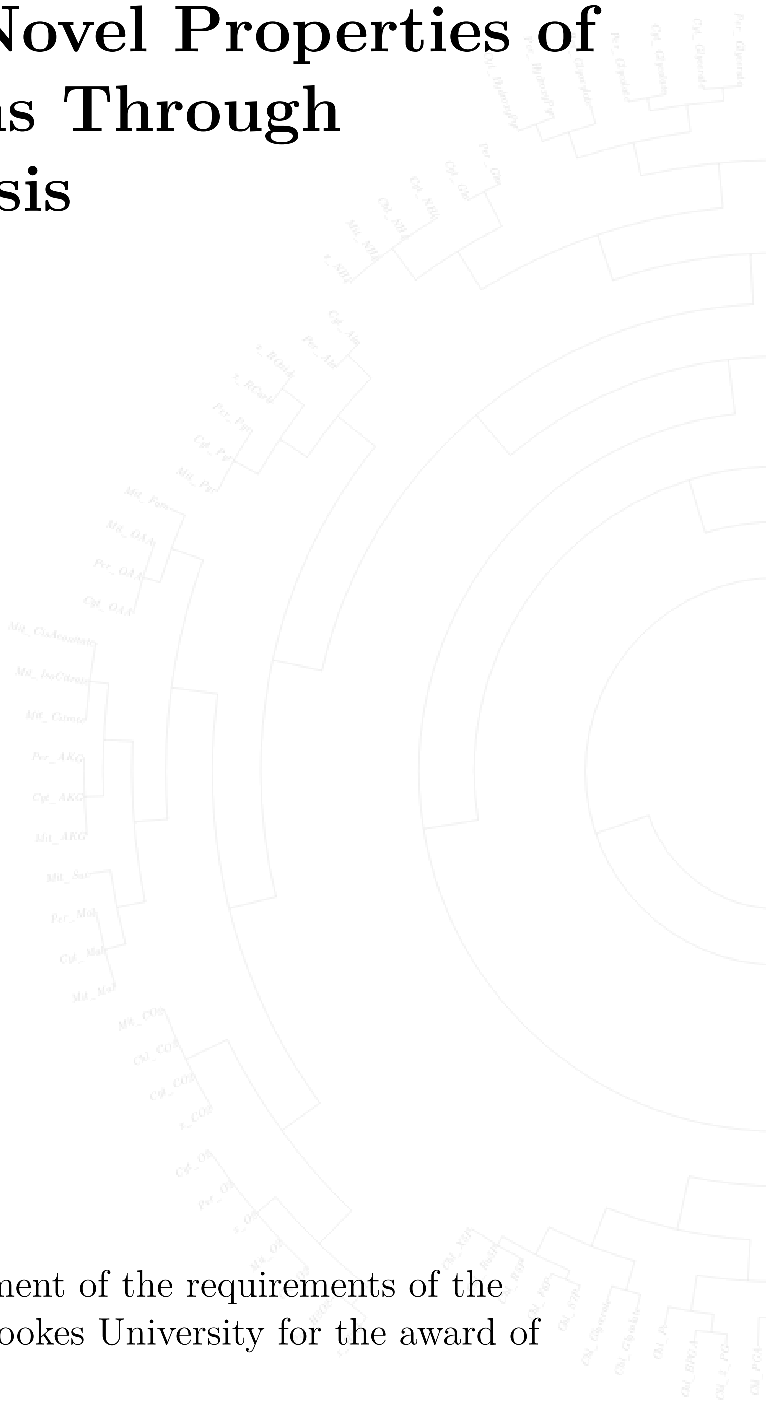
Identification of Novel Properties of Metabolic Systems Through Null-Space Analysis

Yanica Said



A thesis submitted in partial fulfilment of the requirements of the University of Malta and Oxford Brookes University for the award of Doctor of Philosophy.

August 2023

Supervisor / Co-Director of Studies: Professor Cristiana Sebu, University of Malta

Supervisor / Director of Studies: Dr Mark Poolman, Oxford Brookes University

Second Supervisor: Emeritus Professor David Fell, Oxford Brookes University

L-Università ta' Malta

FACULTY/INSTITUTE/CENTRE/SCHOOL__________Science__________

DECLARATION OF AUTHENTICITY FOR DOCTORAL STUDENTS

(a) Authenticity of Thesis/Dissertation

I hereby declare that I am the legitimate author of this Thesis/Dissertation and that it is my original work.

No portion of this work has been submitted in support of an application for another degree or qualification of this or any other university or institution of higher education.

I hold the University of Malta harmless against any third party claims with regard to copyright violation, breach of confidentiality, defamation and any other third party right infringement.

(b) Research Code of Practice and Ethics Review Procedure

I declare that I have abided by the University's Research Ethics Review Procedures. Research Ethics & Data Protection form code __________7253_30112020_Yanica Said__________.

☒ As a Ph.D. student, as per Regulation 66 of the Doctor of Philosophy Regulations, I accept that my thesis be made publicly available on the University of Malta Institutional Repository.

□ As a Doctor of Sacred Theology student, as per Regulation 17 (3) of the Doctor of Sacred Theology Regulations, I accept that my thesis be made publicly available on the University of Malta Institutional Repository.

□ As a Doctor of Music student, as per Regulation 26 (2) of the Doctor of Music Regulations, I accept that my dissertation be made publicly available on the University of Malta Institutional Repository.

□ As a Professional Doctorate student, as per Regulation 55 of the Professional Doctorate Regulations, I accept that my dissertation be made publicly available on the University of Malta Institutional Repository.

Abstract

Metabolic models provide a mathematical description of the complex network of biochemical reactions that sustain life. Among these, genome-scale models capture the entire metabolism of an organism, by encompassing all known biochemical reactions encoded by its genome. They are invaluable tools for exploring the metabolic potential of an organism, such as by predicting its response to different stimuli and identifying which reactions are essential for its survival. However, as the understanding of metabolism continues to grow, so too has the size and complexity of metabolic models, making the need for novel techniques that can simplify networks and extract specific features from them ever more important.

This thesis addresses this challenge by leveraging the underlying structure of the network embodied by these models. Three different approaches are presented. Firstly, an algorithm that uses convex analysis techniques to decompose flux measurements into a set of fundamental flux pathways is developed and applied to a genome scale model of *Campylobacter jejuni* in order to investigate its absolute requirement for environmental oxygen. This approach aims to overcome the computational limitations associated with the traditional technique of elementary mode analysis.

Secondly, a method that can reduce the size of models by removing redundancies is introduced. This method identifies alternative pathways that lead from the same start to end product and is useful for identifying systematic errors that arise from model construction and for revealing information about the network's flexibility.

Finally, a novel technique for relating metabolites based on relationships between their concentration changes, or alternatively their chemical similarity, is developed based on the invariant properties of the left null-space of the stoichiometry matrix. Although various methods for relating the composition of metabolites exist, this technique has the advantage of not requiring any information apart from the model's structure and allowed for the development of an algorithm that can simplify models and their analysis by extracting pathways containing metabolites that have similar composition. Furthermore, a method that uses the left null-space to facilitate the identification of un-balanced reactions in models is also presented.

Astratt

Il-mudelli metaboliċi huma deskrizzjoni matematika tan-netwerks tar-reazzjonijiet bijokimiċi li jsostnu l-ħajja. Fost dawn, il-*genome scale models* jiddeskrivu l-metaboliżmu komplet ta' organiżmu billi jinkludu r-reazzjonijiet bijokimiċi li ġew moqrija mil-ġenetika tiegħu. Dawn il-mudelli huma għodod imprezzabbli għall-esplorazzjoni tal-potenzjal metaboliku ta' organiżmu, per ezempju, billi jbassru ir-rispons tiegħu għal stimuli ambjentali jew jidentifikaw reazzjonijiet li huma essenzjali għal-ħajja. Madankollu, kif l-għarfien tal-metaboliżmu qed ikompli jikber, hekk ukoll id-daqs u l-kumplessità ta' dawn il-mudelli. Dan l-fatt jagħmel l-ħtieġa għal tekniki ġodda li jistgħu jissimplifikaw in-netwerks u jestrattaw karatteristiċi minnhom aktar importanti.

Din it-teżi tindirizza din l-isfida bi tliet modi differenti. L-ewwel metodu hu algoritmu li jiddekomponi imġieba metabolika f'numru ta' *pathways* fundamentali. Dan l-algoritmu hu applikat fuq *genome scale model* ta' *Campylobacter jejuni* sabiex jigi studjat l-effetti tas-saturazzjoni tal-ossiġnu ambjentali fuq dan l-*bacteria*. Dan l-algoritmu għandu l-għan li jegħleb il-limitazzjonijiet komputazzjonali assoċjati mat-teknika tradizzjonali tal-*elementary modes*.

It-tieni metodu 'jnaqqas id-daqs tal-mudelli billi jidentifika u jneħħi *redundancies* fin-*null-space* tal-lemin tal-matriċi tal-istojkjometrija. B'hekk jidentifika *pathways* alternattivi li jwasslu għall-istess prodotti. Dan l-metodu huwa utli biex jiġu identifikati żbalji sistematiċi fill-kostruzzjoni tal-mudelli, u biex tiġi żvelata informazzjoni dwar il-flessibilità tan-netwerks.

Fl-aħħarnett, teknika ġdida biex tirrelata l-metaboliti, bbażata fuq relazzjonijiet bejn il-bidliet fil-konċentrazzjoni tagħhom, jew alternattivament ix-xebh kimiku tagħhom, hi deskritta. Din tuza' l-proprjetajiet invarjanti tan-*null-space* tax-xellug tal-matriċi tal-istojkjometrija. Għalkemm jeżistu diversi metodi simili, din it-teknika għandha l-vantaġġ li ma teħtieġ l-ebda informazzjoni minbarra l-istruttura tal-mudell u b'hekk ippermettiet l-iżvilupp ta' algoritmu ġdid li jista' jissimplifika l-mudelli u l-analiżi tagħhom billi jestratta *pathways* li fihom metaboliti li għandhom kompożizzjoni simili. Barra minn hekk, metodu li juża n-*null-space* tax-xellug biex jidentifika mudelli li fijhom reazzjonijiet li mhumiex ibbilanċjati b'mod tajjeb hu wkoll ppreżentat.

Acknowledgements

I would first like to thank my supervisors, Prof. Cristiana Sebu and Dr Mark Poolman. I am very grateful for their help and guidance. Prof. Sebu and Dr Poolman always took time out of their busy schedule to answer my questions, offer suggestions, and share ideas. Thanks also to Prof. David Fell and Dr Dipali Singh for their invaluable advice, as well as the other members of the Cell Systems Modelling Group at Oxford Brookes University and Department of Mathematics at the University of Malta.

I must express my profound gratitude to my friends: the #PHY3569, *cafe del ħmar*, and *Sheldonian* crews, my Oxford house-mates (human and feline), my colleagues at Oxford Brookes and the University of Malta, friends from Innotargets, and collaborators at Quadram. Thank you for filling my life with joy, laughter, and inspiration, and thank you for always being there when I needed help.

My deepest appreciation goes to my immediate and extended family, and my *miskin*, for their continuous encouragement and for teaching me about what is most important in life. No words can describe how lucky I am to have you — thank you for everything.

The research work disclosed in this thesis is jointly funded by the Tertiary Education Scholarships Scheme, awarded by the Government of Malta, and the Nigel Groome Studentship, awarded by Oxford Brookes University. In this regard, I would especially like to thank Prof. Nigel Groome and Prof. Susan Brooks from the Faculty of Health and Life Science at Oxford Brookes University. I would also like to thank the administrative and estates teams of both universities for their assistance, the Electromagnetics Research Group at the University of Malta for providing me with desk space, as well as EMBL, the UK Biochemical Society, and The Alan Turing Institute for funding my participation at conferences and training events.

Publications

Yanica Said, Dipali Singh, Cristiana Sebu, Mark Poolman. A Novel Algorithm to Calculate Elementary Modes: Analysis of *Campylobacter jejuni* Metabolism, preprint on bioRxiv (under review at *BioSystems*), 2022.

Additional publications unrelated to this thesis:

Meghna Asthana, Robert Blackwell, Sam Davis, Jessica Forsyth, Kasia Kedzierska, Rafael Mestre, Zarreen Reza, Joseph Ribeiro, Pirta Palola, **Yanica Said**, Anna Downie. Center for Environment, Fisheries and Aquaculture Science: Automated identification of sea pens using OpenCV and machine learning, *The Alan Turing Institute Data Study Group Reports*, (to be published in) 2023.

Declaration of Authorship

The work presented in this thesis is the author's own, incorporating direction and feedback from her supervisors. In addition, Dr Dipali Singh (Quadram Institute, Norwich) gave feedback on the biological interpretation of the results involving *C. jejuni* in Chapter 3 and provided Figures 3.5 and 3.6.

Contents

List of Figures

List of Tables

List of Abbreviations

CSM	Cell Systems Modelling Group at Oxford Brookes University
EC	Enzyme Commission
EM	Elementary Mode
EMA	Elementary Modes Analysis
FBA	Flux Balance Analysis
FVA	Flux Variability Analysis
GSM	Genome Scale Model
KEGG	Kyoto Encyclopedia of Genes and Genomes
LP	Linear Programming
MILP	Mixed Integer Linear Programming
SMILES	Simplified Molecular Input Line Entry Specification
SVD	Singular Value Decomposition
tsv	Tab Separated File
WPGMA	Weighted Pair Group Method Using Arithmetic Averaging

List of Symbols

$\mathbf{A}$	Elemental composition matrix
$\mathbf{a}$	Elemental composition vector
C	Flux cone (steady-state solution space)
$\mathbf{E}$	EM matrix
$\mathbf{e}$	EM vector
$\mathbf{G}$	Left null-space matrix
$\mathcal{G}$	External left null-space matrix
$\mathbf{g}$	Left null-space vector
$\mathbf{K}$	Right null-space matrix
$\mathbf{k}$	Right null-space vector
$\mathbf{N}$	Stoichiometry matrix
$\mathcal{N}$	External stoichiometry matrix
$\mathbf{O}$	Zero matrix
P	Flux polyhedron (steady-state solution space with added constraints)
$\mathbf{s}$	Metabolite concentrations vector
$\mathbf{v}$	Flux vector
r_{uv}	Pearson sample correlation coefficient between the vectors $\mathbf{u}$ and $\mathbf{v}$
$\mathbf{R}$	Correlation matrix, where each entry, r_{ij}, is the correlation between metabolites i and j
ρ_{uv}	Pearson correlation coefficient of $\mathbf{u}$ and $\mathbf{v}$, where $\mathbf{u}$ and $\mathbf{v}$ are each a vector of scalars
σ_u	Standard deviation of the elements of $\mathbf{u}$, where $\mathbf{u}$ is a vector of scalars
σ_u^2	Variance of the elements of $\mathbf{u}$, where $\mathbf{u}$ is a vector of scalars
θ_{uv}	Angle between the vectors $\mathbf{u}$ and $\mathbf{v}$
ϕ_{uv}	Cosine of θ_{uv}
$\mathbf{Z}$	Maximal conserved moiety matrix
$\mathbf{z}$	Maximal conserved moiety vector

Chapter 1

Introduction

L-Università ta' Malta

University of Malta Library – Electronic Thesis & Dissertations (ETD) Repository

The copyright of this thesis/dissertation belongs to the author. The author's rights in respect of this work are as defined by the Copyright Act (Chapter 415) of the Laws of Malta or as modified by any successive legislation.

Users may access this full-text thesis/dissertation and can make use of the information contained in accordance with the Copyright Act provided that the author must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the prior permission of the copyright holder.

The processes amongst living beings are all centred around a common goal: the assimilation of resources from the environment to drive growth, generate energy, and eliminate waste. To this end, all living processes happen via chemical reactions, described and studied within the field of metabolism. These reactions form a metabolic network in which every reaction is connected to many others via the chemicals it acts upon, such that the output of one reaction serves as the input of some others. They behave collectively as a system, interacting with one another in complex ways. As a result, understanding the behaviour of metabolic systems is a challenging task, which requires identifying the network's components and characterizing how they are interconnected.

