

Dann berechnet Algorithmus *FastMax* wirklich $1_{L \perp R}$.

Beweis. Es genügt, sich Lemma 3.1 in Erinnerung zu rufen, also:

$$1_{L \perp R} = \sum_{a \in L \perp R} a.$$

Insbesondere ist $\text{Frev} \in L \perp R$ für $R \subseteq \text{NFrev}$. □

Algorithm *FastMax*($R, \mathcal{W}, 1_L, \text{Frev}, \text{NFrev}, \mathcal{C}$)

Input: The set of blocked reactions R and a list of known elements $\mathcal{W} \subseteq L$.

(* optimized version *)

(* $\mathcal{O}(|\mathcal{W}| \cdot n)$ point operations + $\mathcal{O}(n)$ Test *)

1. $\mathcal{W}_{\perp R} \leftarrow \mathcal{W} \cap L_{\perp R}$
2. $\text{lower_bound} \leftarrow \sum_{a \in \mathcal{W}_{\perp R}} a$
3. **if** $R \subseteq \text{NFrev}$
4. **then** $\text{lower_bound} \leftarrow \text{lower_bound} + \text{Frev}$
5. $\text{upper_bound} \leftarrow 1_L$
6. **for** $r \in R$
7. **do** $\text{upper_bound} \leftarrow \text{upper_bound} \setminus \mathcal{C}_r$
8. **for** $i \in [n]$
9. **do if** $i \in \text{upper_bound} \setminus \text{lower_bound}$
10. **then** $a \leftarrow \text{Test}(i, R)$
11. $\text{lower_bound} \leftarrow \text{lower_bound} + a$
12. **if** $i \notin a$
13. **then** $\text{upper_bound} \leftarrow \text{upper_bound} \setminus \mathcal{C}_i$
14. **if** $a \neq 0$
15. **then** $\text{Add}(\mathcal{W}, a)$
16. **return** $1_{L \perp R} \leftarrow \text{lower_bound}$

Einen vollständigen Algorithmus zur FCA erhalten wir analog zu *FewStepsFCA*; wir berechnen zunächst 1_L , indem wir $R = \emptyset$ wählen, und anschließend die \mathcal{C}_i mit $1_{L \perp \{i\}}$ für $i \in 1_L$. Somit ergibt sich für den stationären Flussverband L^F eine theoretische Laufzeiteinschätzung von

$\mathcal{O}(|W| \cdot n^2)$ arithmetische Operationen + $\mathcal{O}(n^2)$ LPs mit $W \in \mathcal{O}(n^2)$. Auch hier wollen wir noch einmal betonen, dass es sich um theoretische Laufzeiten handelt.

Verbandstheorie fasst die meisten Optimierungen aus acht Jahren in 16 Zeilen zusammen. Wir verzichten ganz bewusst auf Kompression (Larhlmi et al. 2012, Goldstein 2009), um uns nicht so auf das Standardmodell einzuschränken. Aufwendige Implikationstabellen wie bei Larhlmi et al. (2012) sind auch nicht nötig – wir nutzen alle ihre Ergebnisse schon implizit, was der interessierte Leser leicht nachvollziehen kann.

16 Zeilen. That's it. Wir brauchen nur noch die Test-Funktion, die wo möglich einen Pfad durch i berechnet.

§7.2 Erweiterte Flusskopplung

§7.2.1 Wer 1 sagt...

Das klassische Verfahren zur FCA hat bis zu $2n^2$ LPs gelöst und ist damit bei genomweiten Netzwerken an den Rand des mit heutiger Computertechnik Möglichen geraten. Die Verbesserungen, die in den Folgejahren vorgestellt wurden, beschleunigten zwar, aber die tatsächliche Laufzeit lag meist immer noch nah an ihrer theoretischen oberen Schranke von $\mathcal{O}(n^2)$ LPs.

Auch unser Verfahren hat eine theoretische Laufzeit von $\mathcal{O}(n^2)$ LPs, in der Realität sind es aber viel, viel weniger. Uns wird eher ein anderer Schritt zum Problem: Die Bestimmung von $W_{\perp R}$. Doch dazu später mehr.

Was machen wir bei der FCA? Wir blockieren eine Reaktion und prüfen, was noch geht. Das war auch vorher schon bekannt, aber nie so klar formuliert wie in der Verbandssprache, wo wir keine einzelnen paarweisen Kopplungen mehr bestimmen, sondern gleich die gesamte Menge unblockierter Reaktionen $1_{L_{\perp\{i\}}}$.

Wer 1 sagt, sollte auch 2 sagen. Zwei Reaktionen blockieren und schauen, was passiert.

Nach altem, naiven Vorgehen müssten wir dafür bis zu n^3 LPs lösen. Bei genomweiten Netzwerken ist $n > 1000$ anzusetzen, wir landen also bei rund 10^9 LPs, die gelöst werden wollen. Selbst wenn wir das LP in einem einzigen Schritt lösen könnten, so würde die Lösung aus n Einträgen bestehen – wir wissen also schon, dass das klassische Vorgehen eine untere Laufzeitschranke von $\Omega(n^4)$ Schreiboperationen hat; das sind gut 10^{12} – eine Billion! – im Falle eines kleineren genomweiten Netzwerkes.

Weg von den Zahlenspielereien, zurück zu den erlebten Laufzeiten. Hat früher eine einfache FCA eine Woche gedauert, so müssten wir mit dem n -fachen der Laufzeit für die auf zwei Reaktionen erweiterte FCA rechnen. Bei 1000 Reaktionen sind das 1000 Wochen oder rund 20 Jahre.

Beim alten Vorgehen wäre einem nichts anderes übrig geblieben, als sich auf eine wesentlich kleinere Teilmenge $R \subseteq [n]$ von Reaktionen zu konzentrieren, und sich nur bei denen die Folgen eines doppelten Reaktionen-Knockouts anzusehen.


Wir können es jetzt aber richtig machen.

§7.2.2 ... kann auch 2 sagen!

Für eine Reaktion sind zwei Zustände möglich: Sie ist blockiert oder sie ist möglich. Zur Berechnung des Maximums müssen wir den Zustand einer jeden Reaktion bestimmen. Das können wir mit unserer Testfunktion direkt, oder durch einige Tricks indirekt.

Prinzipiell müssen wir uns nicht auf zwei Knockouts einschränken. Unser Algorithmus berechnet das Maximum $1_{L \perp R}$ für jede beliebige Teilmenge $R \subseteq [n]$ an Reaktionen. Deshalb können wir eine Erweiterung des Flusskopplungsbegriffs, der von Burgard et al. (2004) geprägt wurde, formal definieren:

$R \xrightarrow{=0} j$



Definition 7.1 (erweiterte Flusskopplung).
 Sei (L, Rev) metabolisches System über $[n]$ und $R \subseteq [n]$, $j \in [n] \setminus R$. Definiere nun wie folgt:

$$R \xrightarrow{=0} j \quad :\Leftrightarrow \quad j \notin 1_{L \perp R} \wedge \forall \tilde{R} \subsetneq R : j \in 1_{L \perp \tilde{R}}.$$

Dann heißt j erweitert gekoppelt an R .

Dadurch, dass wir vom Maximum ausgehen, unterscheiden wir uns auch von Behre, Wilhelm, Kamp, Ruppin & Schuster (2008), die die Robustheit eines Netzwerks untersuchen. Behre et al. messen dies durch die Anzahl an elementaren Flussmoden, die durch einen Knockout von R nicht beeinflusst werden. Offensichtlicher Vorteil unseres Vorgehens ist es, dass wir nicht die riesige Menge an Elementarmoden berechnen müssen.

Anzahl an Testaufrufen

Um eine FCA zu machen, speichern wir Zwischenergebnisse und werden dadurch sehr schnell. Wir können viele Reaktionen direkt als „möglich“ bestimmen. Viele unserer Tests gehen nur noch darum, zu bestimmen, ob eine Reaktion wirklich blockiert ist.

Noch schneller wird es, sobald wir die FCA beendet haben. Wir wollen wieder testen, ob Reaktion i blockiert ist. Wir berechnen $\text{Test}(i, R)$ und addieren den Wert zu unserer unteren Grenze. Ist i enthalten, wissen wir, dass i unblockiert ist und können diesen Zustand auch gleich für alle anderen Elemente in $\text{Test}(i, R)$ festhalten. Sollte aber i nicht enthalten sein, so wissen wir, dass i blockiert ist und damit auch alle an i gekoppelten Reaktionen aus \mathcal{C}_i .

Im Falle eines doppelten Reaktionen-Knockouts $R = \{i, j\}$ können wir vorher schon alle Re-

aktionen bestimmen, die durch den Wegfall von i oder j alleine auch blockiert sind. Alle bisherigen Zeugen, die weder i noch j enthalten, können wir weaternutzen und eine große untere Schranke für $1_{L_{\perp\{i,j\}}}$ ausmachen. Unsere Tests drehen sich jetzt nur noch um die Reaktionen k , die auf i oder j alleine verzichten können, den Wegfall beider gleichzeitig aber nicht verkraften.

Aus Maximumssicht kann das zu interessanten Beobachtungen führen: Zwei Reaktionen, die alleine nicht allzu viele Reaktionen an sich gekoppelt haben, haben gemeinsam einen sehr großen Einfluss auf sich gebündelt. (s. Bild 7.2).

Succ_in	0	0	6	8	4	4	4	8	8	9	9		
Succ-DH	0	0	6	8	4	4	4	8	8	9	9		
Q-oxidase	8	8	6	8	8	8	8	9	10	8	8		
Photo	8	8	8	8	8	8	6	10	9	8	8		
O2_in	6	6	6	8	4	4	8	6	8	4	4		
NADH_in	0	0	6	8	9	9	4	8	8	4	4		
NADH-DH	0	0	6	8	9	9	4	8	8	4	4		
c2-oxidase	8	8	8	10	8	8	8	8	8	8	8		
bc1	6	6	8	8	6	6	6	8	6	6	6		
ATPSynthase	9	9	6	8	0	0	6	8	8	0	0		
ATPdrain	9	9	6	8	0	0	6	8	8	0	0		
			ATPdrain	ATPSynthase	bc1	c2-oxidase	NADH-DH	NADH_in	O2_in	Photo	Q-oxidase	Succ-DH	Succ_in

Abbildung 7.2.: Mögliche Reaktionen nach Doppel-Knockout in *etc_purple*.

Zeugenzahl

Ein Pool an Zeugen und FCA-Resultate machen uns die Maximumbestimmung sehr einfach. Doch leider steckt noch ein Teufel im Detail: Was mit den Zeugen tun?

Jeder Testaufruf liefert uns einen neuen Zeugen – zumal jeder positive Testaufruf uns garantiert einen neuen Zeugen liefert, den wir derzeit nicht zur Verfügung haben.¹ Wie verwaltet man die Zeugen? Und brauchen wir wirklich alle?

Die erste Frage ist einfach: Weil wir im Positivfall nur neue Zeugen produzieren, können wir mit einer einfachen Liste aller Zeugen arbeiten. Wir brauchen keine Verwaltungsstruktur, die auf doppelte Einträge prüft, können also auf einen Binärbaum oder Ähnliches verzichten.

Die schiere Masse an Zeugen wird jedoch zum Problem, wenn wir genomweite Netzwerke betrachten. Jeder Zeuge hat einen Speicherbedarf von $\Omega(n)$, da für jede Reaktion bekannt sein muss, ob sie vorkommt oder nicht.² Für großes n kommen wir bei den immer noch vielen Testergebnissen an Speicherprobleme.

Aber nicht nur aus Speichergründen muss man sich der Frage stellen, wie viele Zeugen man wirklich braucht. Auch aus Laufzeitgründen ist die Frage von gewaltiger Bedeutung: Wir starten jede Berechnung eines Maximums $1_{L \perp R}$ mit der Bestimmung von $W_{\perp R} \subseteq W$. Dabei muss für alle $w \in W$ getestet werden, ob $R \cap w$ leer ist oder nicht.

Eine endgültige und ultimative Antwort auf das Problem ist noch nicht gefunden. Wir haben versuchsweise mit Heuristiken gearbeitet und entsprechend $\text{Add}(W, a)$ (Zeile 15 in *FastMax*) umgesetzt. Doch dazu später mehr im Praxisteil.

Ein Zeuge a hilft uns, wenn zum Zeitpunkt seines Einsatzes $\text{lower_bound} + a > \text{lower_bound}$ gilt. Je größer lower_bound bereits ist (was mit der Anzahl an Zeugen zusammenhängt), desto unwahrscheinlich ist es, dass uns ein weiterer Zeuge helfen kann. Das nutzen wir als Heuristik.

Die Idee ist, dass wir immer weniger Zeugen akzeptieren, wenn die Anzahl steigt. Somit sammeln wir anfangs Zeugen, um unser Verfahren zu beschleunigen. Später akzeptieren wir Zeugen jedoch immer häufiger nur dann, wenn wir sie für nützlich halten.

Der konkrete Nutzen lässt sich nicht vorher festmachen. Man ist nachher klüger. Die Anzahl an Reaktionen in a dürfte jedoch Einfluss haben. Ist $|a|$ zu groß, wird a zu oft beim Test $a \cap R \neq \emptyset$ aussortiert. Ist $|a|$ klein, so können wir a häufiger als Zeugen einsetzen, allerdings wird auch öfter $\text{lower_bound} + a = \text{lower_bound}$ gelten – a hat uns also nichts genutzt.

In unserer Implementierung berechnen wir zunächst die Flusskopplung, ohne Zeugen zu verwenden. Im Worst-Case-Szenario können dabei zwar n^2 Zeugen entstehen, was im realistischen

¹Wir rufen $\text{Test}(i, R)$ auf, wenn $i \notin \text{lower_bound}$. Da lower_bound Summe aller Zeugen in $W_{\perp R}$ ist, ist $a = \text{Test}(i, R)$ neuer Zeuge, wenn $i \in a$.

²Natürlich ist auch eine einfache Auflistung der enthaltenen Reaktionen denkbar, doch macht das die Berechnung der Summe $a_1 + a_2$ langsamer.

Fall aber nie nur ansatzweise vorkam. Erst für die erweiterte Untersuchung mit zwei ausgeschalteten Reaktionen greifen wir auf eine Heuristik zurück. Wir haben gute Erfahrung damit gemacht, eher große Zeugen zu speichern, auch wenn wir diese seltener nutzen können. Insgesamt entscheiden wir für \mathbf{a} mit einer Wahrscheinlichkeit $p_{\mathbf{a}}$, ob wir ihn in \mathcal{W} aufnehmen wollen oder nicht. Hierbei gilt:

$$p_{\mathbf{a}} = \min\left(1, \frac{|\mathbf{a}|}{|\mathcal{W}|}\right).$$

§7.3 Target Prediction – Minesweeper im Zellstoffwechsel

Minesweeper ist ein unter dem Computerbetriebssystem Microsoft Windows beliebtes Spiel für zwischendurch. Jeder, der schon mal Windows benutzt hat, hat es früher oder später schon mal gespielt. Kaye (2000) hat es sogar mathematisch untersucht.

Wofür steht das hier? Ein Vergleich in Kurzfassung:

Minesweeper-Regeln: Treffe sichere Felder. Jeder Treffer liefert dir Informationen über Nachbarfelder. Treffe so schnell wie möglich so viele sichere Felder wie möglich, ohne ein Minenfeld zu treffen. Denn dann ist das Spiel vorbei.

Target Prediction in Metabolomics: Schalte Reaktionen sinnvoll aus. Jede Blockade beeinflusst das Restnetzwerk. Schalte so intelligent wie möglich Reaktionen aus, so dass der verbleibende Stoffwechsel deine Wünsche erfüllt, ohne ganz zusammenzubrechen. Denn dann ist die Zelle tot.

Fragt sich nur: Wie schalten wir Reaktionen *intelligent* aus?

§7.3.1 FBA und Derivate

Target Prediction als Anwendung von Stoffwechselanalyse wird schon lange und ausführlich in der Literatur diskutiert. Dabei ist eine Masse an unterschiedlichen Verfahren entstanden, die sich durch raffinierte Optimierungen von konkurrierenden Vorgehensweisen unterscheiden. Es würde den Rahmen dieser Arbeit sprengen, alle Verfahren vorstellen zu wollen. Gemeinsam ist ihnen fast allen, dass sie für die ausgeschalteten Reaktionen \mathbf{R} eine Flux Balance Analysis machen und prüfen, wie hoch der Flux durch die Wunschreaktionen \mathbf{T} ist. Sie suchen nach \mathbf{R} , das diesen Flux maximiert. Dabei werden verschiedene FBA-Varianten eingesetzt (etwa mit oder ohne Integration regulatorischer Daten) und meist auch gute Erfolge erzielt.

In dieser Arbeit geht es nicht um Flux-Werte. Wir betrachten, was man mit Verbandstheorie schaffen kann. Wir haben die Verfahren nicht auf Praxistauglichkeit getestet, sondern erfreuen uns lieber ihrer Einfachheit und Eleganz, die uns die Verbandstheorie ermöglicht.

Dennoch sollen die Flux-nahen Verfahren nicht unerwähnt bleiben. Stattdessen wird all den Wissenschaftlern hier Respekt gezollt, die an ihnen gearbeitet haben. Die Verfahren sind

sehr gut überlegt ausgearbeitet und mit Liebe zum Detail entwickelt. Sie zeigen vorbildlich, was die Kombination unterschiedlichster Ergebnisse und Verfahren in der interdisziplinären Systembiologie hervorbringen kann. Beispielhaft dafür seien hier (Boghigian, Shi, Lee & Pfeifer 2010) und (Kim & Reed 2010) genannt.

Boghigian et al. (2010) verwenden Elementarmoden, genetische Algorithmen und freie Gibbs-Energie, um die Produktion von Ethanol oder Lycopene in *E. coli* zu maximieren. Kim & Reed (2010) steigert hingegen die Ethanol-Produktion, indem man Zielgene mittels mehrstufiger MILPs bestimmt und hierbei auch noch regulatorische Effekte beachtet.

Einen wieder anderen Ansatz verfolgen Melzer, Esfandabadi, Franco-Lara & Wittmann (2009). Bei ihrem Vorgehen, *Flux Design* genannt, werden nicht nur Zielreaktionen identifiziert, die gehemmt werden sollen, sondern auch welche, die sie hoch regulieren wollen. Dafür untersuchen sie die elementaren Flussmoden, die sie erst einmal berechnen müssen.

Diese drei (und andere Verfahren) enthalten eine FBA als Kern: Sie versuchen konkret einen stationären Zustand zu berechnen, den sie erzwingen können, und der zudem möglichst viel Zielprodukt herstellt. Wenn wir selbst gleich von Zielprodukten reden werden, bestimmen wir die produzierte Kapazität nicht. Treu dem Vereinigungsverband werden wir nur die Aussage treffen: folgende (produzierende) Reaktionen kommen definitiv vor, folgende definitiv nicht.

§7.3.2 Zwei Probleme, ähnlich definiert, gleich gelöst

Stellen wir uns folgendes Szenario vor:


Wir kennen den Stoffwechsel einer Zelle, soll heißen die Metaboliten und die möglichen Reaktionen. Außerdem können wir in das System eingreifen: Für eine Teilmenge $D \subseteq [n]$ der Reaktionen können wir dafür sorgen, dass sie nicht mehr stattfinden. Dabei soll es an dieser Stelle erst einmal keine Rolle spielen, ob wir das durch einen medikamentösen Eingriff, eine genetische Manipulation oder ein anderes Vorgehen schaffen.

Zudem gibt es eine zweite Teilmenge der Reaktionen, $T \subseteq [n]$. Das sind unsere Targets, unsere Zielreaktionen besonderer Bedeutung.

Wir wollen diese „besondere Bedeutung“ nun durch Systemmanipulation betonen. Wir betrachten zwei Fälle:

T soll verhindert werden. Die Reaktionen in T schaden der Zelle und sollen deshalb blockiert werden. Beispielsweise handelt es sich um Reaktionen, die verstärkt in Krebszellen, jedoch kaum in der gesunden Form vorkommen.

Welche Reaktionen $R \subseteq D$ sollten ausgeschaltet werden, damit alle Reaktionen aus T blockiert sind? Wie sollte R gewählt werden, damit zwar T verhindert wird, aber wir das System ansonsten so wenig wie möglich einschränken?



Problem 7.1 (T verhindern).
Finde $R \subseteq D$, das erfüllt:

$$T \cap 1_{L \perp R} = \emptyset$$


$$A \subseteq D, T \cap 1_{L \perp A} = \emptyset \Rightarrow |1_{L \perp A}| \leq |1_{L \perp R}|.$$

Blockierungsproblem

T soll gefördert werden. Wir sind an dem Vorkommen der Reaktionen T interessiert. Das kann daran liegen, dass sie zum Überleben der Zelle gerade noch ausreichen und zusätzlich einen Stoff wie Bioethanol herstellen, an dem wir interessiert sind.

Um die Produktion des Bioethanols zu maximieren, wollen Reaktionen so ausschalten, dass eine minimale überlebensfähige Zelle entsteht, die dennoch Bioethanol produziert.

Welche Reaktionen $R \subseteq D$ können ausgeschaltet werden, damit alle Reaktionen aus T trotzdem noch möglich sind? Wie sollte R gewählt werden, damit T zwar möglich, aber maximal viele andere Reaktionen blockiert sind?



Problem 7.2 (T fördern).
Finde $R \subseteq D$, das erfüllt:

$$T \cap 1_{L \perp R} = T$$

$$A \subseteq D, T \cap 1_{L \perp A} = T \Rightarrow |1_{L \perp A}| \geq |1_{L \perp R}|.$$

Förderungsproblem

Beide Fälle lassen sich mathematisch ähnlich beschreiben und auch ähnlich lösen. Außerdem wäre es in beiden Fällen schön, wenn unsere Lösungsmenge $R \subseteq D$ zusätzlich so klein wie möglich ist. Das senkt im Allgemeinen die Kosten und meist auch das Risiko auf Nebenwirkungen.

Heuristisch?

Die Probleme sind bereits formuliert. Wie kommt man jetzt an die optimalen Lösungen oder zumindest an gute? Unser Lösungsraum besteht aus $2^{|D|}$ Werten. Zu viele, um wirklich jede einzelne Möglichkeit auszuprobieren. Wir diskutieren daher, wie man heuristisch gute Lösungen erhalten könnte, und betrachten danach ein Branch-and-Bound-Verfahren für die genaue Lösung.

Die „Gutheit“ der Lösung wird dabei durch eine Kostenfunktion beschrieben, die es zu minimieren gilt. Jedes Problem, das so eine Kostenfunktion hat, kann mit einer Heuristik angegangen werden. So sind genetische Algorithmen vorstellbar, aber auch andere heuristische Verfahren wie Simulated Annealing könnten eingesetzt werden.

Um eine Heuristik nutzen zu können, brauchen wir eine Kostenfunktion, deren Minima unseren optimalen Lösungen entsprechen. Diese seien hier geliefert:

**Satz 7.3.**

Sei (L, Rev) metabolisches System, $D, T \subseteq [n]$. Sei weiterhin $n + 2 \leq W \in \mathbb{R}$ beliebig und fest.

Seien $c_+, c_- : 2^D \rightarrow \mathbb{R}$ Abbildungen mit:

$$\begin{aligned} c_+ : \quad R &\mapsto |R| + W \cdot |1_{L \perp R}| - W^2 \cdot |1_{L \perp R} \cap T|, \\ c_- : \quad R &\mapsto |R| - W \cdot |1_{L \perp R}| + W^2 \cdot |1_{L \perp R} \cap T|. \end{aligned}$$

Seien nun $R_+, R_- \subseteq D$ mit R_+ globales Minimum von c_+ bzw. R_- Minimum von c_- . Dann ist R_- optimale Lösung des Blockierungsproblems 7.1 und R_+ optimale Lösung vom Förderungsproblem 7.2.

Beweis. Wir beweisen die Aussage, dass R_+ globales Minimum von c_+ optimale Lösung ist. Der Beweis für R_- und c_- erfolgt analog.

Angenommen $A \subseteq D$ mit $|T \cap 1_{L \perp A}| > |T \cap 1_{L \perp R_+}| \neq 0$. Dann:

$$\begin{aligned} c_+(R_+) &= |R_+| + W \cdot |1_{L \perp R_+}| - W^2 \cdot |1_{L \perp R_+} \cap T| \\ &\geq -W^2 \cdot |1_{L \perp R_+} \cap T| \\ &> W \cdot (n + 1) - W^2 \cdot |1_{L \perp A} \cap T| \\ &> n + W \cdot n - W^2 \cdot |1_{L \perp A} \cap T| \\ &\geq |A| + W \cdot |1_{L \perp A}| - W^2 \cdot |1_{L \perp A} \cap T| \\ &= c_+(A) \geq c_+(R_+). \end{aligned}$$

Also kann es kein $A \subseteq D$ geben, das weniger Reaktionen aus T blockiert als R_+ .

Somit wissen wir schon mal, dass unsere Lösung $T \cap 1_{L \perp R_+} = T$ erfüllt, wenn das überhaupt möglich ist für $R \subseteq D$.³

Sei jetzt $A \subseteq D$ beliebig mit $T \subseteq 1_{L \perp A}$. Da R_+ globales Minimum ist, gilt:

$$\begin{aligned} 0 &\leq c_+(A) - c_+(R_+) \\ &= (|A| + W \cdot |1_{L \perp A}|) - (|R_+| + W \cdot |1_{L \perp R_+}|) \\ \Rightarrow W \cdot |1_{L \perp R_+}| &\leq |A| + W \cdot |1_{L \perp A}| \\ &\leq n + W \cdot |1_{L \perp A}| \\ \Rightarrow |1_{L \perp R_+}| &\leq \frac{n}{W} + |1_{L \perp A}|. \end{aligned}$$

Wegen $0 \leq \frac{n}{W} < 1$ und Ganzzahligkeit der Werte gilt $|1_{L \perp A}| \geq |1_{L \perp R_+}|$.

Dass zudem $|R_+|$ minimal ist, ist offensichtlich und somit ist R_+ wirklich optimale Lösung unseres Problems 7.2. \square


³Also genau dann, wenn $T \subseteq 1_L$ oder auch $T \cap \mathcal{B}(L) = \emptyset$.

Oj! Jetzt noch mal richtig...

Wir haben für unsere beiden Szenarien nun Kostenfunktionen, mit denen wir heuristische Verfahren nutzen können. Damit können wir die Laufzeit unseres Verfahrens gut einschränken, haben aber keinerlei Garantie für die Güte unserer Lösungen.

Umgekehrt schaut es aus, wenn wir uns für ein Branch-&-Bound-Verfahren entscheiden, bei dem wir frühzeitig Lösungen abschneiden. Hierbei wissen wir, dass wir garantiert die richtige Lösung finden, müssen aber mit sehr langen Laufzeiten rechnen. Unsere beiden Probleme – mehr noch das Blockierungs- als das Förderungsproblem – bieten jedoch gute Möglichkeiten, frühzeitig nicht-optimale Lösungen abzuschneiden, was wir nun vorstellen.

Folgendes Lemma hilft uns, ein Branch-&-Bound-Verfahren zu entwickeln:




Lemma 7.1.

- * Sei (L, Rev) metabolisches System, $D, T \subseteq [n]$ und $c_- : D \rightarrow \mathbb{R}$ wie in Satz 7.3.
- * Sei $1_{L \perp D} \cap T = \emptyset$. Dann gibt es für alle $R_0 \subseteq D$ mit $1_{L \perp R_0} \cap T \neq \emptyset$ ein $R_0 \subsetneq A_0 \subseteq D$
- * mit $1_{L \perp A_0} \cap T = \emptyset$.
- * Sei weiterhin $R_1 \subseteq D$ mit $1_{L \perp R_1} \cap T = \emptyset$.
- * Dann ist $c_-(R_1) < c_-(A)$ für alle $R_1 \subsetneq A \subseteq D$.

Beweis. Die erste Aussage ist klar: Wähle $A_0 = D$.

Für die zweite Aussage nutzen wir $L \perp A = (L \perp R_1) \perp A$. Damit ist $L \perp A \subseteq L \perp R_1$ Unterhalbverband und deshalb $1_{L \perp R_1} \geq 1_{L \perp A}$ gilt. Der Rest folgt aus der Definition von c_- . □

Damit erschließt sich die Arbeitsweise von Algorithmus *FindBlockingSet*. Um diese nicht ganz triviale Rekursion besser verständlich zu machen, fordern wir o. B. d. A. $D = [d] = \{1, \dots, d\}$.



Satz 7.4 (Rekursive Lösung des Blockierungsproblems).

Sei (L, Rev) metabolisches System und $D = [d], T \subseteq [n]$.

Sei ferner $T \cap 1_{L \perp D} = \emptyset$ und R^* Lösung der Rekursion *FindBlockingSet*, die mit $R = \emptyset, i = 0, \text{upper_bound} = [n]$ gestartet wurde. Dann:

$$T \cap 1_{L \perp R^*} = \emptyset$$

$$A \subseteq D, T \cap 1_{L \perp A} = \emptyset \Rightarrow |1_{L \perp A}| \leq |1_{L \perp R^*}|.$$

Zudem ist $|R^*|$ minimal unter allen $R \subseteq D$, die diese beiden Eigenschaften haben.

Blockierungsproblem

Algorithm *FindBlockingSet*($R, i, \text{upper_bound}$)

Input: Point of branching $R \subseteq \{d_1, \dots, d_{i-1}\}$ with index i of first un-tested reaction.

(* recursive algorithm *)

(* requires global parameters $D = [d], T, \text{Frev}, \text{NFrev}, \mathcal{C}(L)$ *)
(* requires global variables R^*, w^* for the best known solutions and witnesses \mathcal{W} *)
(* $\mathcal{O}(2^{|D|})$ Maximum calculations total running time *)

1. $1_{L \perp R} \leftarrow \text{FastMax}(R, \mathcal{W}, \text{upper_bound}, \text{Frev}, \text{NFrev}, \mathcal{C}(L))$
2. **if** $1_{L \perp R} \cap T = \emptyset$
3. **then if** $(|1_{L \perp R}| > w^*)$ **or** $(|1_{L \perp R}| = w^* \text{ and } |R| < |R^*|)$
4. **then** $R^* \leftarrow R$
5. $w^* \leftarrow |1_{L \perp R}|$
6. **else if** $|1_{L \perp R}| > w^*$
7. **then for** $j \in 1_{L \perp R} \cap \{i+1, \dots, k\}$
8. **do** $\text{ub} \leftarrow 1_{L \perp R} \setminus \mathcal{C}_j$
9. **if** $|\text{ub}| > w^*$
10. **then** $\text{lb} \leftarrow \text{FastMax}(R \cup \{j, \dots, k\}, \mathcal{W}, \text{ub}, \text{Frev}, \text{NFrev}, \mathcal{C}(L))$
11. **if** $\text{lb} \cap T \neq \emptyset$
12. **then break**
13. **else** $\text{FindBlockingSet}(R \cup \{j\}, j, \text{ub})$

break



Bemerkung 7.1 (break im Algorithmus).

Im Algorithmus FindBlockingSet wird vor dem Rekursionsschritt die Maximumsfunktion ein zusätzliches Mal aufgerufen und damit die Anzahl an Aufrufen auf den ersten Blick verdoppelt. Dies ist jedoch eine sehr lohnende Investition:

*Besonders im späteren Verlauf des Algorithmus kann es dazu kommen, dass wir so viele Reaktionen nicht zu R hinzugefügt haben, dass wir das Ziel, T zu blockieren, gar nicht mehr erreichen können. Hierfür dient das **break**; im günstigsten Falle werden dadurch auf einen Schlag viele Äste des Rekursionsbaumes abgeschnitten.*

Dazu später mehr in einem Beispiel.

Unsere beiden Probleme 7.1 und 7.2 lassen sich durch zwei fast identische Kostenfunktionen heuristisch lösen. Es gibt aber auch deterministische Lösungsverfahren, die fast gleich sind. Im Falle des zweiten Problems, der Förderung bestimmter Reaktionen, lässt sich jedoch eine Lösung R^* minimaler Kardinalität nur durch ein Zwei-Schritt-Verfahren bestimmen.

Können wir auf diese Zusatzeigenschaft verzichten, so hilft uns Algorithmus $\text{FindHighlightingSets}$, alle sehr guten Lösungen zu finden.

Förderungs-
problem



Satz 7.5 (Rekursive Lösung des Förderungsproblems).

Sei (L, Rev) metabolisches System und $D = [d], T \subseteq [n]$.

Sei ferner $T \subseteq 1_L$ und \mathcal{R} Lösungsmenge der Rekursion $\text{FindHighlightingSets}$, die mit $R = D, i = 0, \text{upper_bound} = 1_L$ gestartet wurde. Dann gilt für $R^ \in \mathcal{R}$ beliebig:*

$$T \cap 1_{L \perp R^*} = T$$

$$A \subseteq D, T \cap 1_{L \perp A} = T \Rightarrow |1_{L \perp A}| \geq |1_{L \perp R^*}|.$$

Im Allgemeinen ist $|R^|$ aber nicht minimal unter allen $R \subseteq D$, die diese beiden Eigenschaften haben.*

Algorithm *FindHighlightingSets*($R, i, \text{upper_bound}$)

Input: Point of branching $R \subseteq D = [d]$ with index i of first un-tested reaction.

(* recursive algorithm *)

(* requires global parameters $D = [d], T, 1_L, \text{Frev}, \text{NFrev}, \mathcal{C}(L)$ *)

(* requires global variables R^*, w^* for the best known solutions and witnesses \mathcal{W} *)

(* $\mathcal{O}(2^{|D|})$ Maximum calculations total running time *)

1. $1_{L \perp R} \leftarrow \text{FastMax}(R, \mathcal{W}, \text{upper_bound}, \text{Frev}, \text{NFrev}, \mathcal{C}(L))$
2. **if** $1_{L \perp R} \cap T = T$
3. **then if** $|1_{L \perp R}| < w^*$
4. **then** $\mathcal{R} \leftarrow \{R\}$
5. $w^* \leftarrow |1_{L \perp R}|$
6. **else if** $|1_{L \perp R}| = w^*$
7. **then** $\mathcal{R} \leftarrow \mathcal{R} \cup \{R\}$
8. **else if** $|1_{L \perp R}| < w^*$
9. **then for** $j \in \{i + 1, \dots, k\} \setminus \bigcup_{l \in R \cap [i]} \mathcal{C}_l$
10. **do** $\text{ub} \leftarrow \text{FastMax}(R \cap [j - 1], \mathcal{W}, 1_L, \text{Frev}, \text{NFrev}, \mathcal{C}(L))$
11. **if** $\text{ub} \cap T \neq T$
12. **then break**
13. **else** *FindHighlightingSets*($R \setminus \{j\}, j, \text{ub}$)



Beispiel 7.2 (Förderungsalgorithmus).

In diesem Beispiel betrachten wir das komprimierte Netzwerk von *etc_purple*, wie es in Abbildung 6.2 zu sehen ist.

Unser Ziel ist es, die Photosynthese-Reaktion R_7 zu fördern. Hierfür können wir jede Reaktion außer R_6 und R_7 blockieren. Es sei ausdrücklich betont, dass es sich hierbei nur um ein Beispiel handelt, das die Wirkungsweise des Verfahrens vorführen soll – eine biologische Interpretation des Ergebnisses wird nicht vorgenommen.

In Abbildung 7.3 werden die entscheidenden Schritte des Algorithmus sichtbar gemacht.

Wir kommen zu dem Ergebnis, dass ein minimaler Stoffwechsel(-pfad), der die Reaktion R_7 enthält, die Reaktionen R_3, R_6 und R_7 umfasst.

Wir wollen nun insbesondere auf die Bedeutung des **break**-Befehls hinweisen.

In 7.3a ist ein Ablauf des Algorithmus zu sehen, bei dem auf **break** verzichtet wurde. Es mussten insgesamt 27 Maxima von theoretisch nicht mehr als $2^6 = 64$ bestimmt werden.

In 7.3b sind die entscheidenden oberen Schranken hervorgehoben: Wir sehen sofort, dass viele weitere Pfade nicht untersucht werden müssen.

Setzen wir **break** wie in *FindHighlightingSets* vorgesehen um, müssen wir nur noch insgesamt 14 Maxima bestimmen – in Abbildung 7.3c ist der Kurzverlauf des Algorithmus inklusive der **break**-Aufrufe nachzuvollziehen.

Förderung

Ist man nun an dem Optimum von Problem 7.2 interessiert – will also auch noch eine von der Kardinalität her minimale Lösung – so kann man beide Verfahren hintereinander schalten.

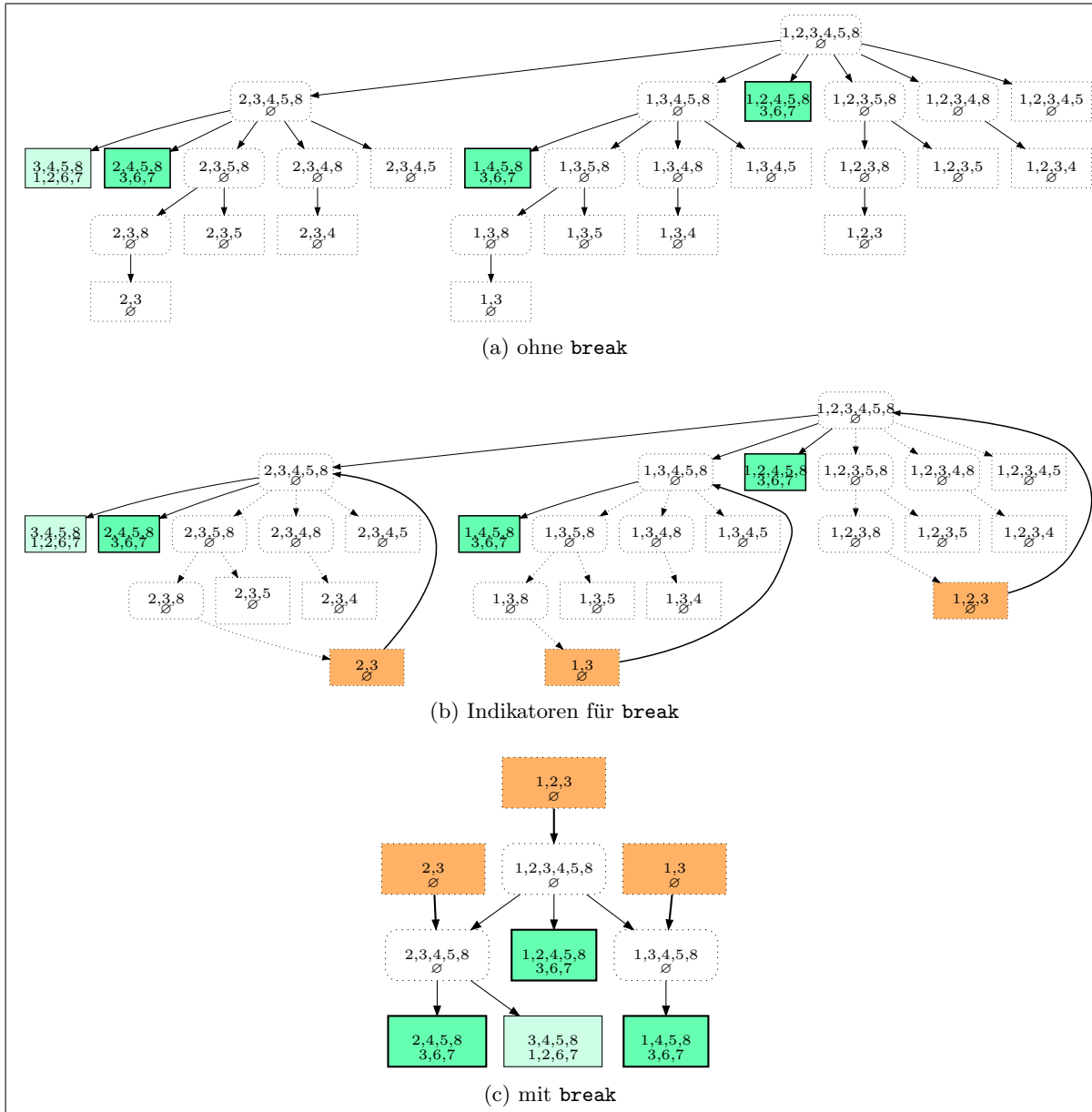


Abbildung 7.3.: Förderung von R_7 im komprimierten Netzwerk von `etc_purple`, $L = L^F$.
 obere Zeile: R – untere Zeile: $1_{L \perp R}$.

 **Bemerkung 7.2.**

Sei (L, Rev) metabolisches System und $D, T \subseteq [n]$.
Ist R_{opt} Lösung von Algorithmus *FindMinimalHighlightingSet*, so ist R_{opt} optimale Lösung kleinster Kardinalität von Problem 7.2.

Algorithm *FindMinimalHighlightingSet*(D, T)

(* requires global parameters $1_L, Frev, NFrev, \mathcal{C}$ *)

(* requires global variables R^*, w^* for the best known solutions and witnesses \mathcal{W} *)

(* $\mathcal{O}(2^{|D|})$ Maximum calculations total running time *)

1. $R_{opt} \leftarrow D$
2. $\mathcal{R} \leftarrow \text{FindHighlightingSets}(D, 0, \emptyset)$
3. **for** $R \in \mathcal{R}$
4. $D \leftarrow T \leftarrow R$
5. $w^* \leftarrow -1$
6. $\tilde{R} \leftarrow \text{FindBlockingSet}(\emptyset, 0, [n])^4$
7. **if** $|\tilde{R}| \leq |R_{opt}|$
8. **then** $R_{opt} \leftarrow \tilde{R}$
9. **return** R_{opt}

Warum die Probleme so trickreich sind

Ignorieren wir den Wunsch nach Minimalität von $|R|$, so erhalten wir folgende Beschreibung:

 **Bemerkung 7.3.**

Sei (L, Rev) metabolisches System, $D, T \subseteq [n]$.
Seien weiter c, c_1, c_2 Funktionen und wie folgt definiert:

$$\begin{aligned} c, c_1, c_2 : \quad & 2^D \rightarrow \mathbb{R}, \\ c_1 : \quad & R \mapsto |1_{L \perp R} \cap T|, \\ c_2 : \quad & R \mapsto |1_{L \perp R}|, \\ & c = W \cdot c_1 - c_2. \end{aligned}$$

Dann:

$$\begin{aligned} R^* \text{ Optimum von 7.1} & \Leftrightarrow R^* = \operatorname{argmin}(c(R) \mid R \subseteq D), \\ R^* \text{ Optimum von 7.2} & \Leftrightarrow R^* = \operatorname{argmax}(c(R) \mid R \subseteq D). \end{aligned}$$

Die Crux an unseren Problemen ist die Ordnungserhaltung der beiden Teilfunktionen: Es gilt $R_1 \subseteq R_2 \Rightarrow c_i(R_1) \leq c_i(R_2)$ für $i \in \{1, 2\}$.