Only a few reactions occur spontaneously in living beings; the majority are catalysed by specialised proteins called enzymes [Campbell and Farrell, 2009, Chapter 6]. Therefore, cellular metabolism can be described as a series of enzyme-catalysed reactions that convert substrates into products, at a rate known as metabolic flux. Every cell contains thousands of different types of enzymes, most of which are extremely precise molecular entities, capable of promoting only one or a particular few chemical reactions and being likewise associated with specific identifiable genes in genetic code. As a consequence, a cell's metabolic network can be inferred from its genetic code [Pitkänen et al., 2010].

In the latter decades of the 20th century, considerable time and effort were devoted to characterising the behaviour of various enzymes and discovering the genes that encode them. More recently, the rise of *omics* technologies has made this task much simpler, allowing for the determination of genetic sequences, thus enabling biologists with the means to map a wide variety of metabolic networks, from simple pathways containing a few reactions to large datasets encompassing entire cells and microorganisms.

The resulting data is used both to understand the mechanisms behind metabolic phenomena and to predict how environmental changes and modifications to the network's structure would affect metabolic behaviour. However, such insights are challenging to come by, owing to the ambiguity and complexity often exhibited by these large datasets.

Conventional approaches attempt to clarify our interpretation of metabolism by partitioning networks into a set of fundamental pathways: groups of reactions that collectively transform one or more input metabolites into one or more output metabolites, thus defining a specific metabolic function. Although this reasoning is useful for designating specialised functions, it ignores the fact that individual reactions are rarely members of a single pathway, and therefore, the components of one pathway can interact with the components of many others. Indeed, the degree of control that a given reaction exerts over the flow of metabolites within a system cannot be predicted by studying its properties in isolation [Thomas and Fell, 1998; Moreno-Sánchez et al., 2008]. For example, when one reaction is perturbed, the metabolic network tends to compensate for the disruption—often in ways that traditional methods find hard to predict. This is a major problem in biotechnology, and in fact, it has been argued that our ability to edit genes is far superior to our ability to predict how the changes will manifest [Fell, 1997, Page 2].

Systems Biology attempts to address this issue by studying the phenomena that emerge from the organised interactions within a network, revealing patterns and relationships that are not evident when observing components individually. These features are referred to as emergent properties and their study is made possible through designing metabolic models that can mimic metabolic behaviour.

Metabolic models are mathematical representations of these complex systems—established through a collection of equations that detail the network's components i.e. the reactions and metabolites, and their way of interaction. Every reaction can be described by the quantitative relationship between the chemical compounds involved (referred to as the reaction's stoichiometry), or by means of rate equations. Once a complete set of reactions has been determined, this information is embedded into the model. Specifically, flux changes are quantified by a series of ordinary differential equations that express the rate of change of the concentration of each metabolic compound in terms of the reactions that consume and produce it.

Models that contain only the stoichiometric information described above are referred to as a structural models. These models embed the rules and conditions that arise from the underlying structure of the network. On the other hand, kinetic models contain additional parameters that incorporate properties such as enzyme rate equations. They allow metabolic systems to evolve from a given initial state by describing how flux and concentration values change over time. However, their applicability is limited because many of the parameters required to construct kinetic models are difficult to obtain. Furthermore, the results of such analysis can be challenging to interpret [Poolman et al., 2004a].

This thesis focuses solely on structural modelling, although every kinetic model must implicitly also contain a structural model.

Once a metabolic model is complete, algorithms condense the complex web of interactions into reliable information that is easy to understand, act upon, and manipulate. In particular, computational simulations can predict how altering some component (such as removing a reaction or specific nutrient source) would affect metabolic behaviour [Gu et al., 2019; Fang et al., 2020]. For example, the deletion of a set of reactions might redirect metabolic flux toward increased production of a desired product [Fatma et al., 2018].

Established structural modelling techniques depend only on the assumption that metabolite flow is constant while the cell is in a steady-state (i.e. the rates of production are counterbalanced by equal rates of consumption, such that no metabolite is indefinitely accumulating within the cell). One particularly useful technique, Elementary Modes Analysis (EMA), details the flux distributions that a network's structure can achieve at steady-state, thus revealing insights on flexibility and metabolic vulnerabilities such as essential genes [Schuster et al., 1999; Schäuble et al., 2011]. Each mode can be understood as a minimal independent pathway, such that any steady-state flux distribution of the system can be constructed via a non-negative linear combination of Elementary Modes (EMs).

In addition, a method called Flux Balance Analysis (FBA), uses Linear Programming (LP) methods to search for the steady-state pathway that best achieves an objective whilst satisfying a set of user-defined constraints (for example, maximizing growth whilst limiting nutrient uptake) [Fell and Small, 1986; Orth and Palsson, 2010]. This technique is often used to determine which reactions are likely to be active when the cell is pursuing a certain goal, and to predict the effects of nutrient/enzyme deficiency.

Using computational simulations is cheaper, safer, much faster, and more ethical than conducting real-world experiments, thereby equipping biologists with ways to make more informed choices later in the lab. Amongst various uses, structural models have been exploited to design strategies for the genetic manipulation of micro-organisms such that they produce valuable products, identify potential drug targets by finding vulnerabilities in disease causing microbes, and to design growth media by predicting the effects that different nutrient combinations have on the growth and secretions of microorganisms like algae and bacteria [Tejera et al., 2020]. For example, Hartman et al. [2014], constructed a genome scale model (GSM) of a *Salmonella typhimurium* which allowed for the identification of reactions whose removal interferes with the organisms ability to grow and generate energy, suggesting that inhibiting these reactions might severely interfere with the organisms ability to respond to anti-biotic challenges or other stress factors. Other applications concern third generation biofuels that aim to generate bio-diesel from photosynthetic microorganisms [Jagadevan et al., 2018; Khan et al., 2019], as well as, using genetically engineered organisms to manufacture materials that would usually be produced from fossil fuels, such as butanediol (an industrial solvent) and bio-nylon [Biz et al., 2019; Van Dien, 2013].

Despite these advances, many challenges still remain, most of which call for multi-disciplinary solutions. One of the biggest obstacles is that many established techniques used to generate valuable insights from small models (tens of reactions), do not scale well for applications on larger systems (hundreds of reactions). For instance, EMA is a useful tool for understanding the underlying architecture of a system. However, this technique is faced with a combinatorial explosion and thus is impractical to apply to large models [Klamt and Stelling, 2002]. On the other hand, FBA is widely applicable but has been criticised for providing specific results that might omit important and useful information. For example, any solution obtained from LP is often one of many, equally optimal, alternative solutions, referred to as multiple optima, the complete set of which is difficult to calculate [Mahadevan and Schilling, 2003]. Knowledge of these optima is useful in cases where the genetic manipulation of one reaction is easier than the manipulation of another, among others. In addition, large models that are generated from online databases often contain various errors which are later corrected through a lengthy process called model curation. This process typically entails many manual steps that could benefit from the creation of novel error-detection algorithms.

1.1 Aims and Structure

The goal of this thesis is to explore novel methods to extract information from the stoichiometric structure of networks (structural modelling), influenced by the necessity for novel tools, which along with aspects from more established methods can generate useful insights whilst still being applicable to larger models.

To this end, this thesis contains the following chapters:

Chapter 2 introduces biochemistry and mathematical modelling methods. This is especially important since this thesis is aimed to be accessible to both biologists and mathematicians.

Chapter 3 describes a method that decomposes flux measurements into a set of EMs. This method addresses the lack of computational efficiency when applying EMA, which, as mentioned above, obtains minimal fundamental pathways that expose the underlying architecture of metabolic networks. Traditional methods aim to accomplish this task by first calculating the entire set of possible EMs attainable by the network. Although the results of such algorithms have a wide range of applications, this methodology is troublesome since enumerating all EMs is not practical in large systems. In contrast, the algorithm described in this chapter obtains a candidate set of EMs quickly without requiring the *a priori* enumeration of all of the EMs, therefore allowing aspects of EMA to be applied to GSMs.

This chapter concludes with the application of this algorithm to a GSM of the bacteria *Campylobacter jejuni* to study this organism's oxygen requirements, the cause of which is still unclear.

Chapter 4 explores methods that reduce the steady-state behaviour of models, by decreasing the number of involved parameters. Two such concepts have been considered.

One relates to the definition of biomass components in models, which are the organic compounds that a cell's metabolism must produce for it to grow. When simulating growth using FBA (where the objective is to minimize total flux), the model's potential behaviour is reduced by enforcing these components to be produced in the proportions in which they are needed by the organism. This can be accomplished in one of two ways: either through the definition of the model's output reactions, or, as part of the definition of the LP problem. Both approaches are helpful in directing FBA towards realistic results. However, determining which approach is the most suitable has been the subject of debate. Chapter 4 resolves this dilemma by showing that FBA solutions obtained from either method are mathematically equivalent.

The second concept regards the creation of an algorithm that reduces the size of models by eliminating redundancies in the right null-space. Historically, such techniques have been used to improve the efficiency of modelling algorithms. In addition to this advantage, the method proposed here also reveals network characteristics of interest, specifically, alternate pathways that lead from the same start to end products. This is achieved by the design of an algorithm that iteratively eliminates two different types of redundancies from the model (enzyme subsets and iso-stoichiometric groups), in contrast to the conventional technique by Pfeiffer et al. [1999] that only eliminate enzyme subsets.

As demonstrated in Chapter 4, this novel method facilitates model curation by identifying erroneous duplicate processes and reactions with incorrectly defined directionality, as well as aids model analysis by identifying redundant pathways and calculating multiple optima in FBA.

Chapter 5 introduces methods that identify information about metabolites by exploiting the left null-space. It has long been apparent that this space reveals conservation relations i.e. sets of metabolites whose sum of molar amounts must remain constant through time [Schuster and Hofer, 1991; Hofmeyr, 2020]. Although such an analysis is a useful way to establish relationships between metabolites, conservation relations are not uniquely defined as the left null-space may be represented by different sets of generating vectors.

The work presented in Chapter 5 addresses this issue by showing that the angles between the rows of the orthonormal left null-space are uniquely defined, thus allowing for the development of a 'similarity measure' that can relate metabolites independently of the choice of left null-space basis. Such a measure has many advantages, for example, by leading to the identification of conserved moieties.

In addition, since information relating the chemical composition of metabolites is embedded in the left null-space, this similarity measure was used to cluster metabolites based on their chemical composition and extract pathways that contain chemically similar metabolites. Notably, this technique relies solely on the network's structure and, therefore, does not require the input of external information about the chemical composition of metabolites. Such information is often difficult to obtain for all metabolites in a model. However, to cater for cases when only the composition of some metabolites is known, a method that integrates this known information with the left null-space to infer the composition of unknown metabolites and to identify the presence of unbalanced reactions is also presented.

Chapter 6 concludes this thesis by discussing how this work addresses the objectives describe above, its contributions to the field of metabolic modelling, and an outlook for future research.

Chapter 2

Theoretical Foundations

As the methods described in this thesis are focused on the analysis of metabolic networks, knowledge of biochemistry and linear algebra methods are essential. Therefore, this chapter contains the following sections:

Section 2.1 introduces cellular metabolism,

Section 2.2 discusses the reductionist approach to understanding metabolic pathways and its shortcomings,

Section 2.3 defines the mathematical definition of structural metabolic modelling,

Section 2.4 introduces the concept of large (genome scale) models,

Section 2.5 describes the software and models used in this thesis.

2.1 An Overview of Cellular Metabolism

Metabolism is a key concept in biochemistry—it refers to the collection of chemical reactions that supply organisms with the energy and compounds required to sustain life.

The energy that drives metabolism in most organisms is ultimately derived from sunlight. Photosynthetic organisms, including plants, algae, and certain bacteria, harvest light energy to fix atmospheric CO_2, thereby producing organic compounds, such as glucose, that serve to store energy and sustain growth.

When these photosynthetic organisms are then ingested as food, their constituent components are broken down into smaller compounds, thereby releasing the stored energy and supplying cells with the chemical building blocks needed to maintain and synthesise cellular components. Additional reactions allow cells to perform tasks such as eliminating waste and transmitting cellular signals.

2.1.1 Metabolites

The small molecules involved in metabolism are metabolites. They, along with the polymers (as defined below) that they combine to form, constitute the total amount of organic material within a cell, which is referred to as its biomass.

Organic compounds consist of carbon atoms linked together through covalent bonds, in combination with some other atoms, the most abundant being hydrogen, oxygen, nitrogen, phosphorus and sulfur. The type and number of atoms in a compound are described by its chemical formula. In addition to the composition, these compounds' physical and chemical properties also depend on their structure (i.e. the three-dimensional arrangement of atoms as defined by chemical bonding). In fact, isomers, meaning compounds which share an identical chemical formula but distinct atomic arrangements, do not necessarily exhibit the same behaviour.

There are four main categories of metabolites: carbohydrates, amino acids, nucleotides, and lipids [Nelson and Cox, 2004, page 16]. The first three categories can act as *monomers*, meaning that numerous compounds of the same category can combine together to create larger molecules called *polymers*.

Carbohydrates serve to store and release energy and to provide cells with the carbon atoms required for synthesising other cellular components. Carbohydrate monomers, the most common being glucose, can be distinguished by being multiples of the chemical formula CH_2O. They can combine to form polymers such as starch (the primary means of energy storage in plants, consisting of multiple glucose molecules).

Amino Acids consist of an amino group (-NH_2), a carboxylic acid group (-COOH), and an -R group which is specific to each type of amino acid. They combine together to create *proteins*. Although there are only 20 distinct proteinogenic amino acids, different permutations and linkages can generate an extremely large number of proteins, each with a specialised function. For example, enzymes catalyse chemical reactions, while keratin is a structural component of hair and nails.

Nucleotides contain a five-carbon sugar attached to a phosphate group and a nitrogenous base. Adenosine triphosphate (ATP) is a nucleotide that provides the energy needed to support many cellular processes. In addition, nucleotides can combine in chains to create two main types of polymers called *nucleic acids*: deoxyribonucleic acid (DNA) and ribonucleic acid (RNA). These polymers store and transmit genetic information, which is encoded as a specific sequence of nucleotide types, each distinguished by containing a different type of nitrogenous base. Genes are short segments of DNA that carry the instructions for synthesising a particular protein, in a process known as *gene expression*.

Lipids such as oils and fats, have diverse structures with a tendency to be generally insoluble in water. *Fatty acids* are lipids that consist of a head and a tail region, where the head contains a carboxyl group and attracts water, while the tail is a hydrocarbon chain that repels water. This property permits phospholipids—molecules that contain 2 fatty acid chains linked to a glycerol backbone—to aggregate when in water, making them important cellular membrane components. Other lipids, such as triglycerides, are used for long-term energy storage, while others, like steroids, are used for signalling.

Furthermore, metals and minerals are present in minor to trace amounts.

2.1.2 Chemical Reactions and Thermodynamics

The transfer of energy in the universe is governed by the fundamental laws of thermodynamics. Therefore, it is these laws that dictate whether a chemical reaction can occur.

Specifically, metabolites may spontaneously react if the transition from substrate to product is thermodynamically favourable, a criterion that depends on the structure of the involved metabolites, as well as variable conditions including metabolite concentrations, environmental temperature, and pressure. These factors are combined to calculate the equilibrium constant, K_{eq}, such that in a closed system, as a reaction progresses, the concentration ratio of substrate to product gradually decreases until reaching the value of K_{eq} [Campbell and Farrell, 2009, chapter 15]. At this point, the system reaches a state of equilibrium, and the substrate/product concentration remains constant. In accordance with this line of reasoning, some reactions are considered to be reversible, meaning they can proceed in either direction depending on the initial concentration ratio of the involved metabolites. In contrast, other reactions are irreversible since the concentrations required for the reverse direction cannot feasibly exist inside living cells. In addition, Hess's law states that the above-defined thermodynamic properties for given net conversion of metabolites are independent of the pathway through which the conversion occurs [Leicester, 1951]. Therefore, different reactions that when combined have the same net-stoichiometry must also have the same overall directionality.

The minimum amount of energy required for a reaction to take place is the activation energy, E_a. This energy is needed to break the existing chemical bonds within substrates, therefore allowing their atoms to rearrange and form new products. The activation energy determines the rate at which a reaction takes place, since, for example, a reaction with a higher E_a requires a greater amount of energy to initiate and so will occur at a slower rate.

2.1.3 Enzyme-Catalysed Reactions

If metabolites are left to react unaided most would not react at all, whilst others would do so at a rate that is too slow to be of use. Thus, specialised proteins called enzymes are needed to drive, regulate, and direct metabolism. Notably, enzymes can increase the rate of chemical reactions by a magnitude ranging from 10^8 to 10^{20} times the original.

Enzymes catalyse reactions by lowering the activation energy, E_a, needed for a reaction to take place. In addition, thermodynamically unfavourable reactions can be driven forward by coupling them with reactions that release energy. Commonly, this is achieved through the hydrolysis of ATP, a molecule that releases energy upon donating a phosphate group (a high-energy bond) to other molecules.

Several mathematical frameworks that define how reaction conditions influence an enzyme's rate of catalysis have been established [Campbell and Farrell, 2009]. Referred to as *enzyme kinetics*, these methods allow reaction rates to be estimated from knowledge of metabolite concentrations and specific enzyme characteristics. They form the basis of kinetic modelling (Section 1), which is beyond the scope of this thesis.

2.1.4 Controlling Metabolism

As discussed above, every reaction is catalysed by one or more enzymes. Therefore, enzymes are central in directing and regulating the flow of material (flux) within the cell. Apart from substrate/product concentration, an enzyme's activity is regulated through multiple factors such as the presence of effectors and co-factors, covalent modification, and gene expression [Campbell and Farrell, 2009, chapter 7].

Effectors are compounds that bind to an enzyme and alter its activity. Furthermore, most enzymes do not function unless assisted by specific molecules called co-factors. These are most often derived from vitamins and minerals, and together with effector molecules, regulate enzyme behaviour depending on the extent of their presence within the cell. In addition, some enzymes can be (de)activated by bonding with additional molecular groups via covalent bonds, a mechanism known as covalent modification.

Finally, the amount of enzymes synthesised within the cell is also essential—a cell may contain hundreds of thousands of copies of one enzyme but only a few copies of another. Therefore, *gene expression*, the mechanism by which the genetic information is used to synthesise proteins (including enzymes) functions as a highly regulated means for influencing metabolic activity.

2.2 Investigating Metabolism Through the Reductionist Approach

The ongoing study of cellular metabolism has identified an immense amount of metabolic processes occurring within living cells. Recent advances in genome sequencing and annotation, a process that determines the order of nucleotides in an organism's DNA and gives it meaning by deducing the enzymes (and corresponding reactions) encoded in the constituent genes, have contributed significantly to this development. As a consequence, databases, like KEGG[1] and BioCyc[2], that list the reactions in thousands of organisms are now available [Kanehisa and Goto, 2000; Karp et al., 2018].

The reductionist approach aims to understand networks by dividing them into smaller modules based on location (via cellular compartments) or specific roles (by assigning specialised pathways). For instance, several key pathways have been studied as being universally important for carrying out particular tasks. However, various problems arise when attempting to use a reductionist approach to predict metabolic behaviour, as is discussed further below.

[1] genome.jp/kegg

[2] biocyc.org

The first pathway to be elucidated was glycolysis [Meyerhof and Junowicz-Kocholaty, 1943], which extracts energy from glucose via a set of catabolic reactions that result in the three-carbon molecule, pyruvate. The energy released by these steps is stored in high-energy ATP and dihydronicotinamide adenine dinucleotide (NADH) molecules that can then be used by enzymes to drive other reactions. Similarly, the pyruvate and other intermediate molecules produced by glycolysis can be used by other pathways. For example, the citric acid cycle (TCA cycle) uses pyruvate, as well as other compounds derived from nutrients, to generate energy and pre-cursors for cellular components. Most of the energy produced by the TCA cycle feeds into the electron transport chain (ETC) pathway which consumes oxygen to generate further energy that is stored in the form of ATP. The creation of ATP from nutrients via the ETC is referred to as respiration.

Cells also contain many important anabolic pathways which synthesise cellular components from smaller compounds. For example, the pentose-phosphate pathway (in the cytosol) uses glucose to produce five-carbon sugars needed for the synthesis of nucleotides, and dihydronicotinamide adenine dinucleotide phosphate (NADPH) which provides the reducing power needed to drive other anabolic processes such as lipid synthesis. The fact that the pentose phosphate pathway has several enzymes in common with the Calvin cycle—the pathway that photosynthetic organisms use to convert light energy to organic compounds—demonstrates the interdependence of metabolic pathways. Indeed, the former pathway breaks down glucose and releases CO_2 and NADPH, whilst the latter uses ATP, NADPH and CO_2 to generate glucose.

Although the above description of pathways implies a degree of separation, it would be deceiving to assume that metabolic networks can be split into separate components that each perform a single function. In fact, most metabolites in a cell can participate in more than one pathway simultaneously, such that any change in the concentration of one metabolite will quickly propagate to affect many others. For example, the pyruvate produced in glycolysis can be converted into different substrates by many enzymes, resulting in different pathways competing for the same pyruvate molecule. The same is also true for many other small molecules within the cell.

It is therefore apparent that an understanding of a metabolic system requires the study

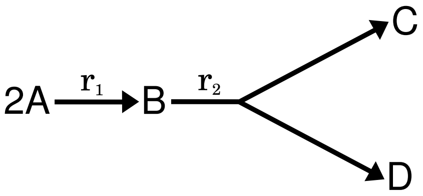


Figure 2.1: A simple metabolic network where A, B, C, and D are metabolites and r_1 and r_2 are reactions.

of the interactions between its components, which motivates the use of metabolic models.

2.3 Investigating Metabolism Through Structural Modelling

2.3.1 Defining Structural Models

The stoichiometry of a reaction specifies the type and number of metabolites participating in it. When defining a metabolic model, the individual stoichiometries of all the network's reactions are combined to form a single system of simultaneous equations that collectively describe the network's structure. As an example, consider the model in Figure 2.1. This metabolic network contains the two reactions:

$$\begin{aligned} r_1: &\quad 2\ \mathrm{A} \rightarrow \mathrm{B}, \\ r_2: &\quad \mathrm{B} \rightarrow \mathrm{C} + \mathrm{D}, \end{aligned} \tag{2.1}$$

where the first reaction, r_1, converts two moles of metabolite A (the substrate) into one mole of metabolite B (the product), whilst the second reaction, r_2, proceeds to split every metabolite B into equal amounts of C and D.

The network formed by these reactions is mathematically defined by characterising changes in metabolite concentrations as a set of ordinary differential equations (functions that describe the rate of change of a quantity with respect to time) [Heinrich and Schuster, 1996, pages 10-13]. Specifically, the rate of change in the concentration of each metabolite is calculated as the difference between the rate at which the metabolite is being produced and the rate at which it is being consumed. Therefore, denoting the flux of reactions r_1 and r_2 as v_1 and v_2 respectively, the metabolic model in Figure 2.1 can be described by the following set of equations:

$$\frac{\mathrm{d}a}{\mathrm{d}t} = -2v_1 \tag{2.2a}$$

$$\frac{\mathrm{d}b}{\mathrm{d}t} = v_1 - v_2 \tag{2.2b}$$

$$\frac{\mathrm{d}c}{\mathrm{d}t} = v_2 \tag{2.2c}$$

$$\frac{\mathrm{d}d}{\mathrm{d}t} = v_2, \tag{2.2d}$$

were a, b, c, and d denote the concentration of metabolites A, B, C, and D within the system, respectively.

The above equations can be equivalently represented in matrix form by introducing the stoichiometry matrix, $\mathbf{N}$, that collectively embodies the stoichiometry of all reactions in the network, such that each element of $\mathbf{N}$, n_{ij}, corresponds to the stoichiometric coefficient of metabolite i in reaction j:

$$\mathbf{N} = \begin{array}{c} \\ \mathrm{A} \\ \mathrm{B} \\ \mathrm{C} \\ \mathrm{D} \end{array} \begin{array}{c} \begin{array}{cc} r_1 & r_2 \end{array} \\ \begin{pmatrix} -2 & 0 \\ 1 & -1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \end{array}, \tag{2.3}$$

and therefore,

$$\begin{pmatrix} \frac{\mathrm{d}a}{\mathrm{d}t} \\ \frac{\mathrm{d}b}{\mathrm{d}t} \\ \frac{\mathrm{d}c}{\mathrm{d}t} \\ \frac{\mathrm{d}d}{\mathrm{d}t} \end{pmatrix} = \begin{pmatrix} -2 & 0 \\ 1 & -1 \\ 0 & 1 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}, \tag{2.4}$$

Denoting $\mathbf{s}$ as the vector of metabolite concentrations and $\mathbf{v}$ as a vector of flux values concisely represents this system as:

$$\frac{\mathrm{d}\mathbf{s}}{\mathrm{d}t} = \mathbf{N}\mathbf{v}. \tag{2.5}$$

In summary, the stoichiometry matrix is a transformation that converts a vector of flux values, $\mathbf{v}$, into a corresponding vector of varying metabolite concentrations, $\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t}$.

A practical advantage of this representation is its invariability. Indeed, although many variables are in a state of constant fluctuation, the underlying structure of the network remains constant. Thermodynamic and kinetic restrictions, such as whether a reaction is reversible, are also unchanging [Heinrich and Schuster, 1996].

The Quasi Steady-State Assumption

Many structural modelling techniques are based on the assumption that the environment within the cell exists in a dynamic steady-state, where the concentration of metabolites within the system stays constant. However, it is important to note that lack of change in metabolite concentrations does not imply that nothing is changing, but rather that the rates of synthesis are counterbalanced by equal rates of degradation, such that no metabolite is indefinitely accumulating within the cell.

Therefore, the quasi steady-state assumption equates the rate of change of the concentration of each metabolite to zero (i.e. $\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t} = \mathbf{0}$), leading to the algebraic system:

$$\frac{\mathrm{d}\mathbf{s}}{\mathrm{d}t} = \mathbf{N}\mathbf{v} = \mathbf{0}, \tag{2.6}$$

meaning that, every feasible steady-state flux distribution, $\mathbf{v}$, allowed by the mathematical structure of a metabolic network must satisfy the equation:

$$\mathbf{N}\mathbf{v} = \mathbf{0}. \tag{2.7}$$

External Metabolites

Metabolic networks facilitate the conversion of nutrients (or other inputs) into valuable outputs that contribute to the cell's functioning. Thus, networks that represent living systems contain two types of metabolites: the *internal* metabolites described above, whose concentration can dynamically evolve through time, and the *external* input/output metabolites, that provide the matter/energy needed to sustain the network's metabolic activity. These external metabolites are transferred between the model and its environment via transporter reactions. Their concentrations are assumed to remain constant and

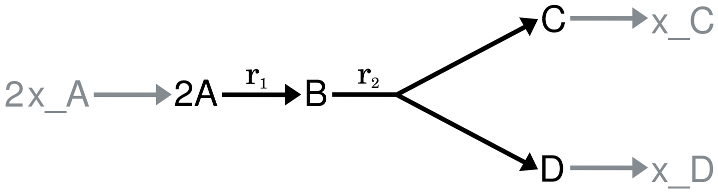


Figure 2.2: A simple metabolic network where A, B, C, and D are internal metabolites, x_A, x_D, and x_C are external metabolites, r_1 and r_2 are internal reactions, and transporter reactions are shown in grey.

therefore they are not included in **N** [Sauro and Ingalls, 2003].

Through these definitions, **N**, can be referred to as the *internal* stoichiometry matrix. Correspondingly, the *external* stoichiometry matrix, denoted in this thesis as $\mathcal{N}$, extends the rows of **N** to include external metabolites, which, in this thesis, are denoted with an 'x_' prefix.

For example, the model in Figure 2.1 can be modified to include the external metabolites x_A, x_C, and x_D, and corresponding transport reactions as shown in Figure 2.2.

The reaction, *A_tx*, that transports metabolite A into the system can be represented by:

$$A\_tx: \qquad 2\,\mathrm{x\_A} \rightarrow 2\,\mathrm{A}. \tag{2.8}$$

Whilst similar reactions exporting C and D are:

$$\begin{aligned} C\_tx: &\qquad \mathrm{C} \rightarrow \mathrm{x\_C} \\ D\_tx: &\qquad D \rightarrow \mathrm{x\_D}, \end{aligned} \tag{2.9}$$

Hence, the internal stoichiometry of this network matrix is:

$$\mathbf{N} = \begin{array}{c} \\ \mathrm{A} \\ \mathrm{B} \\ \mathrm{C} \\ \mathrm{D} \end{array} \begin{array}{c} \begin{array}{ccccc} \mathrm{A}\_tx & r_1 & r_2 & \mathrm{C}\_tx & \mathrm{D}\_tx \end{array} \\ \left(\begin{array}{ccccc} 2 & -2 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{array}\right) \end{array}, \tag{2.10}$$

and the corresponding external stoichiometry matrix is:

$$\mathcal{N} = \begin{array}{c|ccccc|} & \text{A\_}tx & r_1 & r_2 & \text{C\_}tx & \text{D\_}tx \\ \text{x\_A} & -2 & 0 & 0 & 0 & 0 \\ \text{A} & 2 & -2 & 0 & 0 & 0 \\ \text{B} & 0 & 1 & -1 & 0 & 0 \\ \text{C} & 0 & 0 & 1 & -1 & 0 \\ \text{D} & 0 & 0 & 1 & 0 & -1 \\ \text{x\_C} & 0 & 0 & 0 & 1 & 0 \\ \text{x\_D} & 0 & 0 & 0 & 0 & 1 \end{array}. \tag{2.11}$$

Note that **N** denotes an open system that continuously interacts with the environment by exchanging inputs and outputs, whilst the system denoted by $\mathcal{N}$ is closed.

2.3.2 Null-Space Analysis

Once a stoichiometry matrix is established, various methods can be applied to extract information from the network, many of which make use of the null-space (or kernel).

Every matrix is associated with two different null spaces: the right and the left null-spaces. Their use is made possible by following two distinct assumptions about metabolic behaviour: the 'quasi steady-state assumption' and the 'mass conservation rules' respectively, as discussed below.

The Right Null-Space

The number of possible steady-state flux vectors, **v**, (i.e. the vectors, **v**, that satisfy $\mathbf{Nv} = \mathbf{0}$) is infinite, but can be defined by calculating the right null-space (or kernel) matrix of **N** using linear algebra techniques [Heinrich and Schuster, 1996, Section 3.2].

The right null-space matrix, **K**, often referred to as simply the null-space, satisfies the equation:

$$\mathbf{NK} = \mathbf{O}, \tag{2.12}$$

where **O** denotes a zero matrix. The columns of **K** consist of a set of linearly independent vectors called a basis. These vectors span the set of steady-state flux vectors of the system, such that any steady-state flux vector (i.e. a solution, $\mathbf{v}$, of $\mathbf{Nv} = \mathbf{0}$) can be generated by a linear combination of the column vectors of $\mathbf{K}$.

For example, consider the model in Figure 2.2, a right null-space matrix of the corresponding stoichiometry matrix is:

$$\mathbf{K} = \begin{array}{c} \text{A\_}tx \\ r_1 \\ r_2 \\ \text{C\_}tx \\ \text{D\_}tx \end{array} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}, \tag{2.13}$$

such that,

$$\mathbf{NK} = \begin{array}{c|ccccc} & \text{A\_}tx & r_1 & r_2 & \text{C\_}tx & \text{D\_}tx \\ \text{A} & 2 & -2 & 0 & 0 & 0 \\ \text{B} & 0 & 1 & -1 & 0 & 0 \\ \text{C} & 0 & 0 & 1 & -1 & 0 \\ \text{D} & 0 & 0 & 1 & 0 & -1 \end{array} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}. \tag{2.14}$$

Therefore, whenever the metabolic network described by Figure 2.2 is carrying a steady-state flux, all of the reactions in the network must be equal to each other.

It is important to note that the set of basis vectors spanning the null-space is not unique, i.e. $\mathbf{K}$ can be represented through different sets of vectors which span the same space.