Verkleinern wir die eine Funktion, so besteht durchaus auch die Chance, dass dies mit der anderen Teilfunktion ebenfalls passiert. Ebenso, wenn wir eine vergrößern wollen. Da unsere

⁴Die Funktion *FindBlockingSet* muss diesmal angepasst sein auf $D \subseteq [n]$ beliebig.

Gesamtfunktion $c = W \cdot c_1 - c_2$ nun jedoch aus einer Differenz der beiden Funktionen besteht, ist es so schwer, eine geeignete Suchstrategie für R^* zu finden.

§7.3.3 Von Genen, Enzymen und Reaktionen

Die Probleme 7.1 und 7.2 lassen sich beide gleich abstrahieren: Lege dich auf eine Teilmenge der Reaktionen $R \subseteq [n]$ fest, die du verbieten kannst, und prüfe anschließend, ob das neue Verbandsmaximum $L_{\perp R}$ deinen Ansprüchen genügt.

Verbieten können heißt dabei, dass $R \subseteq D$ für eine vorher festgelegte Teilmenge der Reaktionen gelten muss, ansonsten aber beliebig ist. Die Ansprüche an $L_{\perp R}$ werden schließlich vom Problem vorgegeben.

Aber ist das eine sinnvolle Unterteilung?

Was den zweiten Teil angeht, vermutlich ja: Gebe Constraints an das System vor (wobei verbieten viel einfacher als zwingen ist) und prüfe, welche Reaktionen dann vorkommen können. Das ist eine solide Arbeitsgrundlage und etwas viel Besseres werden wir im Vereinigungsverband – der uns keine Flusswerte liefert – wohl kaum finden.

Anders sieht es mit dem ersten Teil und unserer Umsetzung aus: Haben wir überhaupt direkten Einfluss darauf, bestimmte Reaktionen auszuknipsen?

Gene sind der Anfang von allen Zellprozessen. Sie können als Vorlage für Enzyme genutzt werden, die dann jeweils im aktivierten Zustand Reaktionen katalysieren können. Eine Reaktion kann nur dann stattfinden, wenn mindestens ein aktives Enzym anwesend ist, das sie katalysieren kann.

Wenn man all diese Prozesse verstehen will, ist man im Bereich der (Gen-)Regulation angekommen. Wie schon im Motivationskapitel angedeutet, gab es in den vergangenen Jahren unterschiedliche Bemühungen, regulatorische Prozesse in die Stoffwechselbeschreibung zu integrieren. Die meisten davon versuchen Regulation für bessere FBA zu nutzen – weshalb wir darauf in dieser verbandsorientierten Arbeit bisher nicht eingegangen sind.

Erwähnt haben wir jedoch (Kim & Reed 2010) als Beispiel für Verfahren aus dem Bereich Metabolic Engineering. Auch wenn beim vorgestellten *OptORF* wieder einmal FBA Kern des Verfahrens ist, wollen wir uns hier eine seiner äußeren Schichten zu Nutze machen. Die Autoren gehen detailliert darauf ein, warum man nicht Reaktionen, sondern Gene deaktivieren sollte und ziehen hierfür einen Vergleich zu *OptKnock* (Burgard, Pharkya & Maranas 2003), wo eine Kopplung von Wachstum und Zielmetabolit durch Reaktionen-Knockout erreicht wird.

Wir wollen hier nicht alle Details ihres raffinierten Verfahrens kennenlernen. Stattdessen geht es um einen Machbarkeitsbeweis, dass man auch Verfahren, die auf Verbandstheorie basieren, um genetische Informationen ergänzen kann. Dafür nutzen wir wieder einmal die Abstraktionsmöglichkeiten der Mathematik und sagen (ohne uns auf eine konkrete Implementierung festzulegen):

Bemerkung 7.4 (Target Prediction für Gendeaktivierung).

Sei (L, Rev) metabolisches System, $T \subseteq [n]$ Menge an Zielreaktionen.

Sei nun $[g]$ die Menge aller Gene und $\psi : 2^{[g]} \rightarrow 2^{[n]}$ so, dass $\psi(A)$ genau die Reaktionen umfasst, die nicht katalysiert werden, wenn alle Gene aus A deaktiviert sind oder fehlen.

Sei weiterhin ψ ordnungserhaltend bezüglich Inklusion, also $A_1 \subseteq A_2 \Rightarrow \psi(A_1) \subseteq \psi(A_2)$.

Dann lassen sich die Algorithmen *FindBlockingSet*, *FindHighlightingSets* und *FindMinimalHighlightingSet* leicht abwandeln, um optimale Lösungen für die auf Gendeaktivierung umformulierten Probleme 7.1 und 7.2 zu finden.

So ähnlich wie uns die Arbeit mit allgemeinen Vereinigungsverbänden viel Freiheiten lässt, ein Modell zu wählen, das Regeln der eigenen Überzeugung entspricht, so ähnlich sieht es nun auch bei der Übersetzung Gendeaktivierung \rightarrow Reaktionenblockierung aus. Unser Verfahren kann weiter genutzt werden, solange ψ ordnungserhaltend ist.

Doch welche Optionen zur Verfügung stehen, um ψ ordnungserhaltend zu implementieren, soll Frage für künftige Arbeiten bleiben.

§7.3.4 War da nicht was? – Minimal Cut Sets, 2. Akt

Blockiere mit so wenigen Reaktionen wie möglich die Zielmenge. So kann man Problem 7.1 beschreiben, wenn man die Formalia weglässt.

Auf ein ähnliches Problem sind wir schon einmal viel früher gestoßen, auch wenn unsere Umsetzung eine ganz andere war: Die Rede ist von (minimalen) **Überdeckungen** und dem dualen Verband L^\perp .

Der zu einem Vereinigungsverband L duale Verband L^\perp besteht im Wesentlichen aus den Teilmengen $A \subseteq [n]$, die jedes Element aus L treffen. Die Idee war, wenn $A \cap a \neq \emptyset$ für alle $a \in L \setminus \{0\}$ gilt, so gibt es nur noch den trivialen Stoffwechselfad, wenn wir alle Reaktionen aus A blockieren.

Wie bei der Definition von L^\perp schon erwähnt, hat die Forschungsgruppe um Klamt viel an dem Thema gearbeitet. Dort sind im Laufe der Jahre immer raffiniertere Verfahren entwickelt worden, um die minimalen Elemente $\mathcal{M}((L^F)^\perp)$, die Minimal Cut Sets, zu berechnen; zuletzt sogar durch Einführung eines pseudo-dualen Netzwerkes, aus dessen Elementarmoden sich die minimalen Überdeckungen ablesen lassen (Ballerstein et al. 2011). Weiterhin sei auf Hädicke & Klamt (2011) verwiesen, wo ähnliche Probleme wie die hier diskutierten mittels Minimal Cut Sets gelöst werden.

Unsere Definition (und Dualitätsbeweis) unterscheidet vom Original unterscheidet, dass wir nicht auf den stationären Flussverband beschränken. Unsere Ergebnisse können auch für andere Fälle eingesetzt werden. Wenden wir das geschickt an, so kommt folgender Zusammenhang ans Licht:

Satz 7.6.

Sei (L, Rev) metabolisches System und $D, T \subseteq [n]$.

Ist ferner R^* Lösung des Algorithmus *FindBlockingSet*, so gilt:

$$R^* \in \mathcal{M} \left(\left((L_{\parallel T})^\perp \right)_{\subseteq D} \right),$$

$$R \in \left((L_{\parallel T})^\perp \right)_{\subseteq D} \Rightarrow |R^*| \leq |R|.$$

Beweis. $L_{\parallel T}$ ist der Unterhalbverband an Stoffwechselfaden, die mindestens eine Reaktion aus T verwenden.

Da das Ausschalten von R^* sämtliche Reaktionen aus T verhindert, ist somit $R^* \in (L_{\parallel T})^\perp$.

Weiterhin ist $R^* \subseteq D$, also $R^* \in \left((L_{\parallel T})^\perp \right)_{\subseteq D}$.

Da R^* nun auch noch die kleinste Kardinalität unter all diesen Elementen haben soll und Algorithmus *FindBlockingSet* das gewährleistet, folgt der Rest. \square

Hat man die Notation erst einmal verstanden, so sieht man schnell:

Bemerkung 7.5.

Algorithmus *FindBlockingSet* lässt sich leicht abwandeln, so dass er $\mathcal{M} \left(\left((L_{\parallel T})^\perp \right)_{\subseteq D} \right)$ berechnet.

Auch unser Förderungsproblem 7.2 ist mit einer anderen Struktur verwandt. Dafür müssen wir nicht den dualen Verband bemühen, sondern erhalten den Spezialfall:

Satz 7.7.

Sei (L, Rev) metabolisches System und $D = [n], T \subseteq [n]$.

Ist ferner R^* Lösung des Algorithmus *FindMinimalHighlightingSet*, so gilt:

$$1_{L_{\perp R^*}} \in \mathcal{M}(L_{\supseteq T}),$$

$$a \in \mathcal{M}(L_{\supseteq T}) \Rightarrow |1_{L_{\perp R^*}}| \leq |a|,$$

$$R \subseteq [n], 1_{L_{\perp R}} = 1_{L_{\perp R^*}} \Rightarrow |R^*| \leq |R|.$$

Damit wollen wir nun die Welt der maximalen Elemente verlassen. Im Abschlusskapitel dieser Arbeit wollen wir uns einem speziellen Unterhalbverband von L^F widmen, der sich stärker an physikalische Gesetze hält. Im Anhang betrachten wir dann Simulationsergebnisse.

Doch vorher gibt's in aller Ruhe erst noch einen Kaffee.



Kapitel 7 – Kurz und knapp

Maximum

- ▷ Ein schneller Algorithmus zur Berechnung des Maximums führt zu *effizienten Algorithmen* für viele Probleme – etwa FCA.
- ▷ Wir stellen ein einfaches Verfahren vor, das auch für *genomweite Netzwerke* schnell terminiert.
- ▷ Unser Algorithmus ist schnell genug für neue Probleme, wie eine Simulation aller denkbaren *Doppel-Knockouts*.

Target Prediction

- ▷ Netzwerke lassen sich gezielt manipulieren.
- ▷ Das Maximum ermöglicht neue *Optimierungsziele*; wir reden über mögliche Reaktionen statt maximalen Fluss durch eine Reaktion wie bei der FBA.
- ▷ Unsere Target Prediction lässt sich optimal ermitteln, liefert durch *heuristische Verfahren* aber auch schnell erste Lösungen.



KAPITEL 8

Das Starbucks-Lemma

§8.1 Über Geschmack lässt sich nicht streiten

Jedem Menschen schmeckt etwas anderes. Deshalb hat Starbucks¹ schon vor Jahren ein System entwickelt, bei dem jeder Kunde ein Getränk ganz nach seinem Geschmack erhalten kann, ohne dass die Getränkekarte ausufert.

Um die Möglichkeiten übersichtlich zu gestalten, werden alle Getränke zu Basisgetränk und Geschmackszugaben zerlegt. Als Basis wählt man etwa einen Latte Macchiato und wünscht sich dann ganz nach eigenen Vorlieben noch die verwendete Milchsorte, Sahne, Karamellsirup, Kakaopulver oder Zimt dazu oder nicht. Das gesamte System ist modular aufgebaut und so gibt es bei allen Arbeitsroutinen keine Probleme, auf Wünsche des Kunden einzugehen.

Motivation dieser Arbeit ist es, verschiedenste Stoffwechselmodelle miteinander zu vergleichen und in einer einheitlichen Sprache zu beschreiben. Der Sprache der Verbandstheorie. Damit haben wir Untersuchungswerkzeuge geschaffen, deren Benutzung der Anwender nur einmal erlernen muss und sie dann auf ein Modell ganz nach seiner Wahl anwenden kann.

Im Laufe dieser Arbeit haben wir viele Einzelaspekte kennengelernt, die teilweise schon in der Literatur vorkommen. Für jeden Einzelnen haben wir eine verbandstheoretische Formulierung gesucht und oft gleich noch Algorithmen für bestimmte Anwendungen vorgestellt. Jeder einzelne Aspekt ist wichtig: Es gibt nicht DAS eine, perfekte Modell. Jedes bringt Vor- und Nachteile mit sich und so bevorzugen verschiedene Wissenschaftler unterschiedliche Ansätze. Ganz ähnlich wie bei Starbucks jedem Kunden der Kaffee Latte anders schmeckt.

Dieses Kapitel wird nun zeigen, wie viel einfacher die Modellierung durch das Starbucks-Prinzip wird. Ein Prinzip, das sich sehr gut auf Verbände übertragen lässt. Nach Wahl des Grundmodells (z. B. stationäre Zustände) kommen thermodynamische Constraints, Ressourcenbeschränkungen oder eine von vielen anderen Eigenschaften hinzu. Wir erhalten hochgradig individualisierte Modelle, die wir einheitlich untersuchen können.

¹Starbucks Corp. (NASDAQ: SBUX) ist ein auf Kaffeeprodukte spezialisiertes und international tätiges Einzelhandelsunternehmen mit Hauptgeschäftssitz in Seattle in den USA. (Wikipedia 2012c)



Lemma 8.1 (Starbucks-Lemma).

Sei L Verband.

Sei zudem $I = [\alpha^{(1)}, \beta^{(1)}] \times \dots \times [\alpha^{(d)}, \beta^{(d)}]$ Intervall im \mathbb{R}^d und $\varphi : L \rightarrow I$ mit $\varphi_1, \dots, \varphi_d$ ordnungserhaltende Abbildungen, $\varphi_k : L \rightarrow [\alpha^{(k)}, \beta^{(k)}]$ für $1 \leq k \leq d$.
Ausserdem sei $\varphi_k(0) = \alpha^{(k)}$ und $\varphi_k(1_L) = \beta^{(k)}$.

Sei nun $t \in I$ beliebig. Definiere $L_{\varphi \leq t} \subseteq L$ wie folgt:

$$L_{\varphi \leq t} := \left\langle \underbrace{\left\{ a \in L \mid \varphi_k(a) \leq t^{(k)} \quad \forall k \in [d] \right\}}_{P_{\varphi \leq t}} \right\rangle.$$

Dann ist $L_{\varphi \leq t}$ Unterverband von L und es gilt:

$$P_{\varphi \leq t} \subseteq L_{\varphi \leq t} \subseteq \bigcap_{k=1}^d L_{\varphi_k \leq t^{(k)}}.$$

Gleichheit gilt im Allgemeinen nicht; dafür wissen wir über die minimalen / die unzerlegbaren Elemente:

$$\begin{aligned} \mathcal{M}(L_{\varphi \leq t}) &= \bigcap_{k=1}^d \mathcal{M}(L_{\varphi_k \leq t^{(k)}}) \\ &= P_{\varphi \leq t} \cap \mathcal{M}(L), \\ \mathcal{J}(L_{\varphi \leq t}) &= \bigcap_{k=1}^d \mathcal{J}(L_{\varphi_k \leq t^{(k)}}) \\ &= P_{\varphi \leq t} \cap \mathcal{J}(L), \end{aligned}$$

Ist (L, Rev) metabolisches System und $P \subseteq L$ nun beliebige Eigenschaft, so zeigt $1_{\langle P \rangle}$ die Reaktionen an, die unblockiert sind, wenn alle gültigen Pfade die Eigenschaft P haben müssen.

Ausserdem können wir auf Flusskopplung von $i, j \in 1_{\langle P \rangle}$ untersuchen:

$$\begin{aligned} i \xrightarrow{=0} j \text{ (in } P) &:\Leftrightarrow i \notin a \Rightarrow j \notin a \quad \forall a \in P \\ &\Leftrightarrow i \xrightarrow{=0} j \text{ (in } \langle P \rangle) \\ &\Leftrightarrow j \notin 1_{\langle P \rangle \perp \{i\}}. \end{aligned}$$

Damit können wir unseren Algorithmus *FastMax* verwenden, sobald wir eine Test-Funktion $\text{Test}(i, R)$ mit $\text{Test}(i, R) \in \{0\} \cup \hat{P}$ und $\mathcal{J}(\langle P \rangle) \subseteq \hat{P} \subseteq P$ haben.

Beweis. Zunächst zeigen wir die Behauptungen über die geschnittenen Verbandstümpfe, bevor wir uns den blockierten und gekoppelten Reaktionen allgemeiner Eigenschaften zuwenden.

Sei $\mathbf{a} \in L_{\varphi \leq t}$ beliebig. Dann ist $\mathbf{a} = \mathbf{a}_1 + \dots + \mathbf{a}_r$ mit $\mathbf{a}_1, \dots, \mathbf{a}_r \in P_{\varphi \leq t}$. Insbesondere ist damit $\mathbf{a}_i \in P_{\varphi_k \leq t^{(k)}}$ für $1 \leq i \leq r$ und $1 \leq k \leq d$. Damit ist aber auch $\mathbf{a} = \mathbf{a}_1 + \dots + \mathbf{a}_r \in L_{\varphi_k \leq t^{(k)}}$ mit $k \in [d]$ beliebig. Also wissen wir:

$$L_{\varphi \leq t} \subseteq \bigcap_{k=1}^d L_{\varphi_k \leq t^{(k)}}.$$

Dank Satz 3.5 über Verbandstümpfe wissen wir:

$$\begin{aligned} \mathcal{M}(L_{\varphi_k \leq t^{(k)}}) &= \mathcal{M}(L) \cap P_{\varphi_k \leq t^{(k)}}, \\ \mathcal{J}(L_{\varphi_k \leq t^{(k)}}) &= \mathcal{J}(L) \cap P_{\varphi_k \leq t^{(k)}}. \end{aligned}$$

Wir wissen, dass $\mathcal{M}(L_{\varphi \leq t}) \subseteq \mathcal{J}(L_{\varphi \leq t}) \subseteq P_{\varphi \leq t}$ gilt. Es reicht daher, wenn wir jetzt nur noch Elemente aus $P_{\varphi \leq t}$ betrachten. Seien $\mathbf{a}_1, \mathbf{a}_2 \in L$ mit $0 < \mathbf{a}_1 < \mathbf{a}_2$ und $\mathbf{a}_2 \in P_{\varphi \leq t}$. Dann wissen wir, dass $\mathbf{a}_2 \notin \mathcal{M}(L_{\varphi \leq t})$ gilt, da die φ_k ordnungserhaltende Abbildungen sind. Wegen $\mathbf{a}_1 \in L_{\varphi \leq t} \subseteq \bigcap_{k=1}^d L_{\varphi_k \leq t^{(k)}}$ ist damit aber \mathbf{a}_2 auch nicht-minimal in allen $L_{\varphi_k \leq t^{(k)}}$.

Sei nun $\tilde{k} \in [d]$ und $\mathbf{a}_1, \mathbf{a}_2 \in L_{\varphi_{\tilde{k}} \leq t^{(\tilde{k})}}$ mit $0 < \mathbf{a}_1 < \mathbf{a}_2 \in P_{\varphi \leq t}$. Da alle φ_k ordnungserhaltend sind, ist dann aber auch $\mathbf{a}_1 \in P_{\varphi \leq t}$ und somit \mathbf{a}_2 nicht-minimal in $L_{\varphi \leq t}$.

Beides zusammen ergibt:

$$\mathbf{a} \in \mathcal{M}(L_{\varphi \leq t}) \Leftrightarrow \mathbf{a} \in \mathcal{M}(L_{\varphi_k \leq t^{(k)}}) \quad \forall k \in [d].$$

Verwenden wir statt $\mathbf{a}_1 < \mathbf{a}_2$ Zerlegungen $\mathbf{a}_1 + \dots + \mathbf{a}_r = \mathbf{a}$, so zeigen wir analog:

$$\mathbf{a} \in \mathcal{J}(L_{\varphi \leq t}) \Leftrightarrow \mathbf{a} \in \mathcal{J}(L_{\varphi_k \leq t^{(k)}}) \quad \forall k \in [d].$$

Setzen wir Satz 3.5 ein, so erhalten wir die Behauptung.

Wenden wir uns nun den metabolischen Systemen zu. Hier lassen wir beliebige Eigenschaften P zu, wenn wir blockierte und gekoppelte Reaktionen bestimmen wollen.

Eine Reaktion i heißt blockiert, wenn sie in keinerlei Stoffwechselfad vorkommt:

$$i \text{ blockiert in } P \quad :\Leftrightarrow \quad i \notin \mathbf{a} \quad \forall \mathbf{a} \in P.$$

Hierauf wenden wir nun unsere bekannten Aussagen über unzerlegbare Elemente (minimaler

Erzeuger!) und das Verbandsmaximum an. Es ist $\mathcal{J}(\langle P \rangle) \subseteq P$ und damit:

$$\begin{aligned}
 i \text{ blockiert in } P &\Rightarrow i \notin a \forall a \in P \\
 &\Rightarrow i \notin a \forall a \in \mathcal{J}(\langle P \rangle) \\
 &\Rightarrow i \notin 1_{\langle P \rangle} = \sum_{b \in \mathcal{J}(\langle P \rangle)} b \\
 &\Rightarrow i \notin a \forall a \in \langle P \rangle \\
 &\Rightarrow i \notin a \forall a \in P \\
 &\Rightarrow i \text{ blockiert in } P.
 \end{aligned}$$

Schauen wir uns nun Reaktionenkopplung an.

Sei $i \in 1_{\langle P \rangle}$ beliebig. Diesmal zeigen wir die Äquivalenz, indem wir die beiden Einzelrichtungen in ihrer Kontraposition zeigen, und beziehen hierfür auch schon \tilde{P} ein. Wir erinnern uns daran, dass Nicht-Kopplung durch einen einzigen Zeugen gezeigt werden kann und erhalten:

$$\begin{aligned}
 a \in P \text{ mit } i \notin a \ni j &\stackrel{P \subseteq \langle P \rangle}{\Rightarrow} a \in \langle P \rangle \text{ mit } i \notin a \ni j, \\
 a \in \langle P \rangle \text{ mit } i \notin a \ni j &\Rightarrow \exists b \in \mathcal{J}(\langle P \rangle) : i \notin b \ni j \\
 &\Rightarrow \exists b \in \tilde{P} : i \notin b \ni j \\
 &\Rightarrow \exists b \in P : i \notin b \ni j.
 \end{aligned}$$

Reaktionen sind also bei Stoffwechselfaden mit gemeinsamer Eigenschaft genau dann gekoppelt, wenn sie im induzierten Verband aneinander gekoppelt sind. Um dessen Maximum zu bestimmen, reicht es bereits Elemente einer genügend großen Teilmenge der Eigenschaft von Interesse bestimmen zu können. \square

Die ganze Arbeit hindurch standen immer wieder die gleichen drei Strukturen im Vordergrund: Minimale Elemente, unzerlegbare Elemente und Verbandsmaxima. Sie haben sich herauskristallisiert, weil sie besonders wichtig im Verband sind. Sie sind die Augenfänger. Betrachtet man ein Hasse-Diagramm, so betrachtet man am ehesten Fuß und Spitze, das Fundament (eventuell das Dach), und zuletzt vielleicht die Elemente, die nur durch eine Kante erreicht werden.

Das Starbucks-Lemma fasst jetzt so viele Informationen wie möglich zusammen, die wir über solche Elemente haben. Wir schaffen es, mit einem künstlichen Verbandsmaximum für Modelle beliebigen Geschmacks blockierte und gekoppelte Reaktionen zu untersuchen, für eine große Klasse an Eigenschaften kennen wir außerdem noch die erzeugenden Elemente. Kennen wir sie im Ursprungsverband, können wir direkt nach allen Eigenschaften filtern. Durch das Starbucks-Lemma wissen wir, dass wir den Erzeuger der Geschmacks Mischung in Händen halten.

Ein Verband ist komplett durch seine unzerlegbaren Elemente definiert. Aus dem Starbucks-Lemma folgt, dass die Reihenfolge der Eigenschaften unwichtig ist und wir halten fest:

 **Korollar 8.1.**

$$\left\langle \left(L_{\varphi_1 \leq t^{(1)}} \right)_{\varphi_2 \leq t^{(2)}} \right\rangle = \left\langle \left(L_{\varphi_2 \leq t^{(2)}} \right)_{\varphi_1 \leq t^{(1)}} \right\rangle = \left\langle P_{\varphi_1 \leq t^{(1)}} \cap P_{\varphi_2 \leq t^{(2)}} \right\rangle.$$

Es mag elegant sein, ein Maximum in einem Schritt (mit einem MILP) ermitteln zu können. Doch leider legen wir uns damit sofort auf den stationären Flussverband L^F fest.

Unser Algorithmus *FastMax* ist vielseitig genug, um das (hypothetische) Maximum eines jeden Unterverbandes zu finden. Sobald wir eine Implementation $\text{Test}(i, R)$ haben, geht's. Das ist gleichbedeutend damit, einen Pfad durch i (im Subsystem) finden zu können. Eine Aufgabe, die ein Modell erst nutzbar macht.

§8.1.1 Formale Begriffsanalyse

Wie eingangs erwähnt gibt es seit einigen Jahrzehnten den Versuch, die Verbandstheorie leicht verständlich in Anwendung zu bringen. Die Formale Begriffsanalyse basiert auf (endlicher) Verbandstheorie und findet Anwendungen in allen denkbaren informationstechnischen Problemstellungen.

Unser Starbucks-Lemma trifft genau das Herz der Formalen Begriffsanalyse: Wir wollen verstehen, welche Mengenelemente – in unserem Fall Stoffwechselfade – Eigenschaften teilen. Seien es Eigenschaften an das Modell selbst oder Eigenschaften innerhalb eines Modells. So würden wir jetzt für Kegelstümpfe $P = P_{\varphi \leq t}$ feststellen: $\mathcal{M}(\langle P \rangle) = \mathcal{M}(L) \circ_K P$, wobei der Operator \circ_K aus dem Kontextverband stammt, den wir aus unserem Modell bilden können.

Die Formale Begriffsanalyse ist ein sehr spannendes Gebiet, das viele interessante Ergebnisse produziert hat und sicherlich auch weiterhin produzieren wird. In dieser Arbeit wollen wir aber ganz der klassischen Verbandstheorie verschrieben bleiben. Deshalb wollen wir nach dieser kurzen Erwähnung unsere Arbeit wie bisher fortsetzen. Einen richtigen Einstieg in die Konzepttheorie findet der interessierte Leser beispielsweise bei Ganter et al. (2005).

§8.2 Guter Geschmack ist Ordnungsideal

Wir wissen nun, dass wir unser Modell beliebig um Bedingungen ergänzen können, die sich als Verbandsstumpf $L_{\varphi \leq t}$ formulieren lassen, und dennoch Minima (und damit Cut Sets, Target Prediction), Erzeuger (und damit Flusskopplung) und blockierte Elemente (und damit schnelle Algorithmen) untersuchen können. Doch welche Eigenschaften sind das?

Bevor wir diese Frage beantworten, ist es mal wieder Zeit, das Werk anderer Autoren zu würdigen. Wie in den Grundlagenkapiteln zur Verbandstheorie erwähnt, wurde in der Vergangenheit schon viel Arbeit auf diesem Gebiet getan. Auch die Frage, wann der Durchschnitt


zweier Verbände einen Unterverband bildet, wurde bereits diskutiert. Topkis (1976) diskutiert dies sogar für das Produkt mehrerer Verbände ausführlich in seiner Arbeit.

Mit dieser Arbeit soll im Gegensatz zu Topkis (1976) nicht eine mathematische Struktur besser verstanden werden, sondern Einsatzmöglichkeiten einer mathematischen Struktur in der Biologie möglichst praktikabel dargestellt werden. Und so soll auch das Starbucks-Lemma einer möglichst großen Leserschaft vorrangig zeigen, was wir machen können: Eigenschaften kombinieren und dennoch untersuchen. Im Gegensatz zu Topkis müssen solche Eigenschaften alleine nicht Unterverband sein.

Doch zurück zur eigentlichen Frage. Welche Eigenschaften sind kombinierbar?

Der Graphentheoretiker würde wohl von *monotonen* oder *vererblichen* Eigenschaften reden. Gemeint ist damit:

Ordnungs-
ideal




Definition 8.1 (Ordnungsideal).
 Sei L Verband und $P \subseteq L$.

P *Ordnungsideal* $:\Leftrightarrow \forall a_1 \in L \forall a_2 \in P : a_1 \leq a_2 \Rightarrow a_1 \in P$.

Ein Ordnungsideal ist also eine Teilmenge des Verbandes, bei der jedes kleinere Element auch enthalten sein muss. Oder auf metabolische Systeme bezogen: Hat ein gewisser Stoffwechselfad die gewünschte Eigenschaft, so hat das auch jeder kleinere Stoffwechselfad.

Ordnungsideale sind – der Name verrät es schon – eng verwandt mit den **Hauptidealen**. Denn Verbands-**Ideale**, die wir bisher als Hauptideale kennengelernt haben, sind selbst Ordnungsideale.

Ideal




Definition 8.2 (Ideal).
 Sei L Verband und $\mathcal{J} \subseteq L$.

Es ist \mathcal{J} *Ideal* von L , wenn es *Ordnungsideal* und zudem *Unterverband* ist. D. h. es gilt:

$\forall a_1 \in L \forall a_2 \in \mathcal{J} : a_1 \leq a_2 \Rightarrow a_1 \in \mathcal{J},$
 $\forall a_1, a_2 \in \mathcal{J} : a_1 + a_2 \in \mathcal{J}.$


Man sieht schnell ein:



Bemerkung 8.1 (Im endlichen Verband gibt es nur Hauptideale).
 Sei L *endlicher Verband*, dann ist jedes *Ideal* *Hauptideal*.

Wie gesagt, Ordnungsideal sind verwandt mit Hauptidealen. Das folgt schon aus der Definition des Ideals. Man sieht leicht ein, dass das Erzeugnis $\langle P \rangle$ eines Ordnungsideals $P \subseteq L$ zwar Unterverband ist, aber oftmals selbst nicht mehr Ordnungsideal und damit insbesondere kein Ideal ist.

Vererbung ist intuitiv. Das ist sehr nützlich, denn:

 **Satz 8.1** (Verbandsstümpfe sind Ordnungsideale).
 Sei L endlicher Verband und $P \subseteq L$. Dann:

$$P \text{ Ordnungsideal} \Leftrightarrow \exists \varphi : L \rightarrow \mathbb{R} \exists t \in \mathbb{R} : P = P_{\varphi \leq t}, L_{\varphi \leq t} \text{ Verbandsstumpf.}$$

Verbands-
stumpf =
Ordnungs-
ideal

Beweis. Beginnen wir mit der Rückrichtung.

Ist $L_{\varphi \leq t}$ Verbandsstumpf, so ist φ nach Definition monoton. Sei nun $a_1 \leq a_2$ in L mit $a_2 \in P_{\varphi \leq t}$, also $\varphi(a_2) \leq t$. Wegen der Monotonie folgt $\varphi(a_1) \leq \varphi(a_2) \leq t$, also auch $a_1 \in P = P_{\varphi \leq t}$.

Nehmen wir nun an, das $P \subseteq L$ Ordnungsideal ist. Definiere die Abbildung $\varphi_P : L \rightarrow \{0, 1\}$ wie folgt:

$$\varphi_P : L \rightarrow \{0, 1\},$$

$$a \mapsto \begin{cases} 0 & a \in P, \\ 1 & a \notin P. \end{cases}$$


Diese Funktion ist monoton, da P Ordnungsideal ist.

Es gilt $P = P_{\varphi_P \leq 0}$ und wir sind fertig. □

Es reicht also zu zeigen, dass eine Eigenschaft Ordnungsideal ist, und schon wissen wir, dass wir sie als weitere Geschmacksrichtung für unser Modell anbieten können. Je nach Eigenschaft kann es dabei einfacher sein, eine monotone Funktion zu finden, oder nachzuweisen, dass es sich um ein Ordnungsideal handelt.

Erinnern wir uns an die **Hyperpfade** $p = (a_1, \dots, a_t) \in \mathcal{H}$ (Definition 2.6), so liegt die Vermutung nahe, dass es sich auch bei ihnen um ein Ordnungsideal handelt: Schließlich ist jeder abgeschnittene Unterpfad $p_k = (a_1, \dots, a_k)$ auch wieder Hyperpfad.

Doch leider steckt der Teufel auch hier wie so oft im Detail. Hyperpfade sind geordnete Mengen. Auf die Reihenfolge der Elemente kommt es an – nicht *jede* Teilmenge eines Hyperpfades ist wieder Hyperpfad, sondern nur abgeschnittene Teilpfade. Daher bemerken wir:

 **Bemerkung 8.2** (Hyperpfade sind keine Ordnungsideale).
 Es ist $L^{\mathcal{H}} \subseteq L^{\rightarrow 01}$ Unterhalbverband, aber kein Ordnungsideal.
 Bei dem Beispielnetzwerk in Abbildung 8.1 ist $p = (v^+, v^-, u^+, u^-, r, s)$ Hyperpfad und damit $a = \{v^+, v^-, u^+, u^-, r, s\} \in L^{\mathcal{H}}$. Weiterhin ist $\tilde{a} = \{u^+, u^-, r, s\} \in L^{\rightarrow 01}$. Es gibt jedoch keine Anordnung dieser vier Reaktionen, die zu einem gültigen Hyperpfad führen würde. Damit gibt es $\tilde{a} \subsetneq a$ mit $\tilde{a} \notin L^{\mathcal{H}}$, also handelt es sich bei den Hyperpfaden nicht um ein Ordnungsideal.

\mathcal{H} nicht
Ordnungs-
ideal

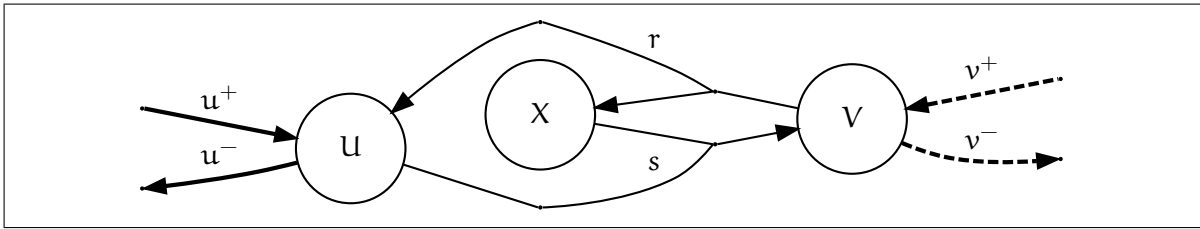


Abbildung 8.1.: Ein Hyperpfad, wo nicht jede Teilmenge wieder Hyperpfad ist.

Ein besonderer Fall von Ordnungsideal ist der Ressourcenmangel. Er war unsere ursprüngliche Motivation, um uns überhaupt mit solchen Teilmengen an Stoffwechselfaden zu beschäftigen.

Die Bereitstellung der Enzyme zum Katalysieren einer Reaktion erfordert Ressourcen, so dass bei Ressourcenknappheit nicht alle gleichzeitig zur Verfügung stehen. Jedes Enzym hat Kosten. Man sieht leicht, dass Ressourcenverteilung Ordnungsideal ist: Verwenden wir weniger Reaktionen, verbrauchen wir weniger Ressourcen.

Formalisiert landen wir bei:

Ressourcen-
mangel

Korollar 8.2 (Ressourcenmangel).

Sei L Vereinigungsverband über $[n]$ und $w : [n] \rightarrow [0, +\infty)$ Ressourcenkosten sowie $\tau > 0$ Vorrat an Ressourcen.

Sei $\varphi_w : L \rightarrow [0, +\infty)$ wie folgt definiert:

$$\varphi_w(\mathbf{a}) = \sum_{i \in \mathbf{a}} w(i).$$

Dann ist $P_{\varphi_w \leq \tau}$ die Menge aller Stoffwechselfade, die durch Verteilungen der Ressourcen möglich sind.

Einen weiteren Fall haben wir bereits unbewusst genutzt, wenn wir schrittweise die kürzesten minimalen Elemente berechnen: Das Kombinationsverbot. Es wird Ausgangspunkt für die meisten Modellanpassungen sein, auf die man stößt.

Kombinations-
verbot

Korollar 8.3 (Kombinationsverbot).

Sei L Vereinigungsverband über $[n]$ und $K \subseteq [n]$.

Sei $P_{-K} = \{\mathbf{a} \in L \mid K \cap \mathbf{a} \neq K\}$.

Dann ist P_{-K} Ordnungsideal und es gilt $P_{-K} = P_{\varphi_{\sum K} \leq |K|-1}$ mit:


$$\varphi_{\sum K}(\mathbf{a}) = |\mathbf{a} \cap K|.$$

Eine interessante Anwendung für das Kombinationsverbot finden Chan & Ji (2011). In ihrer Arbeit stellen sie einen Algorithmus vor, mit dem sie beliebige Flussvektoren $v \in F$ in elementare Flussmoden zerlegen können. Im Gegensatz zu anderen Arbeiten (z. B. Wang, Yang, Ma & Zhao 2007, Song & Ramkrishna 2009) müssen sie dafür nicht im Voraus die gesamte Menge an Elementarmoden E berechnen.

Um das zu schaffen, bedienen sie sich fleißig des Lemmas 4.8 und zerlegen nicht-minimale Flussvektoren solange wie möglich – minimale Flussvektoren sind elementare Flussmoden und sie sind fertig.

Kern ihres Algorithmus ist ein *Decomposability Check*, der als MILP formuliert ist. Um zu testen, ob ein bekannter Flussvektor $v \in F$ zerlegbar ist, beziehen sie sich u. a. auf Figueiredo et al. (2009) und variieren das uns wohlvertraute lineare Programm aus Algorithmus *CalculateMinimalElements* darum, dass der Träger eines neu gefundenen Flussvektors p echt kleiner als der von v sein soll. Gibt es einen solchen Vektor p , so ist v nicht minimal und damit zerlegbar, sonst ist $v \in E$.

Das lässt sich auch verbandstheoretisch ausdrücken. Wir formulieren ein Lemma, dass die Erkenntnisse von Chan & Ji noch ergänzt.



Lemma 8.2 (Test auf Zerlegbarkeit).


Sei L Vereinigungsverband über $[n]$ und $a \in L$ bekannt. Dann sind äquivalent:

$$a \in \mathcal{M}(L) \iff (L_{\subseteq a})_{-a} = \{0\} \iff \forall i \in a : L_{\subseteq a \setminus \{i\}} = \{0\}.$$

Test auf
Zerlegbar-
keit

Die erste Äquivalenz entspricht dem MILP von Chan & Ji. Die zweite hingegen ist neu, so trivial sie auch ist. Sie erlaubt es uns mittels maximal $|\text{supp}(v)|$ LPs zu prüfen, ob $v \in E$ gilt, statt ein MILP lösen zu müssen. Wieder einmal obliegt es dem Nutzer, ob er lieber ein NP-schweres Problem oder lieber $\mathcal{O}(n)$ polynomielle Probleme lösen will, wenn er L^F betrachtet.

Wenden wir das Starbucks-Lemma an, so können wir gleich eine ganze Reihe von Kombinationen verbieten.



Korollar 8.4 (Kombinationenverbot).

Sei L Vereinigungsverband über $[n]$ und $\mathcal{K} \subseteq 2^{[n]} \setminus \{\emptyset\}$.


Sei $P_{-\mathcal{K}} = \{a \in L \mid K \cap a \neq K \ \forall K \in \mathcal{K}\}$.

Dann ist $P_{-\mathcal{K}}$ Ordnungsideal und es gilt:

$$P_{-\mathcal{K}} = \bigcap_{K \in \mathcal{K}} P_{-K}.$$

$P_{-\mathcal{K}}$

Beim Kombinationenverbot handelt es sich um eine Verallgemeinerung des gezielten Reaktionsverbots. Das kann man auch formal zeigen.



Bemerkung 8.3.

Sei L Vereinigungsverband über $[n]$ und $R \subseteq [n]$ beliebig. Dann:

$$L_{\perp R} = P_{-\{\{i\} \mid i \in R\}}.$$

§8.3 Von Kreisen und Thermodynamik


Nach dem Starbucks-Prinzip wird ein Basisgetränk mit Zusätzen auf den Geschmack des Kunden angepasst. Mit diesem Bild vor Augen bieten wir derzeit drei Basisgetränke, nämlich unsere Grundmodelle an. Der Flussverband, der sich aus dem stationären Flusskegel ableitet, der logische Verband, der stöchiometrische Informationen ignoriert und als SAT-Problem formuliert ist, und die Hyperpfade, die Reaktionen anordnen.

Einer der häufigeren Kritikpunkte am stationären Modell ist, dass physikalisch unsinnige Stoffwechselffade zugelassen werden. Es handelt sich um sogenannte Kreise, bei denen Metaboliten sich gegenseitig herstellen und so einen internen Kreislauf in Gang setzen, der zwar Energie verbraucht, aber der Zelle keinerlei Nutzen bringt. „Nutzen“ ist hierbei als Produktion gewisser Endstoffe zu verstehen, für die Nährstoffe verbraucht werden. Selbst bei den Hyperpfaden können wir das nicht immer umgehen, wie unschwer an Abbildung 8.1 zu erkennen ist.

Die meisten Modelle unterscheiden zwischen *internen* Reaktionen und *Transportreaktionen*. Transportreaktionen befördern einen oder mehrere Metaboliten in die Zelle oder aus ihr heraus, interne Reaktionen hingegen interagieren nicht mit der Zellumwelt.

Diese Unterscheidung führt uns zu folgender Definition:

Kreis




Definition 8.3 (Kreis im Stoffwechsel).
Sei (L, Rev) metabolisches System und $[n] = \text{Int} \dot{\cup} \text{Trans}$ disjunkte Zerlegung in interne Reaktionen und Transportreaktionen.
Sei $\alpha \in L$ beliebig, dann:

$$\alpha \text{ Kreis in } L \Leftrightarrow \alpha \subseteq \text{Int} \Leftrightarrow \alpha \in L_{\subseteq \text{Int}}.$$

$L_{\subseteq \text{Int}}$ ist der Unterverband aller Kreise im Stoffwechsel.

Schon früh hat man versucht, Kreise in der Simulation zu verbieten. So beschäftigte sich Price, Famili, Beard & Palsson (2002) mit dem Problem sowohl für extreme Pfade als auch für die FBA.

Das Starbucks-Lemma hat uns nun mit dem Kombinationenverbot ein Konzept zur Verfügung gestellt, mit dem wir nicht nur Kreise selbst verhindern können, sondern auch schon einen Kreis als Unterpfad verbieten können, wenn wir mit Verbänden arbeiten wollen. Als neue „Geschmacksoption“ können wir daher anbieten:


 **Definition 8.4** (kreisfreier Modellverband).
 Sei (L, Rev) metabolisches System und $[\mathfrak{n}] = \text{Int} \dot{\cup} \text{Trans}$ disjunkte Zerlegung in interne Reaktionen und Transportreaktionen. Es sei $L_{\subseteq \text{Int}}^* = L_{\subseteq \text{Int}} \setminus \{0\}$ die Menge aller nicht-trivialen Kreise in L .
 Definiere nun P_{-C} und L_{-C} wie folgt:

$$P_{-C} := P_{-L_{\subseteq \text{Int}}^*},$$

$$L_{-C} := \langle P_{-C} \rangle = L_{-L_{\subseteq \text{Int}}^*}.$$

Dann heißen die Elemente $\mathbf{a} \in P_{-C}$ *kreisfreie Stoffwechselfade* und L_{-C} *kreisfreier Modellverband*.

Diese Wahl als Modell kommt nicht von ungefähr. Es gilt nämlich folgender Satz:

 **Satz 8.2** (Zerlegung).
 Sei (L, Rev) metabolisches System und $[\mathfrak{n}] = \text{Int} \dot{\cup} \text{Trans}$ disjunkte Zerlegung in interne Reaktionen und Transportreaktionen. Gibt es in L auch die *ergänzenden Summanden* aus Lemma 3.4, dann gilt:

$$\forall \mathbf{a} \in L \exists \mathbf{a}_0 \in P_{-C} \exists \mathbf{c} \in L_{\subseteq \text{Int}} : \mathbf{a} = \mathbf{a}_0 + \mathbf{c}.$$

Beweis. Sei $\mathbf{a} \in P_{-C}$, dann wähle $\mathbf{a}_0 = \mathbf{a}, \mathbf{c} = 0$. Analog $\mathbf{a}_0 = 0, \mathbf{c} = \mathbf{a}$ für $\mathbf{a} \in L_{\subseteq \text{Int}}$.


Sei nun $\mathbf{a} \in L \setminus (P_{-C} \cup L_{\subseteq \text{Int}})$. Da \mathbf{a} nicht-kreisfrei ist, gibt es $\mathbf{c}_0 \in L_{\subseteq \text{Int}}$ mit $0 < \mathbf{c}_0 \leq \mathbf{a}$.

Da es in L ergänzende Summanden gibt, können wir schreiben $\mathbf{a} = \mathbf{a}_1 + \mathbf{c}_0$ mit $\mathbf{a}_1 < \mathbf{a}$. Ist $\mathbf{a}_1 \in P_{-C}$ enthalten, so wählen wir $\mathbf{a}_0 = \mathbf{a}_1, \mathbf{c} = \mathbf{c}_0$.

Ansonsten nutzen wir die Rekursion $\mathbf{a}_k = \mathbf{a}_{k+1} + \mathbf{c}_k$ mit $\mathbf{a}_{k+1} < \mathbf{a}_k$ und $\mathbf{c}_k \in L_{\subseteq \text{Int}}$. Diese Rekursion ist endlich, sei also $\tilde{k} \in \mathbb{N}$ so, dass $\mathbf{a}_{\tilde{k}} \in P_{-C}$. Dann wähle $\mathbf{a}_0 = \mathbf{a}_{\tilde{k}}$ und $\mathbf{c} =$

$$\sum_{k=0}^{\tilde{k}-1} \mathbf{c}_k. \quad \square$$

Damit bekommt man sofort folgende Zerlegung:

 **Korollar 8.5.**
 Wenn es in L ergänzende Summanden gibt, dann:

$$\mathcal{M}(L) = \mathcal{J}(L) = \mathcal{M}(L_{-C}) \dot{\cup} \mathcal{M}(L_{\subseteq \text{Int}}).$$



Beispiel 8.1 (kreisfreier Modellverband).

In Abbildung 8.2 ist ein illustrierendes Netzwerk mit zugehörigem Flussverband abgebildet. Es ist t die einzige Transportreaktion.

Es gibt zwei kreisfreie Pfade im stationären Zustand: $\{a, f, u, v\}$ und $\{b, e, u, v\}$. Der kreisfreie Modellverband enthält in diesem Falle zusätzlich das Verbandsmaximum $1_{L-C} = 1_L = \{a, b, e, f, u, v\}$ (s. Abbildung 8.2b).

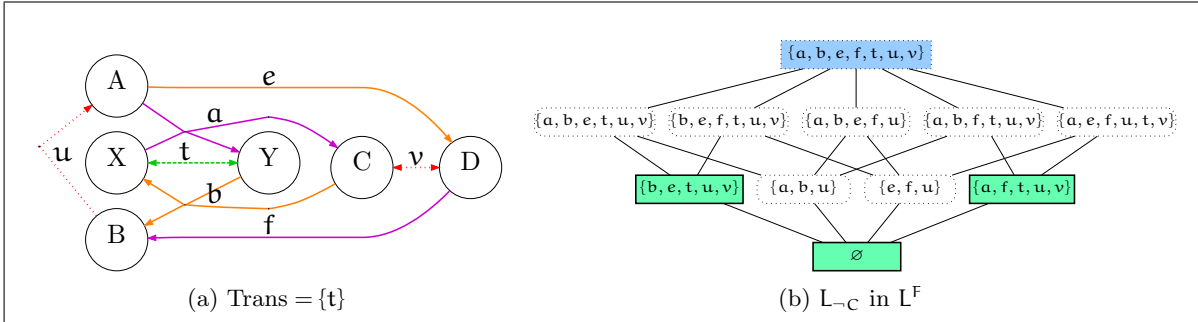


Abbildung 8.2.: Kreisfreie Pfade, Modellverband und Flussverband im Vergleich.



Bemerkung 8.4 (Flusskopplung im kreisfreien Modellverband).

Betrachten wir Netzwerk 8.2a noch einmal ein wenig genauer:

Im ursprünglichen Verband L^F gab es eine *essenzielle Reaktion* $\mathcal{ER}(L^F) = \{u\}$. Da es nur eine Transportreaktion $\text{Trans} = \{t\}$ gibt, so wird auch diese (reversible) Reaktion essenziell. Doch damit nicht genug, auch die interne Reaktion v wird nun essenziell. Im kreisfreien Modellverband gilt nun $\mathcal{ER}(L_{-C}) = \{t, u, v\}$.

Diese drei Reaktionen sind hier aneinander gekoppelt. Im Falle der echten kreisfreien Pfade sogar mit einem ganz neuen Kopplungstyp: Es gilt $v_u = |v_t| = |v_v|$.

Einen Überblick zur veränderten Flusskopplung gibt es in Abbildung 8.3.

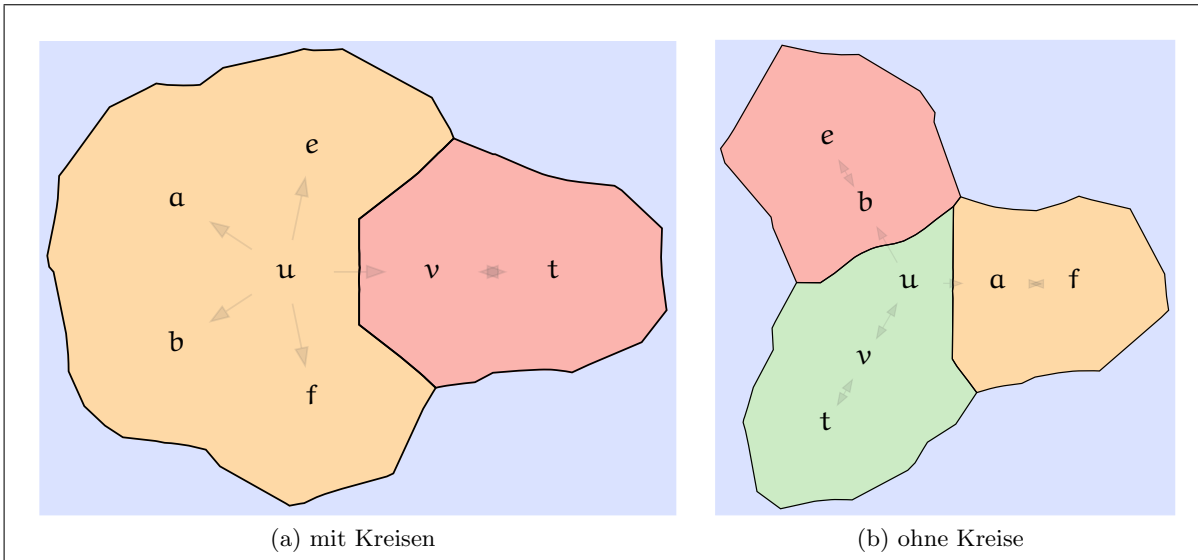


Abbildung 8.3.: Vergleich Flusskopplung mit/ohne Kreise für das Netzwerk 8.2a.

Man ist versucht, den kreisfreien Modellverband fälschlich mit einer anderen, bereits bekannten Struktur auszudrücken, was uns die Untersuchung stark vereinfachen würde. Mit Hilfe der

Zerlegung aus Satz 8.2 sehen wir jedoch:



Lemma 8.3.

Es ist $L_{-C} \subseteq L_{\parallel \text{Trans}}$ Unterverband. Im Allgemeinen liegt jedoch keine Gleichheit vor.

Gibt es in L ergänzende Summanden, dann:



$$P_{-C} = P_{-C}^{(L_{\parallel \text{Trans}})}.$$

Wir können also nicht direkt vereinfachen.

Auch wenn die beiden Strukturen nicht identisch sind, so gibt es zumindest eine große Gemeinsamkeit, nämlich die minimalen Elemente. Durch Zerlegung sehen wir:



Korollar 8.6.

Wenn es in L ergänzende Summanden gibt, dann:

$$\mathcal{M}(L_{\parallel \text{Trans}}) \subseteq P_{-C}.$$

Das liefert uns:



Satz 8.3.

Wenn es in L ergänzende Summanden gibt, dann:

$$\mathcal{M}(L_{-C}) = \mathcal{M}(L_{\parallel \text{Trans}}).$$

L ist atomar, damit nach Starbucks-Lemma auch L_{-C} . Somit:

$$\begin{aligned} \mathcal{M}(L_{-C}) = \mathcal{J}(L_{-C}) = \mathcal{M}(L_{\parallel \text{Trans}}) &\subseteq P_{-C}, \\ L_{-C} &= (L_{\parallel \text{Trans}})^\circ. \end{aligned}$$

Interessieren uns die minimalen kreisfreien Stoffwechselfade, so können wir guten Gewissens mit dem vereinfachten Modell $L_{\parallel \text{Trans}}$ arbeiten:



Korollar 8.7 (Überdeckungen der kreisfreien Stoffwechselfade).

Sei L Verband und c Überdeckung, dann ist c immer auch Überdeckung von L_{-C} .

Ganz allgemein gilt:

$$L_{-C}^\perp = L_{\parallel \text{Trans}}^\perp.$$

Wir können die Überdeckungen der kreisfreien Stoffwechselfade also bestimmen, indem wir den (mathematisch einfacheren) $L_{\parallel \text{Trans}}$ betrachten.

Überdeckung

Interessieren uns die Maxima, so können wir nicht völlig auf eine genauere Untersuchung von L_{-C} verzichten. Jedoch liefert uns $L_{\parallel \text{Trans}}$ gute obere und untere Schranken. Es gilt nämlich:

**Lemma 8.4.**

Sei (L, Rev) metabolisches System und $[n] = \text{Int} \dot{\cup} \text{Trans}$ disjunkte Zerlegung in interne Reaktionen und Transportreaktionen. Sei $R \subseteq [n]$ beliebig, dann gilt:

$$1_{(L-C)_{\perp R}} \subseteq 1_{(L_{\parallel \text{Trans}})_{\perp R}}.$$

Gibt es in L auch stets ergänzende Summanden (Lemma 3.4), dann gilt sogar:

$$1_{L_{\perp R}} \cap \text{Trans} = 1_{(L_{\parallel \text{Trans}})_{\perp R}} \cap \text{Trans} \subseteq 1_{(L-C)_{\perp R}} \subseteq 1_{(L_{\parallel \text{Trans}})_{\perp R}} \subseteq 1_{L_{\perp R}}.$$

Beweis. Es ist $L-C \subseteq L_{\parallel \text{Trans}}$ Unterverband. Lemma 3.7 liefert die erste Aussage. Das gleiche Lemma liefert uns $1_{(L_{\parallel \text{Trans}})_{\perp R}} \subseteq 1_{L_{\perp R}}$.

Sei $i \in 1_{L_{\perp R}} \cap \text{Trans}$, dann ist $1_{L_{\perp R}} \cap \text{Trans} \neq \emptyset$. Damit ist insbesondere $1_{L_{\perp R}} \in (L_{\parallel \text{Trans}})_{\perp R}$; also gilt $1_{L_{\perp R}} \leq 1_{(L_{\parallel \text{Trans}})_{\perp R}}$ und sogar Gleichheit.

Sei nun $i \in 1_{(L_{\parallel \text{Trans}})_{\perp R}} \cap \text{Trans}$. Gibt es ergänzende Summanden, greift Satz 8.2 und wir können schreiben:

$$1_{(L_{\parallel \text{Trans}})_{\perp R}} = \mathbf{a}_0 + \mathbf{c} \quad \text{mit } \mathbf{a}_0 \in P_{-C}, \quad \mathbf{c} \in L_{\subseteq \text{Int}}.$$

Wegen $i \in \text{Trans}$ und $\text{Trans} \cap \text{Int} = \emptyset$ wissen wir, dass $i \in \mathbf{a}_0$ gelten muss. Nun ist $\mathbf{a}_0 \leq 1_{(L-C)_{\perp R}}$ (Summenschreibweise) und wir sind fertig. \square

Viele Informationen über die kreisfreien Stoffwechselfade erfahren wir also bereits durch genaue Untersuchung von $L_{\parallel \text{Trans}}$ oder teilweise sogar nur L . Doch ganz kommen wir nicht darum, auch $L-C$ selbst zu betrachten. Dafür kennen wir bisher jedoch nur die Möglichkeit, alle Kreise in $L_{\subseteq \text{Int}}$ zu berechnen und auszuschließen.

Glücklicherweise ist nicht ganz so viel Aufwand nötig. Ähnlich den Überdeckungen und dem dualen Verband reicht es, die minimalen Kreise zu kennen. Es gilt:

**Lemma 8.5.**

Sei (L, Rev) metabolisches System und $[n] = \text{Int} \dot{\cup} \text{Trans}$ disjunkte Zerlegung in interne Reaktionen und Transportreaktionen. Es seien $\mathcal{M}_C := \mathcal{M}(L_{\subseteq \text{Int}})$ die minimalen Kreise in L . Dann:

$$P_{-C} = P_{-\mathcal{M}_C}.$$

Thermodynamische Grundregeln waren unsere Motivation, nur kreisfreie Stoffwechselfade zu betrachten. Bisher wurden die in der Literatur anders eingeführt. Daher wollen wir uns nun eine andere Formulierung anschauen, die unter dem Begriff *thermodynamische Constraints*

²Vielen Dank an Arne Müller für den Hinweis, dass oftmals $1_{L_{\parallel \text{Trans}}} = 1_L$ gilt.

als Einschränkung des Flusskegels bekannt ist. Beard, Babson, Curtis & Qian (2004) haben dafür eine *Potenzialvariable* μ eingeführt, mit der sichergestellt wird, dass keine Kreisprozesse stattfinden. Idee dahinter ist, dass jede tatsächlich aktive Reaktion die freie Gibbs-Energie reduziert (z. B. Beard, Liang & Qian 2002).

Bei unserem Nachweis werden wir ausnahmsweise das Gegenteil dessen tun, was wir die gesamte Arbeit über getan haben: Wir werden reversible Reaktionen aufsplitten, auch wenn dies Beard et al. in der Vorlage nicht taten. Dadurch schaffen wir es, die Theorie orientierter Matroide (Björner 1993) zu umgehen, und können stattdessen unser Wissen über Verbandstheorie direkt anwenden. Anschließend betrachten wir wieder den uneingeschränkten Fall, lassen also reversible Reaktionen zu. Wir werden sehen, dass wir auch in diesem Fall am besten mit dem kreisfreien Modellverband arbeiten sollten.

Es gilt:


Satz 8.4.

Seien (S, Irr) metabolische Bedingungen und alle Reaktionen irreversibel, also $\text{Irr} = [n] = \text{Int} \dot{\cup} \text{Trans}$.^a Es sei $F = \{v \in \mathbb{R}^n \mid Sv = 0, v \geq 0\}$ der Flusskegel. Dann gelten die folgenden Gleichungen zur Beschreibung des *kreisfreien Verbandsstumpfs*:

$$\begin{aligned}
 P_{-C} &= \{a = \text{supp } v \mid v \in F, |a \cap c| \leq |c| - 1 \forall c \in L_{\subseteq \text{Int}}^F \setminus \{0\}\} \\
 &= \{a = \text{supp } v \mid v \in F, |a \cap c| \leq |c| - 1 \forall c \in \mathcal{M}_C\} \\
 &= \{a = \text{supp } v \mid v \in F, 1_{L_{\subseteq a \cap \text{Int}}^F} = 0\} \\
 &= \left\{ a = \text{supp } v \mid v \in F, 0 = \sum_{i=1}^n c_i \forall c \geq 0 \text{ mit } Sc = 0, c_{[n] \setminus (a \cap \text{Int})} = 0 \right\} \\
 &= \left\{ a = \text{supp } v \mid v \in F, 0 = \max \sum_{i \in \text{Int}} c_i \text{ mit } c \geq 0, Sc = 0, c_{[n] \setminus (a \cap \text{Int})} = 0 \right\} \\
 &= \{a = \text{supp } v \mid v \in F, \beta^\top v_{\text{Int}} = 0, \alpha^\top S_{\cdot, \text{Int}} + \beta^\top \geq 1, \alpha \in \mathbb{R}^m, \beta \geq 0\} \\
 &= \{a = \text{supp } v \mid v \in F, (\mu^\top S_{\cdot, i}) \cdot v_i < 0 \forall i \in a \cap \text{Int}, \mu \in \mathbb{R}^m\}.
 \end{aligned}$$

^aggf. durch Aufsplittung

Beweis. Die erste Gleichung ist Lemma 8.5.

Ist ein Stoffwechselfad kreisfrei, so ist der einzige enthaltene Kreis der triviale. Somit wissen wir $a \in P_{-C} \Leftrightarrow L_{\subseteq a \cap \text{Int}} = \{0\}$. Diese Bedingung ist wiederum äquivalent dazu, dass das Maximum der triviale Pfad 0 ist, womit wir Gleichung 3 erhalten.

Wegen $\text{Irr} = [n]$ gilt $v = 0 \Leftrightarrow 1^\top v = 0$ für $v \in F$ beliebig. Insbesondere ist ein Stoffwechselfad kreisfrei, wenn ein enthaltener Kreis zwangsläufig diese Gleichung erfüllt. Gleichung 4.

Kombinieren wir die Ideen von Gleichung 3 und Gleichung 4 miteinander, so fordern wir äquivalent, dass der maximale enthaltene Kreis Komponentensumme 0 hat. Das ist Gleichung

5, die uns als Ausgang für den Rest dienen soll.

Zunächst vereinfachen wir die Schreibweise und erhalten:

$$\begin{aligned}
 P_{-C} &= \left\{ \mathbf{a} = \text{supp } \mathbf{v} \mid \mathbf{v} \in F, 0 = \max \sum_{i \in \text{Int}} c_i \text{ mit } \mathbf{c} \geq 0, S\mathbf{c} = 0, \mathbf{c}_{[n] \setminus (\mathbf{a} \cap \text{Int})} = 0 \right\} \\
 &= \left\{ \mathbf{a} = \text{supp } \mathbf{v} \mid \mathbf{v} \in F, 0 = \max \mathbf{1}^\top \mathbf{c} \text{ mit } \mathbf{c} \geq 0, S_{\cdot, \text{Int}} \mathbf{c} = 0, \mathbf{c}_{[n] \setminus \mathbf{a}} = 0 \right\} \\
 &= \left\{ \mathbf{a} = \text{supp } \mathbf{v} \mid \mathbf{v} \in F, 0 = \max \mathbf{1}^\top \mathbf{c} \text{ mit } \mathbf{c} \geq 0, S_{\cdot, \text{Int}} \mathbf{c} = 0, \mathbf{c} \leq \mathbf{v}_{\text{Int}} \right\}.
 \end{aligned}$$

Bei der mittleren Gleichung ersetzen wir die Summe durch ein Skalarprodukt. Zudem betrachten wir jetzt nur noch $\mathbf{c} \in \mathbb{R}^{\text{Int}}$ im Gegensatz zu $\mathbf{c} \in \mathbb{R}^n$. Die letzte Gleichung erhalten wir gegebenenfalls durch Skalierung von \mathbf{v} ; das geht, weil uns nur der Träger interessiert.

Die Beschreibung, die uns vorliegt, enthält ein Optimierungsproblem. Für gegebenes (festes) \mathbf{v} soll \mathbf{c} eine gewisse Funktion maximieren. Das enthaltene Maximierungsproblem lässt sich mittels linearer Optimierung lösen. Dieses primäre Problem lässt sich zu einem dualen umformulieren und wir erhalten:

$$\begin{aligned}
 P_{-C} &= \left\{ \mathbf{a} = \text{supp } \mathbf{v} \mid \mathbf{v} \in F, 0 = \max \mathbf{1}^\top \mathbf{c} \text{ mit } \mathbf{c} \geq 0, S_{\cdot, \text{Int}} \mathbf{c} = 0, \mathbf{c} \leq \mathbf{v}_{\text{Int}} \right\} \\
 &= \left\{ \mathbf{a} = \text{supp } \mathbf{v} \mid \mathbf{v} \in F, 0 = \min \alpha^\top \mathbf{0} + \beta^\top \mathbf{v}_{\text{Int}} \text{ mit } \alpha \in \mathbb{R}^m, \beta \geq 0, \alpha^\top S_{\cdot, \text{Int}} + \beta^\top \geq \mathbf{1} \right\}.
 \end{aligned}$$

Durch das Farkas-Lemma wissen wir, dass auch für die nicht-optimalen gültigen Vektoren $\mathbf{c}, \alpha, \beta$ gilt:

$$0 \leq \mathbf{1}^\top \mathbf{c} \leq \alpha^\top \mathbf{0} + \beta^\top \mathbf{v}_{\text{Int}} = \beta^\top \mathbf{v}_{\text{Int}}.$$

Finden wir also ein Paar α, β mit $0 = \beta^\top \mathbf{v}_{\text{Int}}$, so gilt Gleichheit, die Lösung ist optimal. Darum gilt Gleichung 6.

Den Rest zeigen wir separat für beide Richtungen.

Seien $\mathbf{a}, \mathbf{v}, \alpha, \beta$ wie in Gleichung 6 gegeben, $i \in \mathbf{a} \cap \text{Int}$ beliebig. Wegen $\mathbf{v}, \beta \geq 0$ und $\beta^\top \mathbf{v} = 0$ wissen wir $\beta_i = 0$. Damit ist $\alpha^\top S_{\cdot, i} \geq 1$. Wählen wir $\mu = -\alpha$ ist also $\mu^\top S_{\cdot, i} \leq -1$. Aus $\mathbf{v} \geq 0$ folgt wie gefordert $(\mu^\top S_{\cdot, i}) \cdot v_i < 0$.

Seien jetzt $\mathbf{a}, \mathbf{v}, \mu$ aus Gleichung 7 und wieder $i \in \mathbf{a} \cap \text{Int}$ beliebig. Wie zuvor wissen wir $\beta_i = 0$. Wir müssen daher diesmal ein α finden, so dass $\alpha^\top S_{\cdot, i} \geq 1$ gilt.

Wir wählen $\alpha = \frac{\mu}{\max \{\mu^\top S_{\cdot, j} \mid j \in \text{Int}\}}$. Wegen $\mu^\top S_{\cdot, i} < 0$ dreht sich das Ungleichungszeichen um und unsere Konstruktion mit Maximum im Nenner ergibt tatsächlich $\alpha^\top S_{\cdot, \text{Int}} \geq \mathbf{1}$.


Für $i \in \text{Int} \setminus \mathbf{a}$ ist $v_i = 0$. Wir wählen $\beta_i = 1$ und sind fertig. □

Wer sich nur mit kreisfreien Stoffwechselfaden beschäftigen will, hat nun also zwei unterschiedliche Möglichkeiten, sie zu beschreiben: Durch direktes Ausschließen der (zuvor berech-

neten) minimalen **Kreise** oder durch indirektes Verbieten in Form von quadratischer Optimierung oder einer **MILP**-Formulierung.

Die MILP-Formulierung ist bereits in der Literatur bekannt, genauso wie die durch Ausschluss von Kreisen (Einsatz sogenannter kombinatorischer *Benders Cuts*) (Codato & Fischetti 2006, Müller 2012). Hier nun haben wir eine in sich abgeschlossene und stichhaltige Motivation, Formulierung und Begründung gefunden, die sich bis zum Schluss ausschließlich mit verbandstheoretischen Ausdrücken beschreiben und durchführen lässt. Ganz nebenbei sind noch einige neue Erkenntnisse über den Zusammenhang mit $L_{\parallel \text{Trans}}$ und insbesondere die Möglichkeit der **FCA** und Target Prediction wie in den Problemen 7.1 und 7.2 abgefallen.

Lassen wir jetzt reversible Reaktionen wieder zu. An dieser Stelle sei auf Müller (2012) verwiesen, der einen ausführlicheren Beweis dafür erbringt, warum die gleich vorgestellte Menge \mathcal{T} wirklich den thermodynamisch korrekten Stoffwechselfaden entspricht. Wir weichen in der Notation leicht ab und verwenden die Signums-Funktion sign , um nicht in die von Müller genutzten orientierte Matroide einführen zu müssen.




Definition 8.5 (thermodynamisch gültige Stoffwechselfade).
 Seien (S, Irr) metabolische Bedingungen mit Flusskegel F und $[n] = \text{Int} \dot{\cup} \text{Trans}$ disjunkte Zerlegung in interne Reaktionen und Transportreaktionen. Es sei L^F der Flussverband und $L_{\subseteq \text{Int}}$ die Menge aller Stoffwechselkreise.
 Definiere nun $\mathcal{T} \subseteq L^F$ wie folgt:

$$\mathcal{T} := \left\{ \mathbf{a} \in L^F \mid \forall_{\substack{\mathbf{c} \in L_{\subseteq \text{Int}} \\ \mathbf{c} \neq 0}} \exists_{\mathbf{v}, \mathbf{y} \in F} \exists_{\mathbf{i}(\pm) \in \mathbf{c}} \mathbf{a} = \text{supp } \mathbf{v}, \mathbf{c} = \text{supp } \mathbf{y}, \text{sign } v_{\mathbf{i}(\pm)} = \pm \text{sign } y_{\mathbf{i}(\pm)} \right\}.$$

Es ist \mathcal{T} die Menge der thermodynamisch gültigen Stoffwechselfade.

 \mathcal{T}

Diese Definition benötigt Flussvektoren, um zu entscheiden, ob ein Pfad thermodynamisch gültig ist. Wir müssen sicherstellen, dass kein Kreis enthalten ist, bei dem alle Reaktionsrichtungen mit unserem Pfad übereinstimmen. Damit entfernen wir uns von der Leichtigkeit, nur binäre Eigenschaften kennen zu müssen. Folgender Satz hilft uns zurück zu Verbänden:



Satz 8.5 (\mathcal{T} wird kreisfrei erzeugt).
 Sei \mathcal{T} die Menge der thermodynamischen gültigen, P_{-C} die der kreisfreien Pfade.
 Dann ist $P_{-C} \subseteq \mathcal{T} \subseteq L_{-C}$. Damit ist insbesondere $L_{-C} = \langle \mathcal{T} \rangle$ und daher:

$$\begin{aligned} \mathcal{J}(\langle \mathcal{T} \rangle) &= \mathcal{M}(\langle \mathcal{T} \rangle) &= \mathcal{M}(L_{-C}) \\ &= \mathcal{M}(L_{\parallel \text{Trans}}) &= \mathcal{M}(L^F) \cap P_{-C}. \end{aligned}$$

Zudem sind Reaktionen in thermodynamisch gültigen Stoffwechselfaden genau dann gekoppelt, wenn sie es im kreisfreien Modellverband sind:

$$i \xrightarrow{=0} j \text{ (in } \mathcal{T}) \Leftrightarrow i \xrightarrow{=0} j \text{ (in } L_{-C}).$$

 $\langle \mathcal{T} \rangle = L_{-C}^F$

Beweis. Diesen äußerst hilfreichen Satz zeigen wir durch eine Kaskade vorangegangener Ergebnisse. Er beweist sich damit fast von alleine.

Fangen wir an mit $P_{-C} \subseteq \mathcal{T}$. Das ist klar, da nach Definition von \mathcal{T} nur Zusatzbedingungen an Stoffwechselfade gestellt werden, die einen Kreis enthalten. Damit ist $L_{-C} = \langle P_{-C} \rangle \subseteq \langle \mathcal{T} \rangle$.

Sei nun also c Kreis mit $0 < c < a \in \mathcal{T}$. Nach Definition von \mathcal{T} gibt es zwei Reaktionen $i^{(\pm)}$, mit denen wir Lemma 4.9 wiederbeleben können. Wir erhalten eine Zerlegung in kreisfreie Elemente, die damit automatisch auch thermodynamisch gültig sind. Also ist $a \notin \mathcal{J}(\langle \mathcal{T} \rangle)$. Daraus folgt $\mathcal{J}(\langle \mathcal{T} \rangle) \subseteq P_{-C}$ und somit $\langle \mathcal{T} \rangle \subseteq \langle P_{-C} \rangle = L_{-C}$. Die induzierten Verbände stimmen überein und wir sind quasi fertig.

Damit stimmen nämlich die unzerlegbaren Elemente überein, $\mathcal{J}(\langle \mathcal{T} \rangle) = \mathcal{J}(L_{-C})$. Die weiteren Gleichungen haben wir bereits in Satz 8.3 diskutiert; die Schreibweise als Mengenschnitt folgt stammt aus Korollar 8.5 oder alternativ dem Starbucks-Lemma, da die kreisfreien Stoffwechselfade Ordnungsideal von L^F sind.

Die Aussage über gekoppelte Reaktionen stammt schließlich direkt aus dem Starbucks-Lemma. \square



Kapitel 8 – Kurz und knapp

Starbucks-Prinzip

- ▷ Wir können das Modell um Eigenschaften erweitern, ohne neue Verfahren entwickeln zu müssen. Wir erhalten ein Modell *nach unserem Geschmack*.
- ▷ Das Starbucks-Lemma untersucht die minimalen und unzerlegbaren Elemente von *Ordnungsidealen*.
- ▷ FCA ist nun für *beliebige* Modelle möglich.
- ▷ Hauptideale sind Ideale und damit Ordnungsideale.

Kreisfreie Pfade

- ▷ Kreise in Stoffwechselfaden sind thermodynamisch unsinnig, da sie nutzlos Energie verbrauchen.
- ▷ Wir können sie dank Starbucks-Lemma mit Kombinationenverbot ausschließen und unsere Algorithmen weiternutzen.
- ▷ Die Relaxation $L_{\parallel \text{Trans}}$ erlaubt schnellere Laufzeiten und löst einige Probleme vollständig.

Und das zum Abschluss soll als Beweis dafür reichen, dass Verbandstheorie Stoffwechselanalyse durch seine Verallgemeinerungen bereichert.

ANHANG A

So haben wir's gemacht

Kapitelübersicht

A.1	Qual der Wahl	147
A.2	Blaupausen	149
A.3	Verbesserungen und Feintuning	153

Die Theorie zu kennen, ist schön. Doch das schönste Theorem bringt wenig, wenn es in der Praxis nicht funktioniert. Es muss in beidem klappen. In der Vorstellung, aber auch in der Wirklichkeit.

In dieser Arbeit haben wir ein Konzept kennengelernt, mit dem sich diskrete Stoffwechselmodelle vereinheitlichen lassen. Wir haben an Werkzeugen gearbeitet, mit denen sich Zellprozesse vorhersagen lassen. Verbandstheorie als Algebra des Lebens. Nun wollen wir einen kleinen Blick dahin wagen, ob das klappt und wie sich die vorgestellten Verfahren schlagen.

§A.1 Qual der Wahl

§A.1.1 Von Programmiersprachen und Optimierungstools

Wir haben unsere Algorithmen in **C#** implementiert. Diese Kernsprache der **.NET**-Umgebung erlaubte es uns, von Grunde auf objektorientiert zu programmieren und so künftigen Erweiterungen eine gute Integrierbarkeit zu ermöglichen.

Diese Erweiterungen – oder auch Verbesserungen bestehenden Codes – müssen nicht zwangsläufig in **C#** erfolgen, sondern können sich dem gesamten Repertoire der **.NET**-Sprachen bedienen. So können Interessierte ihren optimierten **C++**-Code beitragen, andere mit **F#** funktional programmieren.

Zentrale Frage war zudem die nach einem **LP**-Solver. Hierbei haben wir uns für den vergleichsweise jungen Gurobi (Gurobi Optimization Inc 2012) entschieden.

Gurobi-Modelle sind leicht in viele Programmiersprachen zu inkludieren, auf der Herstellerseite stehen viele ausführlich dokumentierte Beispiele bereit. Entsprechend einfach fiel es uns auch, Gurobi in unser Programm einzubinden.

Vor allem ist Gurobi einer der schnellsten LP- und insbesondere MILP-Solver auf dem Markt. Hinzu kommt, dass das Programm beständig weiterentwickelt wird – so häufig sogar, dass wir mitten in den Berechnungen der folgend diskutierten Simulationen vom Update auf die Version 5.0.0 überrascht wurden. Um miteinander vergleichbare Ergebnisse zu produzieren, haben wir daher alle Berechnungen basierend auf der neuen Version wiederholt.

Gurobi kann kostenlos zu akademischen Zwecken verwendet werden.

§A.1.2 Unterstützte Formate

Unsere Implementierung kann Dateien im Metatool-Format (Kamp & Schuster 2006) sowie in einer Ordnerstruktur einlesen.

Bei Metatool handelt es sich um Textdateien, die sehr der gewöhnlichen Notation von chemischen Reaktionen entsprechen. Nach einer Auflistung der internen/externen Metaboliten und der reversiblen/irreversiblen Reaktionen folgen die Reaktionsgleichungen in Klartext, etwa

`02trans: 02ext = 02.`

Dieses Format erlaubt es dem Nutzer, sehr leicht Änderungen im Eingabnetzwerk vorzunehmen oder sich einfach nur einen Einblick in das Netzwerk zu verschaffen. Auch die Konstruktion kleiner Beispiele ist sehr einfach, da auf die manuelle Erstellung einer stöchiometrischen Matrix verzichtet werden kann.

Alternativ können Ordner eingelesen werden, in denen folgende Informationen (etwa aus SBML-Dateien (Hucka, Finney, Sauro, Bolouri, Doyle, Kitano, and the rest of the SBML Forum.; Arkin, Bornstein, Bray, Cornish-Bowden, Cuellar, Dronov, Gilles, Ginkel, Gor, Goryanin, Hedley, Hodgman, Hofmeyr, Hunter, Juty, Kasberger, Kremling, Kummer, Le Novere, Loew, Lucio, Mendes, Minch, Mjolsness, Nakayama, Nelson, Nielsen, Sakurada, Schaff, Shapiro, Shimizu, Spence, Stelling, Takahashi, Tomita, Wagner & Wang 2003)) abgelegt sind: stöchiometrische Matrix, Reaktionsgeschwindigkeitsober- und untergrenzen, Reaktions- und Metabolitennamen, ggf. berechnete interne Kreise. Aus den unteren Intervallgrenzen werden die reversiblen Reaktionen abgeleitet. Die Reaktionen, die nur einen einzigen Metaboliten produzieren oder konsumieren (und zusätzlich keine Metaboliten konsumieren bzw. produzieren) wurden als die Transportreaktionen definiert.

§A.2 Blaupausen

Diese Arbeit ist durchzogen von dem Gedanken, sich nicht zu sehr auf die Feinheiten eines endgültigen Modellierungswegs, geschweige denn einer Implementierung für eine bestimmte biologische Vorlage zu konzentrieren.

Stattdessen untersuchen wir endliche Verbände, die sich vielseitig nutzen lassen, in unserem Falle diverse biologische Modellierungsansätze zulassen. Sei es Wunsch nach ausschließlich stationären Zuständen im Flussverband L^F oder logische Schlüssigkeit in L^{01} , sei es mit zusätzlichen thermodynamischen Bedingungen, die Kreise verbieten, L_{-C} . Seien es andere, völlig neuartige Wege. Die allermeisten Aussagen beziehen sich auf keinen konkreten Verband; wir sagen nur, dass es ein Verband $L \subseteq 2^{[n]}$ sein soll mit $[n]$ die Menge der Reaktionen. Darauf beziehen sich unsere Algorithmen. So ist unsere Implementierung konstruiert.

Wir liefern die Schnittstellen, die sich einer reinen verbandstheoretischen Sprache bedienen. Umgesetzt haben wir L^F und L_{-C}^F – doch unsere Schnittstellen erlauben einfache Portierung oder Ergänzung um andere Modellkonzepte. Hauptsache, alles ist Verband.

Schauen wir uns an, was heute schon umgesetzt ist.

§A.2.1 Von Kopplungen, Paaren und Maxima – FCA

Wir haben zwei Wege kennengelernt, FCA, Flusskopplungsanalyse, zu betreiben. Einen direkten, der von der ursprünglichen Definition von $i \xrightarrow{=0} j$ abgeleitet war und Reaktionenpaare in den Fokus stellt, und einen zweiten, der durch Verbandstheorie motiviert Maxima $1_{L_{\perp\{i\}}}$ bestimmt und aus diesen \mathcal{C} ableitet.

Unsere Implementierung folgt letzterem Ansatz. Einem hoch entwickelten Algorithmus, der viele bekannten Techniken und mehr noch umsetzt, einfach zu formulieren und daher zu warten ist – Algorithmus *FastMax*. Wir optimieren die Berechnung der Maxima und setzen darauf unseren Fokus. Anschließend kann man frei wählen, ob man mit den Maxima $1_{L_{\perp\{i\}}}$ als Ganzes oder doch mit abgeleiteten Reaktionenpaaren als sogenanntes *Couple* arbeiten will. Auch an dieser Stelle wollen wir die Wahlfreiheit des Nutzers nicht einschränken. Wir liefern eine fertige Infrastruktur, die vom interessierten Anwender, vom geneigten Leser auf Wunsch in Augenschein, ergänzt oder abgewandelt werden kann.

Die Berechnung der Maxima erfolgt in speziellen *Solver*-Klassen, die letztlich die Modellzugehörigkeit festlegen. Die Entscheidung, ob stationäre Zustände verlangt werden oder nicht, wird im Hauptprogramm einmal getroffen und hier schließlich umgesetzt. Ob diese Umsetzung lediglich dem einfachen Algorithmus *SimpleMax* oder dem ausgeklügelteren *FastMax* folgen soll, ob man es vielleicht sogar noch ganz anders schafft, wie bei unserem trickreichen 1-Schritt-Verfahren *CalculateBlock*, kann hier frei entschieden werden.

Unsere Verbandskonzepte sind der Rahmen, für den wir Mosaiksteine liefern, die sich zu einem

Bild zusammensetzen. Unsere Implementierung ist so ein Bild. Ein Bild, das jederzeit durch neue oder veränderte Steine nach eigenen Vorlieben verschönert werden kann, ohne das Bild von ganz von vorne anfangen zu müssen.

Klassische FCA beschäftigt sich mit dem Flusskegel F und Flusskopplungen im stationären Zustand. Darum dreht sich deshalb auch unsere erste Implementierung.

Unser erster Solver berechnet mit Hilfe von Gurobi Elemente, Zeugen $\mathbf{a} \in L_{\perp R}^F$, und setzt sie gemäß *FastMax* zu Maxima zusammen.

Damit haben wir alle Steinchen für das Bild zusammen: Der Nutzer wählt das Eingabeformat, für das wir mehrere Möglichkeiten implementiert haben. Das Hauptprogramm gibt den Auftrag zur FCA und verwaltet alle anfallenden Daten, speichert sie gegebenenfalls statistisch untersuchbar ab. In der nächsten Stufe wird die FCA durchgeführt, nachdem 1_L und F_{rev} berechnet sind. Die bis zu n Maxima $1_{L_{\perp(i)}}$ werden gefunden und die Zeugen verwaltet. Unser Solver findet die Maxima mittels Intervallschachtelung $\text{lower_bound} \leq 1_{L_{\perp R}} \leq \text{upper_bound}$, einzelne Aufträge zur Zeugenfindung werden dafür an Gurobi weitergeleitet. Der Boden unserer Kaskade ist erreicht.