Mathematically, the right null-space matrix embodies the linear dependencies between the columns of $\mathbf{N}$ (i.e. the reactions), such that its dimension corresponds to the number of linearly dependent columns in $\mathbf{N}$, as stated by the fundamental theorem of linear algebra. In fact, let $\mathbf{k}$ be a column vector of the right null-space matrix. Then, the entry, k_i, of $\mathbf{k}$ corresponds to the ith column of $\mathbf{N}$, $\mathbf{n}_i$, such that a linear combination of the columns of

$\mathbf{N}$ with the corresponding entries of $\mathbf{k}$ as coefficients sum to a vector of zeroes:

$$k_1\mathbf{n}_1 + k_2\mathbf{n}_2 + ... + k_r\mathbf{n}_r = \mathbf{0}, \tag{2.15}$$

where r is the number of reactions in $\mathbf{N}$.

Therefore, any column of $\mathbf{N}$ corresponding to a non-zero entry in $\mathbf{k}$, can be obtained through a linear sum of some other columns in $\mathbf{N}$.

Further properties of this space are discussed in Section 2.3.3

The Left Null-Space

An important feature of metabolic interaction is the presence of constraints arising from conservation laws. In a closed system, the flow of material, energy, and redox potential must be conserved at each step [Smith and Missen, 1979; Sauro, 2012]. These constraints lead to mathematical relationships between metabolites that influence the dependence between the rows of $\mathbf{N}$, which in turn, determines the left null-space matrix, $\mathbf{G}$.

This matrix is the kernel of the transpose of the stoichiometry matrix, such that:

$$\mathbf{N}^\top\mathbf{G} = \mathbf{O}, \tag{2.16}$$

or equivalently:

$$\mathbf{G}^\top\mathbf{N} = \mathbf{O}. \tag{2.17}$$

Much like the right null-space, $\mathbf{G}$ is not uniquely defined and its columns span the space of vectors, $\mathbf{g}$, that satisfy:

$$\mathbf{N}^\top\mathbf{g} = \mathbf{0}. \tag{2.18}$$

Every element of such vectors, g_i, corresponds to a metabolite of the system, and it is through this reasoning that $\mathbf{G}$ is seen to reveal conservation relations i.e. groups of metabolites whose linear combination of concentrations remains constant through time [Reder, 1988; Stelling and Klamt, 2006; Schuster and Fell, 2007]. Conservation relations shrink the possible dynamic behaviour of the network: if at the beginning of an experiment the value for the total concentration of metabolites in a conservation relation is known, then this value will stay constant at all times.

For example, consider a left null-space matrix for the model in Figure 2.1:

$$\mathbf{G} = \begin{pmatrix} 1 & 0 \\ 2 & 0 \\ 2 & 1 \\ 0 & -1 \end{pmatrix}, \tag{2.19}$$

which results in the following conservation relations:

$$a(t) + 2b(t) + 2c(t) = \lambda_1, \tag{2.20}$$

and

$$c(t) - d(t) = \lambda_2. \tag{2.21}$$

where a, b, c and d are the concentrations of metabolites A, B, C, and D respectively and λ_1 and λ_2 are constants.

Note that such relationships between metabolites equate to metabolite concentrations in models of single compartments. However, relating concentrations in multi-compartment models requires the relative volume of the different compartments to be taken into account, as described by Hofmeyr [2020].

The identification of conservation relations is essential for the application of kinetic modelling techniques, as many of these methods require a non-singular Jacobian matrix to be calculated from the stoichiometry matrix, which is only possible if $\mathbf{N}$ has full row-rank (a criterion that is ensured by eliminating the linearly dependent rows as identified by the conservation relations [Vallabhajosyula et al., 2006]).

2.3.3 Modelling Steady-State Behaviour

Many metabolic modelling methods exploit properties of the right null-space to reveal insights into the capabilities of an organism's steady-state behaviour. Prominent methods include: the identification of dead reactions, the grouping of coupled reactions into modules called enzyme subsets [Jevremovic et al., 2011; Pfeiffer et al., 1999], flux correlation analysis [Poolman et al., 2004b], and elementary modes analysis (EMA) [Schuster et al., 2000; Gagneur and Klamt, 2004].

2.3.3.1 Enzyme Subsets

Enzyme subsets, as described by Pfeiffer et al. [1999], are groups of reactions that always carry proportional fluxes at a fixed ratio when the system is at steady-state. Therefore, knowledge of the flux of one of the subsets' reactions allows for the calculation of all other fluxes of the subset.

Enzyme subsets can be identified as proportional rows of the right null-space matrix and arise from redundancies within the columns of **N**. For example, consider the right null-space matrix, **K** (Equation 2.13), of the model in Figure 2.2. Since **K** consists of one column corresponding to the vector k_1, the reactions within the model must always carry flux at the proportions specified by k_1 for the system to be at steady-state.

Once enzyme subsets are identified, the stoichiometry matrix can be reduced in size by replacing the reactions of each subset with a single reaction that embodies the subset's overall (or net) stoichiometry. This transformation does not discard any information about steady-state flux, and is useful to improve the efficiency of algorithms that perform calculations on **N**.

Enzyme subsets expose structural couplings between reactions of metabolic networks, which can then be used to deduce details about the underlying regulatory mechanisms of the network [Gagneur and Klamt, 2004; Schuster et al., 2002]. Furthermore, since any increase in the flux of one reaction in a subset is accompanied by a proportional increase in the flux of all other members, enzyme subsets have become useful tools both for metabolic engineering and experimental measurements. For example, the flux of a target reaction may be dramatically increased by causing a small increase in the flux of another reaction. Similarly, deactivating one reaction of the subset concurrently stops steady-state flux from passing through the entire subset, facilitating the development of gene knock-out strategies as the deletion of any enzyme of the subset brings about the same effect (equivalent knock-outs) [Burgard et al., 2004].

2.3.3.2 Dead Reactions

Inconsistencies that arise when defining metabolic models may lead to some reactions being unable to carry flux at steady-state, that therefore should be removed from the model prior to analysis. There are two types of such reactions: (i) reactions identified as rows of zeroes in the right null-space, and (ii) reactions that form part of inconsistent enzyme subsets.

Any reaction that corresponds to a row of zeroes in the right null-space matrix, **K**, can never be assigned flux at steady state since every steady-state flux vector of the network must be created from a linear combination of the columns of **K**. Furthermore, if an enzyme subset is inconsistent, then its enzymes have conflicting thermodynamic constraints such that they cannot operate as required by the subset when at steady-state, leading to every reaction of such subsets being inactive.

2.3.3.3 Steady-State Flux Correlation Analysis

Another way to study the null-space is by analysing the correlation between its rows. In the case of the right null-space, such analysis allows reactions to be hierarchically clustered into modules based on the similarity between their potential steady-state fluxes. Poolman et al. [2004b], used this approach to extend the notion of enzyme subsets by exploiting the fact that the cosine of the angles between the rows of an orthogonal basis for the right null-space, **K**, is equivalent to Pearson's correlation coefficient between all possible steady-state fluxes spanned by **K**.

2.3.4 Elementary Modes Analysis

Describing the steady-state behaviour of a given network through the right null-space is not ideal. Indeed, the vectors in **K** do not necessarily satisfy reaction reversibility criteria (since irreversible reactions can have a negative flux in **K**) and the associated basis cannot be uniquely defined.

These shortcomings are remedied by elementary modes analysis (EMA), a technique that constructs a steady-state flux basis in a way that is most biologically meaningful. Specifically, as described by Schuster et al. [1999], elementary modes (EMs), consist of the irreducible elements of a network that much like the null-space can be combined to form all possible steady-state pathways. In addition, any EM basis is uniquely defined and adheres to reaction reversibility constraints such that any steady-state flux vector, $\mathbf{v}$, obtained from a non-negative linear combination of EMs is stoichiometrically balanced and carries flux in a thermodynamically feasible direction. For example, the model in Figure 2.3 can be decomposed in three EMs that give rise to all of the model potential steady-state behaviour.

EMs are defined as minimal sets of reactions able to operate as a continuous pathway at steady-state, with the respective enzymes weighted by the relative flux that they need to carry for the EM to function [Schuster et al., 2000]. In the context of EMs, the *minimal* property states that a given EM cannot be decomposed into a set of simpler modes since deleting any enzyme will prevent the mode from operating at steady-state.

Definition 2.3.4.1. *A non-negative (or conic) linear combination of a set of vectors* $\mathbf{E} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_t\}$ *is a weighted sum* $\lambda_1\mathbf{e}_1 + \lambda_2\mathbf{e}_2 + \cdots + \lambda_t\mathbf{e}_t$*, where the* λ_i *are non-negative constants. Therefore, let* $\mathbf{E}$ *represent the set of EMs of a given network, then any steady-state vector* $\mathbf{v}$ *can be represented as a non-negative linear combination of the vectors in* $\mathbf{E}$*.*

EMs facilitate the determination of paths that lead from a specific starting material to the desired product, and hence, are widely used for studying the metabolic capabilities of organisms [Trinh et al., 2009; Unrean et al., 2019; Khan et al., 2018]. For example, when engineering recombinant bacteria to produce cyanophycin, EMA was used to study the effect of different carbon sources and oxygen on production [Cardoso Diniz et al., 2006]. This was done by running EMA for each carbon source and evaluating the EMs which were involved in cyanophycin production.

2.3.4.1 The Flux Cone

EMA is achieved by defining the steady-state solution space as a convex polyhedral cone, referred to as the *flux cone*, C, whose set of generators consists of the EMs of the system.

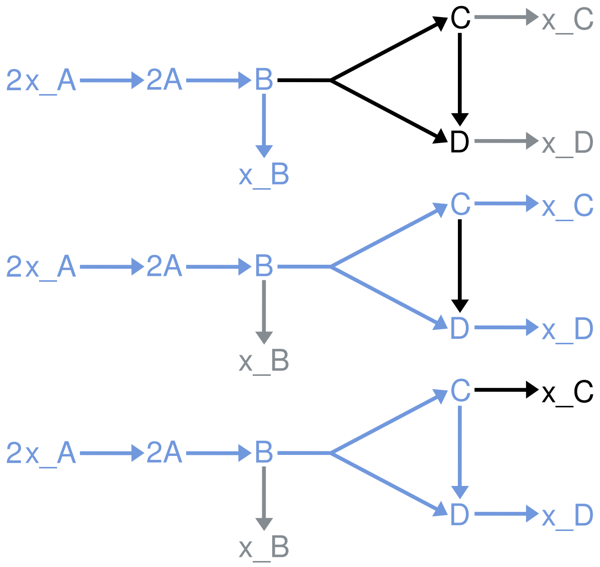


Figure 2.3: A metabolic network that has three EMs. The model is shown three times, with the different EMs outlined in blue. Every possible (input to output) flux distribution of the system can be represented as a positive combination of these EMs.

Thermodynamic consistency is ensured by embedding reaction reversibility constraints into this space as explained below.

Consider a system where all reactions are irreversible. From a geometric perspective, the convex polyhedral cone, C, generated by the steady-state conditions and the reactions' reversibility constraints, is defined as follows:

$$C = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{N}\mathbf{v} = \mathbf{0}, \mathbf{v} \geq \mathbf{0}\}, \tag{2.22}$$

where $\mathbf{v} \geq \mathbf{0}$ means that all the elements of $\mathbf{v}$ are non-negative. A more detailed definition is given by Appendix B and Gagneur and Klamt [2004], which also illustrate how reversible reactions can be included by splitting them into forward and backward components.

Convex cones are the solution set of a system of homogeneous linear inequality constraints (for detailed terminology see [Rockafellar, 1969; Fenchel, 1953]).

Definition 2.3.4.2. *Consider the system of linear inequality constraints, $\mathbf{A}\mathbf{v} \leq \mathbf{b}$ where $\mathbf{v}$ is a vector of variables and $\mathbf{A}$ and $\mathbf{b}$ are a matrix and vector of constants respectively. This system is* homogeneous *if and only if all the elements on the right-hand-side of the equation are zero (i.e. $\mathbf{b} = \mathbf{0}$ such that $\mathbf{A}\mathbf{v} \leq \mathbf{0}$)*

In the case of the flux-cone, every row of $\mathbf{Nv} = \mathbf{0}$ and every reaction reversibility criteria corresponds to a linear constraint. Consider a network with r reactions, such that C (also known as the steady-state solution space) denotes a subset of flux vectors $\mathbf{v}$ in $\mathbb{R}^r$. As shown by Figure 2.4, every constraint shapes the cone by generating a hyper-plane that divides $\mathbb{R}^r$ into two, where one half-space contains the vectors that satisfy the constraint and the other half-space contains the vectors that do not. Therefore, each hyper-plane can be thought of as a mechanism that eliminates the vectors that do not satisfy that constraint from the set of possible solutions of the system (referred to here as the solution-space). In this manner, every constraint of the system gives rise to a corresponding hyper-plane that in turn eliminates more vectors from the solution-space, such that eventually the cone will consist of the set of vectors, $\mathbf{v}$, that simultaneously satisfy all of the constraints imposed by the network's stoichiometry.

Note that since the system of inequalities is homogeneous, every hyper-plane will pass through the origin of $\mathbb{R}^r$, such that the shape created by the intersection of all hyper-planes is that of a cone.

Moreover, every intersection of hyperplanes (an edge of the cone) is referred to as an extreme ray (or in the case of C, an elementary mode). By the Minkowski-Weyl theorem [Fukuda and Prodon, 1996], the complete set of extreme rays is another way to mathematically define the cone. More precisely, every point within the cone can be written as a non-negative linear combination of its extreme rays, such that for a given set of extreme rays, $\mathbf{E}$, the cone can alternatively be represented as:

$$C = \{\mathbf{v} \in \mathbb{R}^r | \quad \mathbf{E}\lambda = \mathbf{v}, \quad \lambda \geq \mathbf{0}\}, \tag{2.23}$$

where λ is a vector of positive constants. For an example, see Figure 2.4

This definition enables steady-state flux vectors to be interpreted as a summation of elementary modes.

Definition 2.3.4.3. *Let the support of a vector denote the set of indices of its non-zero elements. As defined by Fenchel [1953], a vector,* $\mathbf{e}$*, is an extreme ray of* C *if it is not a non-negative linear combination of two linearly independent vectors of* C*, and*

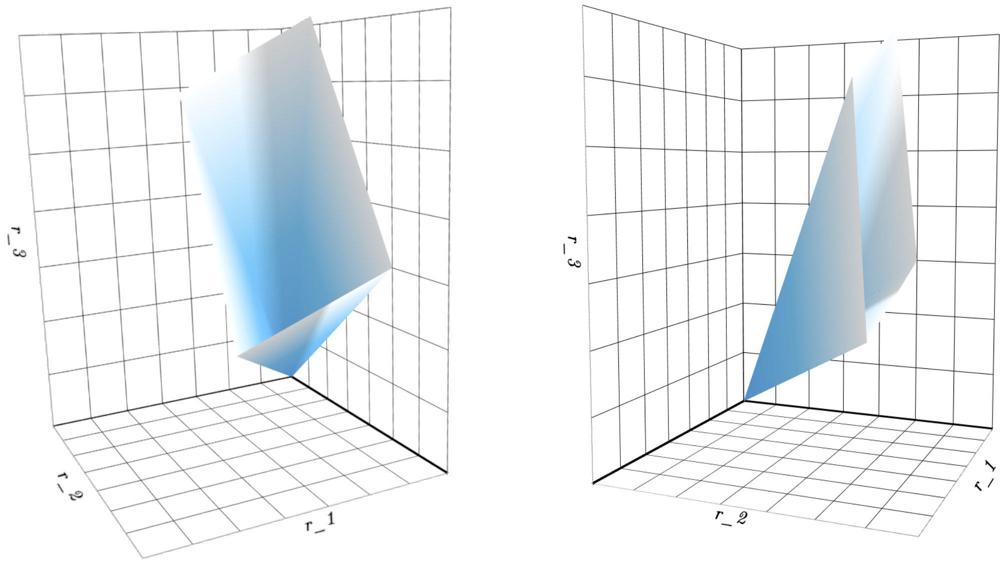


Figure 2.4: A convex cone in three dimensions (r_1, r_2, and r_3), generated by four half-spaces, each corresponding to a constraint. The intersections of the half-spaces are four extreme rays, such that any vector inside the cone can be expressed as a conic combination of these extreme rays.

has minimal-support i.e. if two extreme rays have the same support then they must be identical.