Die Ergebnisse wandern dann von unten nach oben, werden in jeder Stufe angemessen verarbeitet und zusammengefügt. Jede Zwischenschicht kann ausgetauscht oder optimiert werden, ohne die anderen anfassen zu müssen. Sei es Zeugenverwaltung, die in den mittleren Stufen stattfindet, oder doch der konkrete Modellierungsansatz, der einmal im Hauptprogramm festgelegt und dann bis zum Boden durchgereicht wird, ohne Einfluss auf die Prozessabläufe in den inneren Ebenen zu haben.

Bei uns kann alles isoliert voneinander betrachtet werden. Wir können viele Schritte bewusst ausblenden, um konzentriert an lokalen Problemen zu arbeiten. Ein Blick, der Arbeit effektiver macht. Den uns die Verbandstheorie ermöglicht.

§A.2.2 Thermodynamische Nebenbedingungen – tFCA

Als Anwendung des Starbucks-Lemmas haben wir es vorgestellt: Man kann Flusskopplung auch im Verbandsstumpf definieren, bei dem wir aus Gründen der Thermodynamik interne Kreise verbieten. Wir können eine FCA durchführen, die für einige „natürlicher“ wirken dürfte. Weil sie mehr thermodynamischen Regeln folgt, nennen wir sie *tFCA*.

Weil es sich um eine Unterverbandsstruktur handelt, können wir obere Schranken für Maxima im schwerer zu untersuchenden kreisfreien Modellverband $L_{\perp C}$ leicht durch den Ursprungsverband L^F erhalten. Das nutzen wir in unserer Implementierung.

Es gibt mehrere Möglichkeiten, einen garantiert kreisfreien Zeugen zu produzieren; das haben wir ausführlich diskutiert. Wir haben uns hier für die simpelste Methode entschieden: Wir berechnen die minimalen internen Kreise und bauen damit ein MILP. Zeile 2 von Satz 8.4 erlaubt es uns sogar, mit weniger Integer-Variablen als Algorithmus *CalculateMinimalElements* zu arbeiten. Arne Müller schlug folgendes MILP vor. Es funktioniert, da wir nur die Bedingung $\mathbf{a}_j = 0 \Rightarrow \mathbf{v}_j = 0$ benötigen:

$$\begin{aligned}
 & \text{minimize} && 0 \\
 & \text{s.t.} && \mathbf{S} \cdot \mathbf{v} = 0 \\
 & && v_j = 0 && \forall j \in \text{Irr} \\
 & && -M \cdot \mathbf{a}_j \leq v_j \leq M \cdot \mathbf{a}_j && \forall j \in [n] \\
 & && \mathbf{a}_j \in \{0, 1\} && \forall j \in [n] \\
 & && \sum_{j \in c} \mathbf{a}_j \leq |c| - 1 && \forall c \in \mathcal{M}_C \\
 & && v_i \geq 1^1 \\
 & && v_j = 0 && \forall j \in R
 \end{aligned}$$

Unser Wissen aus der Verbandstheorie erlaubt es uns in Kombination mit objektorientierter Programmierung, eine einzige Klasse zu erstellen, die uns statt Maxima $1_{L \perp R}$ die Maxima $1_{(L-C) \perp R}$ liefert. Wir tauschen ein Objekt aus, um auf eine andere Test (i, R) -Funktion zurückgreifen zu können. Mehr muss man von außen nicht wissen.

Der neue Solver nutzt massiv unser Wissen über die Verbände L^F und $L-C$. Er schafft es über einen Trick, der simpler nicht sein könnte: Er verfügt über einen eigenen Solver, der das gewöhnliche Maximum mit gewöhnlichen LPs bestimmt, und setzt das Ergebnis als `upper_bound`. Dieser interne Solver hat dafür Elemente aus L^F bestimmt und als Zeugen gespeichert. Unser neuer Solver filtert aus dieser Zeugenliste all diejenigen heraus, die Kreise enthalten. Was verbleibt wird zur `lower_bound` aufaddiert. Damit landen wir bei einem Intervall `lower_bound` $\leq 1_{(L-C) \perp R} \leq$ `upper_bound`, das in den allermeisten Fällen schon sehr klein ausfällt.

Nur für diese Diskrepanz zwischen unterer und oberer Schranke – sofern es überhaupt eine gibt – müssen MILPs gelöst werden. Ein geschicktes Zeugenmanagement erlaubt es uns so, in vielen, vielen Fällen schwere MILPs zu vermeiden und stattdessen lediglich LPs zu lösen.

§A.2.3 Doppel-Reaktionen-Knockout – EFCA

Um alle paarweisen Reaktionen-Knockouts zu simulieren, muss man eigentlich nur eine Sache tun: Iteriere über alle Reaktionen, blockiere dabei jeweils eine und führe für das Restnetzwerk eine FCA durch. Schließlich ist $L_{\perp\{i,j\}}$ nichts anderes als $(L_{\perp\{i\}})_{\perp\{j\}}$.

Mit dieser Idee im Hinterkopf ist unsere EFCA implementiert: Iteriere über $i \in [n]$ und berechne jeweils die FCA für $L_{\perp\{i\}}$. Bekannte Elemente/Zeugen aus vorangegangenen Schritten können wir übernehmen, müssen aber nicht. Dazu später mehr.

¹oder $v_i \leq -1$.

Doch vor dieser *EFCA*-Phase führen wir stets eine gewöhnliche FCA durch. Allein schon deswegen, weil wir damit in unserer schnellen Maximumberechnung *FastMax* alle gekoppelten Reaktionen ausschalten können, wenn wir eine Reaktion als blockiert identifiziert haben. Wollen wir bis zu $\binom{n}{2}$ Maxima berechnen, so sparen wir uns dadurch jede Menge (MI)LPs.

Wie schon bei der FCA erbt die *EFCA*-Phase einen Maximumberechner, auf den wir über ein Interface zugreifen. So bleibt die Wahl, eine „gewöhnliche“ EFCA durchzuführen, oder doch ein kleines t davorzuhängen und in der tEFCA zusätzlich die thermodynamischen Nebenbedingungen zu respektieren, interne Kreise konsequent auszuschließen.

Dies hier ist der Punkt in unserer Implementierung, wo klarer denn je die Hauptbotschaft dieser Arbeit herausstrahlt! Verbandstheorie erlaubt Modularisierung, Normierung, Vereinheitlichung. Nicht jeder Ansatz braucht sein eigenes Programm, sondern wir können je nach Forschungsfrage Module zusammenstecken. Wir wissen, dass dies funktioniert und dass die Optimierungen der Teilmodule auch in der Komposition klappen werden.

Müssen wir für jedes $i \in [n]$ eine vollständige FCA durchführen? Nein, das müssen wir natürlich nicht. Für $j < i$ müssen wir nicht mehr $1_{(L_{\perp\{i\}})_{\perp\{j\}}}$ berechnen, da wir den Wert zuvor schon berechnet haben, als wir die FCA für $L_{\perp\{j\}}$ durchgeführt haben. Ähnliches gilt für $i = j$; dieses Maximum kennen wir schon, da wir vor der *EFCA*-Phase erst eine gewöhnliche FCA durchführen.

Aus diesem Grund informieren wir den Konstruktor unseres neuen FCA-Objekts nicht nur darüber, dass er die Reaktion i blockieren soll, sondern auch, dass er nur die Maxima für $j > i$ berechnen muss.

Laufzeit ist nicht die einzige Währung, in der man in der Algorithmenwelt bezahlt. Jedes Programm, das man ausführt, hat auch einen Speicherbedarf. Sei es Arbeitsspeicher im RAM, auf den man schnell zugreifen kann, oder heutzutage gewaltiger Festplattenspeicherplatz, den man aber nur um Größenordnungen langsamer verwalten kann.

Arbeitsspeicher wird zu einem Problem, wenn man $\binom{n}{2}$ Objekte mit $\Theta(n)$ Platzbedarf untersuchen will. An einer Laufzeit von $\Omega(n^3)$ kommen wir nicht vorbei. Beim Speicherhunger dürfen wir uns so etwas nicht erlauben, wenn wir tausende Reaktionen untersuchen.

Darum geht unsere Implementierung die Herausforderung sehr pragmatisch an: Wir versuchen gar nicht erst, alle Maxima vorrätig zu halten. Wir berechnen sie, verwerfen sie aber gleich nach jeder Iteration wieder. Uns interessiert momentan nur, wie viele Reaktionen noch möglich sind, wie viele blockiert.

Dieser Ansatz erlaubt es uns, einen guten Überblick darauf zu gewinnen, welche beiden Reaktionen zusammen einen fundamentalen Einfluss auf den Stoffwechsel haben, wenn sie gemeinsam einen Großteil der Reaktionen an sich koppeln.

Zudem haben interessierte Forscher die Möglichkeit, den Quelltext leicht zu modifizieren, einzelne oder alle Maxima abzugreifen, auf eigene Verantwortung zu speichern, bevor sie in der nächsten Iteration verworfen werden.

§A.3 Verbesserungen und Feintuning

Algorithmenentwurf ist ein Geben und Nehmen: Aus einem Gedankenblitz oder gründlicher Überlegung wird eine rohe Skizze auf dem Papier. Man testet sein Verfahren an einigen kleinen Beispielen, findet Logiklücken in der bisherigen Schablone und vervollständigt damit den Ablauf des Algorithmus.

Ist man gegen eine in sich schlüssige Blaupause konvergiert, wird implementiert. Die Suche nach geeigneten Datenstrukturen startet, vielleicht optimiert man seinen Schleifeniterator, um ein `for all a ∈ A \ B` zu optimieren. In der Wunschvorstellung Minuten, in der Realität Stunden, eher Tage vergehen. Danach beginnt die Fehlersuche, *Debugging*.

Algorithmenentwurf ist ein Geben und Nehmen. Man übersetzt einen Arbeitsablauf in eine maschinenlesbare Sprache und wird durch Programmabstürze und kryptische Fehlermeldungen entlohnt. Neben viel Frust widerfahren einem auch viele euphorische Befreiungsmomente. Ein ewiger Hin-und-Her zwischen selbstironischem Wutausbruch wegen verflixten *copy- & paste-Fehlern* und Erleuchtung beim Finden von Angriffs-lücken im Verfahren, die man auf dem Papier nicht gefunden hatte.

Läuft das Programm endlich stabil, greift womöglich der Ehrgeiz zur Optimierung – sei es zur Geschwindigkeit des Verfahrens, oder „nur“ Usability und ein reichhaltiges, einfach bedienbares Angebot an Einstellungen oder Parametern. Mit jeder Idee die kommt und man umzusetzen versucht, zieht man in die nächste Schlacht gegen den Compiler; und seine eigene fehlerlastige Unüberlegtheit.

So auch hier. Die erhofften Minuten wurden zu Stunden, Tagen, mit allem Feintuning schließlich zu Wochen und Monaten. Viel Frust, aber auch viele Ideen für kleine Details kamen heraus, die schließlich zu einem schnelleren oder einfacheren Programm geführt haben.

Hier ein kleines Best-of, damit die Erkenntnisse nicht ganz verloren gehen:

Zeugenfilter Berechnen wir einen neuen Zeugen $a \neq 0$, so wissen wir, dass er definitiv nicht in unserer Zeugenliste steht, wie schon in der Fußnote auf Seite 114 erläutert.

Da wir mit Sicherheit wissen, keine Duplikate zu erstellen, ist ein Filter kein Grund für eine ausgefeilte Datenstruktur zum Speichern der Zeugen.

Vorfilter Es programmiert sich einfacher, wenn eine schlichte Datenstruktur ausreicht. Die so erstellten Programme laufen aber dennoch schneller, wenn man intelligent mit den vorhandenen Ressourcen umgeht.

Unsere EFCA führt nacheinander eine abgekürzte FCA für $i \in 1_L$ durch. Jeder dieser Unterprozesse kann davon profitieren, wenn wir bekannte Elemente als Zeugen weiter-

geben. Um Zeit zu sparen, filtern wir vor: Wir übergeben nie ganz \mathcal{W} sondern $\mathcal{W}_{\perp\{i\}}$. Dadurch beschleunigen wir das Preprocessing in *FastMax*.

teilweise Kopplungen Wir wollen uns an unseren eigenen Rat halten und nutzen aus, dass Flusskopplung zu kennen praktisch ist. Deshalb verwenden wir in Algorithmus *FastMax* alle bekannten gekoppelten Reaktionen so früh wie möglich und beschleunigen damit die Berechnung von Flusskopplung selbst.

Aber da hört es noch nicht auf. Dort verwenden wir Ergebnisse aus der FCA immer nur punktuell, um die Berechnung von Maxima zu beschleunigen. Während der *EFCA*-Phase können wir noch einiges draufsatteln. Zur Berechnung einzelner Maxima verwenden wir den gleichen Trick weiter und vermeiden so LP-Solver-Aufrufe. Und manchmal – wenn nämlich $i \longleftrightarrow j$ teilweise aneinander gekoppelt sind – vermeiden wir nicht nur die Berechnung einzelner Maxima, sondern ganzer Teil-FCAs! Wir wissen, dass dann $L_{\perp\{i\}} = L_{\perp\{j\}}$ gilt und können die gesamte Iteration überspringen. Schneller geht's kaum.

thermodynamische Relaxation Bei der t(E)FCA konnten wir zusätzlich viel Laufzeit sparen, indem wir zweimal relaxieren:

1. Wir vereinfachen das MILP auf $\alpha_i = 0 \Rightarrow v_i = 0$. Damit schaffen wir es, nur n statt $3n$ Integer-Variablen zu nutzen und ersparen uns auch das Aufsplitten der v_i .
2. Nutze $L_{-C} \subseteq L^F$ und verzichte damit auf MILPs, wo auch immer möglich. Wir lösen lieber erst LPs und schauen anschließend, welche der dortigen Lösungen auch thermodynamisch gültig sind. Nur in Fällen, wo wir Lösungen verwerfen müssen, brauchen wir wirklich ein MILP.



Kapitel A – Kurz und knapp

Implementierung

- ▷ Wir haben FCA, tFCA und EFCA umgesetzt.
- ▷ Das Programm ist objektorientiert gegliedert und in C# geschrieben. Lineare Probleme lösen wir mittels Gurobi.
- ▷ Es liest Metatool-Dateien und Ordner ein.

Prozessverlauf

- ▷ Es gibt mehrere Prozessschichten. Die mittleren verwalten, die unterste berechnet Maxima mittels Intervallschachtelung *FastMax* und Gurobi.
- ▷ Die Ergebnisse aus der FCA beschleunigen die wahlweise anschließende *EFCA*-Phase massiv.
- ▷ Die Option thermodynamischer Nebenbedingungen steht als Parameter offen, ohne den Prozessverlauf zu beeinflussen.

ANHANG B

Graph & Zahl

Kapitelübersicht

B.1	Was wir gerechnet haben	155
B.2	FCA	158
B.3	EFCA	171

§B.1 Was wir gerechnet haben

§B.1.1 Getestete metabolische Systeme

Bei der Wahl der metabolischen Netzwerke, an denen wir unsere Algorithmen getestet haben, orientierten wir uns am Usus (z.B. David et al. 2011, Larhlimi et al. 2012). Die folgend beschriebenen Berechnungen beziehen sich auf folgende sechs Systeme:

E. coli (iAF1260) (Feist, Henry, Reed, Krummenacker, Joyce, Karp, Broadbelt, Hatzimanikatis & Palsson 2007),

H. pylori (Thiele, Vo, Price & Palsson 2005),

M. barkeri (Feist, Scholten, Palsson, Brockman & Ideker 2006),

M. tuberculosis (Jamshidi & Palsson 2007),

S. aureus (Becker & Palsson 2005),

S. cerevisiae oder einfach Hefe (Duarte, Herrgård & Palsson 2004).

Anfangs wollten wir noch ein anderes Modell – die größere **E. coli**-Variante **iJ01366** (Orth, Conrad, Na, Lerman, Nam, Feist & Palsson 2011) – verwenden. Wie wir jedoch gleich erklären werden, haben wir unsere Untersuchung dieses Systems abgebrochen und verworfen.

§B.1.2 Wie man repräsentativ berechnet

Bevor wir uns gleich die Laufzeitergebnisse anschauen, wollen wir noch einmal betonen, dass wir mit den hier diskutierten Implementationen kein Wettrennen gewinnen wollten.

Diese Arbeit soll Verbandstheorie dem Anwender nahe bringen. Wir haben Modelle als Verbände ausgedrückt, Eigenschaften standardisiert, diese auf andere Modelle übertragen und neue theoretische Ergebnisse erhalten. Außerdem haben wir Algorithmen vorgestellt, die sich auf allgemeine Verbandsmodelle übertragen lassen. Hier haben wir sie nur für den Fall des Flussverbandes L^F implementiert, um so das Vorgehen klassischer FCA-Solver wie F2C2 (Larhlimi et al. 2012) nachzubilden.¹

Unser Ansatz soll einfach zu bedienen sein und trotzdem effiziente Algorithmen liefern. Mit effizient meinen wir hier, dass wir Laufzeiten in ähnlicher Größenordnung wie etablierte Programme haben – mit dem Unterschied, dass wir unser objektorientiertes Programm in Windeseile auf andere Modelle anpassen können. Modelle, die nicht mal notwendigerweise Verband sein müssen, wie wir dank dem Starbucks-Lemma wissen.

Wir sind an den Größenordnungen interessiert, nicht am letzten Prozentpunkt. Um alltags-taugliche Zahlen nennen zu können, messen wir daher keine Prozessorzeit, sondern die echt vergangene Zeit während der verschiedenen Programmphasen. Bei der FCA sind das die Berechnung des Maximums, der vollständig reversiblen Reaktionen und eben der Reaktionskopp-lungen (als Maxima der Unterverbände $L_{\perp\{i\}}$). Bei der EFCA kommt anschließend noch die Laufzeit der entsprechenden Phase hinzu.

Von der Messung ausgeschlossen haben wir das Einlesen der Daten und Speichern der Ergeb-nisse. Im Falle der thermodynamisch korrekteren tFCA haben wir auch nicht den Zeitraum zur Berechnung der inneren Kreise erfasst, die wir im Voraus berechnet haben. Wir wollen hier eine praxistaugliche FCA haben und nicht eine optimal schnelle Kreisberechnung.

Da die tatsächliche Rechenzeit auf einem Computer Einflüssen unterliegt, die zum Teil nicht kontrolliert werden können, haben wir unsere Simulationen wiederholt durchgeführt. Die FCA jeweils 11 mal, die doppelte Knockout-Simulation EFCA aus Zeitgründen 5 mal. In unseren vorgestellten Tabellen und Graphiken, die nur einen Zahlenwert zuließen, wählten wir in allen Fällen den Median einer Messgröße, um uns nicht von einzelnen Ausreißern beeinflussen zu lassen. Dort, wo es ging, haben wir jedoch in den Graphiken alle Datenpunkte dargestellt. Hier wird man sehen, wie die vergleichsweise kurze Laufzeit der FCA zum Teil starken Schwan-kungen unterlegen ist, die der EFCA nur kaum fassbare (relative) Schwankungen innehat.

§B.1.3 Computer können nicht rechnen

Die Vereinfachung des Maximum-Algorithmus als Intervallschachtelung durch Aufsummierung kombiniert mit diversen Detailverbesserungen im Arbeitsablauf haben zu einem Verfahren geführt, mit dem man die FCA in extrem kurzer Zeit durchführen kann – gemessen am Zeitaufwand, der vor wenigen Jahren noch nötig war.

¹Es sei wiederholt angemerkt, dass wir nur rein qualitative Eigenschaften betrachten; die quantitative feste Flusskopplung bestimmen wir nicht.

Damals wie heute ist das Lösen linearer Probleme Herz und Mittelpunkt des Verfahrens. Alle Optimierungen beschäftigen sich damit, diesen Kernschritt möglichst selten durchführen zu müssen. Vor allem, weil er auch mit modernen LP-Solvern zeitaufwendig ist im Vergleich zu einer gewöhnlichen Grundrechenart. Zudem ist er die Schnittstelle zwischen theoretischem Modell und echten biologischen Daten – das gilt insbesondere für unseren verbandstheoretischen Ansatz, wo nur in der Test-Funktion mit Stöchiometrien und echten Reaktionsgeschwindigkeiten gearbeitet werden muss.

Wir können heute Modelle untersuchen, die tausende von Reaktionen umfassen. Die Zeit ist fast kein Constraint mehr. Die Rechengenauigkeit von Computern ist es jedoch.

Fordern wir stationäre Zustände, also $Sv = 0$, führt das durch Rechenungenauigkeiten zu $\|Sv\| \sim \varepsilon$ mit ε klein gemessen an der Stabilität des Verfahrens. Bei einem Modell mit 2000 Reaktionen, wie `iJ01366`, kann dieses ε schnell dem 2000-fachen einer eingestellten Toleranz entsprechen, die vom Solver in der Grundeinstellung meist auf Werte in der Größenordnung 10^{-6} gewählt sind. Was nun tun bei $v_i = 0.001$? Findet die Reaktion statt oder nicht?

Dies kann zu Fehlvorhersagen führen. Eine gut begründete Fuzzy-Logik muss her, die nicht aus den Augen verliert, dass wir Verbandseigenschaften wie ergänzende Summanden (und damit Atomarität) für den stationären Flussverband nachgewiesen haben (Lemma 4.8). Wir haben in Bemerkung 4.12 schon darauf hingewiesen, dass man den Flusskegel bedachtsam auf einen Kegelstumpf einschränken kann, doch ist die Handhabung nicht ganz einfach.

Als Konsequenz haben wir entschieden, das große `E. coli`-Modell `iJ01366` aus unserer Untersuchung zu entfernen. In einer früheren Version berechneten wir die minimalen Kreise für die tFCA noch mit Algorithmus *CalculateMinimalElements*, der eine ähnliche MILP-Formulierung besitzt wie unsere tFCA. Egal welche Toleranzbereiche wir einstellten, es kam zu reproduzierbar unsinnigen Ergebnissen oder Programmabstürzen. Für die verbliebenen Elemente haben wir uns schließlich für eine Toleranz von $i \in \text{supp } v \Leftrightarrow |v_i| \geq \text{tol} = 0.1$ entschieden. Zudem haben wir die obere Schranke $M = 1000$ gewählt.

Doch selbst wenn es bei anderen Modellen zu keinen solch auffälligen Fehlern gekommen ist, möchten wir hier den Anwender dazu auffordern, Simulationsergebnisse stets mit diesem Bewusstsein zu untersuchen: Das hier geschilderte Problem ist keines unseres Ansatzes speziell, sondern systemimmanent, sobald man irgendeine Art mathematischer Computerberechnung² durchführen will.

²auf indiskreten Realräumen

§B.2 FCA

§B.2.1 Laufzeiteigenschaften

Wagen wir zuerst einen Blick auf die verschiedenen Abschnitte der FCA, also (1) die Berechnung der blockierten Reaktionen gefolgt von (2) den vollständig reversiblen Reaktionen, um uns schließlich (3) den eigentlichen Kopplungen zu widmen.

Es ist zu erwarten, dass die Berechnung von \mathcal{F}_{rev} ähnlich anspruchsvoll oder ein wenig einfacher ist als die von \mathcal{B} , da in beiden Fällen lediglich ein (Unter-)Verbandsmaximum berechnet werden muss. \mathcal{C} hingegen sollte sich als wesentlich schwerer erweisen, da hierfür n Maxima bestimmt werden müssen.

Um dies zu überprüfen wählen wir zunächst eine Darstellung, bei der alle absoluten Zahlen wegfallen – dies erlaubt uns einen besseren Blick auf die Verhältnisse. Tatsächlich bestätigt Abbildung B.1 unsere Vermutungen.

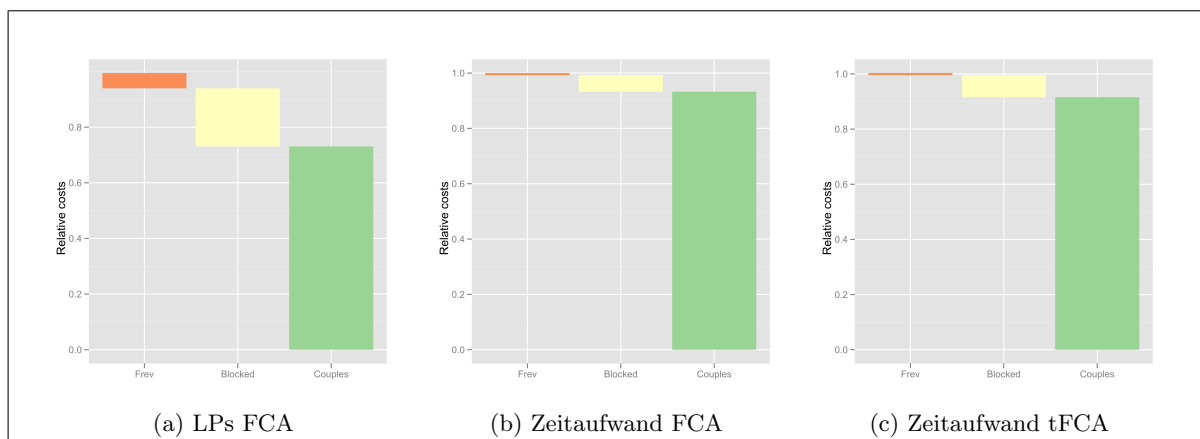


Abbildung B.1.: Relativer Aufwand der verschiedenen Etappen der FCA.

In allen drei Unterabbildungen – gelöste LPs, Zeitaufwand ohne/mit Kreisausschluss – beansprucht die Berechnung der Kopplungen den Löwenanteil an Ressourcen. \mathcal{F}_{rev} lässt sich am einfachsten berechnen. Interessant ist ein direkter Vergleich von B.1a mit B.1b. Denn hier offenbart sich, dass sich die Laufzeit nicht linear aus der Anzahl zu lösender LPs bestimmen lässt.

Das Verhältnis gelöster LPs zwischen *Blocked* und *Couples* lässt sich mit etwa 1:4 schätzen. Betrachtet man nun jedoch das gleiche Verhältnis im Zeitaufwand, so bietet sich 1:10 als konservative Schätzung an. Es vergeht also mehr Zeit in der *Couples* Phase von einer LP-Berechnung bis zur nächsten, als dies in der Startphase der Fall ist.

Eine mögliche Erklärung hierfür ist die vorbereitende Phase, in der vor allem die untere Schranke bestimmt wird. Hierfür betrachten wir alle bereits berechneten Zeugen in $|\mathcal{W}| \cdot \Omega(1)$

und vereinen alle geeigneten Verbandselemente mit einem Zeitaufwand von $|\mathcal{W}_{\perp\{i\}}| \cdot \Theta(n)$ in unserer Implementierung. Dieser Zeitaufwand steigt mit wachsender Anzahl an bekannten Elemente über den Punkt hinaus, wo er nicht mehr zu verachten ist im Vergleich zum (eigentlich sehr komplexen) Lösen von LPs.

Wir müssen weniger LPs lösen – wie wir gleich noch genauer sehen werden –, was wir uns jedoch durch einen sich aufblasenden Verwaltungsapparat erkaufte haben. Dies ist sicherlich noch verbesserbar.

Verzichtbar ist es indes nicht: Unser Verfahren schafft ein Verhältnis von weniger als 1:10 bei Modellen, die 500 bis 2000 Reaktionen umfassen. Bei trivialen Vorgehen ohne Zeugen gäbe es deswegen ein Verhältnis von 1:500 bis 1:2000 – zu massiven Kosten der Laufzeit.

Dieser Unterschied in den Verhältnissen zeigt auf gewisse Weise aber auch, wie effizient Gurobi dabei ist, LPs zu lösen. Schauen wir nämlich auf die Abbildung [B.1b](#), so sehen wir, dass sich das Verhältnis minimal verschoben hat. Grund dafür könnte sein, dass wir diesmal nicht nur LPs, sondern auch die wesentlich komplexeren MILPs lösen mussten.

Die dargestellten Werte sind die relativen Mediane der Werte. Das soll heißen, dass wir für jedes Modell bestimmt haben, welchen prozentualen Anteil LP-Anzahl bzw. Laufzeit ausgemacht haben. Aus den Werten der unterschiedlichen Modelle haben wir schließlich den Median dargestellt. Das erklärt auch, warum sich die einzelnen Blöcke nicht auf exakt 1 aufaddieren. Wir haben uns deshalb für diese Darstellung entschieden, da wir so alle Modelle gleich berechtigt einfließen lassen können – wir vermeiden, dass das aufwendigste Modell das Ergebnis vorgibt.

Relationen kennen wir nun für einen gemittelten Blick, bei dem die Ergebnisse aller unserer Modelle genutzt wurden.

In [Figur B.2](#) sehen wir nun die Aufteilung der Laufzeit für die diversen Modelle separat. Hier stellen wir fest, dass die vertraute Reihenfolge an Aufwand ($\text{Frev} < \mathcal{B} < \mathcal{C}$) bestehen bleibt und in allen Fällen die *Couples*-Phase mehr als zwei Drittel der Laufzeit ausmacht.

Ein genauer Blick führt zu weiteren Beobachtungen. Es lässt sich keine Tendenz feststellen, ob es wesentliche Unterschiede zwischen FCA und tFCA gibt. Bei einem der beiden Ausreißermodelle *M.barkeri* (besondere lange *Blocked*-Phase in der FCA) und *E.coli* (extrem lange *Couples*-Phase für beide Verfahren) ist die Rechenzeit zur Bestimmung der Kopplungen beim tFCA länger, bei den anderen Modellen ist es ausgeglichener/andersrum. Zudem ist zu bemerken, dass die Aufteilung bei dreien der Modelle sich sehr stark ähneln und sich abzüglich der beiden genannten Ausnahmen ein sehr harmonisches Bild ergibt.

Eine Übersicht über die genauen Laufzeiten und Werte gewähren die Tabellen [B.1](#) (FCA) und [B.2](#) (tFCA). Für die genannten Laufzeiten ist der Median aus den wiederholten Simulationen gewählt worden, um einen möglichst geringen Bias der Daten zu erreichen.

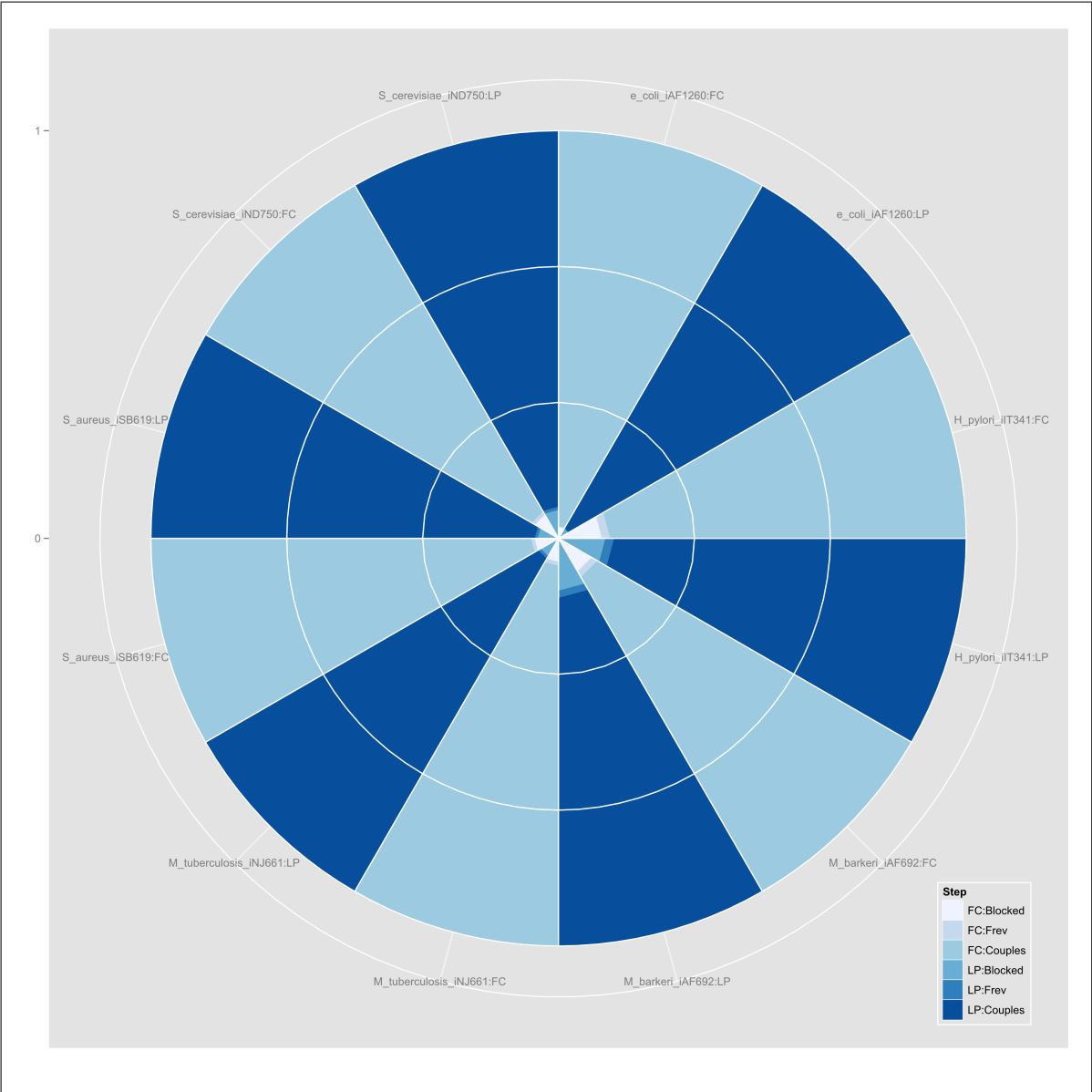


Abbildung B.2.: Relativer Zeitaufwand im Vergleich für verschiedene Modelle.

Model	Value	Number	LPs	Witnesses	Time
E_coli_iAF1260	Total	2382	11233	4067	228.1
	1 _L	1460	1635	436	5.8
	Frev	3	320	0	0.5
	C	13175	9278	3631	221.8
H_pylori_iIT341	Total	554	1978	295	3.4
	1 _L	246	426	46	0.4
	Frev	6	110	0	0.1
	C	4651	1442	249	2.9
M_barkeri_iAF692	Total	690	1778	477	4.8
	1 _L	287	552	63	0.6
	Frev	14	116	0	0.1
	C	5413	1110	414	4.0
M_tuberculosis_iNJ661	Total	1025	3425	1148	21.5
	1 _L	564	694	160	1.0
	Frev	27	180	0	0.1
	C	3048	2551	988	20.4
S_aureus_iSB619	Total	743	2499	634	7.7
	1 _L	375	506	76	0.4
	Frev	0	163	0	0.1
	C	4060	1830	558	7.2
S_cerevisiae_iND750	Total	1266	3822	1283	28.1
	1 _L	625	1013	166	1.9
	Frev	35	207	0	0.2
	C	3085	2602	1117	25.9

Tabelle B.1.: FCA Benchmark – Berechnung der Kopplungen ohne Kreisabschluss.
Zeitangaben in Sekunden.

Model	Value	Number	MILPs	Witnesses	Time
E_coli_iAF1260	Total	2382	377	4063	253.2
	1_L	1451	27	426	7.1
	Frev	0	3	0	0.5
	\mathcal{C}	13392	347	3637	245.6
H_pylori_iIT341	Total	554	43	284	3.8
	1_L	234	0	42	0.4
	Frev	6	21	0	0.1
	\mathcal{C}	4472	16	242	3.4
M_barkeri_iAF692	Total	690	129	552	6.6
	1_L	286	8	58	0.6
	Frev	0	14	0	0.1
	\mathcal{C}	5611	107	494	5.9
M_tuberculosis_iNJ661	Total	1025	167	1248	23.3
	1_L	558	21	150	1.3
	Frev	0	24	0	0.2
	\mathcal{C}	3146	122	1098	21.7
S_aureus_iSB619	Total	743	15	573	6.9
	1_L	374	2	75	0.4
	Frev	0	0	0	0.1
	\mathcal{C}	4061	13	498	6.4
S_cerevisiae_iND750	Total	1266	220	1352	33.7
	1_L	620	19	158	2.3
	Frev	35	35	0	0.3
	\mathcal{C}	3118	166	1194	31.1

Tabelle B.2.: tFCA Benchmark – Berechnung der Kopplungen unter Kreisabschluss.
Zeitangaben in Sekunden.

§B.2.2 Zeitlicher Verlauf

Hier wollen wir die Rechenphase genauer betrachten, die den Löwenanteil unseres Aufwands ausmacht: Die Berechnung der Kopplungen, wenn 1_L und Frev bereits bekannt sind.

Diese Phase lässt sich sehr gut in Stufen unterteilen: Für jede unblockierte Reaktion $i \in 1_L$ gibt es eine Stufe, die wir nacheinander besteigen. In jeder Stufe wird das zugehörige Maximum $1_{L_{\perp(i)}}$ berechnet. Die Abbildungen B.3, B.4, B.5 und B.6 machen sich dies zu Nutze; dort werden wieder unsere drei integralen Messgrößen präsentiert. Allen Abbildungen ist gemein, dass sie nicht den zusätzlichen Aufwand pro Schritt, sondern den Gesamtaufwand bis zum Schritt i darstellen.

Zeitlicher Aufwand

In allen sechs Zellmodellen dauern FCA und tFCA ähnlich lang ausfällt, wobei die tFCA grob geschätzt 15% länger dauert. Das dürfte an den MILPs liegen.

Wir haben das Konzept der Zeugen eingeführt, um durch gute untere Schranken einen guten Ausgangspunkt für eine Intervallschachtelung zu haben, die unsere Laufzeit verkürzen soll. Daher ist davon auszugehen, dass mit steigender Anzahl bekannter Verbandselemente die Rechenzeit für eines einzelnen Maximums schrumpfen sollte. Dies ist bei unseren vier kleineren Modellen gut zu beobachten, deren Gesamtlaufzeit sich sehr gut durch eine rechtsgekrümmte, ansteigende Parabel über die bereits getesteten Reaktionen i darstellen lässt.

Anders verhält es sich für unsere beiden größeren Modelle, *S.cerevisiae* und insbesondere *E.coli*. Wo sich bei ersterer eine perfekte Regressionsgerade in die Laufzeit legen lässt, approximiert man unser größtes System am besten durch eine linksgekrümmte Parabel. Dafür gibt es keine einfache Erklärung. Grund könnte die bereits erwähnte Laufzeitsteigerung durch verwaltungstechnische Schritte sein. Mit steigender Zahl an Zeugen wächst die Laufzeit im Preprocessing, ohne dass notwendigerweise die Anzahl an zu lösenden LPs wesentlich fällt. Das werden wir später bei der EFCA testen.

Ein Indiz für diese Vermutung lässt sich dem direkten Vergleich der Abbildungen B.3a mit B.3b entnehmen. An zwei Stellen zur Laufzeit – etwa bei Reaktion 850 und dann wieder kurz vor 1250 – steigt die Anzahl an gelösten LPs (und gefundenen Zeugen) in der FCA sprunghaft an. Sollten die LPs wesentlicher Treiber der Rechenzeit sein, müssten entsprechende Sprünge auch in der Darstellung feststellbar sein. Dies ist jedoch nicht der Fall!

Es ist wie so häufig bei experimentellen Daten: Wenn man sie sehen will, so erkennt man leichte Knicke in den Laufzeitgraphen an den Stellen, wo besonders viele LPs gelöst und Elemente gefunden wurden. Ein kleiner Sprung hier, weil Gurobi dort ja lange braucht, etwas steiler dort, weil wir ja jetzt mehr Zeugen verwalten müssen.

Aus Sicht des Autors kann auf eine quantitative Überprüfung verzichtet werden – dafür unterstützen die Daten hier an dieser Stelle noch nicht kräftig genug die Behauptung, dass die Anzahl an Zeugen Auswirkung auf die Laufzeit hat.

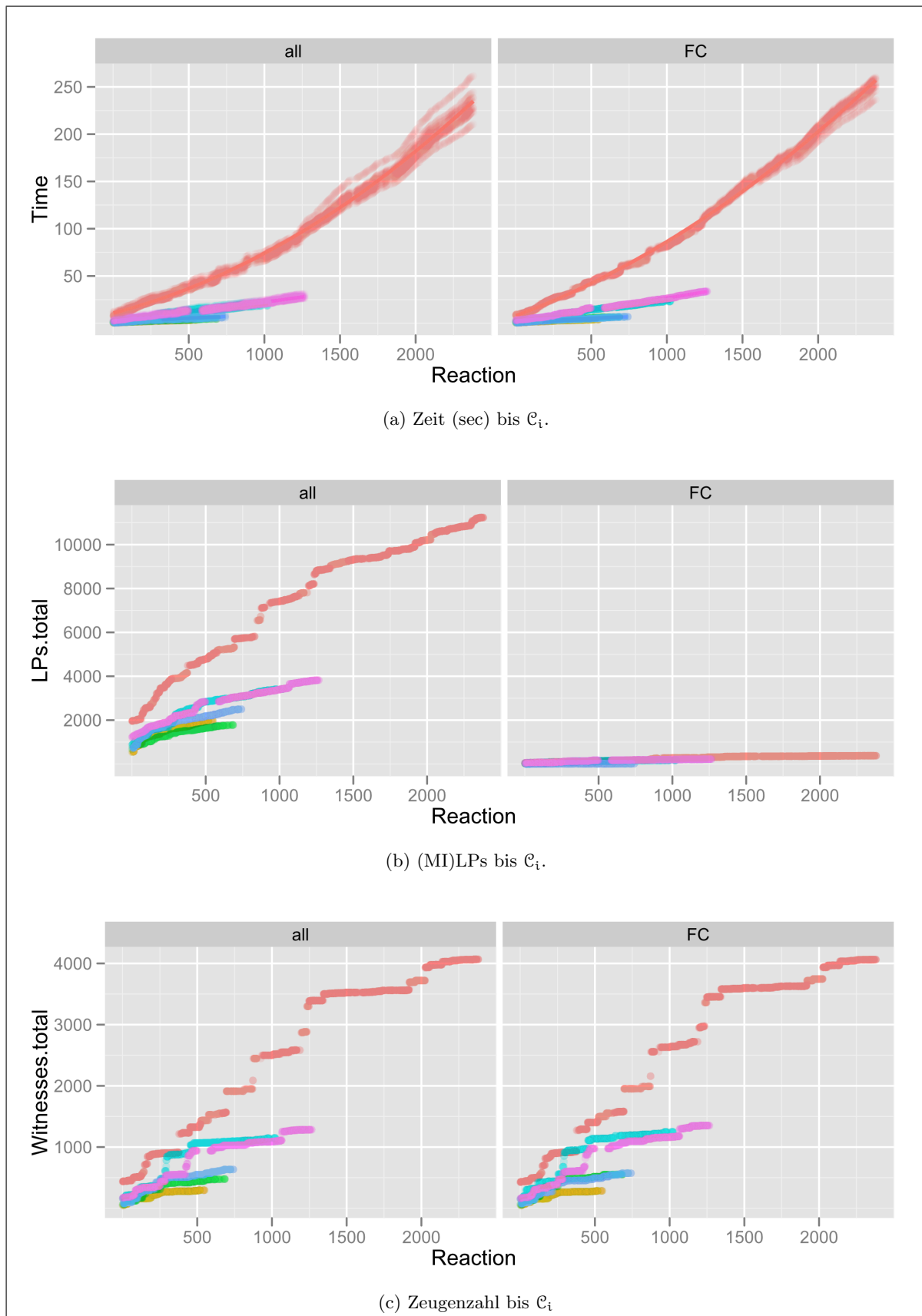


Abbildung B.3.: Zeitaufwand, Test-Aufrufe und Zeugenanzahl im Verlauf der FCA. Vergleich von FCA mit tFCA.