Therefore, the cone C can be equivalently described by Equation (2.22) or Equation (2.23), where Equation (2.22) is referred to as the $\mathcal{H}$-representation (where $\mathcal{H}$ denotes a half-space), while Equation (2.23) is the $\mathcal{V}$-representation (where $\mathcal{V}$ denotes a vertex). These two definitions are referred to as the double description of the cone and, correspondingly, calculating the entire set of EMs of a system entails obtaining the $\mathcal{V}$-representation from the corresponding $\mathcal{H}$-representation, a computationally challenging problem referred to as *extreme ray enumeration* [Gagneur and Klamt, 2004; Guil et al., 2020].

Unfortunately, the enumeration of the complete set of EMs pertaining to large models is a challenging endeavour, accounting for thousands to millions of EMs that might be present [Ullah et al., 2019; Acuña et al., 2009]. In fact, enumerating the vertices of general (unbounded) polyhedra is NP-hard, whilst enumerating the vertices of bounded polyhedra remains an open problem [Khachiyan et al., 2008]. Most EM enumerating algorithms rely on the Double Description Method [Fukuda and Prodon, 1996; Gagneur and Klamt, 2004], that iteratively inserts constraints and counts the generated extreme rays, whilst deleting any previous rays which are no longer extreme. Alternatively, a Depth-First-Search can be used to move from ray to ray along the edges of the polyhedron. In this case, enumeration is identical to the transversal of a graph whose nodes are bases and edges are pivots.

General speed-ups include parallel computing [Terzer, 2009, Chapter 5], as well as, reducing the size of the stoichiometry matrix by removing column redundancies. Furthermore, more recent trends abandon the quest of complete enumeration and instead aim to solely obtain a subset of EMs with desirable properties [Röhl et al., 2019]. Alternative methods such as Random Sampling [Herrmann et al., 2019], and meta-heuristic optimisation algorithms [Hon et al., 2019], as well as, complementary approaches such as Minimal Cut Set Analysis [Hädicke and Klamt, 2011; Röhl and Bockmayr, 2019]), can also be used to explore this space.

2.3.5 Flux Balance Analysis and the Flux Polyhedron

Although calculating the entire set of EMs of a network is difficult, individual flux vectors with desirable properties can be easily obtained using Linear Programming (LP), as explained below.

The stoichiometric structure of a network may allow for many possible flux pathways, but only a few of these are likely to exist in nature. For example, in a growing cell, the reaction rates are capped by the number of enzymes and metabolites that can fit in the space available. The availability of the cell's surface area also limits the rate through which metabolites can be exchanged with the environment. These facts, along with other limiting factors (such as, the amount of available nutrients) allow the number of possible steady-state flux vectors to be reduced by adding flux constraints. For example, if a nutrient is unavailable then its corresponding transporter reaction can be constrained such that it is always inactive.

Non-zero flux constraints are not homogeneous, and therefore, each such constraint generates a hyperplane that does not pass through the origin of $\mathbb{R}^r$. Consequently, their addition to the steady-state solution space restricts the flux cone into a convex polyhedron, referred to as the flux polyhedron, P, whose $\mathcal{H}$-representation is defined as:

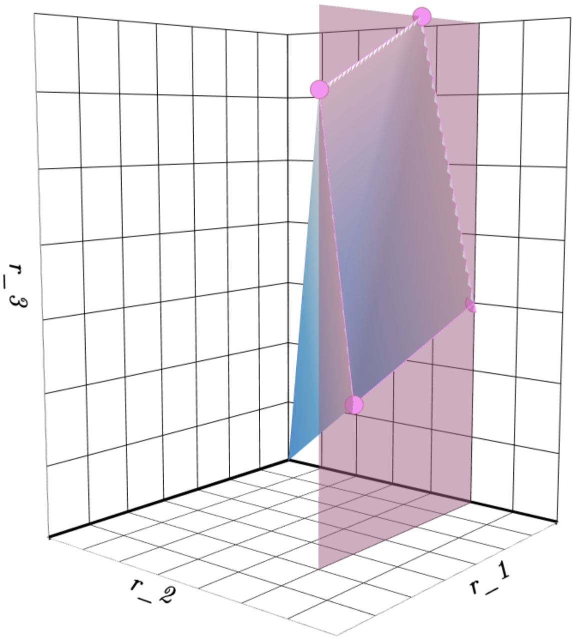


Figure 2.5: A convex polyhedron created by the addition of the plane caused by the constraint $r_2 \leq 5$ to the cone in Figure 2.4. The vertices are marked in pink.

$$P = \{\mathbf{v} \in \mathbb{R}^r | \quad \hat{\mathbf{N}}\mathbf{v} \leq \mathbf{b}, \quad \mathbf{v} \geq \mathbf{0}\}, \tag{2.24}$$

where $\hat{\mathbf{N}}$ is a modified stoichiometry matrix and $\mathbf{b}$ a vector of constants, whose rows contain the constraints imposed by $\mathbf{Nv} = \mathbf{0}$ (as described above), as well as additional rows corresponding to the novel flux constraints (as described in Chapter 3.3.3).

The intersections of flux constraints with the flux-cone are vertices as shown in Figure 2.5.

Definition 2.3.5.1. *A point* $\mathbf{x} \in \mathbb{R}^r$ *is a vertex of the polyhedron, P, if* $\mathbf{x} \in P$ *and* $\mathbf{x}$ *does not lie on the line segment between any two other points,* $\mathbf{y}$ *and* $\mathbf{z}$*, in P (i.e. for* $0 \leq \alpha \leq 1,$ *if* $\mathbf{x} = \alpha\mathbf{y} + (1-\alpha)\mathbf{z},$ *then* $\mathbf{y} = \mathbf{z} = \mathbf{x}$*).*

Once P is defined, LP can be applied to identify the flux vector in P that best achieves a desired biological goal a method is referred to as Flux Balance Analysis (FBA) [Fell and Small, 1986].

LP is an optimisation technique originally devised for finding the best allocation of limited resources. Each LP is defined as a set of constraints (in this case, P), and a given objective function. The choice of objective function is subjective, and is often chosen to correspond to the possible endpoints of evolutionary pressure, such as maximising growth [Feist and Palsson, 2010]. Other objectives may seek flux vectors that maximise the production of a desirable compound. In this thesis, LP are formulated to minimise intracellular flux, as this corresponds to the minimisation of the cell's total protein investment.

The solution of such an FBA problem must be a vertex of the flux polyhedron P. Therefore, LP algorithms, such as the simplex methods, initiate from a starting vertex of P, and incrementally explore the space, moving from vertex to vertex, in pursuit of solutions that best satisfy the objective function.

Note that FBA solutions obtained after the addition of just one flux constraint to C will correspond to EMs [Maarleveld, 2015, page 63]. This can be understood intuitively since the vertices of P will consist of the sites where the extreme rays (and hence EMs) of the flux cone meet the hyperplane created by the constraint as shown in Figure 2.5. Adding more constraints violates this statement, as the vertices will no longer all incorporate extreme rays, but also consist of intersections between the constraints themselves, as shown in Figure 2.6.

Flux Constraint Scanning

A common application of FBA is flux constraint scanning, a method that iteratively solves an LP whilst varying specific conditions. This techniques was first described by Poolman et al. [2009] who used it to study the plant Arabidopsis's response to varying energy demands. Another example, by Singh [2017, Chapter 4], regards varying light intensity on algae and observing changes in the optimal steady-state flux. Such methods reveal how the metabolism re-adjusts to account for changes in the environment, and was deployed for drug target identification by Hartman et al. [2014], by pinpointing the enzymes whose joint inhibition stops salmonella from regenerating ATP. The set of flux values resulting from flux-scanning may be fed into algorithms such as Principal Component Analysis that identify the reactions that account for most of the variance [Vijayakumar et al., 2020; Culley et al., 2020].

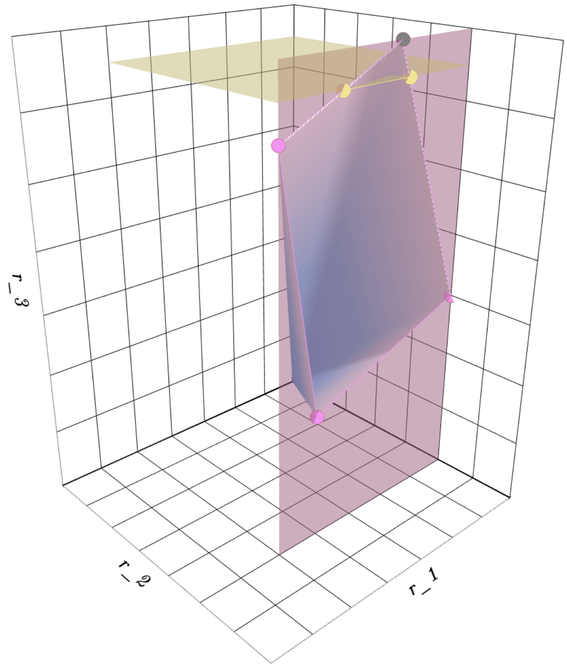


Figure 2.6: The Flux Polyhedron created by the addition of the constraints $r_3 \leq 6.5$ and $r_2 \leq 5$ on the cone, C, in Figure 2.4. The vertices that correspond to original extreme rays of C are marked in pink, the vertices created by the intersections of the flux constraints are in yellow, whilst eliminated extreme rays of C are in grey.

Flux Variability Analysis

FBA is often used to determine which reactions are likely to be active when the cell pursues a certain goal. However, the redundancy in metabolic networks mean that any given FBA solution is likely to be one of many alternate pathways that achieve the same objective using different routes (a problem known as multiple optima). Consequently, reactions that do not appear in a solution may still be relevant. Furthermore, knowledge of these optima is useful such as to determine network flexibility, or in cases where the gene-editing of one enzyme may be easier than another.

A way to counteract this obstacle is by re-solving the problem repeatedly for varying conditions through methods such as Flux Variability Analysis (FVA), which identifies the range of values each flux can vary whilst still preserving the optimal value as identified by the original LP [Maarleveld et al., 2015; Guebila, 2020]. This method determines network flexibility by identifying fluxes which are fixed (cannot vary), and the level of flexibility in the other fluxes. Essential reactions are identified as those which must always carry flux in order for the system to achieve the optimal objective value.

This method was used by Kelk et al. [2012], when maximizing biomass in GSMs, to identify sub-networks that can achieve the same net stoichiometry using different internal flux distributions.

2.4 Construction and Analysis of Metabolic Models

The techniques and tools outlined in the previous sections have allowed for the complete metabolic capability of several cells and processes to be investigated. In fact, current metabolic models range from small-scale hand-built models that define specific behaviour of well-known pathways, such as the Calvin cycle or glycolysis, to very large computationally-built, GSMs that encapsulate the metabolism of entire organisms, such as bacteria and algae [Tejera et al., 2020; Mesfin and Fell, 2019], or complex processes within human cells [Masid et al., 2020].

Building structural metabolic models is a laborious process that involves characterising reactions as described in Section 2.3.1. It is important to note that most such models are built to serve a specific purpose, and thus include many simplifying assumptions.

First, a draft is automatically constructed from annotated genetic sequencing data and metabolomic measurements (found in databases such as KEGG and BioCyc). However, the set of reactions extracted from such databases usually contains various inconsistencies that require human adjustments—a difficult and time-consuming task which aims to ensure that every reaction in the model has the correct stoichiometry and directionality. Common errors that arise from database artefacts include:

- metabolites that are not denoted by the same identifier across all of the reactions of the model,
- reactions with incorrect stoichiometry or missing substrates/products, where some are identified as violating the law of mass conservation, as described by Gevorgyan et al. [2008],
- dead reactions, identified as described in Section 2.3.3.2,
- reactions with incorrect directionality, where some are identified as part of incon-

sistent enzyme subsets, as described in Section 2.3.3.2, or as part of internal cycles that violate the first law of thermodynamics by creating energy (in the form of ATP/NAD(P)H) from no external input, as described by Fritzemeier et al. [2017],

- stoichiometrically disconnected reactions that arise from annotated genomes in which not all enzymes had been identified. These errors are remedied by manual/automatic *gap-filling* methods that make assumptions about which reactions to add to the model such that all metabolites are connected, as described by Orth and Palsson [2010].

Following these corrections, the model's components are then iteratively improved to achieve behaviour that agrees with experimental measurements, for example, by ensuring that the model is capable of producing the organism's known biomass components.

Both model construction process described above and the subsequent interrogation is facilitated by many computational tools, such as ScrumPy[3] (used within this thesis), ModelSEED[4], COBRA[5], and COPASI[6].

2.5 Software and Models Used as Part of this Thesis

2.5.1 ScrumPy Metabolic Modelling Software

The algorithms presented in this thesis are implemented in Python as add-ons to the open-access ScrumPy metabolic modelling software, which is developed and maintained by the Cell Systems Modelling (CSM) Group at Oxford Brookes University [Poolman, 2006]. ScrumPy is user-accessible through a Python IDLE[7] shell and has an object-oriented implementation. It facilitates both kinetic and structural modelling through specially defined classes and the integration of third party open access software.

Classes frequently used in the thesis are described below:

- *DataSets*: A two-dimensional data structure, with labelled rows and columns.

[3] mudshark.brookes.ac.uk/ScrumPy

[4] modelseed.org/

[5] opencobra.github.io

[6] copasi.org

[7] python.org/3/library/idle

- *LP*: Generates and solves an FBA problem for a given model. Attributes include functions that add flux constraints and define the LP objective function. The LP functionality is provided by the GNU Linear Programming Kit (glpk[8]).
- *ElModes*: Calculates the elementary modes of a given model using the algorithm described by Schuster and Hilgetag [1994].
- *EnzSubsets*: Calculates the enzyme subsets of a given model as described in Section 2.3.3.1.
- *StoMat*: A two dimensional stoichiometry matrix, with metabolites labelled as rows and reactions as columns. Attributes of this class include functions that calculate its right and left null-space.
- *Model*: Models are stored within *.spy* text files, each *Model* object has an associated *StoMat* and functions that return the model's associated *ElModes*, *EnzSubsets*, and *LP* objects amongst others.

2.5.2 Metabolic Models

Throughout the course of this thesis, several metabolic models were used to evaluate the accuracy of the novel presented algorithms, and to derive biological insight. All of these models were obtained from the archives of the CSM Group at Oxford Brookes University. They consisted of three hand-built small scale models and four large GSMs, as described by Table 2.1 and below.

Calvin cycle. This model contains the Calvin cycle and the oxidative part of the pentose phosphate pathway, as described by Figure 2.7. The Calvin cycle occurs in the chloroplast of plants, algae, and some bacteria. It is concerned with fixing environmental CO_2 into sugars, in a process referred to as carbon fixation. The enzyme RuBisCo incorporates CO_2 into the five-carbon sugar ribulose-1,5-biphosphate (RuBP) to create two molecules of the three-carbon sugar 3-phospho-D-glycerate (PGA). PGA is then converted into other three-carbon sugars, D-glyceraldehyde-3-phosphate (GAP) and dihydroxyacetone phosphate (DHAP), in a process that uses energy harvested from sunlight.

[8] gnu.org/software/glpk/

Table 2.1: The seven metabolic models used as described in this chapter.

Model	**Reactions**	**Metabolites**	**Inputs**	**Output Products**	**Output Bi-products**	**Reference**
Calvin Cycle	21	28	5	4	1	[Poolman et al., 2003]
Simplified plant	75	77	3	2	4	[Poolman et al., 2007]
Photorespiration	90	113	7	1	0	[Huma et al., 2018]
Campylobacter jejuni	1150	1105	44	51	12	[Tejera et al., 2020]
Cupriavidus necator	1358	1454	109	1	9	[Pearcy et al., 2022]
Geobacillus thermoglucosidasius	1125	1198	8	54	6	[Ahmad et al., 2017]
Escherichia coli	1659	1714	6	46	7	unpublished

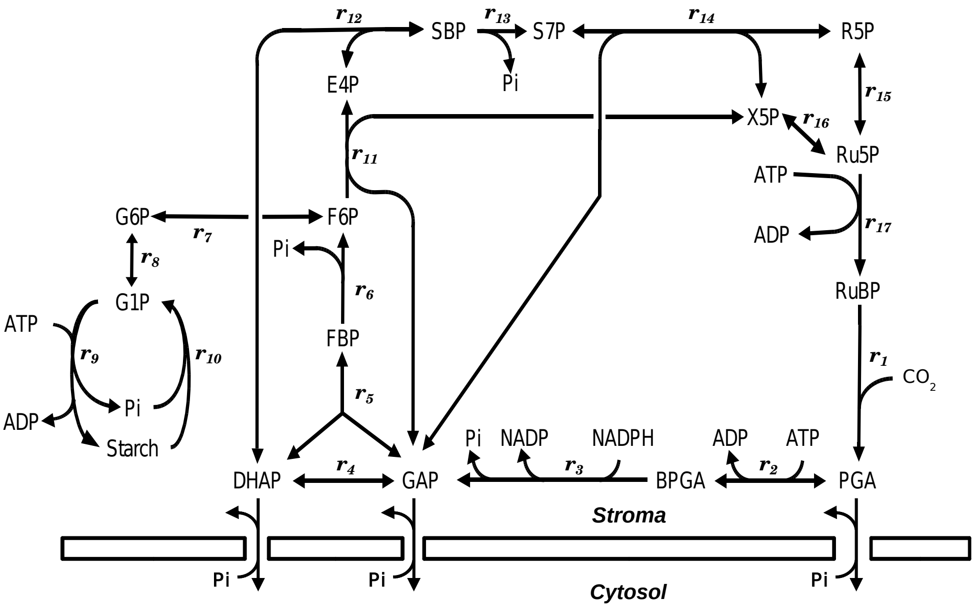


Figure 2.7: The model of the Calvin cycle. See Appendix A.1 for reaction and metabolite abbreviations. Replicated with permission of Poolman et al. [2004a].

Some of these three-carbon sugars are exported to supply the cell with the organic material that it needs to build and maintain cellular structures, whilst the rest are used to (i) create a fourth product, starch, which is retained in the chloroplast as a reserve of carbon and energy during periods of darkness, and (ii) to regenerate RuBP, the starting product of the cycle.

This model was first described by Poolman et al. [2004b] who used it to investigate energy generation in plants, such as by quantifying the effects that different light conditions have on the formation/degradation of starch.

Simplified plant model. This model was developed by Poolman et al. [2007] and is a simplification of a model of potato carbohydrate metabolism described in Poolman et al. [2004a]. This model describes how photosynthetic glucose, produced at the leaves during growth, is transported to root cells where it is stored as starch within *amyloplast* compartments. This starch serves as an energy reserve that allows plants to survive periods of starvation, such as seasonal hibernation.

This model includes two chloroplast compartments, that each contain identical copies of the Calvin cycle model, the cytosol (containing glycolysis and sucrose synthesis, which are necessary for growth), and an amyloplast (starch synthesis). The chloroplast compartments simulate carbon fixation in the leaves by creating the three-carbon sugar PGA which is then transported to the cytosol. A portion of the PGA is used to generate energy via glycolysis, leading to the bi-product pyruvate, while the remainder is converted to sucrose in the cytosol, and finally, starch in the amyloplast.

Photorespiration. This model extends that of the Calvin cycle to account for the behaviour exhibited by plants, as shown in Figure 2.8. It captures the consequence of the enzyme RuBisCo fixing O_2 to RuBP instead of CO_2. This reaction has two products: PGA, which can be incorporated into the Calvin cycle, as well as, the two-carbon sugar PG which must be converted into a three-carbon sugar before it can be used by the cell. This conversion happens via a process that consumes energy and releases CO_2.

In a study by Huma et al. [2018], EMA was applied to this model to analyse the impact of photorespiration on the cell's metabolism (in terms of O_2 consumption/production relative to other net conversions), where it was noted that photo-respiration enhances the rate of photosynthesis since the amount of CO_2 released by this process must be equivalently recaptured. The authors also identified reactions that are essential for photorespiration to occur and discussed how photorespiration provides plants with the means to dissipate excess energy.

C. jejuni. This organism is one of the most common causes of food poisoning worldwide. Its GSM was first described by Tejera et al. [2020] and used to design an optimal minimal media to promote its growth [Tejera et al., 2020], a result achieved by identifying the substrates that have the best effect on biomass generation.

C. necator. *C. necator* is of interest as it can produce the bio-plastics polyhydroxyalkonoates (PHA) when growing in nitrogen/oxygen limited conditions (where carbon is abundantly available). Pearcy et al. [2022] developed this GSM and used it to understand the regulatory mechanisms that lead to PHA production and to identify potential genetic engineering strategies that can improve the yield of PHA.

G. thermoglucosidasius. This GSM was designed by Ahmad et al. [2017] to investigate this organism's capability to produce commercially valuable chemicals when feeding

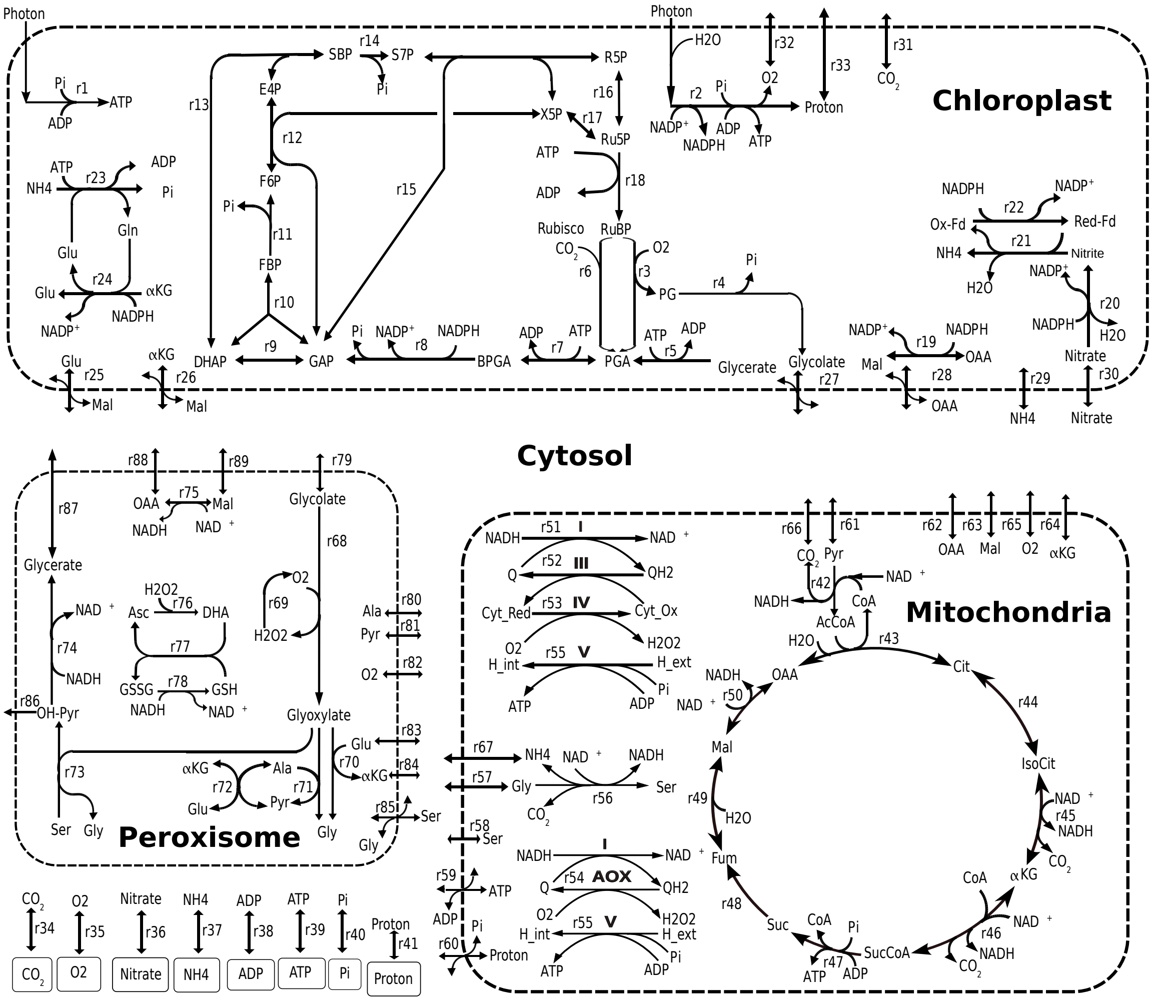


Figure 2.8: The model of photorespiration, where: Chloroplast: r1-r2:the cyclic and non-cyclic photophosphorylation reactions, r3-r5:the plastidic reactions involved in the C2 cycle namely, RuBisCo oxygenase,PGLP1 and GLYK, r6-r18:the C3/Calvin cycle, r19:MalDH, r20-r22:N-assimilation cycle, r23-r24:GS2 and Fd–GOGAT cycle, r25:DiT2.1, r26:DiT1, r27:PLGG1, r28:AtpOMT1, r29-r33:plastidic transporters for exchange of extracellular resource metabolites. r34-r41:cytosolic exchange reactions for extracellular resource metabolites. Mitochondrion: r42-r50:reactions of the TCA cycle,(r44 represents combined reaction catalysed by aconitase) r51:Complex I, r52-r53:COX, r54:AOX, r55:ATP synthase, r56:combined reaction for GDC and SHMT1, r57-r67:metabolite exchange reactions. Peroxisome: r68-r74:C2 cycle, r75:MalDH, r76-r78:GSH-ASC cycle, r79-r89:metabolite exchange reactions. Reproduced with permission of Huma et al. [2018].

on rice straw hydrolysate (a waste product of the farming industry). By simulating the metabolism of *G. thermoglucosidasius* under different growth conditions, it was shown that this organism can convert the sugars present in rice straw hydrolysate into products such as ethanol and acetate.

E. coli. The GSM of *E. coli* was developed as part of INNOTARGETs[9], an ongoing project that seeks to understand the bacterial metabolic response to antibiotics. This model is currently in development and is therefore unpublished.

[9] `https://innotargets.ku.dk`

Chapter 3

Decomposing Flux Vectors into Elementary Modes

3.1 Introduction

This chapter presents a novel method that efficiently decomposes flux vectors, obtained from experiments or simulations, into a set of EMs, without requiring the calculation of the entire set of EMs of the network.

As described in Section 2.3.4, EMs consist of the smallest non-decomposable pathways within a metabolic system, that can combine to form every possible steady-state flux. Decomposing flux measurements obtained from experiments or simulations into such a set of constituents allows for their interpretation as a weighted sum of pathways (each with associated external input and output), thus revealing how potential routes within the network can contribute to the observed overall system behaviour, as well as providing the means to assign relative flux values to EMs.

A number of methods for this task have been previously described. However, most rely on having calculated a complete set of EMs *a priori*, which is not computationally practical for large metabolic models. Furthermore, since a given flux vector can be generally decomposed into more than one distinct combination of EMs, these techniques can be distinguished by the type of decomposition that they seek to achieve.

3.1.1 Decomposing Flux Vectors by Calculating All EMs in the Network *A Priori*

The first such method was developed by Poolman et al. [2004b] who used it to reveal changes in the relative flux assigned to EMs during different stages of the fermentation progress in *Lactobacillus rhamnosus*. This approach decomposes a flux vector $\mathbf{v}$ into a weighted sum of EMs in two steps. First, a matrix of EMs, $\mathbf{E}$, is calculated. Then, the pseudo-inverse of this matrix is multiplied with $\mathbf{v}$, to obtain a vector of weights, $\mathbf{w}$, such that $\mathbf{Ew} = \mathbf{v}$. Alternatively, Schwartz and Kanehisa [2005] defined a quadratic optimisation problem that when applied to a matrix of EMs $\mathbf{E}$ and steady-state flux vector $\mathbf{v}$, finds a vector of weights, $\mathbf{w}$, such that the sum of square weights is minimal and $\mathbf{Ew} = \mathbf{v}$.

3.1.2 Decomposing Flux Vectors without Calculating All EMs in the Network

Despite the difficulty of calculating a network's entire set of EMs, individual EMs that satisfy some desired attributes can be easily determined using optimisation techniques [Song et al., 2017; de Figueiredo et al., 2009]. Consequently, methods such as LP and Mixed Integer Linear Programming (MILP) have been previously utilized to decompose flux-vectors into EMs, without needing to determine the complete set of EMs *a priori* [Oddsdóttir et al., 2015; Hung et al., 2011; Jungers et al., 2011; Ip et al., 2011].

Oddsdóttir et al. [2015] designed an algorithm that calculates EMs that account for a collection of observed transporter fluxes. The algorithm begins with a matrix, $\mathbf{E}$, whose columns, $\mathbf{e}_i$, contain a small sub-set of EMs, and a corresponding weighting vector, $\mathbf{w}$. Then, two optimisation problems are simultaneously solved: a least-squares data fitting master problem that iteratively improves the weighting vector $\mathbf{w}$, seeking to find a product, $\mathbf{Ew}$, that is consistent with the external flux measurements, as well as an additional sub-problem that after every iteration of the master, calculates a new EM that when added to $\mathbf{E}$ improves upon the least-square fit. The algorithm terminates when adding more EMs to $\mathbf{E}$ no longer improves this fit. This method solves a quadratic program at each iteration, which is inherently more computationally costly than LP, limiting this algorithm's applicability to large models.

An alternative algorithm, introduced by Hung et al. [2011] decomposes steady-state flux vectors, $\mathbf{v}$, through a series of iterations. At each phase, a succession of MILPs reduce the reaction fluxes in $\mathbf{v}$ such that the number of zero fluxes in $\mathbf{v}$ continues to increase until a solution that cannot be reduced further—and therefore is an EM— is found. This EM is subtracted from $\mathbf{v}$, and the previous steps are repeated until all constituent EMs are identified. A disadvantage of this algorithm is its reliance on MILP, which, similarly to the method described previously, is more computationally expensive than LP.

Another example, by Jungers et al. [2011], uses LP to decompose a steady-state flux vector, $\mathbf{v}$, into a minimal number of EMs. This algorithm extracts EMs from the solution space created by combining the stoichiometry of the network with $\mathbf{v}$. The first EM is randomly extracted from this space, while additional EMs are chosen in a way that the distance between them and the previous EMs is maximal. This technique gradually reduces the dimension of the solution space, and therefore, the number of constituent EMs in the final decomposition cannot be greater than the dimension of the null-space. This algorithm is likely to yield different results when applied to the same vector more than once, making replication of findings challenging.

As of July 2023, the algorithm by Poolman et al. [2004b] was publicly available as part of the open source ScrumPy software (Section 2.5.1). Similarly, a Matlab implementation of the algorithm by Hung et al. [2011] was available on GitHub[1] as an add-on to the COBRA[2] toolbox. Implementations of the other aforementioned algorithms were not found to be publicly accessible.

3.1.3 Aims and Objectives

The algorithm presented in this chapter, referred to as LPEMs, uses LP to decompose a flux vector, $\mathbf{v}$, into a linear combination of EMs, such that reactions carrying a proportionally large flux in $\mathbf{v}$ are expected to contribute to a large number of EMs within the decomposition.

Constituent EMs are obtained by solving a sequence of Linear Programs that iteratively eliminate reactions from $\mathbf{v}$. At each iteration, the LP problem is formulated such that the reaction with the smallest flux in $\mathbf{v}$, v_{min}, carries the same flux as $\mathbf{v}$ within the LP solution, and, therefore, subtracting the LP solution from $\mathbf{v}$ eliminates v_{min}. At each stage, the LP solution is either an EM or a combination of EMs that can be efficiently calculated. This process repeats until all reactions in $\mathbf{v}$ are eliminated, and therefore all constituent EMs are obtained.

This algorithm is applied to a GSM of *C. jejuni* [Tejera et al., 2020]. This gram-negative, microaerophilic bacteria is recognised as one of the primary causes of bacterial gastroenteritis. It is commonly associated with poultry and, once ingested, infects the intestines causing fever and diarrhoea. An unusual feature of *C. jejuni*'s metabolism is its use of oxygen: although this organism has the capacity for anaerobic respiration, it still requires

[1] github.com/shjchan/DecompFlux

[2] opencobra.github.io

small amounts of oxygen to grow [Sellars et al., 2002].