11 Wiederholungen, teils ergänzt um quadratische Regression.

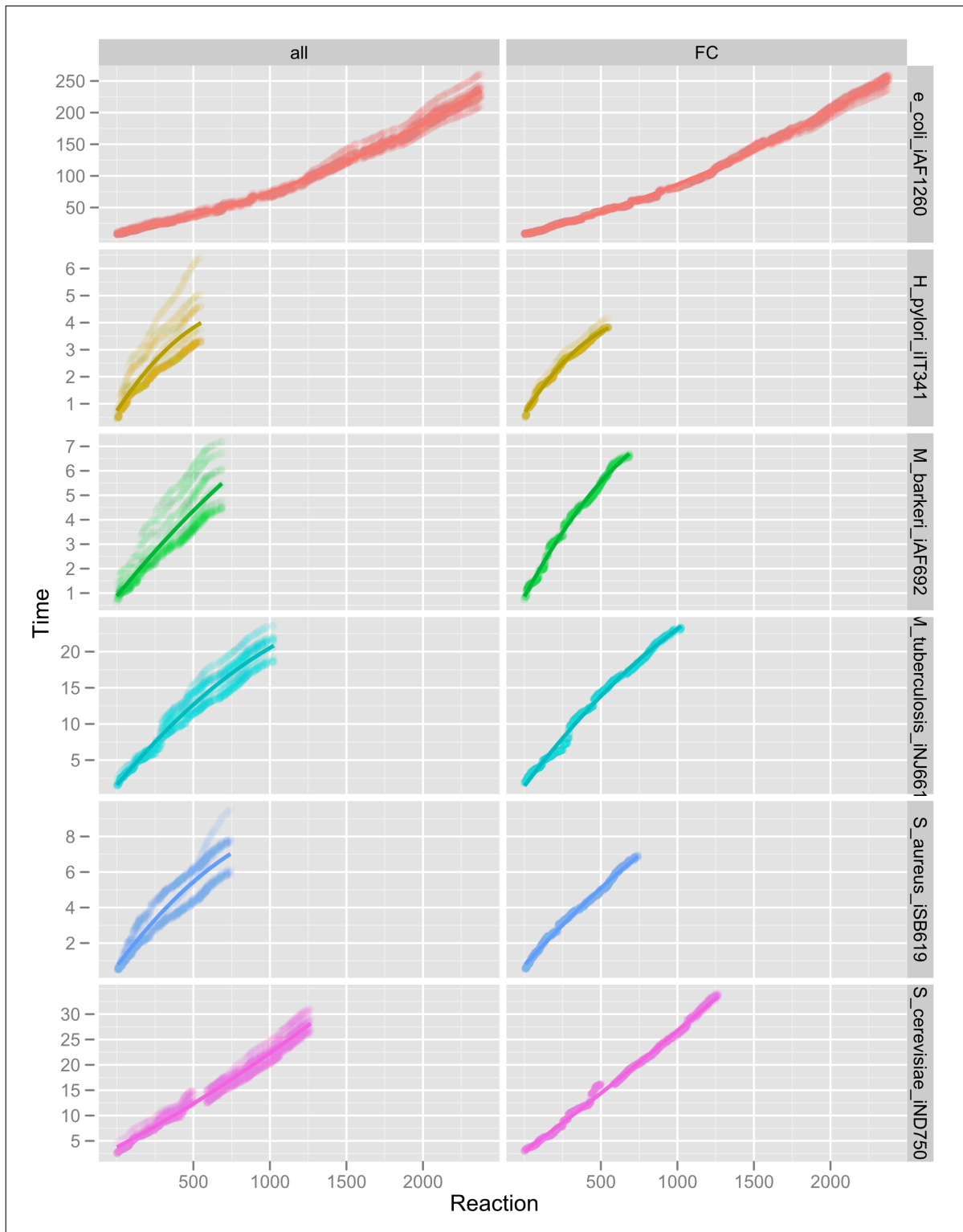


Abbildung B.4.: Zeitaufwand zur Berechnung der ersten i Maxima $1_{L_{\perp}(i)}$.

LPs, MILPs & Zeugen

In Abbildung B.3b ist auf der linken Seite dargestellt, wie viele LPs gelöst werden müssen, um diesen Punkt zu erreichen. Auf der rechten Seite betrachten wir die tFCA. Auch hier ist es interessant zu wissen, wie viele LPs zu lösen sind, da wir ja im Hintergrund eine FCA laufen lassen, deren Zeugen wir erst im Nachhinein auf Kreisfreiheit testen. Doch das führt uns zur interessanteren Frage: Wie oft finden wir denn Kreise vor? Und wie oft müssen wir deshalb ein MILP lösen, das wesentlich komplexer ist? Die Antwort auf diese Frage ist auf der rechten Seite abgebildet. Um es kurz zu machen: Unglaublich wenige.

Daher überrascht es kaum, dass die Graphen in B.6 beinahe identisch ausfallen. Hier vergleichen wir Zeugenanzahl der FCA mit der der tFCA. Da wissen wir: Jeder Zeuge aus der FCA, der keinen Kreis enthält, ist auch Zeuge der tFCA. Aus B.3b wissen wir, dass das fast alle sind. Nur wenige müssen wir aussortieren und ersetzen sie gegebenenfalls durch ein kreisfreies Äquivalent. Daher sollte und ist die Zeugenanzahl der tFCA fast identisch zu der der FCA.

Auf die Besonderheiten des *E.coli*-Systems sind wir im vorangegangenen Abschnitt detailliert eingegangen, weshalb wir uns jetzt auf die fünf anderen Modelle konzentrieren wollen.

Allen ist gemein, dass sie unsere Idee, mit steigender Laufzeit und gewachsener Anzahl bekannter Pfade, weniger LPs lösen zu müssen, vorbildlich demonstrieren. In der Einstiegsphase müssen viele LPs gelöst werden, werden viele neue Elemente entdeckt, im späteren Verlauf nehmen beiden Zahlen ab. Was jedoch neu ist und überrascht, ist ein Knick in der Anzahl an Zeugen. Man kann vielleicht sogar von einem *Sweetspot* reden, ab dem die Anzahl an Zeugen massiv abnimmt, sowohl bei FCA als auch bei tFCA.

Diesen Sweetspot vorhersagen zu können, würde enorme Vorteile bringen. Künftige Algorithmen zur FCA könnten gefundene Elemente bis zu diesem Punkt der Zeugenliste hinzufügen und alle folgend gefundenen Elemente direkt nach ihrer Auswertung verwerfen. Dies scheint ein optimaler Angriffspunkt zu sein, um die ausufernde Rechenzeit zur Zeugenverwaltung zu deckeln ohne die Vorteile einer voroptimierten Intervallschachtelung zu verlieren.

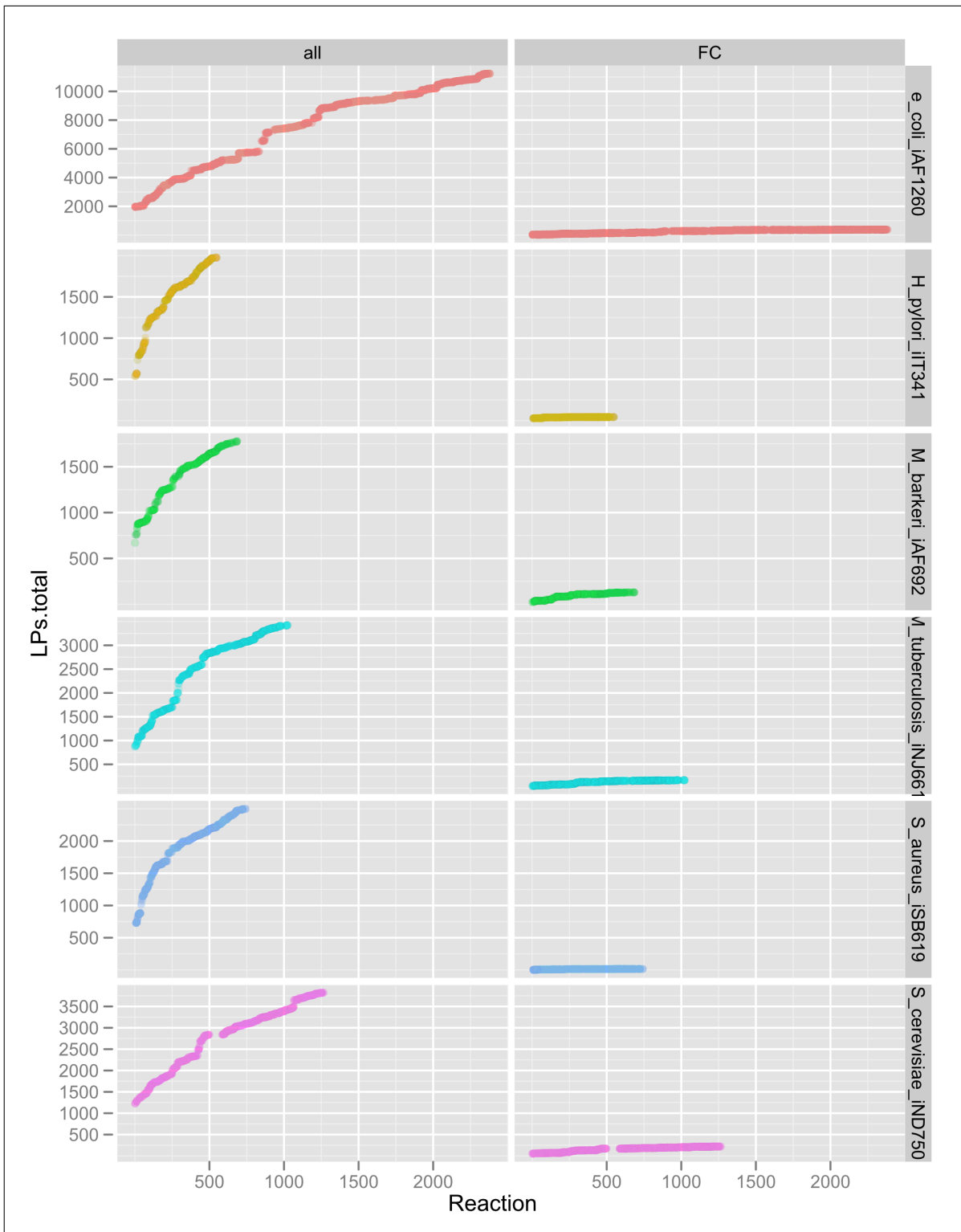


Abbildung B.5.: Gelöste LPs/MILPs zur Berechnung der ersten i Maxima $1_{L_{\perp\{i\}}}$.

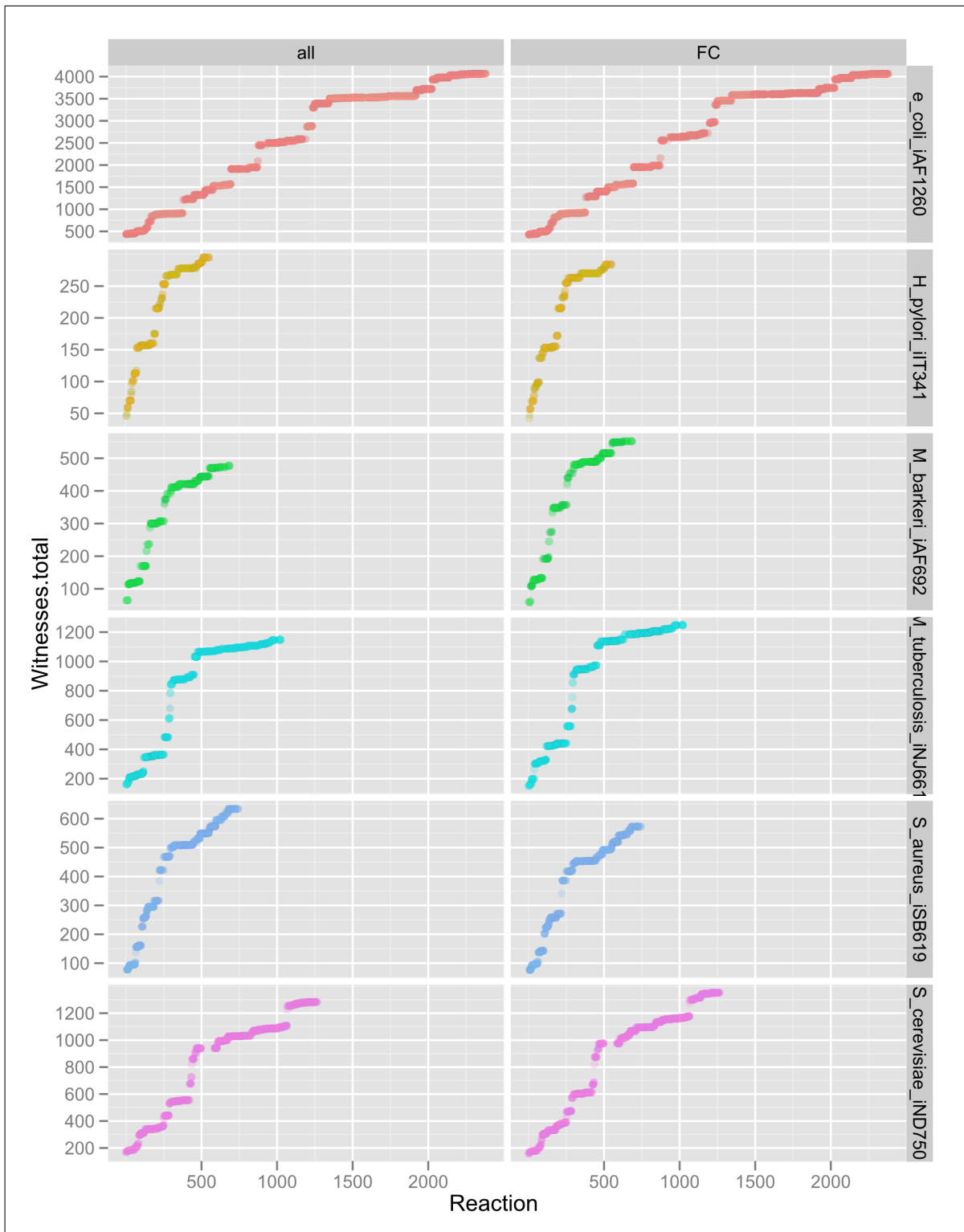


Abbildung B.6.: Gefundene Elemente zur Berechnung der ersten i Maxima $1_{L_{\perp\{i\}}}$.

§B.2.3 Zeugen

Werfen wir abschließend einen letzten Blick auf die gefundenen Elemente selbst.

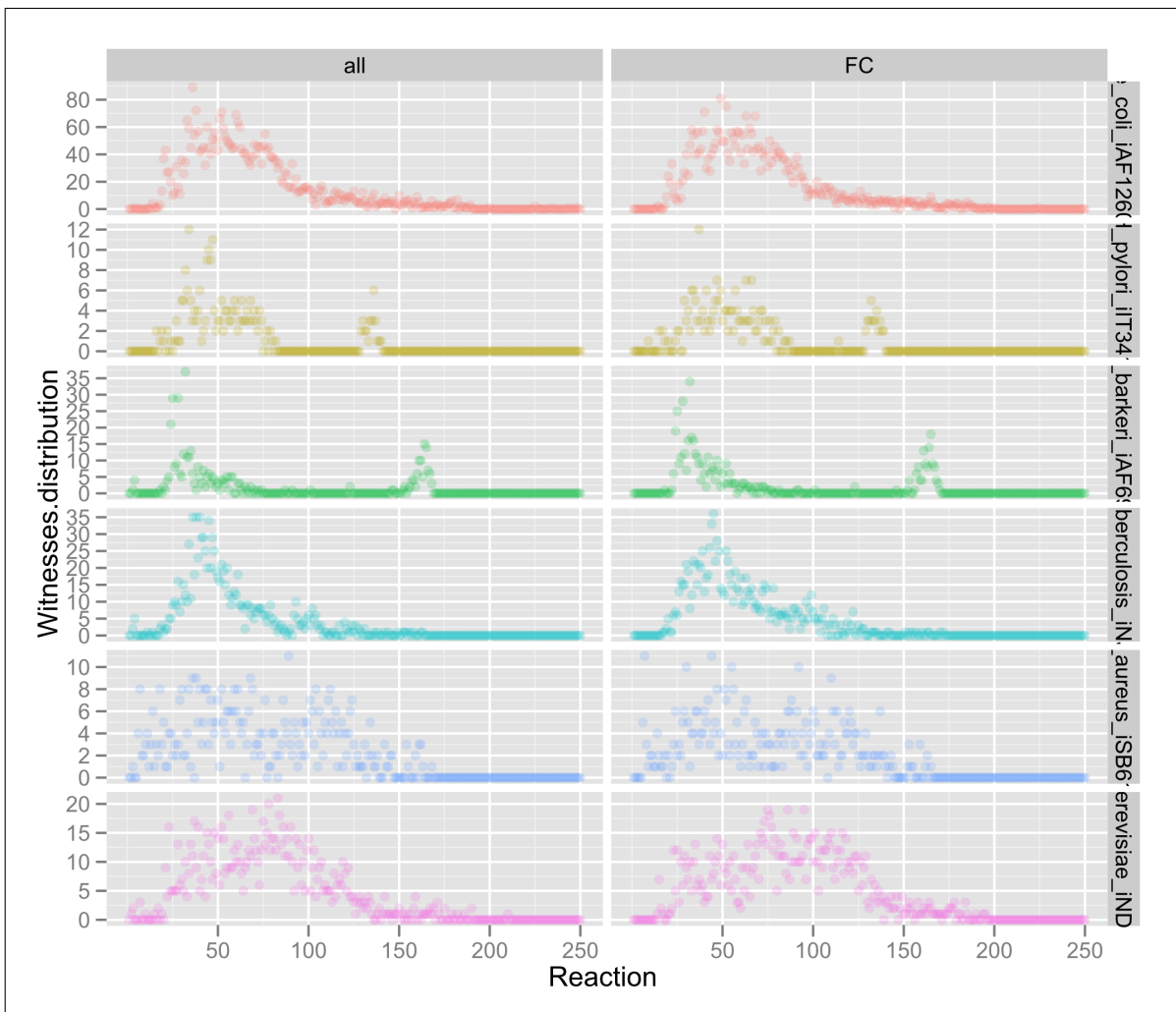


Abbildung B.7.: Anzahl gefundener Verbandselemente (Zeugen) aufgeteilt nach Kardinalität.

Wie wir Abbildung B.7 entnehmen können, enthalten die meisten unserer gefundenen Stoffwechselfade zwischen 50 und 100 Reaktionen; mehr als 200 ist so gut wie ausgeschlossen. Dies ist insofern interessant, da wir Systeme mit 550 bis 2300 Reaktionen betrachten und die Komplexität der gefundenen Phase davon so gut wie unabhängig ist.

Zudem ist man versucht, eine Log-Normalverteilung der Pfadlänge mit maximaler Wahrscheinlichkeit von 50-100 zuschreiben zu wollen. Dem in die Quere kommen die beiden Modelle *H_pylori* und *M_barkeri*, die über zweite kleine Peaks bei 130 bzw. 160 verfügen.

Wir wissen nun in etwa, was für Elemente gefunden werden. Aber wie häufig passiert das? In Abbildung B.3 deutet es sich schon an, in Abbildung B.8a noch einmal genau. Hier stellen wir die gefundenen Elemente der Anzahl gelöster LPs in der FCA (jedoch nicht in der tFCA) gegenüber.

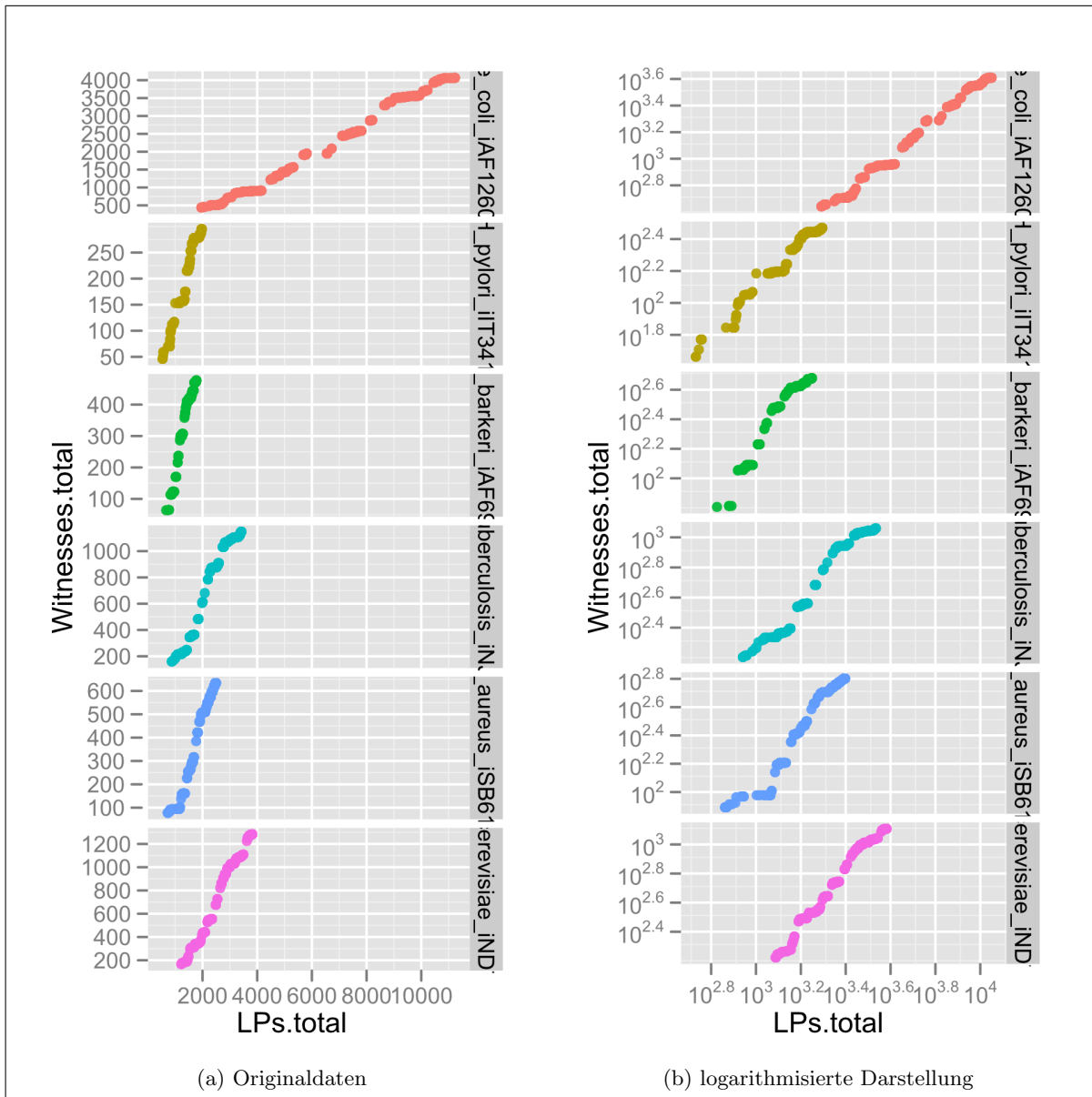


Abbildung B.8.: Anzahl gefundener Verbandselemente (Zeugen) über Anzahl gelöster LPs.

Offensichtlich gibt es einen linearen Zusammenhang zwischen Anzahl zu lösender LPs und gefundener Elemente. Es scheint also eine Art Modell-spezifische Dichte ρ zu geben mit $\rho \sim \frac{|L_{\perp\{i\}}|}{n}$ Wahrscheinlichkeit, dass Reaktion j nicht an i (beliebig) gekoppelt ist. Denn solche Zeugen finden wir ja immer nur dann, wenn unser Test eine Nicht-Kopplung befindet.

Vielleicht gibt es solch eine Dichte, womöglich ist es aber doch nur das Gesetz der großen Zahlen. Interessant ist es auf alle Fälle.

§B.3 EFCA

Bei der EFCA simulieren wir sämtliche denkbaren Reaktionen-Doppel-Knockouts. Da hierfür rund $\frac{n^2}{2}$ Maxima bekannt sein müssen, ist ein effektiver Zeugeneinsatz besonders wichtig.

Unsere Laufzeitanalyse der FCA hat angedeutet, dass es einen *Sweetspot* an vorhandenen Zeugen gibt, bei dem das Verhältnis LP-Vermeidung zu Verwaltungsaufwand besonders günstig ist. Die EFCA haben wir nun in doppelter Ausführung implementiert und getestet: Bei der ersten Fassung werden nach einem heuristischen Verfahren einige Zeugen aus einem Rechenschritt in den nächsten vererbt³, bei der zweiten werden nur die Zeugen aus der vorangegangenen gewöhnlichen FCA als Ausgangslage verwendet.

Wir wissen, dass die zweite Implementierung mehr LPs lösen muss als die erste. Doch der direkte Vergleich wird uns helfen zu verstehen, wie wichtig das Vorhalten möglichst vieler Stoffwechselfade wirklich ist. Der Blick auf die Laufzeitgraphen wird uns zudem noch zeigen, wie sehr LPs und wie sehr Zeugenmanagement zum Rechenaufwand beitragen.

§B.3.1 Laufzeiteigenschaften

Betrachten wir wieder zunächst den relativen Aufwand. Diesmal konzentrieren wir uns voll auf die „gewöhnliche“ Knockout-Simulation im stationären Zustand ohne zusätzliche thermodynamische Nebenbedingungen (feasibility constraints, FC).

Wir erhalten ein ähnliches Bild, wie wir es bereits in Abbildung B.1 gesehen haben. Die Komplexität der Teilprobleme wächst entsprechend unserer Erwartung, wobei die neu hinzugekommene *EFCA*-Phase konsequent am aufwendigsten ist. Wieder gibt es feine Nuancen zwischen Aufwandsverhältnissen im Vergleich der gelösten LPs zur echten Laufzeit.

Wesentliche Unterschiede zwischen den Simulationen, wo die Zeugen unmittelbar verworfen wurden, zu denjenigen, wo wir heuristisch einige in Folgeschritten übernommen haben, sind nicht erkennbar. Erst bei genauem Vergleich der beiden Abbildungen B.9a und B.9d erkennt man, dass die *EFCA*-Phase von etwa 88% Anteil gelöster LPs auf 91% Anteil ansteigt, wenn wir Zeugen unmittelbar nach Beendigung einer Unter-FCA verwerfen. Mit anderen Worten: Wir müssen dann mehr LPs lösen.

Bei dieser Überlegung sollten wir jedoch nicht vergessen, dass wir hier Mediane der Relationen von lediglich fünf Modellen betrachten, die sich nicht auf 100% aufaddieren (da wir, wie gesagt, Mediane betrachten). Wir können daher nur mit Sicherheit sagen, dass sich das Bild nicht wesentlich verändert.

Betrachten wir die detaillierte Aufteilung des Zeitaufwandes (Abb. B.10a) und der gefundenen Elemente (Abb. B.10b), so gibt es auch hier keine großen Überraschungen. Aus Abbildung

³Wir übernehmen den Zeugen a mit einer Wahrscheinlichkeit von $\frac{|a|}{|W|}$.

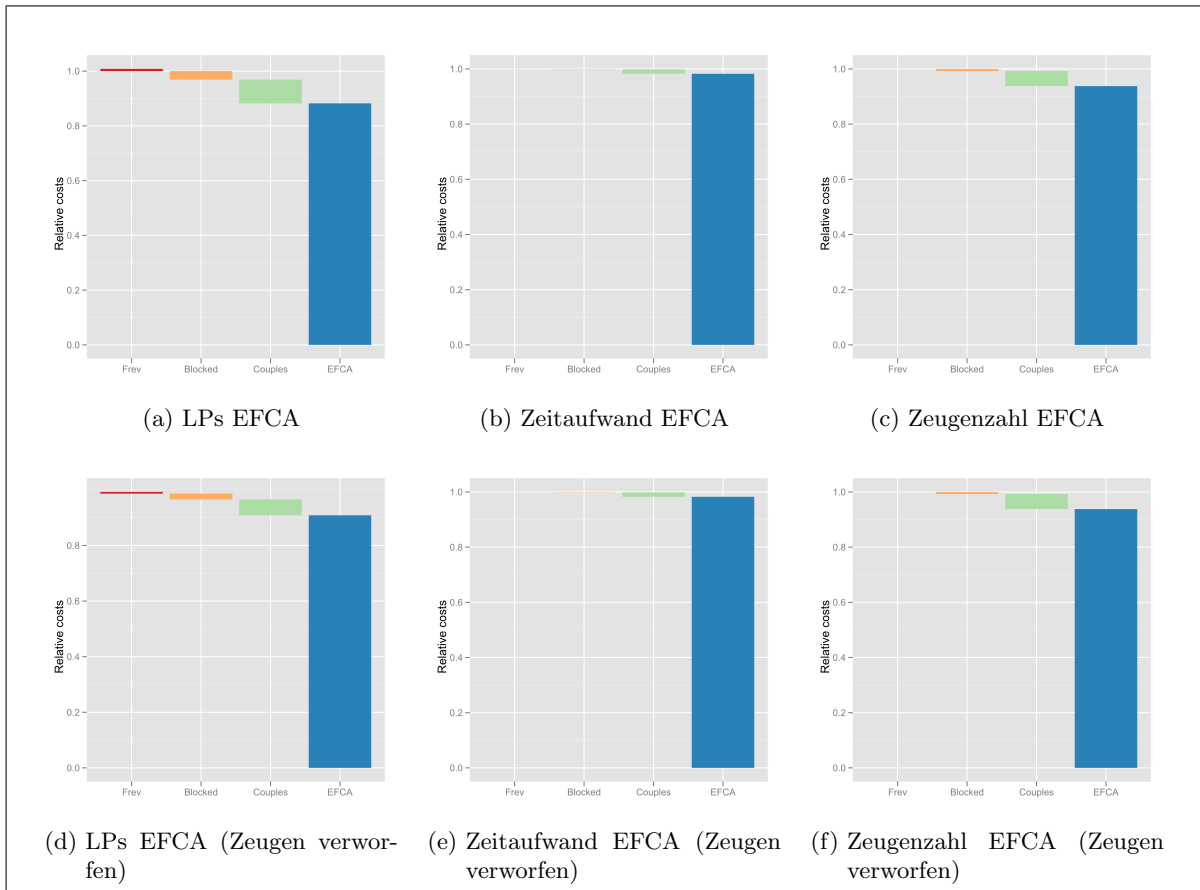


Abbildung B.9.: Relativer Aufwand der verschiedenen Etappen der EFCA. Vergleich Zeugenauswahl mit Zeugenverwerfung.

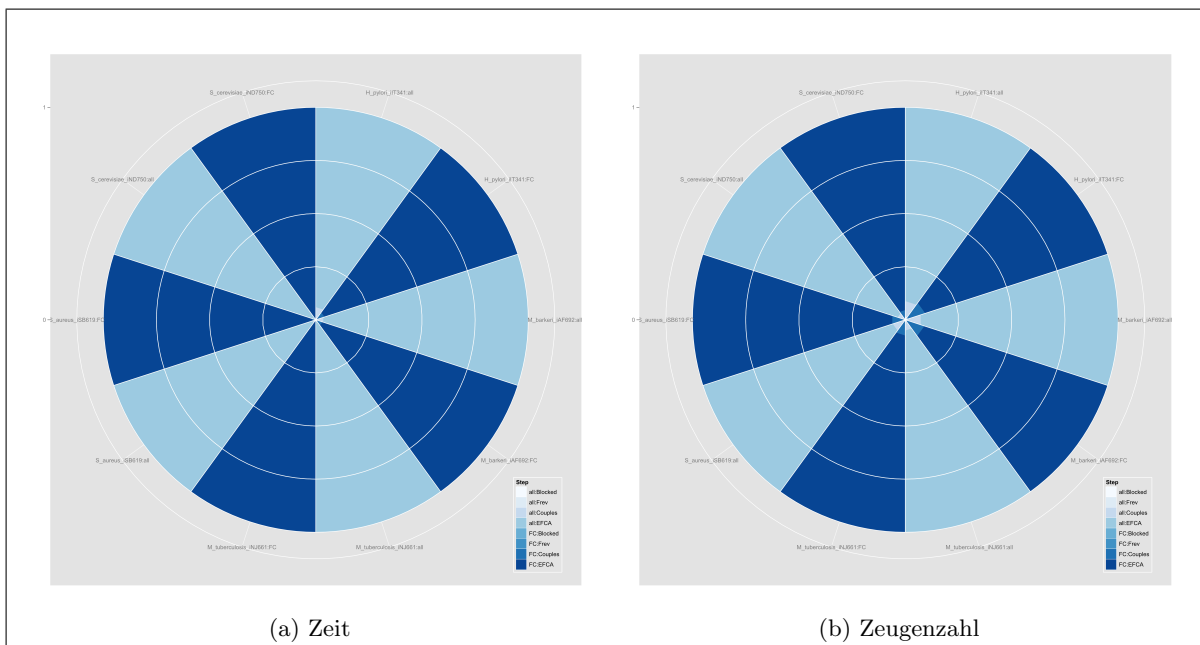


Abbildung B.10.: Relativer Aufwand im Vergleich für verschiedene Modelle. (heuristische Zeugenauswahl)

B.9 wissen wir bereits, dass die *EFCA*-Phase das alles Dominierende ist, beim Zeitaufwand mehr noch als bei der Zahl gefundener Elemente.

So kommt es auch dazu, dass lediglich bei zwei der fünf untersuchten Modelle ein nennenswerter Zeitaufwand für die *Couples*-Phase erkennbar ist. Dabei handelt es sich um die Modelle *H.pylori* und *M.barkeri*. Vergleichen wir Zeitaufwand mit Anzahl gefundener Zeugen, so sehen wir, dass dies auch die beiden Modelle sind, bei denen relativ wenige Elemente erst in der vierten Phase gefunden wurden. Im Gegensatz zur Laufzeit ist in Abbildung B.10b in allen fünf Modellen die *Couples*-Phase mit bloßem Auge erkennbar.

Wesentliche Unterschiede zwischen *EFCA* und *tEFCA* sind nicht erkennbar. Wie schon bei der *FCA* fällt einmal hier, einmal dort der Aufwand beim gleichen Modell größer aus. In jedem Fall ist die neu hinzugekommene *EFCA*-Phase ähnlich aufwendig, egal ob wir thermodynamische Nebenbedingungen inkludieren oder nicht.

§B.3.2 Zeitlicher Verlauf

Wie uns die *FCA* bei der *EFCA* hilft und was die *EFCA* über die *FCA* verrät

Bevor wir uns wieder die schrittweise Entwicklung der Laufzeit, LPs und gefundener Elemente ansehen, wollen wir mit etwas anderem starten: Den wiederverwertbaren Zeugen.

Wieder und wieder dreht es sich darum, wie viele Elemente zu speichern, wie viele als Zeugen zu nutzen sind. Verwaltungsaufwand, Intervallschachtelung, ... Kennen wir schon. Aber was passiert denn nun genau?

Abbildung B.11 sieht auf den ersten Blick wie ein bunter Schneeschauer aus. Tatsächlich aber können wir sehr viel daran ablesen.

Die Spalte *all* umfasst die Simulation im stationären Flussverband mit Anwendung unserer Heuristik, bei *allo* verwerfen wir neue Zeugen. Das sieht man direkt in der Abbildung: Wo bei *all* die maximale Anzahl an wieder-nutzbaren Stoffwechselfaden allmählich auf etwa das Doppelte ansteigt, so bleibt sie bei *allo* konstant.

Doch wir erkennen mehr: Die Schritte, bei denen wir nahe am jeweiligen Maximalwert liegen, sind regelmäßig verteilt über die gesamte Laufzeit: Wir kennen nicht mehr oder weniger „gute“ Elemente, die für kleines i in $L_{\perp\{i\}}$ enthalten sind, als dies für großes i vorkommt.

Dies hätte durchaus der Fall sein können. Wir haben bereits festgestellt, dass wir gerade in der Startphase der *FCA* besonders viele LPs lösen und Elemente finden. In dieser Startphase werden gezielt Elemente aus $L_{\perp\{i\}}$ mit kleinem i gesucht. Die Spalte *allo* in Abbildung B.11 liefert uns also weitere Erkenntnisse über die *FCA*, da hier nur Zeugen aus diesem Algorithmenschritt wiederverwendet werden.

Noch etwas fällt auf: Beim Vergleich mit Abbildung B.6 erkennen wir zudem, dass unsere Obergrenze von *allo* mehr oder weniger dem vermutlichen Sweetspot-Niveau entspricht. In

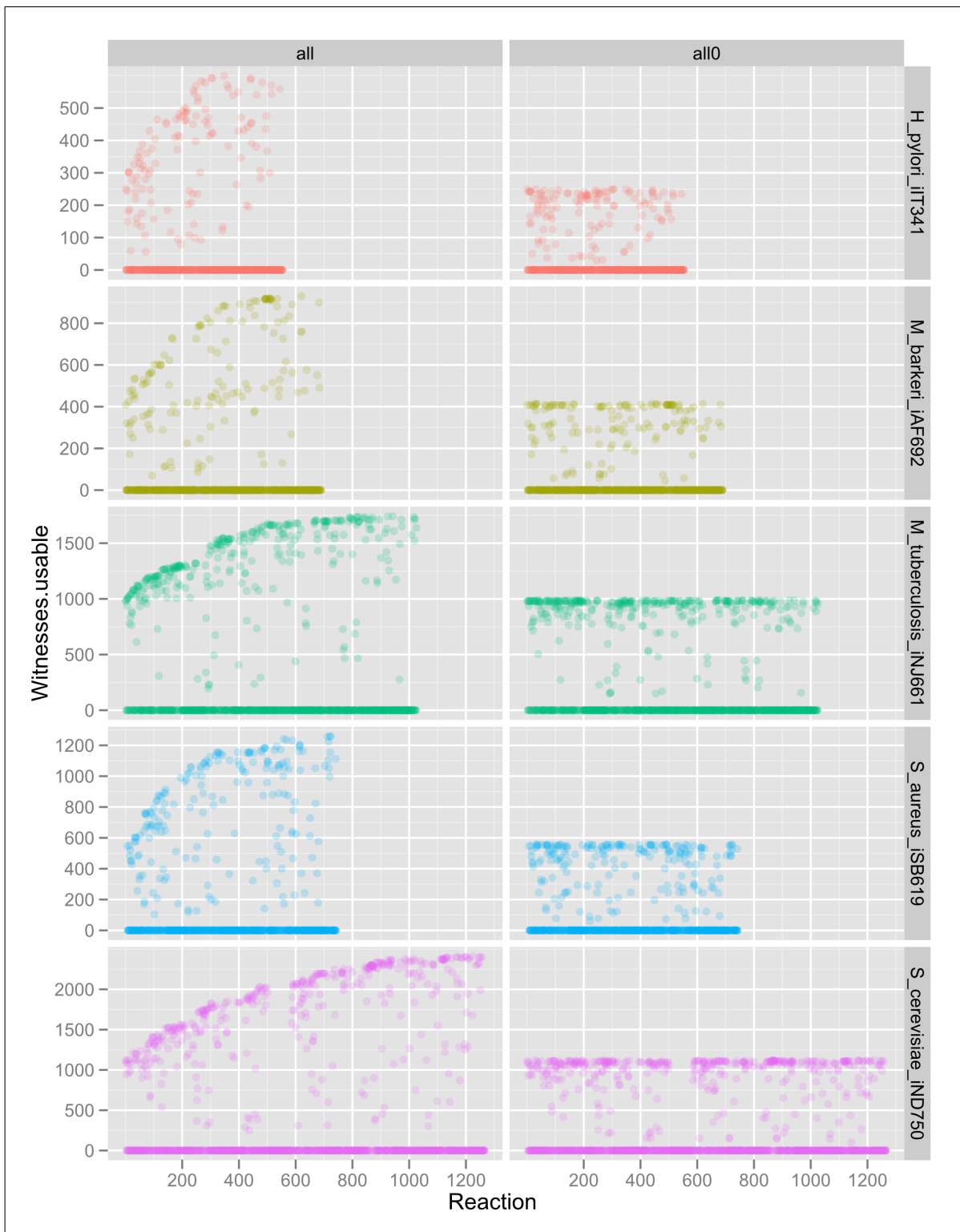


Abbildung B.11.: Anzahl bekannter Elemente, die wir im i -ten Schritt wiederverwerten. Vergleich heuristische Zeugenauswahl mit direkter Verwerfung. 1 Simulation ohne Wiederholung.

drei der fünf Modelle treffen wir ihn genau, bei den beiden weiteren nutzen wir einige Elemente mehr. Gehen wir davon aus, dass die FCA angewandt auf $L_{\perp\{i\}}$ – was dem i -ten Schritt in der *EFCA*-Phase entspricht – sich nicht wesentlich anders verhält als die FCA von L^F selbst, so starten wir in vielen Fällen gleich von Anfang an mit der nötigen Anzahl an Elementen, um schnell arbeiten zu können. Vielleicht kennen wir in der *EFCA*-Phase sogar schon von Beginn an genug Elemente, um zusätzliche Zeugen für die i -te FCA gar nicht erst speichern zu müssen. Dies sollte in einer künftigen Version mit seiner Auswirkung auf die Gesamtlaufzeit getestet werden.