The study of the GSM of *C. jejuni* presented here investigates this oxygen requirement with regards to biomass precursors, leading to the identification of various routes for energy generation in *C. jejuni* and indicating that oxygen is required for the production of pyridoxal phosphate (PLP), a biomass precursor that is essential for growth.

Initially, FBA (Section 2.3.5) was applied to simulate *C. jejuni*'s growth under different oxygen conditions. However, as typically expected of LP solutions that constrain for many products, the resultant flux vector was large and therefore difficult to examine. Hence, the production of each biomass precursor was investigated, one precursor at a time, through a sequence of LPs that included the production of a single precursor as their only output flux constraint. This yielded smaller, more understandable pathways corresponding to EMs (Section 2.3.4) . However, these EMs did not necessarily reflect the paths used by a cell to simultaneously produce all biomass.

The LPEMs algorithm described here was able to tackle this challenge by decomposing the FBA solution that simultaneously produced all biomass into a set of EMs that each produce a subset of the biomass precursors, whilst taking the background demand for all biomass into account. This algorithm simplified the analysis of the original FBA solution, and exemplified how FBA solutions that only simulate the production of a single product of interest must be interpreted with caution, by bearing in mind that the cell is likely to refine its metabolism to simultaneously consider multiple constraints and objectives.

3.1.4 Chapter Structure Overview

This chapter contains the following sections:

Section 3.2 outlines the novel algorithm.

Section 3.3 introduces mathematical concepts relating to it. For example, a theorem that states that, although the number of EMs in a system can be extreme, for every steady-state flux there exists a minimal decomposition that contains at most r EMs, where r is the number of active (non-zero) reactions in the measured flux vector.

Section 3.4 applies this algorithm to metabolic models of the Calvin cycle and *C. jejuni*.

Section 3.5 discusses the results with emphasis on *C. jejuni*'s oxygen requirements.

The work presented in this chapter follows closely from the pre-print *A Novel Algorithm to Calculate Elementary Modes: Analysis of Campylobacter jejuni Metabolism*, in Appendix F, and a research paper with the same title submitted to *BioSystems*.

3.2 Methodology

The algorithm described within this section operates by exploiting the feasible steady-state flux space that was introduced in Chapter 2.3.4.1. First the algorithm is described, then the mathematical concepts related to are expanded upon in the subsequent section.

3.2.1 The Algorithm

Let the flux vector to be decomposed into EMs be denoted by $\mathbf{v} \in \mathbb{R}^r$, where r is the number of reactions. Then, the output of the algorithm is a matrix, $\mathbf{E}$, whose column vectors, $\mathbf{e}_i$, consist of EMs, such that:

$$\sum_{i=1}^{t} \mathbf{e}_i = \mathbf{v}, \tag{3.1}$$

where t is the number of EMs. Note that the EMs are not normalised in this instance, such that the magnitude of each $\mathbf{e}_i$ reflects the contribution of that EM to $\mathbf{v}$.

Prior to starting the decomposition, $\mathbf{v}$ is ensured to be at steady-state. If this is not the case, $\mathbf{v}$ is approximated to the closest vector that satisfies $\mathbf{N}\mathbf{v} = \mathbf{0}$, where $\mathbf{N}$ is the stoichiometry matrix, and the excess is saved as an error vector.

Then, the algorithm proceeds to iteratively obtain EMs, whilst simultaneously eliminating components of the flux vector (starting from the smallest first). Consider the loop described by Algorithm 1, at each iteration, the following LP problem is used to obtain a

Algorithm 1 Decomposing $\mathbf{v}$ into a set of EMs, $\mathbf{E}$

```
 1: while $|\mathbf{v}| > \mathbf{0}$ do
 2:   $v_{\text{min}}$ = minimum value in $\mathbf{v}$
 3:   $\mathbf{v}' = \mathcal{F}(\mathbf{v}, \mathbf{N})$ given by Equation (3.2)
 4:   if $\mathbf{v}'$ is not an EM then
 5:     decompose $\mathbf{v}'$ into a set of EMs
 6:     append these EMs to $\mathbf{E}$
 7:   else
 8:     append $\mathbf{v}'$ to $\mathbf{E}$
 9:   end if
10:   $\mathbf{v} - \mathbf{v}' \rightarrow \mathbf{v}$
11: end while.
```

solution, $\mathbf{v}'$, that has specific properties that depend on $\mathbf{v}$:

$$\mathcal{F}(\mathbf{v}, \mathbf{N}) = \operatorname{argmin} \quad \textstyle\sum_{i=1}^{r} |\mathbf{v}'_i|$$
$$\text{subject to} \quad \begin{cases} \mathbf{N}\mathbf{v}' = \mathbf{0}\,, \\ v'_{\text{min}} = v_{\text{min}}\,, \\ |v'_i| \leq |v_i|,\ \operatorname{sign}(v'_i) = \operatorname{sign}(v_i),\ \text{for all } i \in \{1, 2, \ldots, r\}\,. \end{cases} \tag{3.2}$$

The LP is constrained such that the reaction with the smallest non-zero flux value in $\mathbf{v}$, v_{min}, has the same flux as in $\mathbf{v}$ within the solution (i.e. $v_{\text{min}} = v'_{\text{min}}$). In addition, the value of every other reaction in the solution, v'_i, is constrained to not exceed the flux values found within $\mathbf{v}$, and the LP objective is to minimise the total flux in the solution.

Once an LP solution is obtained, the Rank Test (Section 3.3) is applied to determine whether $\mathbf{v}'$ is an EM (Step 4 in Algorithm 1).

If the solution is not an EM, it is decomposed into constituent EMs by creating a sub-matrix of $\mathbf{N}$ that contains only the reactions present within $\mathbf{v}'$ and enumerating its EMs. Once the EMs of this sub-system are obtained, a conventional algorithm (described by Poolman et al. [2004b]) assigns a set of fluxes to the EMs by obtaining the pseudo-inverse of the EM matrix, $\mathbf{E}$ (EMs as columns reactions as rows), such that $\mathbf{E}\mathbf{w} = \mathbf{v}'$ where $\mathbf{w}$ is a vector of flux weightings assigned to each EM.

The one or more obtained EMs are saved, and the loop restarts using the modified flux vector $\mathbf{v}$, that is obtained by subtracting the solution from $\mathbf{v}$ such that v_{min} is eliminated:

$$\mathbf{v} - \mathbf{v}' \to \mathbf{v}, \tag{3.3}$$

or,

$$\begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_{\text{min}} \\ \vdots \\ v_r \end{pmatrix} - \begin{pmatrix} v'_1 \\ v'_2 \\ \vdots \\ v'_{\text{min}} \\ \vdots \\ v'_r \end{pmatrix} = \begin{pmatrix} v_1 - v'_1 \\ v_2 - v'_2 \\ \vdots \\ 0 \\ \vdots \\ v_r - v'_r \end{pmatrix}. \tag{3.4}$$

The loop continues to iteratively append EMs to $\mathbf{E}$ whilst simultaneously eliminating components of the flux vector $\mathbf{v}$, until the final flux vector $\mathbf{v}$ is equal to $\mathbf{0}$. The elimination of components starting by the smallest first is based on the assumption that the smallest component is the most likely to contribute to the least amount of EMs.

3.2.2 Implementation

This algorithm is implemented in Python and is publicly accessible as part of release 3 of ScrumPy[3] (Section 2.5.1).

Rounding errors are treated as follows. Following Step 10 in Algorithm 1, absolute values in $\mathbf{v}$ that are less than 10^{-7} are set to zero, where the value of 10^{-7} was chosen with the assumption that the smallest possible absolute value in the original $\mathbf{v}$ is 10^{-8}.

At each stage, the numerical accuracy of the results is verified by ensuring that the difference $(\mathbf{v} - \mathbf{v}')$ is a steady-state flux of the system. If due to rounding errors, this is not the case then the decomposition is treated to have failed and the residual from steady-state (corresponding to $\mathbf{N}(\mathbf{v} - \mathbf{v}')$) is returned to the user.

Finally, when the algorithm terminates, the numerical accuracy of its result is verified by checking that Equation (3.1) is satisfied (i.e. the EMs sum to $\mathbf{v}$).

[3] gitlab.com/MarkPoolman/scrumpy/-/blob/master/ScrumPy/Structural/LPEMs.py

3.3 Mathematical Considerations

This section presents mathematical aspects behind Algorithm 1.

Recall the following considerations:

1. EMs are obtained by defining the set of potential steady-state fluxes of a metabolic system (i.e. the set of flux vectors, $\mathbf{v}$, that satisfy the reaction reversibility criteria and $\mathbf{Nv} = \mathbf{0}$) as a convex polyhedral cone, C (referred to as the flux cone), whose set of generators (i.e. the extreme rays) are EMs.

2. When carrying out FBA, it is common to add more constraints to the system in order to eliminate fluxes with undesirable properties, thus reducing the flux cone into a smaller sub-space known as the flux polyhedron, P, that is generated by a set of vertices (or corners), such that the resulting LP solution will consist of the vertex that best achieves a user defined objective. This vertex will be either an EM or a conic combination of EMs.

Consider a system defined by an $m \times r$ stoichiometry matrix, $\mathbf{N}$, and a corresponding steady-state vector, $\mathbf{v} \in \mathbb{R}^r$.

As discussed in Section 2.3.4, $\mathrm{S}(\mathbf{v})$ denotes the set of indices of the non-zero elements of $\mathbf{v}$ (sometimes referred to as the support), i.e.

$$\mathrm{S}(\mathbf{v}) = \{i \in \mathbb{N} | 1 \leq i \leq r \,\mathrm{and}\, v_i \neq 0\}, \tag{3.5}$$

additionally, let $\mathrm{Z}(\mathbf{v})$ denote the set of indices of the zero elements of $\mathbf{v}$ (referred to as the zero-set) i.e.

$$\mathrm{Z}(\mathbf{v}) = \{i \in \mathbb{N} | 1 \leq i \leq r \,\mathrm{and}\, v_i = 0\}, \tag{3.6}$$

and let $\mathbf{N}_\mathrm{S}$ denote a sub-matrix of $\mathbf{N}$ that contains only the reactions in $\mathrm{S}(\mathbf{v})$.

This algorithm relies on the following three properties:

- **The rank test,** Step 4 in Algorithm 1. The vector, $\mathbf{v}$, is an EM if $\mathbf{N}_\mathrm{S}$ has a null-space of dimension one i.e. $\dim(\ker(\mathbf{N}_\mathrm{S})) = 1$ (proved in Lemma 2 of Gagneur and Klamt [2004]).

- **Decomposing $\mathbf{v}$ into EMs,** Step 5 in Algorithm 1. The EMs of the sub-model generated by $\mathbf{N}_\mathrm{S}$ are a sub-set of the EMs of the original model. More specifically, $\mathbf{N}_\mathrm{S}$ contains the EMs of $\mathbf{N}$ that only carry non-zero flux in the reactions of $\mathrm{S}(\mathbf{v})$, and therefore can form part of a conic combination of EMs that sum to $\mathbf{v}$. This reasoning is extended to show that there exists a minimal decomposition of $\mathbf{v}$ into at most $\dim(\ker(\mathbf{N}_\mathrm{S}))$ EMs.

- **Calculating a linear combination of steady-state fluxes,** Step 10 in Algorithm 1. Consider two steady-state flux vectors, $\mathbf{v}$ and $\mathbf{v}'$, where $\mathbf{v} \geq \mathbf{v}'$, then their difference, $\mathbf{v} - \mathbf{v}'$, is a steady-state flux vector of the system.

Additional results regarding FBA solutions that support statements made in Section 3.4 of this chapter, are:

- FBA solutions obtained from the addition of only one non-zero flux constraint to C are EMs, and

- FBA solutions obtained from the addition of more than one non-zero flux constraint to C are not necessarily EMs.

Please note that throughout this chapter, all flux values are assumed to be strictly non-negative (i.e. all reactions are treated as irreversible) because of the nature of convex geometry computation (further information about handling reversible reactions can be found in Gagneur and Klamt [2004] and in Appendix B).

3.3.1 Decomposing a Flux Vector into Elementary Modes

This subsection aims to show that a vector, $\mathbf{v} \in \mathbb{R}^r$, can be decomposed into EMs by enumerating the entire set of EMs of a sub-model that only contains the reactions that carry non-zero flux in $\mathbf{v}$.

Theorem 3.3.1.1. *Consider the steady-state flux vector* $\mathbf{v}$ *defined above, then the EMs of* $\mathbf{N}$ *that can make up* $\mathbf{v}$ *are equivalent to the entire set of EMs of* $\mathbf{N}_\mathrm{S}$.

Proof. Let the complete set of EMs of $\mathbf{N}$ be denoted by $\mathfrak{E} = \{\mathbf{e}_1, \mathbf{e}_2, \ldots, \mathbf{e}_t\}$.

Since $\mathbf{v}$ can be represented as a conic combination of the EMs in $\mathfrak{E}$, there exist non-negative constants λ_i such that

$$\mathbf{v} = \lambda_1\mathbf{e}_1 + \lambda_2\mathbf{e}_2 + \cdots + \lambda_t\mathbf{e}_t. \tag{3.7}$$

By the nature of conic combinations, if a reaction carries zero flux within $\mathbf{v}$, then that reaction must also be zero in all of $\mathbf{v}$'s constituent EMs. Therefore, if $\mathbf{e}_i$ is such an EM, then $\mathrm{Z}(\mathbf{v}) \subseteq \mathrm{Z}(\mathbf{e}_i)$.

Let all such EMs of $\mathbf{N}$ be denoted by the set $\mathfrak{E}_s$, i.e.

$$\mathfrak{E}_{\mathfrak{s}} = \{\mathbf{e}_i \in \mathfrak{E} | \quad Z(\mathbf{v}) \subseteq Z(\mathbf{e}_i)\}. \tag{3.8}$$

As discussed by Terzer [2009], $\mathfrak{E}_{\mathfrak{s}}$ are the EMs of the convex cone obtained when restricting the flux cone, C, such that the fluxes corresponding to the reactions in $\mathrm{S}(\mathbf{v})$ are zero, i.e.

$$C = \{\mathbf{u} \in \mathbb{R}^r | \mathbf{N}\mathbf{u} = \mathbf{0}, \mathbf{u} \geq \mathbf{0} \,\text{and}\, u_i = 0 \,\text{for all}\, i \in \mathrm{Z}(\mathbf{v})\}. \tag{3.9}$$

Without loss of generality, let

$$\mathbf{u}_\mathrm{Z} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_k \end{pmatrix}, \tag{3.10}$$

correspond to the subset of reaction fluxes constrained to zero, also, let the remaining elements of $\mathbf{u}$ be denoted by $\mathbf{u}_\mathrm{S}$. Similarly, let the reactions in $\mathbf{u}_\mathrm{Z}$ correspond to the first k columns of $\mathbf{N}$, denoted by $\mathbf{N}_\mathrm{Z}$, whilst the remaining columns are the sub-matrix $\mathbf{N}_\mathrm{S}$ (as defined previously), such that the system of equalities satisfied by C can be written as:

$$\begin{pmatrix} \mathbf{N}_\mathrm{Z} & \mathbf{N}_\mathrm{S} \end{pmatrix} \begin{pmatrix} \mathbf{u}_\mathrm{Z} \\ \mathbf{u}_\mathrm{S} \end{pmatrix} = \mathbf{0}. \tag{3.11}$$

By Fourier-Motzkin elimination, the variables of $\mathbf{u}_\mathrm{Z}$ can be eliminated from the system whilst preserving the original solutions over the remaining variables [Williams, 1986]. The first step is achieved by moving $\mathbf{N}_{\mathrm{Z}}\mathbf{u}_{\mathrm{Z}}$ to the left hand side of the equation:

$$\mathbf{N}_{\mathrm{S}}\mathbf{u}_{\mathrm{S}} = \mathbf{0} - \mathbf{N}_{\mathrm{Z}}\mathbf{u}_{\mathrm{Z}}. \tag{3.12}$$

But since $\mathbf{u}_{\mathrm{Z}} = \mathbf{0}$, this system is equivalent to:

$$\mathbf{N}_{\mathrm{S}}\mathbf{u}_{\mathrm{S}} = \mathbf{0}. \tag{3.13}$$

where $\mathbf{u}$ can be calculated from $\mathbf{u}_{\mathrm{S}}$ by appending $r - k$ zero elements to $\mathbf{u}_{\mathrm{S}}$ (at the appropriate indices).