Damit nicht genug. Diese Abbildung, die in der *EFCA*-Phase entstanden ist, gibt uns Informationen über die *FCA*-Phase. Genauer: Wie viele Elemente nutzen wir jeweils, um `lower_bound` zu berechnen. In der Spalte *allo* werden nur Elemente gelistet, die Zeugen in der *FCA*-Phase waren. Doch auch wenn wir nicht alle Elemente von Anfang an kennen, so wissen wir aus Abbildung B.6, dass wir sehr früh die meisten Elemente berechnet haben. Nennen wir $S = |\text{Sweetspot}|$, so gibt uns $\Theta(S \cdot n)$ eine gute Einschätzung, wie häufig wir den $+$ Operator zur Berechnung der unteren Schranken in der FCA verwenden. Unsere Implementierung schafft so eine Addition in $\Theta(n)$ Punktoperationen. Daraus ergibt sich eine Einschätzung der Gesamtlaufzeit der FCA von $\Theta(S) \text{ LPs} + \Theta(S \cdot n^2 + |W| \cdot n \cdot 1) = \Theta(S) \text{ LPs} + \Theta(S \cdot n^2)$.

Für diese Abschätzung verwenden wir zwei Vermutungen: Erstens, dass es einen Sweetspot in der Zahl bekannter Elemente gibt, von wo an wir nur noch unwesentlich wenige LPs lösen müssen. Zweitens, es gibt einen linearen Zusammenhang zwischen Anzahl gelöster LPs und gefundener Elemente.

Schließlich wollen wir einen letzten kleinen Punkt aus Abbildung B.11 mitnehmen. Wir sind gut. Die bunte Abbildung zeigt uns nämlich nicht nur, dass aus einer dichten Wolkenschicht Schneeflocken herabsinken, sondern diese auf einen festen, durchgängigen Boden auftreffen. Der bedeutet für uns nur das Allerbeste – jeder Datenpunkt, der auf dem 0-Niveau liegt, ist einer, wo wir auf den Schritt in der *EFCA* verzichten können. Oftmals, weil die Reaktion blockiert ist, aber auch – und das ist das Gute – weil wir auf diesen Schritt verzichten können, da er redundant wäre. Wir kennen den Ausgang schon, etwa wegen einer vorliegenden Kopplung $i \leftrightarrow j$. Der Boden ist fest und stabil, darum sind wir schnell und schaffen die *EFCA*, den Reaktionen-Doppel-Knockout, in kürzerer Zeit, als vor wenigen Jahren noch für die gewöhnliche FCA nötig war.

Laufzeitverhalten

Wie schon bei der FCA gibt es eine zusammenfassende Graphik, bei der Laufzeit, gelöste (MI)LPs und gefundene Elemente modellübergreifend verglichen werden können. Dafür haben wir in Abbildung B.12 die Ergebnisse von je fünf Simulationen für unsere vier Möglichkeiten (*EFCA* und *tEFCA*, Zeugen auswählen/verwerfen) gesammelt. Um repräsentative Ergebnisse zu ermöglichen, gab es jeweils fünf Wiederholungen. Bei vier verschiedenen Szenarien gab es somit insgesamt 20 Simulationen pro Modell; aus Zeitgründen haben wir auf diese akribische Auswertung für *E.coli_iAF1260* verzichtet.

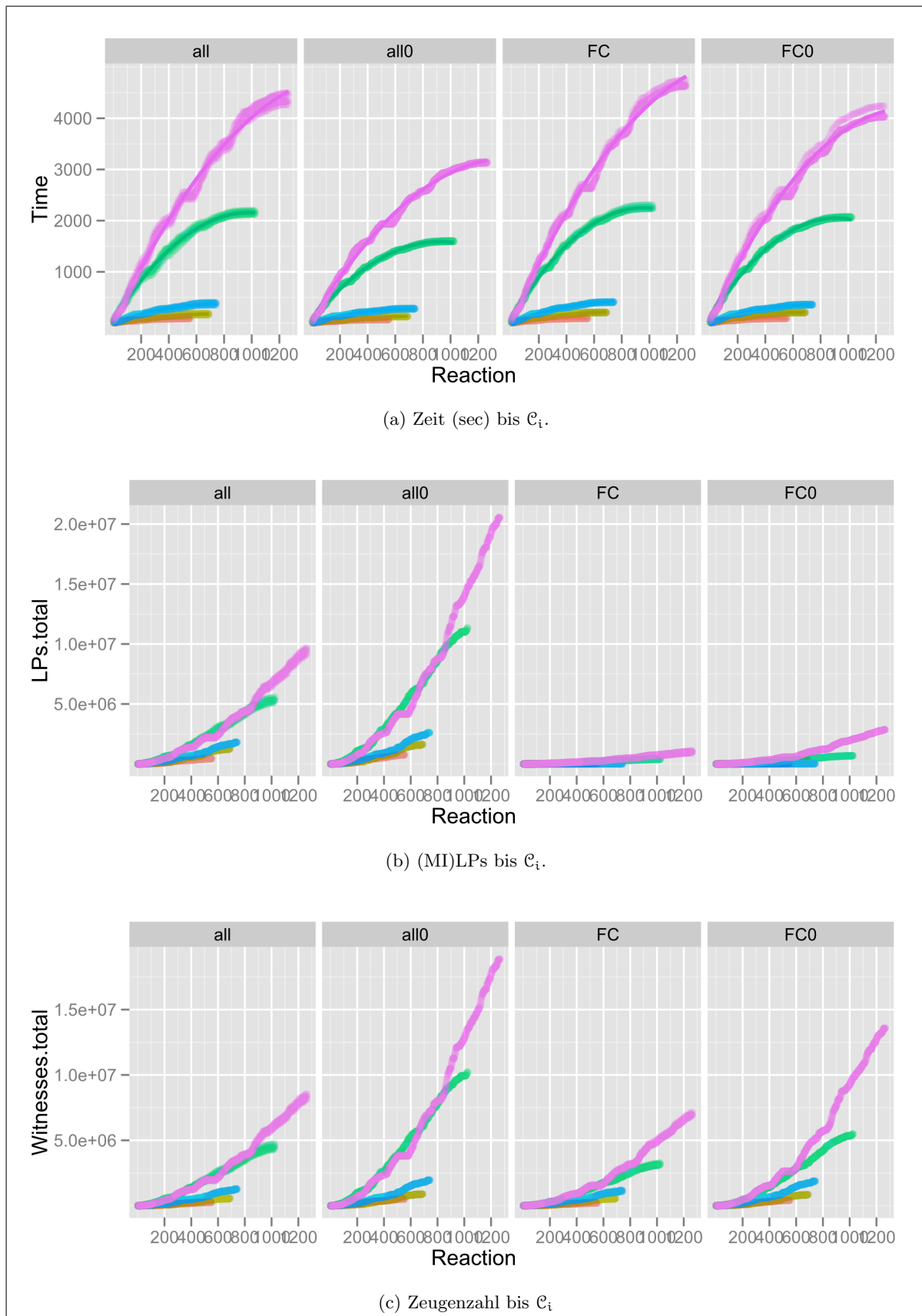


Abbildung B.12.: Zeitaufwand, Test-Aufrufe und Zeugenanzahl im Verlauf der FCA. Vergleich von EFCA mit tEFCA, Zeugenauswahl mit Zeugenverwerfung. 5 Wiederholungen, teils ergänzt um quadratische Regression.

Zeitlicher Aufwand

Wir haben in Abbildung B.4 gesehen, dass sich der zeitliche Rechenaufwand einer FCA nicht in ein perfektes Modell packen lässt. Es sollte eine monoton steigende Rechtskurve sein, was wir in vier von sechs Fällen durch eine Parabel approximieren konnten. Warum dies nicht immer geht, wollen wir hier kein zweites Mal diskutieren. Aber seien wir ehrlich: Auch lineare Approximation würde ganz gut funktionieren, selbst wenn sie nicht so einfach logisch begründbar wäre. Viel zu unterschiedlich ist der Verwaltungsaufwand der Zeugen in den ersten von den letzten Rechenschritten.

Hier wollen wir aber nicht über die FCA, sondern die EFCA reden. Die kleine Einleitung war nötig, da wir es wieder mit der FCA zu tun haben. Jeder Teilschritt in der EFCA-Phase entspricht nämlich mehr oder weniger einer abgekürzten FCA. Wesentlicher Unterschied ist, dass wir von Anfang an mit einem großen Zeugensatz starten und später nur noch wenige LPs lösen müssen.

Dieser Unterschied sollte uns jetzt aber endgültig in die Position bringen, linear zu approximieren. Für jedes Mal gibt es eine theoretische Laufzeit τ für die FCA, so dass sich der i -te Schritt (bei dem die Kopplungen an die Reaktionen $i + 1, \dots, n$ berechnet werden) entsprechend in $\frac{n-i}{n} \cdot \tau$ ablaufen sollte. Die Laufzeit eines einzelnen Schritts fällt linear, also summiert sich die Gesamtlaufzeit zu einer quadratischen Funktion auf, den Parabelscheitelpunkt im n -ten Schritt. So viel zur Theorie.

Die Praxis sehen wir in B.13. Eine bessere Vorhersage hätten wir nicht machen können! Egal welcher Implementierung wir folgen, welches Modell wir betrachten, ob wir thermodynamische Nebenbedingungen integrieren oder nicht – uns strahlen perfekte Halbparabeln an. Eine bessere Unterstützung unserer Laufzeithypothese kann es nicht geben.

Betrachten wir deshalb nun die beiden Implementierungen. Dabei sehen wir Erstaunliches, das uns aber doch nicht mehr überrascht.

Lange Zeit war die Laufzeit einer FCA davon getrieben, viele LPs lösen zu müssen, was vergleichsweise zeitaufwendig ist. Es musste alles daran gesetzt werden, LP-Solver-Aufrufe zu vermeiden.

Zeiten ändern sich. Und wie wir sehen, kann man es auch übertreiben. In den beiden Datenspalten *all0* und *FC0* verzichten wir darauf, besonders viele Flussmuster als Zeugen vorzuhalten, und werden mit einer rund 25% kürzeren Laufzeit in der EFCA (Spalten *a11* und *a110*, alle Stoffwechselfade sind zugelassen) belohnt. Weniger Technik, mehr harte LP-Solver-Arbeit, kürzere Laufzeit. So kann's gehen. Bei der tEFCA (*FC*, *FC0*) sind wir auch schneller, aber nur wenige Prozentpunkte. Hier müssen nämlich nicht nur zusätzliche LPs, sondern auch die viel komplexeren MILPs gelöst werden (Abbildung B.14).

Was wir daraus lernen: LPs kann man schnell lösen, MILPs nicht. Wir können schneller sein, wenn wir am Verwaltungsapparat abspecken, auch wenn wir dafür zusätzliche lineare Probleme lösen müssen. Die unterschiedliche Beschleunigung bei EFCA und tEFCA passt perfekt in das Bild, dass wir effizienter programmieren müssen, denn mittlerweile bremst uns

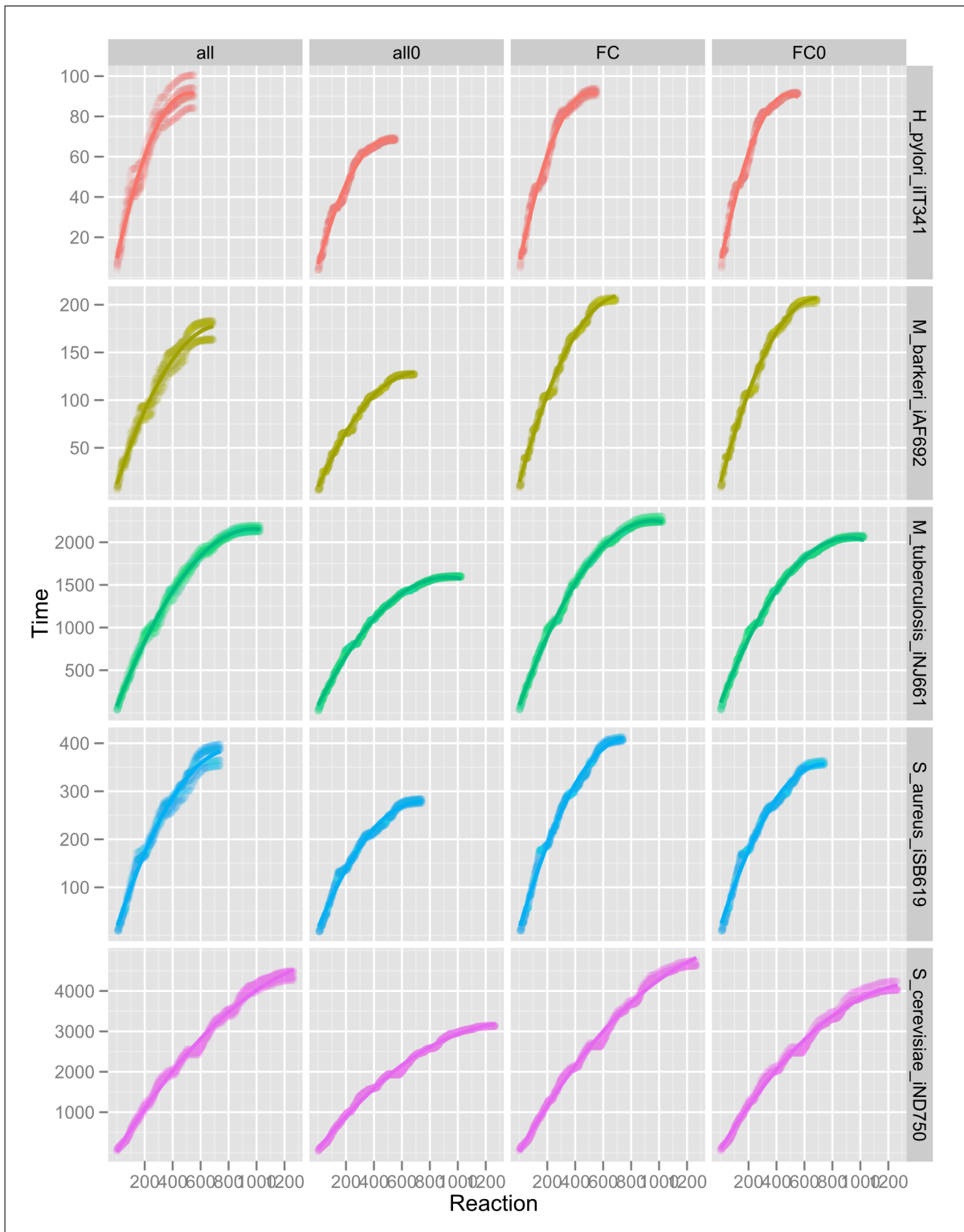


Abbildung B.13.: Zeitaufwand zur Berechnung der ersten i Maxima $1_{L_{\perp\{i\}}}$.

die Theorie schon aus.

Was wir daraus mitnehmen: Künftige Implementierungen werden wohl um einiges schneller laufen können, da wir eine schlanke Verwaltung als Hauptankerpunkt identifiziert haben. So viel Ehrlichkeit muss sein, aus Bequemlichkeit wurde hier oftmals auf Ressourcenschonung verzichtet.

LPs, MILPs & Zeugen

Nach all der Euphorie und Erkenntnisgewinn ein kleiner Stimmungsdämpfer. Es gibt keine logische Erklärung für die Abbildungen B.14 und B.15.

Wir können wie auch bei der Laufzeit versuchen, eine Prognose zu machen. Wir wissen von der FCA, dass bis zum Erreichen eines Sweetspot in kurzer Zeit sehr viele LPs gelöst werden, anschließend nur noch vernachlässigbar wenige.

Übertragen auf die *EFCA*-Phase könnte man also denken, dass es einen Großteil an Schritten gibt, bei denen zum Sweetspot aufgefüllt werden kann, wofür nach Abbildung B.11 in etwa gleich viele (wenige) LPs nötig sein müssten. Erst wenn unser i so groß, unsere Unter-FCA so kurz wird, dass wir den Sweetspot nicht erreichen, sollte die Anzahl an neugefundenen Elementen, an gelösten LPs nur noch langsam wachsen.

Zwei Phasen: Eine Hauptphase linearen Wachstum, eine gebremste Abschlussphase.

Doch ein Blick auf die Abbildungen B.14 und auch B.15 spricht eine andere Sprache.

Ja, es gibt eine Hauptphase linearen Wachstums (mehr oder weniger); wenn man sie denn sehen will, dann erkennt man auch eine gebremste Abschlussphase.

Aber vor allem sieht man eine Beschleunigungsphase, eine Linkskurve im Anfang der EFCA. Ganz am Anfang lösen wir weniger LPs als wir später müssen. Das gilt nun nicht nur bei der EFCA für die LPs, sondern auch für die tEFCA und ihre MILPs.

Wir wissen, dass es nicht an besonders vielen vorliegenden Zeugen liegen kann. Das haben wir in Abbildung B.11 bereits ausgeschlossen. Ganz im Gegenteil: Diese Abbildung würde uns sogar prognostizieren, dass wir in den Spalten *all* und *FC* ein gebremstes, nicht-lineares Wachstum beobachten sollten. Da wir dort im Laufe des Verfahrens immer mehr Zeugen als Startgrundlage haben, sollten wir immer weniger (MI)LPs lösen müssen.

Die EFCA gibt uns also noch ein großes Rätsel mit auf den Weg. Des Pudels Kern gilt noch gefunden zu werden. Ein interessanter Ausgangspunkt für künftige Untersuchungen.

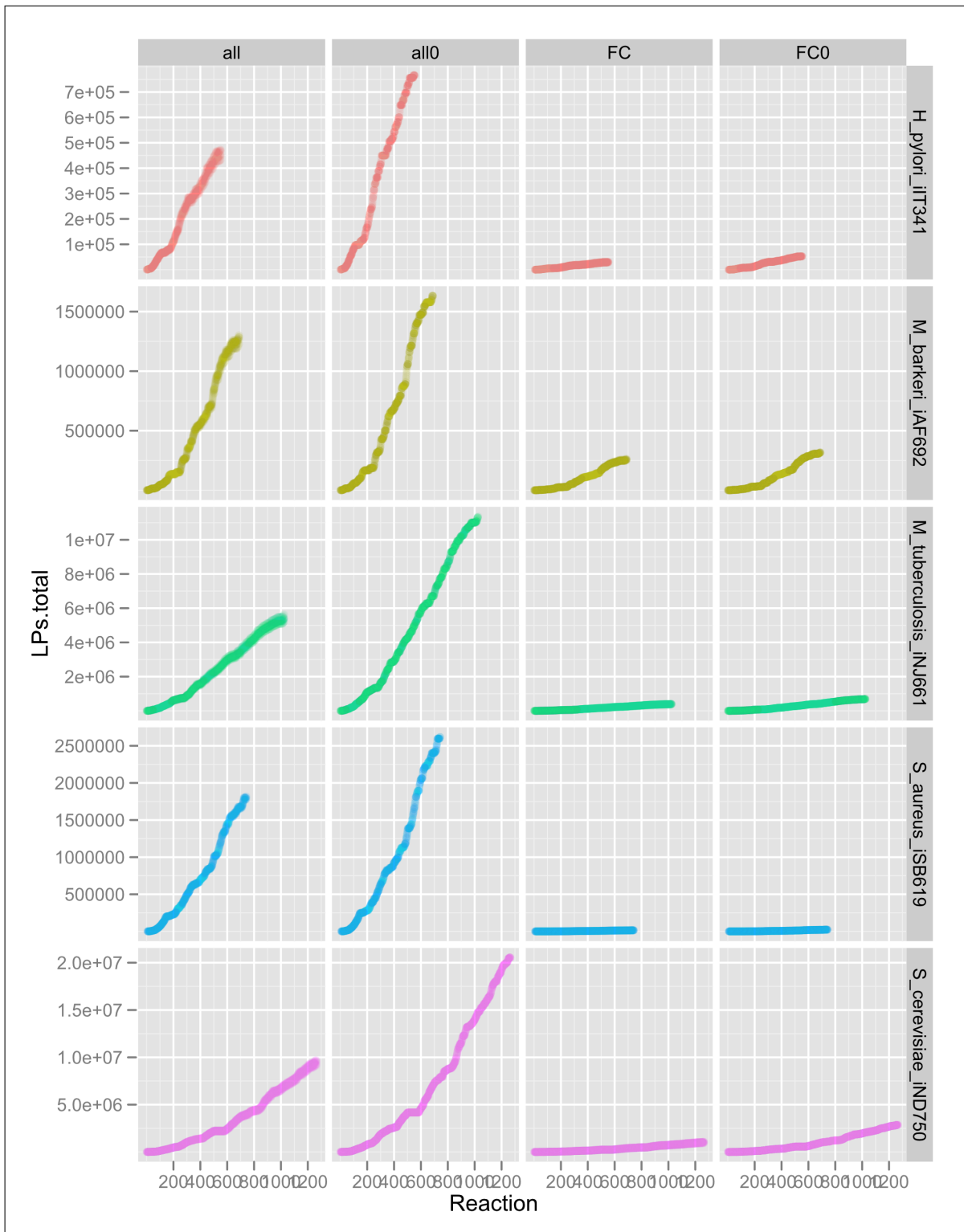


Abbildung B.14.: Gelöste LPs/MILPs zur Berechnung der ersten i Maxima $1_{L_{\perp(i)}}$.

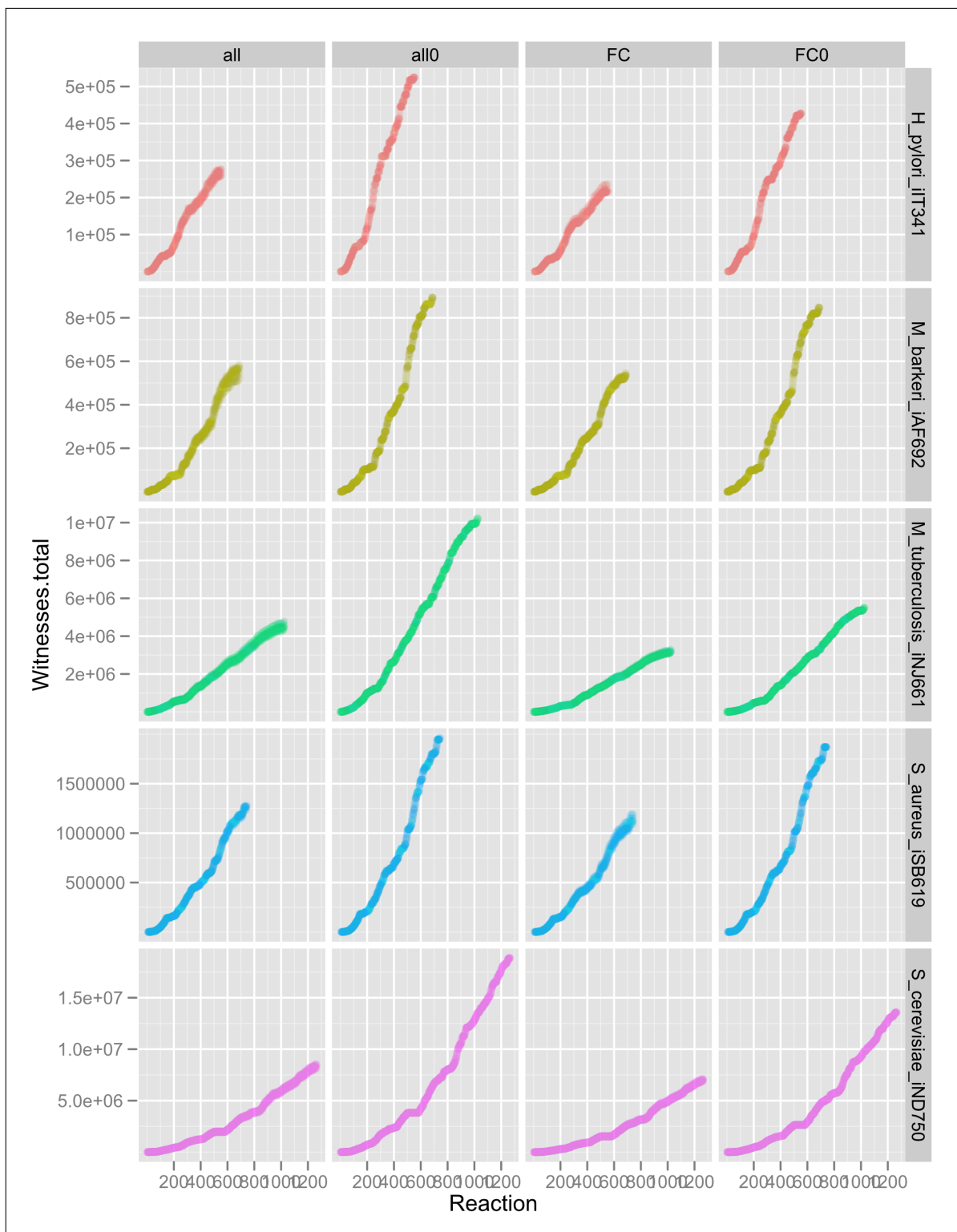


Abbildung B.15.: Gefundene Elemente zur Berechnung der ersten i Maxima $1_{L_{\perp}(i)}$.

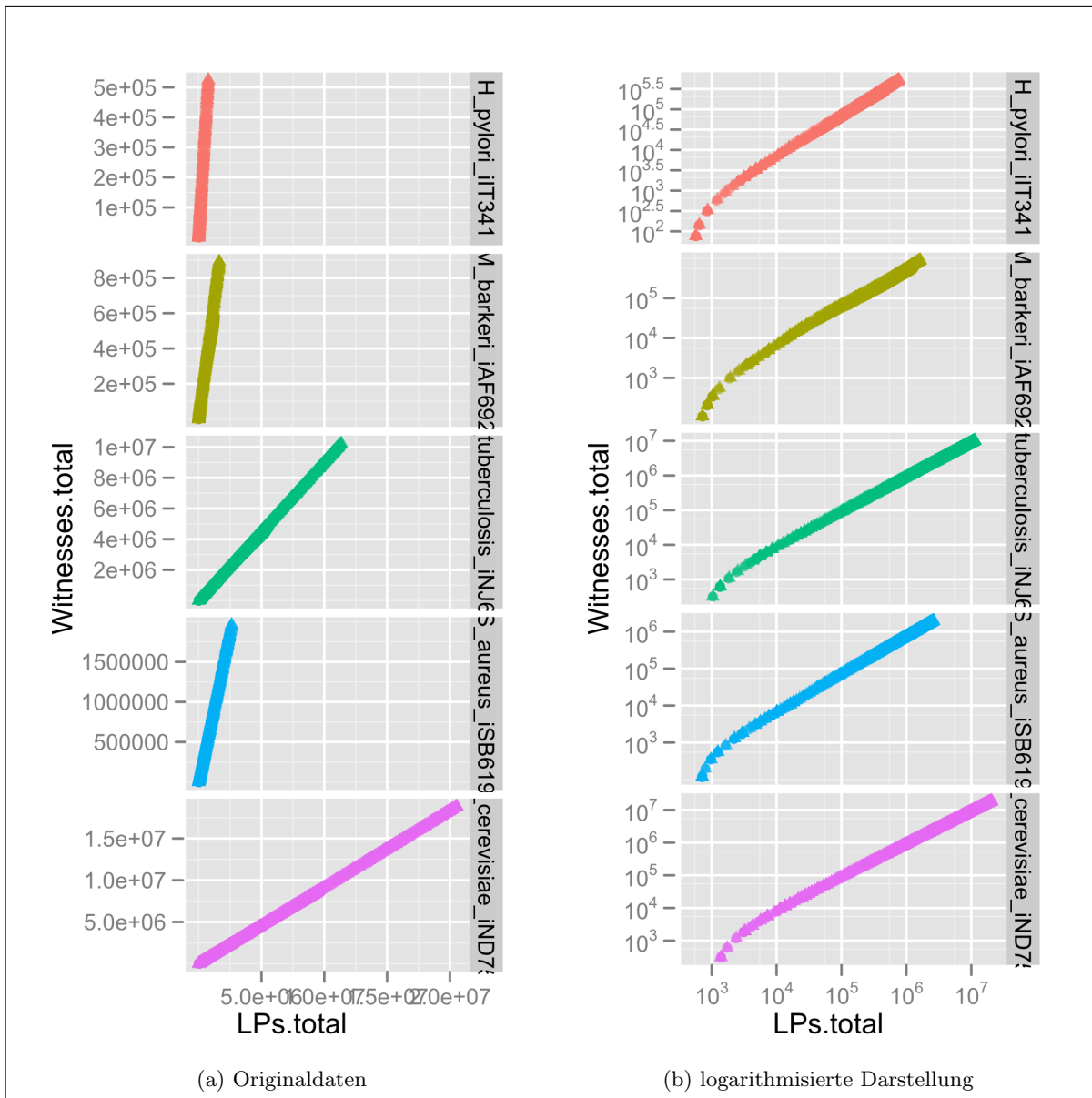


Abbildung B.16.: Anzahl gefundener Verbandselemente (Zeugen) über Anzahl gelöster LPs.

§B.4 Zur Testumgebung

Die Tests wurden allesamt unter einem Windows 7 Professional, Service Pack 1 (64-bit) System mit .NET Framework 4.0.30319 durchgeführt. Als (MI)LP-Solver wurde Gurobi 5.0.0 in der win64-Version genutzt und über die mitgelieferten .NET-dlls aufgerufen.

Hardware-seitig standen ein Intel Core i7-2600 (3.4 GHz, 4 Kerne, Hyperthreading) sowie 4 GB Arbeitsspeicher zur Verfügung.



Kapitel B – Kurz und knapp

Beobachtungen

- ▷ Die Komplexität der Einzelphasen entspricht der Erwartung. *FCA* ist nur um eine Größenordnung komplexer als die Bestimmung des Maximums.
- ▷ Die Vorhersage, die *EFCA*-Phase hat einen quadratischen Zeitverlauf, wurde bestätigt.
- ▷ Während der *FCA* werden Zeugen ohne Reaktionenpräferenz berechnet. Die meisten gefundenen Stoffwechselfade sind kürzer als 150 Reaktionen und kreisfrei.

Erkenntnisse

- ▷ Ein intelligenter Einsatz von Zeugen erlaubt wirklich schnelle (E)*FCA*.
- ▷ Es gibt wohl einen *Sweetspot* an bekannten Elementen, die ausreichen, um schnell die Maxima der *FCA* zu berechnen.
- ▷ Unsere Zeugenverwaltung verbraucht zu viel Rechenzeit gemessen am Zeitbedarf von Gurobi.



ANHANG C

Was übrig bleibt

Auch der praktische Teil dieser Arbeit neigt sich nun dem Ende zu. Zeit, zu reflektieren.

Zu reflektieren, wie es weiter geht. In der Praxis.

§C.1 Was zu testen bleibt

Wir können schnell Maxima berechnen und damit Flusskopplung auffinden. Das haben wir diskutiert, das haben wir implementiert, das haben wir laufzeitanalysiert.

Was zu testen bleibt, ist, wie gut unsere Vorhersagen mit Experimentalbeobachtungen zusammenpassen. Bei welchen der getesteten Modellen gibt es große Übereinstimmungen? Bei welchen noch nicht getestet? Bei welchen nicht? Fehlen in unserem Modell etwa noch Reaktionen? Oder stimmt die Grundannahme stationärer Zustände ganz einfach nicht? Was schlägt sich besser: Kreisfreiheit oder kein Verbot von Kreisen? Hier haben wir getestet, ob unser Algorithmus bei alltagstauglichen Modellen terminiert – die Qualität der Ergebnisse stellen wir nicht auf die Probe. Das Überlassen wir lieber den Profis aus den Bio-Wissenschaften.

Welche neuen Erkenntnisse gewinnen wir tatsächlich aus der EFCA? Bei welchen Fragen verhilft sie möglicherweise zu einem Durchbruch?

Auch Optimierungsprobleme haben wir vorgestellt, aber bisher noch nicht untersucht. Denn hier ist es umso wichtiger, Hand in Hand mit Praktikern vorzugehen. Welche Zellkulturen wollen wir untersuchen? Welchen Stoffwechsel wollen wir beeinflussen und was wollen wir dabei erreichen? Welche Reaktionen können wir überhaupt blockieren, wozu sind wir technisch in der Lage?

§C.2 Was zu verbessern bleibt

Wie schaut's denn nun aus: Sollten wir uns so viele Zeugen merken wie möglich oder nicht? Wie viele müssen wir wirklich kennen? Und wie speichern wir sie?

Derzeit wissen wir, dass wir nie ein Element berechnen werden, das wir gespeichert haben. Wir arbeiten mit einer Liste, an der wir uns langhangeln, wenn wir `lower_bound` für das Maximum berechnen. Aber geht das nicht auch intelligenter?

Wie wäre es, mit anderen Datenstrukturen zu arbeiten: Jedes bekannte Element $a \in L$ wird Listen L_i zugeordnet, wenn $i \in a$ gilt. Oder $i \notin a$. Wenn wir mit Zeigern arbeiten statt mit Kopien der Objekte kriegen wir möglicherweise den Speicherhunger in den Griff. Oder vielleicht auch nicht, bei n sehr groß. Wie schaut es aus mit anderen Datenstrukturen?

Als Hauptlaufzeitbedarf hatten wir vor der Implementierung das Lösen der LPs ausgemacht. Die Wahl eines modernen Solvers wie Gurobi, so dachten wir, wird schon dafür sorgen, dass uns die Rechenzeit nicht um die Ohren fliegt. Wie sich herausgestellt hat, sollten wir Recht behalten. Wir müssen nur noch so wenige lineare Programme lösen (ob kontinuierlich oder teils ganzzahlig), dass ein kleiner Eingriff in unsere Zeugenverwaltung der EFCA massiv Auswirkung auf die Laufzeit hat. Ein Beweis, dass wir unsere Implementierung überdenken sollten.

C# ist eine Programmiersprache, die sehr bequeme Implementierungen ermöglicht. Verbände haben eine Halbgruppenoperation $+$: $L \times L \rightarrow L$, und so haben wir auch unsere `Fluxpattern`, unsere Flussmuster implementiert. Das ist bequem und einfach zu nutzen, aber es kostet Rechenzeit. Dort ist noch sehr viel herauszuholen, weil wir so viele Elemente zu unteren Schranken aufaddieren.

Diese Problemzonen kennen wir bereits. Jetzt müssen wir uns ehrlich gegenüber sein und fragen: Welche Schlaglöcher haben wir uns sonst noch in die Laufzeit gerissen?

Doch Rechenzeit allein ist noch nicht alles. Unsere Algorithmen sind nun konzeptionell schnell genug, dass wir Luxuswünsche stellen dürfen.

Ein Wunsch, der eigentlich nicht Luxus sein darf, ist die Zuverlässigkeit der Ergebnisse. Gurobi lieferte uns bei den Kreisen von `E_coli_iJ01366` einige böse Überraschungen, die uns unmittelbar auffielen. Doch was für Rechenfehler verbergen sich sonst noch in den Datenbergen, die wir produziert haben, und bisher noch unsichtbar sind?

Gurobi ist einer der besten (MI)LP-Solver, aber es gibt auch andere. Deshalb sollten wir auch mit den anderen arbeiten und die Ergebnisse vergleichen. Fehler werden wir nicht ausschließen können. Aber wir sollten genau aufpassen, wo wir sie selbst vermeiden können. So stellen etwa Cook, Koch, Steffy & Wolter (2011) ein vielversprechendes Verfahren vor, um MILPs exakt zu lösen.

§C.3 Was zu verstehen bleibt

Viele unserer Beobachtungen entsprachen unseren Erwartungen, konnten wir gut erklären oder sogar vorhersagen. Dennoch gab es einige Überraschungen.

Vor allem der *Sweetspot* stellt uns vor Rätsel. (Warum) gibt es für jedes Modell eine bestimmte Anzahl an Zeugen, an Stoffwechselfaden, die man kennen muss, um künftig nur noch sehr wenige Zusatzinformationen zum Bestimmen der maximalen Flusspfade ermitteln zu müssen? Oder sehen unsere Graphen nur zufällig so aus, weil es generell einen exponentiell gebremsten Erkenntnisgewinn gibt, der nur gerade in unserem Fall besonders scharfkantig aussieht?

Hat es etwas damit zu tun, was für Stoffwechselfade Gurobi bestimmt? Ist es nur Zufall, der sich bei anderen Modellen nicht wiederholen würde? Gibt es vielleicht doch eine biologische Erklärung dafür, weshalb das Wissen über eine (gemessen an Elementarmoden) geringe Anzahl an gültigen Flussmustern bereits gewaltige Beschreibungsmacht über Flusskopplung hat?

Wieso verhalten sich die Graphen bei der EFCA so außerordentlich? So außerordentlich gut vorhersagbar, was die Laufzeit betrifft, aber so außerordentlich seltsam, wenn es um die Zahl zu lösender linearer Probleme geht?

Unsere einfachen Berechnungen haben uns schon vor diese zwei neuen Frageblöcke gestellt, wenn wir die Theorie nur ein wenig praktisch umsetzen. Realisieren wir noch mehr Verbandstheorie, stoßen wir sicherlich auf noch viele weitere interessante Probleme.

§C.4 Was zu sagen bleibt

In diesem letzten Abschnitt gab es keine Feststellungen. Keine Beweise, keine Schlussfolgerungen, keine Grundannahmen. Wir haben Fragen gestellt. Wir haben Fragen gestellt, denn darum dreht sich die Wissenschaft. Jede Antwort führt zu neuen Fragen.

Auf manche findet man Antwort. Manche revolutionieren die Welt. Manche sind Zahnrad im Getriebe, das schließlich Therapien für Krankheiten produziert.

Auf manche findet man keine. Manchmal erst spät, manchmal nie.

Mathematik hilft, Antworten zu finden, die immer gültig sein werden, solange die Grundannahmen bestehen bleiben.

Biologie beschreibt das Leben. Rettet es manchmal.

Wissenschaft stellt Fragen und bleibt dabei niemals stehen. Was übrig bleibt, sind meistens mehr Fragen als das, womit man gestartet ist. So ist es auch uns passiert.

Verbandstheorie hat uns auf einen Weg geführt, dessen Ziel wir vorher nicht kannten.

Wir wollten mit Mathematik Stoffwechselanalyse vereinheitlichen. Ein Framework schaffen, das einfach zu verstehen und zu bedienen ist. Das Komplexität vermeidet, so oft wie möglich anwendbar ist. Uns einfache Werkzeuge zur Verfügung stellt und neue Erkenntnisse bringt.

Verbandstheorie hat uns auf einen Weg geführt, der uns zu vielen neuen Erkenntnissen gebracht hat. Der uns zeigt, wie man mit einem Ansatz stationäre Zustände genauso beschreiben kann wie ein logisches Modell und dabei Ressourcenmangel oder thermodynamische Regeln so einfach hinzufügen kann wie Zucker und Milch zum Kaffee.

Die Verbandstheorie hat uns Algorithmenentwürfe näher gebracht, bei denen schwer erkämpfte Verbesserungen vieler Jahre wie selbstverständlich vom Himmel fallen, und zu Abwandlungen, die Methoden wie die EFCA ermöglichen, für die es dank Maximumsuntersuchung jetzt ein natürliches Konzept gibt.

Immer wieder hat sie uns zu neuen Fragen geleitet. Fragen, auf die wir jetzt einige Antworten kennen, aber auch Fragen, die wir noch untersuchen sollten. Manchmal haben wir schon Ideen, wie die Antworten lauten könnten. Doch irgendwann mussten wir den Schlussstrich ziehen, ein Kapitel beenden, diese Arbeit abschließen. Denn auch diese Antworten hätten zu neuen Fragen geführt, die wir beantworten wollen. Vermutlich wieder mit Verbandstheorie.

Immer wieder war vom großen Bild die Rede. Das große Bild, das die Welt der diskreten Stoffwechselmodellierung zeigt. Die Verbandstheorie war uns am Ende viel, viel mehr als bloßer Rahmen.

Die meisten Publikationen über qualitative Eigenschaften – seien es nun Zerlegung von Pfaden in Elementarmoden, Minimal Cut Sets, Flusskopplung oder viele andere – laufen schließlich auf minimale, unzerlegbare oder maximale Elemente im Verband hinaus. Weil die Verbandstheorie so perfekt dafür geeignet ist, Stoffwechsel zu abstrahieren, erklärt sich auch auf einen Schlag, warum es keine wesentlich anderen Konzepte gibt. Alle qualitativen Anwendungen laufen letztlich auf diese Höflinge im Verband hinaus, weil sie die wichtigsten und auffälligsten Elemente der Verbandstheorie sind. Der Rahmen erklärt das Bild.

Und wenn auch du, werter Leser, wissen willst, wohin sie uns noch führen wird, dich das Entdeckungsfieber gepackt hat – dann hast auch du sicher die Magie gespürt, zu der die Verbandstheorie, zu der die Mathematik im Stande ist.

Glossar

Abschluss

Maximaler Vereinigungsverband über $[n]$, der $\mathcal{M}(L)$ als Menge minimaler Elemente enthält. Es ist $\tilde{\alpha} \subseteq [n]$ genau dann in \bar{L} , wenn $\alpha \leq \tilde{\alpha}$ für ein $\alpha \in L$. 55

Antikette

Eine Teilmenge einer partiell geordneten Menge, bei der je zwei Elemente unvergleichbar sind. Im Falle des Metabolismus also eine Menge an Stoffwechselfaden, bei der kein Pfad in einem anderen enthalten ist. 33, 35, 39, 66

Atom

s. minimales Element. 35

atomar

s. atomarer Verband. 71

atomarer Verband

Gilt $\mathcal{M}(L) = \mathcal{J}(L)$, so heißt L *atomarer* Verband. 37

Aufspaltung

Unter der Aufspaltung reversibler Reaktionen versteht man eine Modifikation des Netzwerkes, in dem man eine reversible Reaktion durch zwei gegenläufige Reaktionen gleicher Stöchiometrie ersetzt. Die beiden möglichen Reaktionsrichtungen werden dadurch strikt voneinander getrennt und als verschiedene Prozesse betrachtet.

Das Aufsplitten reversibler Reaktionen vereinfacht die Formulierung und Untersuchung vieler Algorithmen, erschwert sie aber für das Gesamtmodell. Oftmals wird daher in der Modellbeschreibung mit reversiblen Reaktionen gearbeitet, für die Algorithmen jedoch in einem Preprocessing eine Aufspaltung in die entsprechenden irreversiblen Reaktionen durchgeführt. Ein Postprocessing überträgt dann die Ergebnisse auf das Ursprungsmodell. 13, 141

Biomasse

Ein (hypothetischer) Metabolit, der das erfolgte Wachstum einer Zelle symbolisiert. Dient meist als Grundlage der FBA. 22, 23

blockierte Reaktion

Reaktion $i \in [n]$, die an keinem gültigen Stoffwechselfad beteiligt ist.

i ist blockiert genau dann, wenn $i \notin 1_L$.

Bei der FCA wird untersucht, was passiert, wenn man eine Reaktion gezielt blockiert.

Es ist $i \xrightarrow{=0} j$ genau dann, wenn j blockiert ist in $1_{L_{\perp\{i\}}}$. 90

dualer Verband

Erzeugnis der minimalen Überdeckungen von L . Es ist $(L^\perp)^\perp = L^\circ$.
Unterscheidet sich vom dualen Verband im Sinne von Kategorien. 64

EFM

s. elementarer Flussmodus. 24

elementarer Flussmodus

Flussvektor mit kleinst möglichem, nicht-trivialen Träger.
 $e \in F$ ist elementarer Flussmodus genau dann, wenn $\text{supp } e \in \mathcal{M}(L^F) = \mathcal{J}(L^F)$. Deshalb bestimmen die elementaren Flussmoden die meisten Eigenschaften des Flusskegels/-verbandes wie Flusskopplung und Überdeckungen/Minimal Cut Sets.
Elementarmoden sind bis auf Vielfache eindeutig durch ihren Träger bestimmt. 24, 77

Elementarmodus

s. elementarer Flussmodus. 24

entlang R

Unterhalbverband von L , der außer dem neutralen Element 0 genau die Elemente enthält, die mindestens eine Zahl aus der korrespondierenden Menge R enthalten. 55

ergänzender Summand

Gibt es zu $\alpha_1 < \alpha$ ein nicht-triviales Element $\alpha_2 < \alpha$ mit $\alpha_1 + \alpha_2 = \alpha$, so ist α_2 ergänzender Summand. Alle Verbände, bei denen es zu jedem Paar $\alpha_1 < \alpha$ ergänzende Summanden gibt, sind atomar. 37, 70, 137

Erzeuger

Eine Teilmenge $B \subseteq L$, deren Erzeugnis $\langle B \rangle$ ganz L ist. 38, 39, 85

Erzeugnis

Das Erzeugnis $\langle A \rangle$ einer Teilmenge $A \subseteq L$ entspricht dem kleinsten Unterhalbverband von L , der A enthält. Es besteht aus den Elementen, die sich als Summe von Elementen aus A schreiben lassen. 38, 45

essenzielle Reaktion

Reaktion $i \in [n]$, die in jedem gültigen Stoffwechselfad (außer dem leeren) vorkommt. 91, 138

FBA

s. Flux Balance Analysis. 23

FCA

s. Flusskopplung. 26, 89, 96, 108, 143

Flusskegel

Menge aller Flussvektoren im stationären Zustand, also die Lösungsmenge von $Sv = 0$, $v_{\text{Irr}} \geq 0$. 20, 77

Flusskopplung

Betrachtet die gegenseitigen Abhängigkeiten von Reaktionen im Verband oder in Unterstrukturen.

Die Reaktion j ist an i gekoppelt, wenn alle Stoffwechselfade, die j verwenden, auch i verwenden. Flusskopplung ist eindeutig durch die unzerlegbaren Elemente bestimmt ebenso wie durch die Maxima der Unterverbände $L_{\perp\{i\}}$.

Man unterscheidet zwischen einfacher, teilweiser sowie vollständiger Kopplung. 26, 92, 96

Flussmuster

Träger eines Flussvektors. 20

Flussvektor

Die Elemente aus dem Flusskegel F . Es gilt $v \in F \Leftrightarrow Sv = 0, v_{\text{Irr}} \geq 0$. 20

Flussverband

Menge aller Träger der Flussvektoren in F . 20, 21, 77

Flux Balance Analysis

Mathematisches Verfahren, das nach Flussvektoren im (beschränkten) Flusskegel sucht, die eine gegebene Wachstumsfunktion – meist Biomasseproduktion – maximiert. Dies soll den optimalen stationären Zustand darstellen, den eine Zelle annehmen kann.

Zur Berechnung wird meist ein LP gelöst. 23

Flux Coupling Analysis

s. Flusskopplung. 26

gekoppelte Reaktion

s. Flusskopplung. 92

Halbverband

Endliche, idempotente Halbgruppe. Hat sie ein neutrales Element 0 , definiert sie eindeutig einen Verband. Deshalb können wir in dieser Arbeit Verbände untersuchen, auch wenn wir nur die Halbverbandseigenschaft zeigen, solange der leere Stoffwechselfad auch möglich ist. 7

Hauptideal

Spezieller Unterverband von L , der durch Verknüpfung aller Elemente $a \in L$ mit seinem Maximum $a \circ 1_U$ bestimmt werden kann.

Im endlichen Verband ist jedes Ideal Hauptideal. Damit sind Hauptideale auch Ordnungsideale.. 41–43, 45, 47, 56, 132

Hyperpfad

Geordnete Menge an Reaktionen $\mathbf{p} = (\alpha_1, \dots, \alpha_t)$, bei denen alle Metaboliten, die durch Reaktion α_k abgebaut werden, durch die vorangegangenen Reaktionen produziert worden sind. Das Ergebnis ist ein schrittweiser Pfad durch den Stoffwechsel.

Der induzierte Verband $L^{\mathcal{J}^c}$ ist ein Unterhalbverband von $L^{\rightarrow 0^1}$. 15, 16, 62, 133

Ideal

s. Hauptideal. 132

in R

Hauptideal in L, das genau die Elemente enthält, die Teilmengen der korrespondierenden Menge R sind. 56

Innere

Erzeugnis der minimalen Elemente und somit atomar.

Bestimmt eindeutig die Überdeckungen und somit den dualen Verband L^\perp . 47, 55

irreduzibles Element

s. unzerlegbares Element. 36

Kreis

Stoffwechselfad, der keine Austauschreaktionen enthält, also lediglich interne Metaboliten ineinander umwandelt. Aus thermodynamischer Sicht sind solche Pfade nicht als stationärer Zustand möglich. 136, 140, 143

kreisfrei

Ein Stoffwechselfad, der keinen Kreis als Unterpfad enthält, heißt *kreisfreier* Pfad. 137, 141

kreisfreier Modellverband

Verband L_{-C} , der von den kreisfreien Stoffwechselfaden erzeugt wird.

P_{-C} ist der Verbandsstumpf aller kreisfreien Pfade.

Für die thermodynamisch gültigen Pfade \mathcal{T} gilt $P_{-C} \subseteq \mathcal{T} \subseteq L_{-C}$. 137

logisches Modell

Modell, das man als logische Formel C^{01} formulieren kann. Gültige Stoffwechselfade entsprechen gültigen Belegungen von C^{01} , also $C^{01} = \mathbf{true}$.

Sofern nicht anders genannt ist eine Belegung logisch erfüllbar, wenn Metaboliten genau dann produziert werden, wenn sie auch abgebaut werden. 10, 11

logischer Verband

Verband, dem ein logisches Modell zu Grunde liegt.

Sofern nicht anders genannt, handelt es sich um L^{01} basierend auf C^{01} . 11, 21

LP

Lineares Programm. Mathematische Formulierung eines Optimierungsproblem in Form

eines Kostenvektors $c \in \mathbb{R}^n$ und eines Polyeders P .

Ist in der Komplexitätsklasse \mathcal{P} enthalten und kann von LP-Solvern in der Regel sehr „schnell“ gelöst werden. 67, 69, 145

Maximum

Eindeutig bestimmtes maximales Element im Verband. Lässt sich als Summe aller unzerlegbaren Elemente und somit insbesondere als Summe aller Elemente des Verbandes schreiben. 30, 31, 89, 91, 97, 106, 108, 131

metabolisches System

Vereinigungsverband über die n Reaktionen mit Aufteilung $[n] = \text{Rev} \dot{\cup} \text{Irr}$. Dient als Grundlage für unsere Untersuchungen des Stoffwechsels. 8

metabolische Bedingungen

Informationen über die Stöchiometrien der Reaktionen sowie die Reversibilitäten der Reaktionen. Definieren eindeutig den Flusskegel, der alle Zustandsvektoren im stationären Zustand umfasst, sowie den logischen Verband. 9

Metabolit

Stoffwechselprodukt. Kann Zwischenprodukt in einem Stoffwechselfad sein oder Molekül, das in die Zelle transportiert wurde oder in ihr für externen Gebrauch hergestellt wird. Daher unterscheidet man zwischen *internen* Metaboliten, die in der Zelle vorkommen, und *externen* Metaboliten, die die Prozesse in der Zelle beeinflussen oder ermöglichen können.

Sofern nicht anders genannt, gehen wir von m internen Metaboliten aus. 3, 4

MILP

Lineares Programm, das auch ganzzahlige Variablen enthält. Das Lösen allgemeiner MILPs ist \mathcal{NP} -schwer, ist durch geeignete MILP-Solver jedoch meist „schnell“ möglich. 69, 92–95, 143

Minimal Metabolic Behaviour

Erzeugendensystem des Flusskegels, das auf eine Aufspaltung des Kegels in linearen Unterraum und Facetten besteht.

Es gilt $\mathcal{M}(L_{\text{Irr}}^F) = \mathcal{J}(L_{\text{Irr}}^F) = \mathcal{MMB}$. 86, 87

minimales Element

Nach Verbandsordnung kleinstes, nicht-triviales Element. Die Menge der minimalen Elemente bestimmen in Vereinigungsverbänden eindeutig den dualen Verband.

Alle minimalen Elemente sind automatisch auch unzerlegbar. In atomaren Verbänden gilt $\mathcal{M}(L) = \mathcal{J}(L)$. 24, 35

Null-Verband

Verband, der durch den Kern einer (stöchiometrischen) Matrix induziert wird.

Es ist $L^F \subseteq L^{F,0}$ Unterhalbverband. 83, 94, 97

ohne R

Hauptideal in L , das genau die Elemente enthält, die keine Zahl aus der korrespondierenden Menge R enthalten.

Entspricht dem metabolischen Verband, den man erhält, wenn die Reaktionen in R blockiert werden. Daher ermöglicht die Untersuchung von $L_{\perp R}$ Erkenntnisgewinn über Flusskopplung. 56

Ordnung

Jeder Halbverband verfügt über eine durch die Operation induzierte partielle Ordnung. Es gilt $a \leq b \Leftrightarrow a + b = b$. 30, 31

Ordnungsideal

$P \subseteq L$ ist Ordnungsideal, wenn aus $a_2 \in P$ automatisch $a_1 \in P$ folgt für alle $a_1 \leq a_2$

P ist Ordnungsideal genau dann, wenn man P als Verbandsstumpf $P = P_{\varphi \leq t}$ schreiben kann.

Für Informationen über die minimalen und unzerlegbaren Elemente des von einem Ordnungsideals erzeugten Verbands $\langle P \rangle$ s. Starbucks-Lemma. 132, 133

Projektion

Verbandshomomorphismus $P_R : L \rightarrow L_R$ mit L Vereinigungsverband, der jedes Element auf seine Elemente in $R \subseteq [n]$ projiziert, also $P_R : a \mapsto a \cap R$. 59

Reaktion

Dreh- und Angelpunkte dieser Arbeit. Biochemische Reaktionen wandeln Metaboliten in bestimmten Verhältnissen ineinander um und ergeben so gemeinsam den gesamten Stoffwechsel.

Wir unterscheiden *irreversible* Reaktionen, die nur in eine Richtung ablaufen können, von *reversiblen* Reaktionen. Zudem wird manchmal noch in Transport-/Austauschreaktionen sowie interne Reaktionen unterteilt, die Metaboliten in oder aus der Zelle transportieren bzw. nur interne Umwandlungen vornehmen.

Sofern nicht anders genannt gehen wir von n Reaktionen aus und unterteilen in $[n] = \text{Rev} \cup \text{Irr} = \text{Trans} \cup \text{Int}$. 3, 4

irreversible Reaktion

Reaktion, die nur in Vorwärtsrichtung möglich ist (im Gegensatz zur reversiblen Reaktion). 6, 13

reversible Reaktion

Reaktion, die in Vorwärts- und Rückwärtsrichtung möglich ist (im Gegensatz zur irreversiblen Reaktion). 6, 13

SAT

SAT, kurz für *satisfiability*, ist der Name des Entscheidungsproblems, ob eine logische Formel eine gültige Belegung besitzt, also erfüllt werden kann.

SAT ist ein in der Komplexitätstheorie sehr bekanntes Problem, da es das erste war, das bewiesenermaßen \mathcal{NP} -vollständig ist. D. h. jedes Problem in \mathcal{NP} kann in polynomieller

Zeit gelöst werden, sollte ein polynomieller Algorithmus für das Entscheidungsproblem SAT gefunden werden. 14, 101

stationärer Zustand

Zustand einer Zelle, in dem die einzelnen Reaktionsgeschwindigkeiten so aufeinander angepasst sind, dass die Metabolitenkonzentrationen in der Zelle konstant bleiben. Also alle Metaboliten genauso häufig abgebaut werden, wie sie produziert werden. Die Menge aller stationären Zustände bilden den Flusskegel F . 5

stöchiometrische Matrix

Matrix, deren Komponenten Aufschluss über die Beteiligung der Metaboliten an verschiedenen Reaktionen liefern. Sofern nicht anders genannt, gilt $S \in \mathbb{R}^{m \times n}$. 8, 9

Stoffwechselfad

Kombination an Reaktionen, die zusammengenommen einen „sinnvollen“ Stoffwechselfad ergeben. Die Überlegungen, was eine sinnvolle Kombination sein soll, sind Grundlage des Verbandes, den wir als Modell zur Untersuchung nutzen. 5, 6, 13

Träger

Indizes der Nicht-0-Komponenten eines Vektors. Z. B. $\text{supp}(2, 0, 3)^\top = \{1, 3\}$. 20, 78

über R

Unterhalbverband von L , der außer der 0 genau die Elemente enthält, die Obermengen der korrespondierenden Menge R sind. 55

Überdeckung

Teilmenge $c \subseteq [n]$, die jedes Element $a \in L \setminus \{0\}$ schneidet. Die Menge aller Überdeckungen $\overline{C}(L)$ bildet einen Verband und ist eindeutig durch $\mathcal{M}(L)$ definiert.

Es ist $\overline{C}(L) = \overline{L}^\perp$ Abschluss des dualen Verbandes von L . 63–65, 123

Unterhalbverband

Eine Teilmenge $U \subseteq L$, die bezüglich der Addition abgeschlossen ist.

Jeder Unterhalbverband, der die 0 enthält, ist selbst wieder ein Verband und es gilt $a_1 \circ_U a_2 \leq a_1 \circ a_2$. Gilt stets Gleichheit, so ist U Unterverband. 40, 42, 54

Unterverband

Eine Teilmenge $U \subseteq L$, die bezüglich der beiden Verbandsoperationen $+$ und \circ abgeschlossen ist. Jeder Unterverband ist Unterhalbverband. 40, 42, 132

unzerlegbares Element

Verbandselement, das sich nicht als Summe kleinerer Elemente darstellen lässt. Die Menge aller unzerlegbaren Elemente ist der eindeutig bestimmte minimale Erzeuger des Verbandes. Bestimmen viele Eigenschaften des Verbandes, z. B. Flusskopplung. Die minimalen Elemente sind eine Teilmenge der unzerlegbaren Elemente. In atomaren Verbänden gilt $\mathcal{M}(L) = \mathcal{J}(L)$. 36

Verband

Endliche, idempotente Halbgruppe mit Addition $+$, neutralem Element 0 sowie Maximum 1 . Definiert eindeutig eine zu $+$ duale Zweitoperation \circ . Elemente repräsentieren gültige Stoffwechselfade. Weiterhin definiert $+$ eine partielle Ordnung auf L und es gilt $\mathbf{a} \leq \mathbf{b} \Leftrightarrow \mathbf{a}_1 + \mathbf{a}_2 = \mathbf{a}_2$. 19, 30, 32

Verband der Hyperpfade

Der von Hyperpfaden induzierte Verband.

Es ist $\mathbf{a} \in L^{\mathcal{H}}$, wenn es einen Hyperpfad $\mathbf{p} = (\mathbf{a}_1, \dots, \mathbf{a}_t) \in \mathcal{H}$ gibt mit $\mathbf{a} = \{|j| \mid j \in \mathbf{p}\}$.
16

Verbandsstumpf

Teilmenge von L , deren Bild unter der monoton steigenden Funktion φ durch \top beschränkt ist. Das Erzeugnis $L_{\varphi \leq \tau} = \langle P_{\varphi \leq \tau} \rangle$ ist Unterverband von L und es gilt $\mathcal{M}(L_{\varphi \leq \tau}) = \mathcal{M}(L) \cap P_{\varphi \leq \tau}$ sowie $\mathcal{J}(L_{\varphi \leq \tau}) = \mathcal{J}(L) \cap P_{\varphi \leq \tau}$.

Jeder Verbandsstumpf ist vererbliche Eigenschaft und umgekehrt. 45–47, 133, 141

Vereinigungsverband

Teilmenge der n -Menge $[n]$, die die leere Menge enthält und abgeschlossen ist bzgl. Vereinigung.

Es ist $\mathbf{a}_1 + \mathbf{a}_2 = \mathbf{a}_1 \cup \mathbf{a}_2$ und $\mathbf{a}_1 \leq \mathbf{a}_2 \Leftrightarrow \mathbf{a}_1 \subseteq \mathbf{a}_2$. Weiter ist $\mathbf{a}_1 \circ \mathbf{a}_2 = 1_{L_{\subseteq \mathbf{a}_1 \cap \mathbf{a}_2}} \subseteq \mathbf{a}_1 \cap \mathbf{a}_2$ größtes Verbandselement, das Teilmenge des Durchschnitts ist.

Bei der Untersuchung des Stoffwechsels arbeiten wir mit Vereinigungsverbänden, da wir so Stoffwechselfade durch die Indizes ihrer beteiligten Reaktionen repräsentieren können. 54, 58, 68

Zeuge

Einige Verbandseigenschaften, wie die Blockiertheit einer Reaktion oder der Kopplung eines bestimmten Reaktionspaares, sind durch die Nicht-Existenz von Elementen mit gewissen Eigenschaften definiert. Ein Element ist Zeuge für die Nichtgültigkeit dieser Eigenschaft, wenn es sich um so ein Element handelt.

Es ist $i \in [n]$ blockiert in L , wenn $i \notin \mathbf{a}$ für alle $\mathbf{a} \in L$. Betrachte z. B. $\mathbf{a} = \{1, 23, 42\} \in L$. Dann ist \mathbf{a} Zeuge dafür, dass die Reaktionen 1, 23, 42 nicht blockiert sind. 102, 106, 112

Symbolverzeichnis

1_L	Maximum. 31, 89, 91
$\alpha \circ L$	Hauptideal. 41
$\mathcal{B}(L)$	Menge der blockierten Reaktionen. 90, 91
C^{01}	logisches Modell. 11
$\overline{C}(L)$	Menge aller Überdeckungen. 63, 66
E	Menge der elementaren Flussmoden. 77, 78
$\langle A \rangle$	Erzeugnis von $A \subseteq L$. 38
F	Flusskegel. 20, 68
\mathcal{H}	Menge aller Hyperpfade. 15, 16
$i \xrightarrow{=0} j$	Flusskopplung; j ist gekoppelt an i . 26, 92
$i \longleftrightarrow j$	teilweise Kopplung von i und j . 92
$i \longleftrightarrow_{\lambda} j$	vollständige Kopplung von i und j . 92
$\mathcal{J}(L)$	Menge der unzerlegbaren Elemente. 36, 39, 66, 77, 137
Int	Indizes der internen Reaktionen. 136, 137
Irr	Indizes der irreversiblen Reaktionen. 6, 9, 87
\leq	partielle Ordnung der Verbandselemente. 31, 54
\circ	Zweitoperation im Verband. Wird eindeutig durch $+$ definiert. Es ist $\alpha_1 \circ \alpha_2$ das maximale Element, das kleiner als α_1 und α_2 ist. 32, 58
L^{01}	logischer Verband. 11, 40, 91, 101
\overline{L}	Abschluss von L . 55, 66
L^{\perp}	dualer Verband. 64, 66, 123
$L_{\parallel R}$	L entlang R . 55
$L_{F,0}^{\perp}$	Null-Verband. 83, 94
$L^{\mathcal{H}}$	Menge aller Hyperpfade. 16, 62
$L_{\subseteq R}$	L in R . 56
L°	Innere von L . 47, 55, 66

L_{-C}	kreisfreier Modellverband. 137
$L_{\perp R}$	L ohne R. 56, 106, 108
L_R	Projektion von L auf R. 59
$L_{\supseteq R}$	L über R. 55
$L_{\varphi \leq \tau}$	Erzeugnis von $P_{\varphi \leq \tau}$. 45
L^F	Flussverband. 20, 68, 91
(L, Rev)	metabolisches System. 8
$\mathcal{M}(L)$	Menge der minimalen Elemente. 24, 35, 39, 66, 77, 137
\mathcal{M}_C	Menge der minimalen Kreise. 140
\mathcal{MMB}	Menge der Minimal Metabolic Behaviours. 87
\mathcal{NP}	Klasse aller Entscheidungsprobleme, die in polynomieller Zeit verifizierbar sind. 69, 101
\mathcal{P}	Klasse aller Entscheidungsprobleme, die in polynomieller Zeit entscheidbar sind. 14
$P_{\varphi \leq \tau}$	Verbandsstumpf (φ 1-dimensional) oder Durchschnitt mehrerer Verbandsstümpfe ($\varphi : L \rightarrow \mathbb{R}^d$, s. Starbucks-Lemma). 45
P_R	Projektionsabbildung von L nach L_R . 59
Rev	Indizes der reversiblen Reaktionen. 6
S	stöchiometrische Matrix. 9
(S, Irr)	metabolische Bedingungen. 9
supp	Trägerfunktion. 20, 68, 77, 78
\mathcal{T}	Menge der thermodynamisch gültigen Stoffwechselfade. 143
Trans	Indizes der Transportreaktionen. 136, 137
W	Zeugen – Liste bereits bekannter Elemente des (allgemeinsten) Verbandes. 106, 112

Abbildungsverzeichnis

2.1.	Ein metabolisches Netzwerk (<code>etc_purple</code>).	4
2.2.	Ein Stoffwechselfad (<code>etc_purple</code>).	5
2.3.	Ein logischer Stoffwechselfad (<code>etc_purple</code>).	10
2.4.	Beispielnetzwerk <code>adenin_compressed</code> .	21
	(a). Netzwerk von <code>adenin_compressed</code> .	21
	(b). Hasse-Diagramm vom logischen Verband sowie vom stationären Flussverband von <code>adenin_compressed</code> .	21
2.5.	Flusskopplung im System <code>etc_purple</code> .	26
3.1.	Einige Unterstrukturen des Flussverbands zu <code>adenin_compressed</code> .	42
	(a). Flussverband von <code>adenin_compressed</code> .	42
	(b). Ein Unterhalbverband U_1 .	42
	(c). Ein Unterverband U_2 .	42
	(d). Ein Hauptideal U_3 .	42
3.2.	Schnitte von Unterstrukturen von L^{01} zu <code>adenin_compressed</code> .	44
	(a). Schnitt zweier Unterverbände.	44
	(b). Schnitt zweier Hauptideale.	44
3.3.	Ein Verbandsstumpf von $L = L^{01}$ zu <code>adenin_compressed</code> . ($P_{\varphi \leq \tau}$ gerahmt)	46
3.4.	Das Innere von $L = L^{01}$ zu <code>adenin_compressed</code> .	48
3.5.	Ein Verbandshomomorphismus $\varphi = P_{\{1,2,3\}}$ zu <code>adenin_compressed</code> .	51
	(a). Urbilder vom Homomorphismus in <code>adenin_compressed</code> .	51
	(b). Verband L_2 , auf den φ abbildet.	51
	(c). Vergleichsverband $L_{\subseteq \{1,2,3\}}$.	51
4.1.	Einige Unterstrukturen des Flussverbands zu <code>adenin_compressed</code> .	57
	(a). L entlang R .	57
	(b). L über R .	57
	(c). L ohne R .	57
	(d). L in R .	57
4.2.	Beziehungen zwischen Verband L , Projektion $P_R(L)$ und Untersystem $L_{\subseteq R}$.	61
	(a). Beziehungen beim allgemeinen Verband L .	61
	(b). Beziehungen beim atomaren Verband L .	61
6.1.	Die gestrichelte Reaktion ist blockiert in L^F , aber nicht in L^{01} .	92
6.2.	Das komprimierte Netzwerk zu <code>etc_purple</code> .	101
7.1.	Bestimmung des Verbandsmaximums 1_L in wenigen Schritten.	109
	(a). Bestimmung des Maximum von <code>adenin_compressed</code> .	109
	(b). Bestimmung einer blockierten Reaktion.	109
7.2.	Mögliche Reaktionen nach Doppel-Knockout in <code>etc_purple</code> .	113

7.3. Förderung von R_7 im komprimierten Netzwerk von `etc_purple`, $L = L^F$.
 obere Zeile: R – untere Zeile: $1_{L \perp R}$ 122
 (a). Suchbaum des Förderungsalgorithmus ohne `break`. 122
 (b). Indikatoren, die zum Abschneiden von Ästen mittels `break` genutzt werden können. 122
 (c). Suchbaum von `FindHighlightingSets` mit `break`-Aufrufen. 122

8.1. Ein Hyperpfad, wo nicht jede Teilmenge wieder Hyperpfad ist. 136
 8.2. Kreisfreie Pfade, Modellverband und Flussverband im Vergleich. 140
 (a). Netzwerk zur Illustration kreisfreier Stoffwechselfade. 140
 (b). Flussverband L^F und kreisfreier Modellverband $L-C$ des Beispielnetzwerks. 140
 8.3. Vergleich Flusskopplung mit/ohne Kreise für das Netzwerk 8.2a. 140
 (a). Flusskopplung im Flussverband. 140
 (b). Flusskopplung im kreisfreien Modellverband. 140

B.1. Relativer Aufwand der verschiedenen Etappen der FCA. 158
 (a). Gelöste LPs der FCA ohne Ausschluss von Kreisen (relativer Median) . . 158
 (b). Zeitaufwand der FCA ohne Ausschluss von Kreisen (relativer Median) . . 158
 (c). Zeitaufwand der tFCA unter Ausschluss von Kreisen (relativer Median) . 158
 B.2. Relativer Zeitaufwand im Vergleich für verschiedene Modelle. 160
 B.3. Zeitaufwand, Test-Aufrufe und Zeugenzahl im Verlauf der FCA.
 Vergleich von FCA mit tFCA.
 11 Wiederholungen, teils ergänzt um quadratische Regression. 164
 (a). Gesamtrechnzeit bis \mathcal{C}_i 164
 (b). Gelöste LPs (MILPs) bis \mathcal{C}_i 164
 (c). Gefundene nicht-triviale Elemente bis \mathcal{C}_i 164
 B.4. Zeitaufwand zur Berechnung der ersten i Maxima $1_{L \perp \{i\}}$ 165
 B.5. Gelöste LPs/MILPs zur Berechnung der ersten i Maxima $1_{L \perp \{i\}}$ 167
 B.6. Gefundene Elemente zur Berechnung der ersten i Maxima $1_{L \perp \{i\}}$ 168
 B.7. Anzahl gefundener Verbandselemente (Zeugen) aufgeteilt nach Kardinalität. . . 169
 B.8. Anzahl gefundener Verbandselemente (Zeugen) über Anzahl gelöster LPs. . . . 170
 (a). Verhältnis im kartesischen Koordinatensystem 170
 (b). Verhältnis in logarithmisierte Darstellung 170
 B.9. Relativer Aufwand der verschiedenen Etappen der EFCA.
 Vergleich Zeugenauswahl mit Zeugenverwerfung. 172
 (a). Gelöste LPs der EFCA ohne Ausschluss von Kreisen (relativer Median) . 172
 (b). Zeitaufwand der EFCA ohne Ausschluss von Kreisen (relativer Median) . 172
 (c). Gefundene Elemente bei EFCA ohne Ausschluss von Kreisen (relativer Median) 172
 (d). Gelöste LPs der EFCA ohne Ausschluss von Kreisen (relativer Median, Zeugen verworfen) 172
 (e). Zeitaufwand der EFCA ohne Ausschluss von Kreisen (relativer Median, Zeugen verworfen) 172
 (f). Gefundene Elemente bei EFCA ohne Ausschluss von Kreisen (relativer Median, Zeugen verworfen) 172
 B.10. Relativer Aufwand im Vergleich für verschiedene Modelle.
 (heuristische Zeugenauswahl) 172
 (a). Vergleich: Laufzeit 172

(b). Vergleich: Anzahl gefundener Elemente	172
B.11. Anzahl bekannter Elemente, die wir im i -ten Schritt wiederverwerten. Vergleich heuristische Zeugenauswahl mit direkter Verwerfung. 1 Simulation ohne Wiederholung.	174
B.12. Zeitaufwand, Test-Aufrufe und Zeugenzahl im Verlauf der FCA. Vergleich von EFCA mit tEFCA, Zeugenauswahl mit Zeugenverwerfung. 5 Wiederholungen, teils ergänzt um quadratische Regression.	176
(a). Gesamtrechenzeit bis \mathcal{C}_i	176
(b). Gelöste LPs (MILPs) bis \mathcal{C}_i	176
(c). Gefundene nicht-triviale Elemente bis \mathcal{C}_i	176
B.13. Zeitaufwand zur Berechnung der ersten i Maxima $1_{L_{\perp\{i\}}}$	178
B.14. Gelöste LPs/MILPs zur Berechnung der ersten i Maxima $1_{L_{\perp\{i\}}}$	180
B.15. Gefundene Elemente zur Berechnung der ersten i Maxima $1_{L_{\perp\{i\}}}$	181
B.16. Anzahl gefundener Verbandselemente (Zeugen) über Anzahl gelöster LPs.	182
(a). Verhältnis im kartesischen Koordinatensystem	182
(b). Verhältnis in logarithmisierte Darstellung	182

Literaturverzeichnis

- Acuña, V., Chierichetti, F., Lacroix, V., Marchetti-Spaccamela, A., Sagot, M.-F. & Stougie, L. (2009), 'Modes and cuts in metabolic networks: Complexity and algorithms', *Biosystems* **95**(1), 51–60.
- Acuña, V., Marchetti-Spaccamela, A., Sagot, M.-F. & Stougie, L. (2010), 'A note on the complexity of finding and enumerating elementary modes', *BioSystems* **99**, 210–214.
- Ballerstein, K., Kamp, A. v., Klamt, S. & Haus, U.-U. (2011), 'Minimal cut sets in metabolic networks are elementary modes in a dual network', *Bioinformatics* .
- Beard, D. A., Babson, E., Curtis, E. & Qian, H. (2004), 'Thermodynamic constraints for biochemical networks', *Journal of theoretical biology* **228**(3), 327–333.
- Beard, D. A., Liang, S.-d. & Qian, H. (2002), 'Energy Balance for Analysis of Complex Metabolic Networks', *Biophysical Journal* **83**(1), 79–86.
- Becker, S. A. & Palsson, B. Ø. (2005), 'Genome-scale reconstruction of the metabolic network in *Staphylococcus aureus* N315: an initial draft to the two-dimensional annotation', *BMC Microbiology* **5**(1), 8.
- Becker, S., Price, N. & Palsson, B. (2006), 'Metabolite coupling in genome-scale metabolic networks', *BMC Bioinformatics* **7**(1), 111.
- Behre, J., Wilhelm, T., Kamp, A. v., Ruppin, E. & Schuster, S. (2008), 'Structural robustness of metabolic networks with respect to multiple knockouts', *Journal of Theoretical Biology* **252**(3), 433–441.
- Biere, A. (2008), 'PicoSAT Essentials', *Journal on Satisfiability, Boolean Modeling and Computation (JSAT)* **4**, 75–97.
- Björner, A. (1993), *Oriented matroids*, Cambridge University Press, Cambridge and New York.
- Boghigian, B. A., Shi, H., Lee, K. & Pfeifer, B. A. (2010), 'Utilizing elementary mode analysis, pathway thermodynamics, and a genetic algorithm for metabolic flux determination and optimal metabolic network design', *BMC Systems Biology* **4**(1), 49.
- Brøndsted, A. (1983), *An introduction to convex polytopes*, Springer-Verlag, New York.

- Burgard, A. P., Nikolaev Evgeni V., Schilling, C. H. & Maranas, C. D. (2004), ‘Flux Coupling Analysis of Genome-Scale Metabolic Network Reconstructions’, *Genome Research* **14**(2), 301–312.
- Burgard, A., Pharkya, P. & Maranas, C. (2003), ‘OptKnock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization’, *Biotechnology and Bioengineering* **84**, 647–657.
- Carbonell, P., Fichera, D., Pandit, S. B. & Faulon, J.-L. (2012), ‘Enumerating metabolic pathways for the production of heterologous target chemicals in chassis organisms’, *BMC Systems Biology* **6**(1), 10.
- Carlson, R. P. (2007), ‘Metabolic systems cost-benefit analysis for interpreting network structure and regulation’, *Bioinformatics* **23**(10), 1258–1264.
- Chan, S. H. J. & Ji, P. (2011), ‘Decomposing flux distributions into elementary flux modes in genome-scale metabolic networks’, *Bioinformatics* **27**(16), 2256–2262.
- Codato, G. & Fischetti, M. (2006), ‘Combinatorial Benders’ Cuts for Mixed-Integer Linear Programming’, *Operations Research* **54**(4), 756–766.
- Cook, W., Koch, T., Steffy, D. E. & Wolter, K. (2011), ‘An Exact Rational Mixed-Integer Programming Solver’, *Lecture Notes in Computer Science* **6655**, 104–116.
- Covert, M. W. (2002), ‘Transcriptional Regulation in Constraints-based Metabolic Models of Escherichia coli’, *Journal of Biological Chemistry* **277**(31), 28058–28064.
- Davey, B. A. & Priestley, H. A. (1990), *Introduction to lattices and order*, 1 edn, Cambridge University Press.
- David, L., Marashi, S.-A., Larhlimi, A., Mieth, B. & Bockmayr, A. (2011), ‘FFCA: a feasibility-based method for flux coupling analysis of metabolic networks’, *BMC Bioinformatics* **12**(1), 236.
- Drösser, C. (2004), ‘Stimmt’s? – Verlotterte Jugend’.
- Duarte, N. C., Herrgård, M. J. & Palsson, B. Ø. (2004), ‘Reconstruction and validation of *Saccharomyces cerevisiae* iND750, a fully compartmentalized genome-scale metabolic model’, *Genome research* **14**(7), 1298–1309.
- Edwards, J. S. & Palsson, B. O. (2000), ‘Metabolic flux balance analysis and the in silico analysis of *Escherichia coli* K-12 gene deletions’, *BMC Bioinformatics* **1**, 1.
- Een, N. & Sörensson, N. (2005), ‘MiniSat v1.13 - A SAT Solver with Conflict-Clause Minimization’, *Competition: SAT 2005* .
- Feeman, T. G. (2010), *The mathematics of medical imaging: A beginner’s guide*, Springer, New York.

- Feist, A. M., Henry, C. S., Reed, J. L., Krummenacker, M., Joyce, A. R., Karp, P. D., Broadbelt, L. J., Hatzimanikatis, V. & Palsson, B. Ø. (2007), ‘A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information’, *Molecular systems biology* **3**, 121.
- Feist, A. M., Scholten, J. C. M., Palsson, B. Ø., Brockman, F. J. & Ideker, T. (2006), ‘Modeling methanogenesis with a genome-scale metabolic reconstruction of *Methanosarcina barkeri*’, *Molecular Systems Biology* **2**.
- Figueiredo, L. F. d., Podhorski, A., Rubio, A., Kaleta, C., Beasley, J. E., Schuster, S. & Planes, F. J. (2009), ‘Computing the shortest elementary flux modes in genome-scale metabolic networks’, *Bioinformatics (Oxford, England)* **25**(23), 3158–3165.
- Freese, R. S., Ježek, J. & Nation, J. B. (1995), *Free lattices*, American Mathematical Society, Providence and R.I.
- Füredi, Z. (1988), ‘Matchings and covers in hypergraphs’, *Graphs and Combinatorics* **4**(1), 115–206.
- Gagneur, J. & Klamt, S. (2004), ‘Computation of elementary modes: a unifying framework and the new binary approach’, *BMC Bioinformatics* **5**(175).
- Ganter, B., Stumme, G. & Wille, R. (2005), *Formal concept analysis: Foundations and applications*, Springer, Berlin.
- Gatterburg, A. (2010), ‘Ich pubertiere, also bin ich’.
- Gianchandani, E. P., Chavali, A. K. & Papin, J. A. (2010), ‘The application of flux balance analysis in systems biology’, *Wiley interdisciplinary reviews. Systems biology and medicine* **2**(3), 372–382.
- Goldstein, Y. A. B. (2009), ‘Zur mathematischen Theorie elementarer Flussmoden: Diplomarbeit’.
- Grünbaum, B. (1967), *Convex polytopes*, Interscience, London and New York [etc.].
- Grunwald, S., Speer, A., Ackermann, J. & Koch, I. (2008), ‘Petri net modelling of gene regulation of the Duchenne muscular dystrophy’, *Bio Systems* **92**(2), 189–205.
- Gurobi Optimization Inc (2012), ‘Gurobi 5.0’.
<http://www.gurobi.com>
- Hädicke, O. & Klamt, S. (2011), ‘Computing complex metabolic intervention strategies using constrained minimal cut sets’, *Metabolic Engineering* **13**(2), 204–213.
- Herrgård, M. J., Swainston, N., Dobson, P., Dunn, W. B., Arga, K. Y., Arvas, M., Blüthgen, N., Borger, S., Costenoble, R., Heinemann, M., Hucka, M., Le Novère, N., Li, P., Liebermeister, W., Mo, M. L., Oliveira, A. P., Petranovic, D., Pettifer, S., Simeonidis, E.,

- Smallbone, K., Spasić, I., Weichart, D., Brent, R., Broomhead, D. S., Westerhoff, H. V., Kürdar, B., Penttilä, M., Klipp, E., Palsson, B. Ø., Sauer, U., Oliver, S. G., Mendes, P., Nielsen, J. & Kell, D. B. (2008), ‘A consensus yeast metabolic network reconstruction obtained from a community approach to systems biology’, *Nature Biotechnology* **26**(10), 1155–1160.
- Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., and the rest of the SBML Forum: Arkin, A. P., Bornstein, B. J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., Gilles, E. D., Ginkel, M., Gor, V., Goryanin, I. I., Hedley, W. J., Hodgman, T. C., Hofmeyr, J.-H., Hunter, P. J., Juty, N. S., Kasberger, J. L., Kremling, A., Kummer, U., Le Novère, N., Loew, L. M., Lucio, D., Mendes, P., Minch, E., Mjolsness, E. D., Nakayama, Y., Nelson, M. R., Nielsen, P. F., Sakurada, T., Schaff, J. C., Shapiro, B. E., Shimizu, T. S., Spence, H. D., Stelling, J., Takahashi, K., Tomita, M., Wagner, J. & Wang, J. (2003), ‘The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models’, *Bioinformatics* **19**(4), 524–531.
- Jahberg, H. (2011), ‘Wirtschaft – Was Kinder dürfen’.
- Jamshidi, N. & Palsson, B. Ø. (2007), ‘Investigating the metabolic capabilities of *Mycobacterium tuberculosis* H37Rv using the in silico strain iNJ661 and proposing alternative drug targets’, *BMC Systems Biology* **1**(1), 26.
- Jol, S. J., Kümmel, A., Terzer, M., Stelling, J. & Heinemann, M. (2012), ‘System-level insights into yeast metabolism by thermodynamic analysis of elementary flux modes’, *PLoS computational biology* **8**(3), e1002415.
- Jozefczuk, S., Klie, S., Catchpole, G., Szymanski, J., Cuadros-Inostroza, A., Steinhauser, D., Selbig, J. & Willmitzer, L. (2010), ‘Metabolomic and transcriptomic stress response of *Escherichia coli*’, *Molecular Systems Biology* **6**(364).
- Kaleta, C., Figueiredo, L. F. d. & Schuster, S. (2009), ‘Can the whole be less than the sum of its parts? Pathway analysis in genome-scale metabolic networks using elementary flux patterns’, *Genome research* **19**(10), 1872–1883.
- Kamp, A. v. & Schuster, S. (2006), ‘Metatool 5.0: fast and flexible elementary modes analysis’, *Bioinformatics* **22**(15), 1930–1931.
- Kaye, R. (2000), ‘Minesweeper is NP-complete’, *Mathematical Intelligencer* **22**(2), 9–15.
- Kim, J. & Reed, J. L. (2010), ‘OptORF: Optimal metabolic and regulatory perturbations for metabolic engineering of microbial strains’, *BMC Systems Biology* **4**(1), 53.
- Klamt, S. (2006), ‘Generalized concept of minimal cut sets in biochemical networks’, *Bio Systems* **83**(2-3), 233–247.
- Klamt, S., Gagneur, J. & Kamp, A. v. (2005), ‘Algorithmic approaches for computing elemen-

- tary modes in large biochemical reaction networks', *Syst Biol (Stevenage)* **152**(4), 249–255.
- Klamt, S. & Gilles, E. D. (2004), 'Minimal cut sets in biochemical reaction networks', *Bioinformatics* **20**(2), 226–234.
- Klamt, S., Grammel, H., Straube, R., Ghosh, R. & Gilles, E. D. (2008), 'Modeling the electron transport chain of purple non-sulfur bacteria', *Molecular Systems Biology* **4**.
- Klamt, S. & Stelling, J. (2002), 'Combinatorial complexity of pathway analysis in metabolic networks', *Molecular biology reports* **29**(1-2), 233–236.
- Klipp, E., Liebermeister, W., Wierling, C., Kowald, A., Lehrach, H. & Herwig, R. (2009), *Systems biology: A textbook*, Wiley-VCH, Weinheim.
- Kreuzer, A. (2008), 'Jugendgerichte – Gefängnis ist nicht gleich Erziehung'.
- Larhlimi, A. & Bockmayr, A. (2006), 'A New Approach to Flux Coupling Analysis of Metabolic Networks', *Computational Life Sciences II* **4216**, 205–215.
- Larhlimi, A. & Bockmayr, A. (2009), 'A new constraint-based description of the steady-state flux cone of metabolic networks', *Discrete Applied Mathematics* **157**(10), 2257–2266.
- Larhlimi, A., David, L., Selbig, J. & Bockmayr, A. (2012), 'F2C2: a fast tool for the computation of flux coupling in genome-scale metabolic networks', *BMC Bioinformatics* **13**(1), 57.
- Llaneras, F. & Picó, J. (2010), 'Which metabolic pathways generate and characterize the flux space? A comparison among elementary modes, extreme pathways and minimal generators', *Journal of biomedicine & biotechnology* **2010**, 753904.
- Mahadevan, R. & Schilling, C. (2003), 'The effects of alternate optimal solutions in constraint-based genome-scale metabolic models', *Metabolic Engineering* **5**(4), 264–276.
- Marashi, S.-A. (2011), Constraint-based Analysis of Substructures of Metabolic Networks, PhD thesis, Berlin.
http://www.diss.fu-berlin.de/diss/servlets/MCRFileNodeServlet/FUДИSS_derivate_000000009508/thesisOnline.pdf?hosts=
- Marashi, S.-A. & Bockmayr, A. (2011), 'Flux coupling analysis of metabolic networks is sensitive to missing reactions', *Biosystems* **103**(1), 57–66.
- Marashi, S.-A., David, L. & Bockmayr, A. (2012a), 'Analysis of Metabolic Subnetworks by Flux Cone Projection', *Algorithms for molecular biology : AMB* **7**(1), 17.
- Marashi, S.-A., David, L. & Bockmayr, A. (2012b), 'On flux coupling analysis of metabolic subsystems', *Journal of Theoretical Biology* **302**, 62–69.
- Melzer, G., Esfandabadi, M., Franco-Lara, E. & Wittmann, C. (2009), 'Flux Design: In silico

- design of cell factories based on correlation of pathway fluxes to desired properties', *BMC Systems Biology* **3**(1), 120.
- Motzkin, T. S., Raiffa, H., Thompson, G. L. & Thrall, R. M. (1953), 'The double description method: Contributions to the theory of games', *Annals of Mathematics Studies* **28**, 51–73.
- Müller, A. (2012), 'Thermodynamic Constraints in Metabolic Networks: Master Thesis'.
<http://page.mi.fu-berlin.de/arnem/theses/master.pdf>
- Nation, J. B. (2012), 'Revised Notes on Lattice Theory'.
<http://www.math.hawaii.edu/~jb/>
- Nirenberg, S. & Pandarinath, C. (2012), 'Retinal prosthetic strategy with the capacity to restore normal vision', *Proceedings of the National Academy of Sciences* .
- Notebaart, R. A., Teusink, B., Siezen, R. J. & Papp, B. (2008), 'Co-Regulation of Metabolic Genes Is Better Explained by Flux Coupling Than by Network Distance', *PLoS Computational Biology* **4**(1), e26.
- Oliveira Dal'Molin, C. G. d., Quek, L.-E., Palfreyman, R. W., Brumbley, S. M. & Nielsen, L. K. (2010), 'AraGEM, a Genome-Scale Reconstruction of the Primary Metabolic Network in Arabidopsis', *Plant Physiology* **152**(2), 579–589.
- Oliveira, J. S., Bailey, C. G., Jones-Oliveira, J. B. & Dixon, D. A. (2001), 'An algebraic-combinatorial model for the identification and mapping of biochemical pathways', *Bulletin of mathematical biology* **63**(6), 1163–1196.
- Orth, J. D., Conrad, T. M., Na, J., Lerman, J. A., Nam, H., Feist, A. M. & Palsson, B. Ø. (2011), 'A comprehensive genome-scale reconstruction of Escherichia coli metabolism–2011', *Molecular systems biology* **7**, 535.
- Orth, J. D., Fleming, R. M. & Palsson, B. Ø. (2010), 'The Core E. coli Model'.
<http://systemsbiology.ucsd.edu/Downloads/EcoliCore>
- Orth, J. D., Thiele, I. & Palsson, B. (2010), 'What is flux balance analysis', *Nature Biotechnology* **28**, 245–248.
- Palsson, B. (2009), 'Metabolic systems biology', *FEBS Letters* **583**(24), 3900–3904.
- Papadimitriou, C. H. (1994), *Computational complexity*, Addison-Wesley, Reading and Mass.
- Partridge, J. D., Sanguinetti, G., Dibden, D. P., Roberts, R. E., Poole, R. K. & Green, J. (2007), 'Transition of Escherichia coli from Aerobic to Micro-aerobic Conditions Involves Fast and Slow Reacting Regulatory Components', *Journal of Biological Chemistry* **282**(15), 11230–11237.
- Planes, F. & Beasley, J. (2009), 'Path finding approaches and metabolic pathways', *Discrete Applied Mathematics* **157**(10), 2244–2256.

- Price, N. D., Famili, I., Beard, D. A. & Palsson, B. (2002), ‘Extreme Pathways and Kirchhoff’s Second Law’, *Biophysical Journal* **83**, 2879–2882.
- Savinell, J. M. & Palsson, B. O. (1992), ‘Network analysis of intermediary metabolism using linear optimization. I. Development of mathematical formalism’, *Journal of Theoretical Biology* **154**(4), 421–454.
- Schellenberger, J., Park, J. O., Conrad, T. M. & Palsson, B. (2010), ‘BiGG: a Biochemical Genetic and Genomic knowledgebase of large scale metabolic reconstructions’, *BMC Bioinformatics* **11**(213).
- Schilling, C. H., Letscher, D. & Palsson, B. O. (2000), ‘Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective’, *Journal of theoretical biology* **203**(3), 229–248.
- Schrijver, A. (2003), *Combinatorial optimization*, Springer, Berlin [u.a.].
- Schuster, S. (1994), ‘On elementary flux modes in biochemical reaction systems at steady state’, *Journal of Biological Systems* **2**(2), 165.
- Schuster, S., Hilgetag, C., Woods, J. H. & Fell, D. A. (2002), ‘Reaction routes in biochemical reaction systems: algebraic properties, validated calculation procedure and example from nucleotide metabolism’, *Journal of mathematical biology* **45**(2), 153–181.
- Shlomi, T., Eisenberg, Y., Sharan, R. & Ruppin, E. (2007), ‘A genome-scale computational study of the interplay between transcriptional regulation and metabolism’, *Molecular systems biology* **3**, 101.
- Soh, T. & Inoue, K. (2012), Finding Minimal Reaction Sets in Large Metabolic Pathways, in A. Dovier, A. D. Palù & S. Will, eds, ‘WCB10’, Vol. 4 of *EPiC Series*, EasyChair, pp. 38–52.
- Song, H.-S. & Ramkrishna, D. (2009), ‘Reduction of a set of elementary modes using yield analysis’, *Biotechnology and bioengineering* **102**(2), 554–568.
- Terzer, M. & Stelling, J. (2008), ‘Large-scale computation of elementary flux modes with bit pattern trees’, *Bioinformatics* **24**(19), 2229–2235.
- Thiele, I., Vo, T. D., Price, N. D. & Palsson, B. Ø. (2005), ‘Expanded metabolic reconstruction of *Helicobacter pylori* (iT341 GSM/GPR): an in silico genome-scale characterization of single- and double-deletion mutants’, *Journal of bacteriology* **187**(16), 5818–5830.
- Topkis, D. M. (1976), ‘The structure of sublattices of the product of n lattices.’, *Pacific J. Math.* **65**(2), 525–532.
<http://projecteuclid.org/euclid.pjm/1102866810>
- Voss, K., Heiner, M. & Koch, I. (2003), ‘Steady state analysis of metabolic pathways using Petri nets’, *In silico biology* **3**(3), 367–387.

- Wang, Q., Yang, Y., Ma, H. & Zhao, X. (2007), ‘Metabolic network properties help assign weights to elementary modes to understand physiological flux distributions’, *Bioinformatics* **23**(9), 1049–1052.
- Watson, M. R. (1984), ‘Metabolic maps for the Apple II’, *Biochemical Society Transactions* **12**, 1093–1094.
- Wikipedia (2012a), ‘Dissertation – Wikipedia, Die freie Enzyklopädie’.
["http://de.wikipedia.org/w/index.php?title=Dissertation&oldid=107590523"](http://de.wikipedia.org/w/index.php?title=Dissertation&oldid=107590523)
- Wikipedia (2012b), ‘KISS principle — Wikipedia, The Free Encyclopedia’.
http://en.wikipedia.org/w/index.php?title=KISS_principle&oldid=496157551
- Wikipedia (2012c), ‘Starbucks — Wikipedia, Die freie Enzyklopädie’.
<http://de.wikipedia.org/w/index.php?title=Starbucks&oldid=106043523>
- Wikipedia (2012d), ‘Weniger ist mehr — Wikipedia, Die freie Enzyklopädie’.
http://de.wikipedia.org/w/index.php?title=Weniger_ist_mehr&oldid=101348117
- Wille, R. (2009), Restructuring Lattice Theory: An Approach Based on Hierarchies of Concepts, *in* D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, S. Ferré & S. Rudolph, eds, ‘Lecture Notes in Computer Science’, Springer Berlin Heidelberg, Berlin and Heidelberg, pp. 314–339.
- Zhang, A., Borodovsky, M., Ozsoyoglu, G., Mikler, A., Ay, F. & Kahveci, T. (2010), Functional similarities of reaction sets in metabolic pathways, *in* ‘Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology - BCB ’10’, ACM Press, p. 102.
- Zhao, Q. & Kurata, H. (2009), ‘Genetic modification of flux for flux prediction of mutants’, *Bioinformatics* **25**(13), 1702–1708.

Zur Entstehung dieser Arbeit

Indes bitte ich euch darum auch noch recht sehr,
ihr Athener, und bedinge es mir aus, wenn ihr
mich hört mit ähnlichen Reden meine
Verteidigung führen, wie ich gewohnt bin, auch
auf dem Markt zu reden bei den Wechlertischen,
wo die meisten unter euch mich gehört haben, und
anderwärts, – dass ihr euch nicht verwundert noch
mir Getümmel erregt deshalb!

Sokrates (in *Die Apologie des Sokrates* von Platon)

Laut Platon beginnt Sokrates die Gerichtsverhandlung,¹ in der über sein Leben oder Tod verhandelt wurde, mit der Bitte darum, dass ihm seine Wortwahl verziehen werde. Er sei – so redet er weiter – mit nun mehr 70 Jahren zum ersten Male vor Gericht und wisse nicht, wie man korrekterweise reden sollte. Die Richter, deren Aufgabe das Richten sei, mögen bitte darüber hinwegsehen, da er sich dafür auf seine Aufgabe, eine Wiedergabe der Wahrheit konzentrieren werde.

Bei Fertigstellung dieser Arbeit bin ich noch keine 70, sondern 25 Jahre alt. Hierbei geht es um keine Verhandlung um Leben und Tod, noch um sonst ein Gerichtsverfahren. Auch kann ich mich nicht darauf berufen, nicht zu wissen, wie man sich in der Wissenschaft, in der Mathematik, in der Systembiologie auszudrücken pflege. Dafür habe ich zu viele Artikel gelesen, dafür habe ich erste Erfahrung durch eine Diplomarbeit erworben.

Ähnlich wie Sokrates fühle ich mich jedoch dazu verpflichtet, meinen Sprachgebrauch, ja, die ganze Erscheinung dieser Arbeit einmal kurz zu erläutern.

Mir persönlich war es wichtig, dass man mich in dieser Arbeit wiederfindet. Ich wollte einen Text schreiben, den ich auch in zehn oder 20 Jahren gerne einmal wieder anfasse und sehe, wie ich die Welt mit 25 gesehen habe.

Meine offizielle Aufgabe war es, eine Doktorarbeit im Fach der Mathematik zu erstellen. Das findet sich vielfach in der Gliederung wieder. Definition – Satz – Beweis – Bemerkung – Definition – Lemma ... Selbstredend kommen diese Kernmerkmale einer mathematischen Arbeit hier wieder und wieder vor.

Doch es gibt auch viel Zwischentext und Erläuterungen. Es sollte eine Arbeit werden, die Mathematiker lesen können, die zugleich aber auch an Biowissenschaftler gerichtet ist. Dar-

¹Original und Übersetzung unter <http://www.gottwein.de/Grie/plat/apoll7a.php>.

um habe ich auf eine ausgewogene Mischung von Motivation und Formalisierung wertgelegt. Manchmal auch ein doch sehr bildliches Beispiel gewählt.

Dieses Anderssein sollte sich auch im Layout widerspiegeln. Der Leser sollte die Arbeit aufschlagen und auf einen Blick sehen, dass sie anders ist, als allgemein von einer Dissertation in der Mathematik erwartet.

Gleich werde ich einige Worte dazu verlieren, woher die einzelnen Elemente stammen oder wie sie erstellt wurden. Doch zuvor will ich es noch einmal Sokrates gleichtun und den Leser darum bitten, über meinen äußeren Stil – vielleicht ist er schlechter, vielleicht auch wohl gar besser – hinwegzusehen und nur auf den Inhalt zu schauen.

§D.1 Genutzte Software

Diese Arbeit ist mit Hilfe des Textsatzprogramms $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X } 2_{\epsilon}$ entstanden.² Das Layout stammt im Kern von einer ehemaligen Kommilitonin, Katja Hanowski, die so freundlich war, es mir zur Verfügung zu stellen. Danke dafür!

In Einzelfällen wurden jedoch Änderungen vorgenommen. So wurden etwa die Mathematik-Boxen für Definition, Satz, etc. um Piktogramme ergänzt. Diese stehen allesamt unter einer Creative Commons Lizenz, können also frei genutzt werden. Sie entstammen der *Open Clip Art Library*.

Die anderen Bilder und Graphen sind selbst erstellt. Bis auf wenige Ausnahmen liegen sie als Vektorgraphiken im *svg*-Format vor, die mittels *Inkscape* in ein Latex-freundliches Format umgewandelt wurden.

Viele dieser Vektorgraphiken wurden durch das Graphen-Visualisierungsprogramm *Graphviz* erstellt, so etwa die Beispiele für metabolische Netzwerke und die Verbände. Auch die Landkarten-Darstellung der Flusskopplung entspringt den Algorithmen von *Graphviz*.

Graphviz ist eine Software, die Graphen als Text-Dateien im *dot*-Format akzeptiert und basierend darauf eine zweidimensionale Darstellung der Ecken und Kanten erstellt. Die Arbeit erleichtert hat mir dabei in fast allen Fällen das Programm *Instaviz* für das iPad, mit dem sich Graphen mit Fingergesten malen und anschließend beschriften lassen.

Die Kombination *Instaviz* \rightarrow *Graphviz* \rightarrow *Inkscape* deckt die meisten der im Theorieteil gezeigten Graphiken ab.

²Für die in diesem Kapitel genannte Computer-Software werden keine Einzelquellen genannt. Sie sind jedoch allesamt schnell und eindeutig durch gängige Internetsuchmaschinen auffindbar. In vielen Fällen handelt es sich um offene Software, die frei genutzt und abgewandelt werden kann.

Im Praxisteil (aber auch zuvor schon bei einem Beispiel zur EFCA) übernahm dann das Statistikprogramm *R*, das auch zur Auswertung der Daten genutzt wurde.

Die Graphen selbst wurden mittels des Pakets *ggplot2* erstellt, das unglaublich viele Möglichkeiten zur graphischen Aufarbeitung von Daten ermöglicht.

Wir sind fast durch, jetzt fehlen nur noch zwei Programme.

Die beiden Abbildungen 4.2 wurden mittels *yEd* erstellt; einem Programm, das es einem ermöglicht, mittels Maus und Tastatur Graphen zu zeichnen.

Das Mindmap von Seite 77 schließlich entstand mithilfe der kostenlosen Version des Programms *SimpleMind+*, das ebenfalls auf einem iPad genutzt wurde.

Zur Literaturverwaltung wurde das Programm *Citavi* eingesetzt.

§D.2 Erstellte Software

Das im Theorieteil hergeleitete und im Praxisteil diskutierte Programm zur FCA, tFCA und EFCA wurde, wie erwähnt, in **C#** geschrieben und nutzt Gurobi.

Zeitgleich zur Fertigstellung dieser Arbeit wird es auf GitHub öffentlich zugänglich gemacht. Es steht unter Creative Commons NC-SA-AT Lizenz und darf somit für nicht-kommerzielle Zwecke unter Nennung des Ursprungs frei genutzt und abgewandelt werden.

Es findet sich unter folgendem Link:

https://github.com/goldsteiny/tEFCA_Lattice_Solver.git



Ein paar Worte zum Abschluss

Die Jugend von heute liebt den Luxus, hat schlechte Manieren und verachtet die Autorität. Sie widersprechen ihren Eltern, legen die Beine übereinander und tyrannisieren ihre Lehrer.

Sokrates

Odi et amo. quare id faciam, fortasse requiris. nescio, sed fieri sentio et excrucior.

Catull

Wenn man als Werkzeug nur einen Hammer hat, sieht jedes Problem wie ein Nagel aus.

Abraham Maslow

Wie bedankt man sich bei den Menschen, die einen die drei Jahre über begleiteten, in denen man sich hauptsächlich mit der Erstellung einer Doktorarbeit beschäftigte? Sowohl bei den Menschen, die thematisch dabei waren und Hinweise geben konnten, als auch bei denen, die sich inhaltlich zurückhielten, dafür jedoch ideell oder moralisch unterstützten. Ich will es erst mit meiner kleinen Sicht auf die Dinge versuchen, bevor ich brav Danke sage.

Laut Wikipedia (2012a) geht es bei einer Doktorarbeit in Inhalt und Form um Folgendes:

Je nach Fach beschäftigt sich eine Dissertation entweder theoretisch mit einem Thema oder beschreibt und interpretiert empirisch oder experimentell gewonnene Erkenntnisse.

Eine Dissertation soll belegen, dass der Kandidat selbständig wissenschaftlich zu arbeiten versteht. Sie muss im Regelfall neue Erkenntnisse zu dem gewählten Gegenstand enthalten und methodisch einwandfrei sein. Eine Dissertation ist im Normalfall also eine Forschungsarbeit. Wichtig zum Nachweis der Fähigkeit zum eigenverantwortlichen wissenschaftlichen Arbeiten sind auch die Kenntnis der relevanten Forschungsliteratur, der üblichen Arbeitsweise des Fachgebiets, das Ziehen belastbarer Rückschlüsse sowie die Einbettung der eigenen Arbeiten in den wissenschaftlichen Kontext. Zur Textmenge einer Dissertation gibt es üblicherweise keine Vorschriften. Der übliche Rahmen liegt zwischen etwa 150 und 500 Seiten.

Die Seitenzahl dieser Arbeit liegt demnach im üblichen Rahmen. Und wie jeder stolze Autor ist für mich ganz klar, dass diese Arbeit viele neue Erkenntnisse enthält und methodisch einwandfrei ist – doch ob dem auch objektiv so ist, das müssen die Gutachter noch entscheiden.

Diese abschließenden Worte beginnen mit drei Zitaten. Mindestens eines davon ist falsch.

Während diese Arbeit entstand, wurde in den Medien viel darüber diskutiert, wie in der Wissenschaft heutzutage gearbeitet werde. Wie viel Wert der akademische Doktorgrad noch haben solle, welche selbständige Leistung wirklich gefordert werden müsse. Angefangen hat diese Diskussion mit der Dissertation eines Politikers, die Jahre nach ihrer Publikation als vermeindliches Sammelsurium undeklarer Zitate enttarnt wurde. Seitdem geschah Ähnliches bei weiteren Politikern in Deutschland, von denen mehrere Ämter niederlegten.

Als die Medien das Thema noch pushten, gaben mir mehrere Bekannte und Freunde den scherzhaft besorgten Ratschlag, es diesen Politikern nicht nachzutun, sondern immer schön brav meine Quellen anzugeben. Sicherlich war ich nicht der einzige Doktorand der letzten Jahre, der damit schon bald keinen Spaß mehr verstand.

In der Wissenschaft wird zitiert, Punkt. So soll es sein, so muss es sein. Es geht dabei um mehrere Dinge – darum, nicht fremde Verdienste als die eigenen auszugeben, aber vielleicht viel mehr noch darum, dem Zitierten die Anerkennung zu geben, die er für seine Arbeit verdient. Dass man mit einem Quellenverweis die getroffene Aussage noch faktisch absichern will, wird dabei schon Pragmatismus.

Über den Diebstahl geistigen Eigentums will ich hier nicht schreiben. Dieses Thema beschäftigt Teile der Bevölkerung seit Jahren und wird es voraussichtlich noch lange tun. Das hier ist nicht der Platz dafür.

Wertschätzung und klare Fakten wären jedoch in der gesamten Welt wünschenswert. Wäre es nicht schön, wenn alles gedruckte Wort doch stimmte?

Ich bemühe Sokrates, ohne Sokrates zu bemühen. Das genutzte Pseudo-Zitat wird ständig gerne genutzt, meist um eine Aussage über die Jugend einzuleiten; auch von qualitativ hochwertigeren Medien wie *Zeit* (Kreuzer 2008), *Spiegel* (Gatterburg 2010), *Tagesspiegel* (Jahberg 2011) und vielen weiteren. Der Haken: Es stammt nach Kenntnis der Wissenschaft nicht von Sokrates, wie Drösser (2004) zu überprüfen gewagt hat. Dabei wäre es doch so schön, wenn man einen so lockeren und aktuellen Spruch einem der bekanntesten Weisen der Weltgeschichte zuschreiben könnte.

Hassliebe empfindet vermutlich jeder Wissenschaftler früher oder später für seine Arbeit. Auch ich erlebte in den letzten Jahren eine Achterbahnfahrt der Gefühle, während ich mein Geisteskind, diese Forschungsarbeit, sich mal schneller, mal langsamer entwickeln sah. Von abwechselnden Momenten der Freude und des Frusts war bereits einmal im praktischen Teil über die Programmierung die Rede. Ähnlich – nur noch größer – ist das bei einem Forschungsprojekt, das einen wichtigen Abschnitt im Leben eines Doktoranden ausfüllt. Kleine und große Durchbrüche wechseln sich dabei ab mit Wochen des Stockens oder der zu späten Feststellung, dass das Ergebnis doch schon längst in der Literatur bekannt ist.

Catull 85 ist für Viele eines der schönsten Gedichte aller Zeiten. Auch mich bewegt es seit Jahren. Darum bin ich froh, über die Hassliebe des Wissenschaftlers zu seiner Arbeit einen Weg gefunden zu haben, es in meiner Dissertation unterzubringen.

Die Danksagung meiner Diplomarbeit 2009 leitete Albert Einstein ein, diesmal tut er es für die Gesamtheit dieses bescheidenen Werkes. Mir ist es immer wieder Überraschung, was für eine weitere Weisheit Einstein zu seinen Lebzeiten von sich gegeben hat. Sein Zitat könnte Mantra dieser Arbeit sein, hätte ich es nicht vor wenigen Tagen erst gefunden.

Das dritte Zitat dieser Abschlussworte und somit das letzte Wort in dieser Arbeit überlasse ich jedoch einem anderen jüdischen Wissenschaftler, dem amerikanischen Psychologen Abraham Maslow. Denn so schön Einsteins Worte sind, Maslows fassen diese Arbeit umso prägnanter zusammen.

Ich wollte ein neues Werkzeug in die Welt setzen, mit dem man in der Stoffwechselmodellierung arbeiten kann. Einfachste Grundlagen der Verbandstheorie haben dafür gereicht und sind dennoch schon machtvolles Hilfsmittel. Doch so großartig ich sie selbst auch finde – und hier spricht wieder der stolze Autor und Studiosus – so bewusst bin ich mir, dass auch ich fast in die Falle getappt wäre, alles als Verband zu sehen.

Dass dem nicht ist, verdanke ich vielen Menschen.

§E.1 Danksagung

Zu aller erst bedanke ich mich bei Professor Alexander Bockmayr, der mich nicht nur während meiner Promotionszeit betreut hat, sondern auch schon zuvor meine Diplomarbeit.

Alexander, ich danke dir für die vielen Freiheiten die du mir gegeben hast.

Du hast mich wildern lassen in Gebieten, wie es mir beliebte, aber zugleich eine Auge darauf behalten, dass ich dabei in sinnvollen Grenzen blieb. Den Weg zur Verbandstheorie hätte ich ohne diesen Freiraum nicht gefunden.

Von Anfang an standest du mir bei meinen Unternehmungen zur Seite und hast sie unterstützt, wo es dir sinnvoll erschien. Und dank deiner vielen Hinweise und vor allem Literaturverweise war das sehr oft der Fall.

Ohne dich wäre ich heute nicht akademisch dort, wo ich es bin. Danke sehr!

Doch nicht nur Prof. Bockmayr hat mir auf die Finger geguckt, dass ich mich nicht zu weit vom Usus entferne. Mehr als nur einmal haben mich meine lieben Mitdoktoranden Laszlo David und Arne Müller auf aktuelle Publikationen verwiesen, Vor- und Nachteile meiner Ideen diskutiert.

Es hat lange gedauert, aber zumindest das letzte Jahr über haben auch wir es endlich geschafft, ab und an gemeinsam Mathematik zu machen. Einige nette und einige praktische Ergebnisse sind dabei herausgekommen. Laszlo und Arne, ich danke euch für eure Unterstützung!

Oftmals arbeiten Doktoranden vor allem im Rahmen ihrer eigenen Arbeitsgruppe. Ich glaube aber sehr daran, dass man mit Problemen/Herausforderungen gern auch zum Experten gehen kann. Und so hat mich mein kleiner Ausflug zu den Überdeckungen von Stoffwechselfaden/Hypergraphen zu einer anderen Arbeitsgruppe unseres Instituts geführt.

Es ist zwar schon etwas länger her, aber hiermit möchte ich mich auch heute noch einmal bei meinem alten Bekannten Roman Glebov und der gesamten Arbeitsgruppe von Tibor Szabo bedanken, für die Zeit, die sie sich genommen haben, mir ein wenig Nachhilfe zu Überdeckungen von Hypergraphen zu geben. Danke euch sehr!

Doch es gibt noch eine weitere Arbeitsgruppe, die meine Zeit als Doktorand bereichert hat. Eine, die nicht zur Freien Universität Berlin gehört. Nicht einmal in Berlin angesiedelt ist.

Dear Richard Bonneau, I thank you so much for your invitation to NYU two years ago. You welcomed me so warmly in your team and introduced me to another field of research in systems biology. I had four amazing months in New York, that I don't want to have missed. Thank you very much, I hope we will be able to work again in the near future!

Meine Zeit in New York verbinde ich einerseits mit meinem kleinen Einblick ins Bonneau Lab, aber auch mit einigen anderen Dingen. Dem Times Square, der es mir einfach angetan hat. Der liebevollen und energiegeladenen Salsa-Gemeinschaft der Stadt, insbesondere den beiden Tanzschulen Baila Society und die von Eddie & Maria Torres. Den jüdischen Studentenbewegungen, denen nichts gleich kommt, was ich in Deutschland jemals erlebt habe. Aber auch ein wenig mehr Akademisches.

Zeitgleich zu meinem New York Aufenthalt experimentierte einer meiner besten Schulfreunde für seine Master-Arbeit in Biochemie in Princeton. Er beschäftigte sich mit der Vielfältigkeit von Herpes-Viren. Dafür hatte er eine neue Theorie entwickelt, die er experimentell nachweisen wollte. Doch damit nicht genug, er wollte seine Überlegungen auch simulieren; da kam ich ins Spiel.

Paul Brodersen kontaktierte mich und gleich nach der ersten Salsa-Party startete eine tagelange Diskussion darüber, was zu programmieren sei. Ein schönes Beispiel von interdisziplinärem Kauderwelsch. Am Ende stand ein Programm, dessen Ergebnisse uns beide begeisterte. Publiziert haben wir es jedoch immer noch nicht.

Wenn ich von Paul rede, dauert es nicht lange, bis Katharina Becker erwähnt werden muss. Ironischerweise sind wir drei mit einem Faible für die Biowissenschaften ausgerechnet in der Schülerzeitung aufeinander gestoßen. Kurz vor unserem Abitur plante Paul bereits unsere gemeinsame künftige Forschungsgruppe. Der Ausgang dieses Plans ist immer noch offen.

Doch unabhängig davon, wie viel oder wenig wir in Zukunft gemeinsam forschen werden – die letzten Jahre haben wir alle darauf aufgepasst, dass die jeweils anderen etwas Sinnvolles mit ihren Talenten anstellen.

Katharina und Paul sind die größten Zweifler am stationären Zustand. Steady state? Sowas gibt's nicht!

Ein wenig dieser Skepsis habe ich übernommen, auch wenn Stationarität eine *der* Grundannahmen in meinem Forschungszweig überhaupt ist. Mit den Verbänden versuche ich sie zu umgehen, indem ich an allgemeinen qualitativen Modellen zu arbeiten erlaube.

Katharina und Paul, ich danke euch für all euren Rat, eure Literaturtipps, und all den anderen Quark, den ihr mir die letzten Jahre gereicht habt. Wenn Biowissenschaften Gesichter für mich haben, dann sind es eure.

An Freunden in der Wissenschaft habe ich jedoch nicht nur Biotechnologen und Biochemikern zu danken, sondern auch Mathematikern. Und da sei an erster Stelle Alexandra Kraft genannt.

Als wir uns in unserem dritten Semester in der Einführung zur Stochastik kennenlernten, sind wir wohl beide nicht davon ausgegangen, wie intensiv wir eines Tages miteinander arbeiten würden. Doch als Alexandra mir vor ziemlich genau einem Jahr erzählte, sie sei auf der Suche nach einem neuen Diplomarbeitsthema in der angewandten Mathematik, am besten Graphentheorie, mit einem kleinen Schuss Algorithmen und notfalls auch etwas Programmieren, fiel mir ein Tipp nicht schwer. Nach einer kurzen Einführung in die Idee der logischen Modelle stellte sie sich Professor Bockmayr vor, der von da an um eine Diplomandin reicher war.

Seit einem halben Jahr stehen wir in ständigem Austausch. Gaben und geben uns Ideen und Literaturtipps, haben Verbesserungsvorschläge parat, lesen Korrektur, etc. . . .

Vor allem haben wir dabei aber auch jeweils einen Zweiten gefunden, der versteht, an dieser Schwelle zwischen angewandter, exakter Mathematik und theoretischer Biologie zu arbeiten. Der den Frust kennt und die Freude. Der dabei motivieren kann und auch für das ein oder andere Schmunzeln sorgen kann, auch wenn das mal total abseits vom Thema ist.

Alexandra, ich hätte das kaum geglaubt, aber von all meinen Freunden bist ausgerechnet du es geworden, die am meisten zum endlichen Abschluss beigetragen hat. Vielen Dank für alles, für deine Zeit, deine Kommentare – ja sogar für deinen IrrElephant.

Vielen Dank für die Blumen. Du verstehst, was ich meine.

Alexandra und ich haben Forschungsthemen, die eng verwandt sind, und alleine deshalb schon viel miteinander gearbeitet. Auch ohne unsere Sympathien hätten wir beide davon profitiert.

Anders sieht das bei Jan und Jana de Wiljes und Dani Freund aus. Die ersten beiden selbst Doktoranden der Mathematik, letzterer ein junger Aspirant des Faches, der gerade seit drittes Studienjahr in England beginnt.

Alle drei wurden während der Entstehungszeit dieser Dissertation immer mal wieder über

Zwischenstände informiert und bekamen Zwischenfassungen zu Gesicht. Nahmen sich die Zeit dafür, sie einmal von vorne bis hinten durchzulesen und mir nach bestem Verständnis Tipps und Anmerkungen, manchmal akribische Verbesserungsvorschläge zu geben.

Manchmal kommt einem alles trivial vor, wenn man sich jahrelang mit dem gleichen Thema beschäftigt. Deshalb sind solche Einsichten von außen unglaublich hilfreich. Auch euch dreien deshalb meinen herzlichen Dank.

Eine meiner ältesten Freundinnen ist Martine Bindernagel, die kürzlich auch ihren Weg in die Mathematik gefunden hat. Vielen Dank für deine moralische Unterstützung! Ich weiß, dass ich immer auf dich zählen kann, und bin mir sicher, dass deine Schüler in Zukunft schneller lernen werden, als es ein gewisser Lehrer dir damals zugetraut hat.

Martine kenne ich seit über elf Jahren; ähnlich lang wie einen weiteren guten Freund.

Martine kenne ich noch aus der Schule, aus der Schach-AG genauer gesagt. Benni Barth hingegen kenne ich von einer jüdischen Jugendfahrt. Viele Jahre konnten wir uns nur sporadisch sehen, wohnte er doch im Ruhrpott, ich hingegen in Berlin. Dennoch hatte er stets meine größte Hochachtung und ist bis heute theologische Anlaufstelle. Ich kenne wenige Menschen in unserem Alter, die sich so gut auf das Thema verstehen, und zudem noch bereit sind, sich auf stundenlange Gespräche und detaillierte Diskussionen darüber einzulassen. Habe ich eine Frage zur Religion, dann weiß ich, wer mich sofort mit passenden Antworten aus dem Talmud versorgen und zugleich unterhalten kann.

Benni hat mich schließlich nach Düsseldorf geführt, wo ich seit einem Jahr gern gesehener Gast bin. Dort habe ich viele Leute kennen und schätzen gelernt. Darunter waren etwas vertraute Gesichter, wie das von Naomi und Esti Reichard, aber auch neue, wie Ganna Epshteyn. Ganna ist seitdem eine meiner engsten und besten Vertrauten geworden. Auch einige ihrer Piesackereien haben dafür gesorgt, diese Arbeit endlich zu einem Ende zu bringen. Da haben sich zwei gefunden, könnte man sagen.

Düsseldorf, bleib so toll wie du bist! Und keine Sorge, ihr seht mich bald wieder.

Zurück nach Berlin. Da gibt es nämlich noch zwei Kameraden aus Schulzeiten, die nicht vergessen werden sollten.

Matthias Tacke und Felix Rautenberg kenne ich, seitdem ich zehn war. Und ähnlich lange sind wir schon befreundet.

Diese Arbeit über war viel von Wegen die Rede. Man weiß nie, wohin es einem im Leben noch verschlägt. Es gibt Wege, die man gemeinsam beschreitet, wie wir drei unsere neun Jahre am Gymnasium Steglitz. Uns drei verband nicht nur schließlich die gleiche Abiturnote, sondern zuvor die Freude an einer guten Partie Skat oder Doppelkopf.

Matthias und ich haben seit einigen Jahren einen weiteren hochgradig unakademischen Weg gefunden, uns von unserem medizinischen bzw. mathematischen Alltag abzulenken. Wir treffen uns monatlich im Kino. Immer Hollywood-Produktionen, immer Originalversion. Abwechslung muss sein, und diese Arbeit wäre nicht möglich gewesen, hätte ich mich 24/7 auf sie konzentrieren müssen.

Einige Wege im Leben verschiedener Personen kommen sich näher, einige trennen sich wieder, ohne dass man weiß, wann man sich wiederbegegnet. Lieber Felix, ich danke dir für viele schöne Erinnerungen und hoffe sehr, dass sich unsere Pfade wieder kreuzen.

Ganz am Anfang dieser Danksagung stelle ich die rhetorische Frage, wie man sich bei Menschen bedanke, die Jahre lang bis zum Ende für einen da waren. Doch gerade stehe ich vor dem Problem, einen Menschen erwähnen zu müssen, der eben nicht dabei war von der ersten bis zur letzten Zeile. Ohne den es diese Arbeit aber nicht gäbe.

Dalia, die mir zur Diplomarbeit, aber auch noch anfangs zur Doktorarbeit Muse war. In ihrer Gesellschaft kam mir die Idee, es mal algebraisch, mal mit Verbänden zu versuchen.

Dalia, wir haben darüber gesprochen. Auch Monate nach der Erstfassung weiß ich nicht, was dir hier und jetzt zu schreiben ist. Über das schwierige Dankeschön.

Diese Arbeit bleibe dir gewidmet.

Einfacher, Danke zu sagen, fällt es mir bei meiner Familie. Sie war immer für mich da, ist immer für mich da und wird immer für mich da sein.

Es wäre ein Leichtes, jetzt über und über Gutes zu schreiben. Schönes genauso wie Amüsantes. Wer bis hierher gekommen ist, weiß, dass ich kein Mensch der kleinen Worte bin.

Vielleicht tu ich es gerade deshalb nicht. Egal, wie viel ich schriebe, es wäre nicht genug.

Darum mache ich es kurz und danke euch einfach für all die Liebe und für Alles, was ihr mir gegeben habt.

§E.2 Dank an Institutionen

Diese Arbeit wäre nicht ohne die vielen Menschen zu Stande gekommen, denen ich hiermit danken wollte. Aber es gab auch Institutionen, die mich unterstützt und meine Promotion so erst ermöglicht haben.

Diese Dissertation ist im Rahmen des Graduiertenprogramms der Berlin Mathematical School (BMS) entstanden. Das bot mir die Möglichkeit, talentierte Mathematiker aus der ganzen Welt bei regelmäßigen Veranstaltungen wie den BMS-Fridays, aber auch sogenannten Softskill-Seminaren und Trainings kennenzulernen.

Im Rahmen eines Mentorenprogramms standen mir anfangs Frau Professor Caroline Lasser, später Herr Professor Ralf Kornhuber bei akademischen Fragen zu Seite.

Die Förderung durch die BMS war aber nicht nur rein ideell. Mein Aufenthalt in New York wäre ohne eine finanzielle Unterstützung durch sie kaum möglich gewesen. Deshalb will ich an dieser Stelle auch noch einmal der BMS, ganz besonders Nadja Wisniewski danken, die mir leicht und unkompliziert zu einem Auslandsstipendium verholfen und mir auch gleich noch ein paar Überlebensstipps mitgegeben haben.

Der BMS verdanke ich viel ideelle Förderung und Unterstützung fürs Ausland. Noch viel mehr verdanke ich der Gerhard C. Starck Stiftung.

Diese private Stiftung, dessen Vermögen aus dem Nachlass von Gerhard C. Starck und seiner Frau stammt, hat sich zum Ziel gesetzt, die Aus- und Weiterbildung junger Juden in Deutschland zu fördern und so zu einem Wiederaufkommen eines vielseitigen jüdischen Lebens in Deutschlands beizutragen.

Im Herbst 2006 wurde ich als einer der ersten Stipendiaten in die Förderung der Starck Stiftung aufgenommen, die mir anfangs vor allem finanziell sehr stark zur Seite stand. Sowohl mein gesamtes Studium, als auch im direkten Anschluss meine Promotion wurden durch die Starck Stiftung finanziert und dadurch zu großem Teil mit ermöglicht. Damit bin ich meinem Kenntnisstand nach Dienstältester unter den Stipendiaten.

Im Frühjahr 2011 kamen schließlich die *Starckies* zum ersten Mal zusammen, an die 100 Stipendiaten und Alumni, aber auch der Vorstand sowie Frau Inge Frey, die mit viel Herz weit mehr tut als nur zu verwalten. Es wurde ein interessantes Wochenende im idyllischen Fischbachau, bei dem wir nicht nur tolle Menschen trafen und wiedersahen, sondern uns auch wissenschaftlich austauschen konnten und über die Geschichte der Stiftung lernten. Einer der Höhepunkte war sicherlich die bewegende Rede von Frau Dr. Charlotte Knobloch, die über die Geschichte des jüdischen Lebens in Deutschland in den letzten Jahrzehnten erzählte.

Auch im Frühsommer 2012 kamen wir alle wieder zu einem ähnlich interessanten Treffen zusammen und hatten die Möglichkeit, über aktuelle Themen aus Politik und Forschung zu reden, und genossen einfach ein schönes Beisammensein.

Lieber Vorstand, liebe Frau Frey, ich danke Ihnen für all die Unterstützung der vergangenen Jahre! Ich hoffe sehr, dass meine Ausbildung und Entwicklung auch aus Ihrer Sicht im Sinne der Stiftung verlaufen ist und freue mich darauf, in den kommenden Jahren als Alumnus aktiv werden zu können.

Danke euch allen!

Yaron Goldstein

(New York, September 2012 – Berlin, Januar 2013.)



Selbständigkeitserklärung

Hiermit versichere ich, alle genutzten Quellen, Hilfsmittel und Hilfen angeben zu haben, und versichere weiterhin, auf dieser Grundlage diese Arbeit selbständig verfasst zu haben.

Die vorliegende Arbeit ist nicht schon einmal in einem früheren Promotionsverfahren eingereicht worden.

Berlin im September 2012.