Therefore, the EMs in $\mathfrak{E}_{\mathfrak{s}}$, i.e. the set of EMs of $\mathbf{N}$ that can constitute $\mathbf{v}$, are equivalent to those of $\mathbf{N}_{\mathrm{S}}$, i.e. the generators of the cone C_s:

$$C_s = \{\mathbf{u}_{\mathrm{S}} \in \mathbb{R}^{r-k} | \mathbf{N}_{\mathrm{S}}\mathbf{u}_{\mathrm{S}} = \mathbf{0}, \mathbf{u}_{\mathrm{S}} \geq \mathbf{0}\}. \tag{3.14}$$

□

Corollary 3.3.1.1. *For any flux vector* $\mathbf{v}$*, given the above definitions, there exists a minimal decomposition of at most* $\dim(\ker(\mathbf{N}_{\mathrm{S}}))$ *EMs.*

Proof. Let, $\mathbf{v}$, be a flux vector with $k \leq r$ reactions that carry a zero flux. Then, by Proposition 3.3.1.1, all of the possible constituent EMs of $\mathbf{v}$ can be found in the convex cone C_s:

$$C_s = \{\mathbf{u}_{\mathrm{S}} \in \mathbb{R}^{r-k} | \mathbf{N}_{\mathrm{S}}\mathbf{u}_{\mathrm{S}} = \mathbf{0}, \mathbf{u}_{\mathrm{S}} \geq \mathbf{0}\}. \tag{3.15}$$

The dimension of such a cone is $z = \dim(\ker(\mathbf{N}_{\mathrm{S}}))$, assuming that $\mathbf{N}_{\mathrm{S}}$ has full row-rank.

By Carathéodory's Theorem for polyhedral cones [Cook and Webster, 1972; Grotschel et al., 1988], for any point, $\mathbf{v}$, lying within a cone, C_s, of dimension z, being generated by the conic hull of a set of EMs, $\mathfrak{E}_{\mathfrak{s}}$, there exists a minimal composition of at most z of these EMs.

In fact, let $\mathbf{v} = \sum_{i=1}^{\ell} \lambda_i \mathbf{e}_i$ be a minimal decomposition for $\mathbf{v}$ with a set of coefficients

$\lambda_i > 0$.

Suppose that $\ell > z$, then $\ell - z$ of the $\mathbf{e}_i$'s must be linearly dependant, such that there must exist a linear dependence $\mathbf{0} = \sum_{i=1}^{\ell} \mu_i \mathbf{e}_i$ for a set of coefficients μ_i, some of which are non-zero.

Hence, a sufficiently small $\alpha \in \mathbb{R}$ can be found such that:

- $\lambda_q + \alpha\mu_q = 0$ for at least one $q \in \{1, 2, \ldots, \ell\}$, and
- and $\lambda_i + \alpha\mu_i > 0$ for all other i.

This creates conic decomposition, $\mathbf{v} = \sum_{i=1}^{l}(\lambda_i + \alpha\mu_i)\mathbf{e}_i$, with non-negative coefficients, $(\lambda_i + \alpha\mu_i)$, one of which must vanish.

But $\mathbf{v} = \sum_{i=1}^{\ell} a_i \mathbf{e}_i$ was chosen to be minimal, thus resulting in a contradiction.

□

3.3.2 Calculating a Linear Combination of Steady-State Fluxes

Proposition 3.3.2.1. *Consider two steady-state flux vectors,* $\mathbf{v}$ *and* $\mathbf{v}'$*, where* $\mathbf{v} \geq \mathbf{v}'$*, then their difference,* $\mathbf{v} - \mathbf{v}'$ *is a steady-state flux vector.*

Proof. Let $\mathbf{v}$ and $\mathbf{v}' \in \mathbb{R}^r$ be two steady-state flux vectors corresponding the metabolic network defined above. Suppose that $\mathbf{v} \geq \mathbf{v}'$ and all reactions are irreversible such that $\mathbf{v}, \mathbf{v}' \geq \mathbf{0}$.

Consider the vector $\mathbf{v}''$ obtained by the linear combination:

$$\mathbf{v}'' = \mathbf{v} - \mathbf{v}', \tag{3.16}$$

Since any steady-state flux vector of this metabolic network must be an element of the right null-space of $\mathbf{N}$, then

$$\mathbf{N}\mathbf{v} = \mathbf{0}, \tag{3.17}$$

and

$$\mathbf{N}\mathbf{v}' = \mathbf{0}. \tag{3.18}$$

Therefore,

$$\mathbf{N}\mathbf{v}'' = \mathbf{N}(\mathbf{v} - \mathbf{v}') = \mathbf{N}\mathbf{v} - \mathbf{N}\mathbf{v}' = \mathbf{0}. \tag{3.19}$$

Hence, $\mathbf{v}''$ is an element of the right null-space of $\mathbf{N}$.

Furthermore, since $\mathbf{v} \geq \mathbf{v}'$, then $\mathbf{v}'' \geq \mathbf{0}$. Consequently, $\mathbf{v}''$ satisfies the reaction reversibility criteria of the system.

Hence, since $\mathbf{v}''$ satisfies the constraints arising from the stoichiometry and reaction reversibility criteria of the system, then $\mathbf{v}''$ is in the flux cone and is therefore a steady-state flux of $\mathbf{N}$.

□

3.3.3 FBA Solutions and EMs

Proposition 3.3.3.1. *FBA solutions obtained after adding only one non-zero flux constraint to C are EMs.*

Proof. Consider a Flux Cone C, generated by the $m \times r$ stoichiometry matrix, $\mathbf{N}$, as defined above. Without loss of generality consider the addition of a minimum flux constraint, v_{min}, to the flux v_1, i.e. $v_1 \geq v_{\text{min}}$.

This new constraint corresponds to the half-space H, defined as follows

$$H = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{a}.\mathbf{v} \leq -v_{\text{min}}, \quad \mathbf{v} \geq \mathbf{0}, v_{\text{min}} \geq 0\}, \tag{3.20}$$

where

$$\mathbf{a} = \begin{pmatrix} -1 & 0 & 0 & \dots & 0 \end{pmatrix}, \tag{3.21}$$

since

$$\mathbf{a}\mathbf{v} = \begin{pmatrix} -1 & 0 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_n \end{pmatrix} = -v_1 \leq -v_{\text{min}}. \tag{3.22}$$

The convex polyhedron, P, containing the set of flux vectors, $\mathbf{v}$, that satisfy both the constraints imposed by $\mathbf{N}\mathbf{v} = \mathbf{0}$, as well as this new constraint is the intersection of C and H:

$$P = C \cap H, \tag{3.23}$$

This space, P, can be expressed as a system of inequalities:

$$P = \{\mathbf{v} \in \mathbb{R}^r | \hat{\mathbf{N}}\mathbf{v} \leq \mathbf{b}, \quad \mathbf{v}, \mathbf{b} \geq \mathbf{0}\}, \tag{3.24}$$

where

$$\hat{\mathbf{N}} = \begin{pmatrix} \mathbf{a} \\ \mathbf{N} \\ -\mathbf{N} \end{pmatrix} \text{ and } \mathbf{b} = \begin{pmatrix} -v_{\min} \\ \mathbf{0} \end{pmatrix}. \tag{3.25}$$

Maximum flux constraints can be added in a similar manner.

The addition of such a constraint restricts the flux cone by one hyperplane, H, such that the vertices of P will consist of the intersections between the extreme rays of C and the hyperplane. Therefore, given that FBA solutions must be vertices of P, it follows that such solutions must be EMs.

□

Corollary 3.3.3.1. *FBA solutions obtained after adding more than one non-zero flux constraint to C are not necessarily EMs.*

Proof. Consider now a set of flux constraints expressed as a system of inequalities generating a polyhedron Q:

$$Q = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{A}\mathbf{v} \leq \mathbf{v}_{\min}, \quad \mathbf{v} \geq \mathbf{0}\}, \tag{3.26}$$

where $\mathbf{A}$ is a matrix whose rows consist of vectors each constructed in the same manner as $\mathbf{a}$ in Proposition 3.3.3.1 above.

Note that since Q contains multiple constraints that each generate a hyperplane, then this system is itself a convex polyhedron with rays and vertices that arise from the intersection of the imposed constraints.

As before, the solution space, denoted by P, consists of the intersection of the flux cone, C, and convex polyhedron, Q:

$$P = C \cap Q, \tag{3.27}$$

that can be expressed as a system of inequalities representing P:

$$P = \{\mathbf{v} \in \mathbb{R}^r | \hat{\mathbf{N}}\mathbf{v} \leq \mathbf{b}, \quad \mathbf{v} \geq \mathbf{0}\}, \tag{3.28}$$

where

$$\hat{\mathbf{N}} = \begin{pmatrix} \mathbf{A} \\ \mathbf{N} \\ -\mathbf{N} \end{pmatrix}, \text{ and } \mathbf{b} = \begin{pmatrix} \mathbf{v}_{\min} \\ \mathbf{0} \end{pmatrix}. \tag{3.29}$$

This new system consists of the least convex polyhedron containing extreme rays and vertices of both C and Q [Halbwachs et al., 2006]. However, since the extreme rays and vertices of Q are not defined as EMs, then the extreme rays and vertices of P are not necessarily EMs. Therefore FBA solutions obtained from P are either EMs or a conic combination of EMs. □

It should be noted that FBA solutions obtained when only adding two constraints to the same reaction (i.e. a minimum and a maximum) must be EMs since the hyperplanes corresponding to these constraints are parallel and therefore do not intersect.

3.4 Applications to Models

Two metabolic models were analysed: a small model of the Calvin cycle and a GSM of *C. jejuni* (see Table 2.1 in Section 2.5.2 for a numerical description).

The Calvin cycle model was chosen since its behaviour is well understood and its EMs are documented [Poolman et al., 2003]. Therefore, it was used to compare the results obtained from the LPEMs algorithm with the expected biological behaviour, as well as with the results obtained from the algorithm by Poolman et al. [2004b] that decomposes flux vectors into EMs by calculating the complete set of EMs *a priori* (described in Section 3.2.1).

Once the correctness of the algorithm was tested on the Calvin cycle, attention was turned to the much larger *C. jejuni* model. As discussed in Section 3.1, this organism has a requirement for oxygen, the reasons for which are still unclear. Moreover, the scale of the model means that conventional EMA cannot be used to examine it, a problem that motivated the application of the LPEMs algorithm.

For both models, traditional FBA methods were initially applied to obtain flux vectors of interest, that were then used as input for Algorithm 1.

3.4.1 Analysis of the Calvin cycle

3.4.1.1 The Metabolism of the Calvin cycle

The Calvin cycle (Figure 2.7) is concerned with fixing environmental CO_2 into three-carbon sugars, in a process referred to as carbon fixation [Poolman et al., 2003].

The three-carbon sugars produced by this cycle (3-phospho-D-glycerate (PGA), glyceraldehyde 3-phosphate (GAP), and dihydroxyacetone phosphate (DHAP)) supply the cell with the organic material that it needs to build and maintain cellular structures. Each of these sugars is individually transported out of the chloroplasts via a triose-phosphate translocator (TPT) reaction. A fourth product, starch, is retained within the chloroplast, where it is used as a reserve of carbon and energy during periods of darkness.

By the principle of conservation of mass, the export of material from the system must be balanced by the input of an equal amount of atoms. Furthermore, energy must be provided to some of the enzymes driving the cycle in the form of ATP and NADPH created by photosynthesis's light-dependent reactions. Specifically, the enzyme RuBisCo fixes one carbon atom from CO_2 to form two molecules of PGA, that are then reduced to GAP and DHAP in a process that requires the input of two molecules of ATP and NADPH (one pair for each PGA molecule). Whilst a further input of one ATP molecule is needed such that a portion of the three-carbon sugars is recycled back into the cycle's starting compound, thus allowing the cycle to continue and more carbon to be fixed. Alternatively, carbon

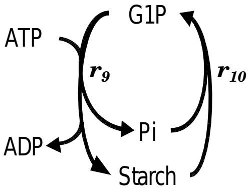


Figure 3.1: An EM which is a futile cycle that dissipates excess ATP.

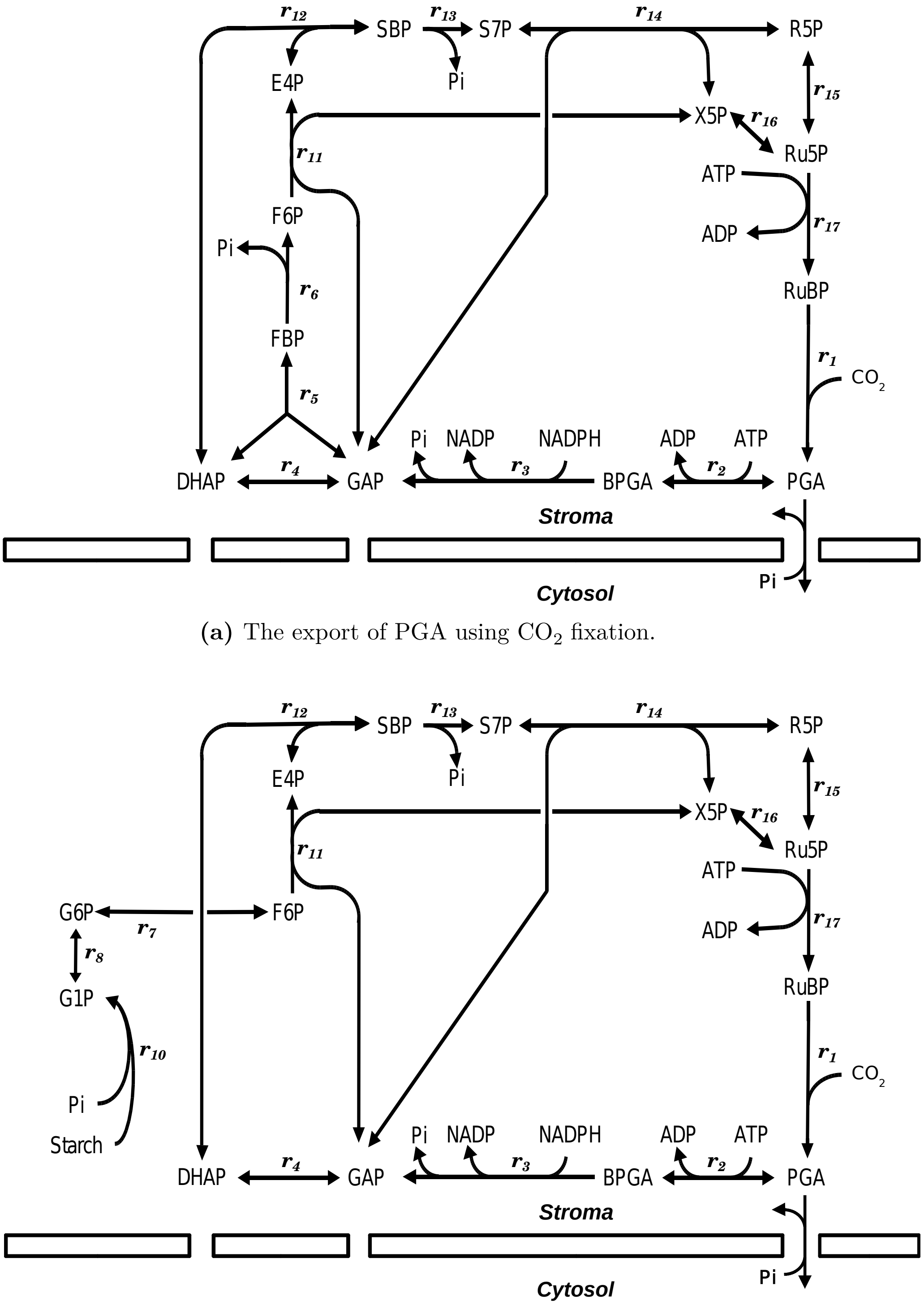
may also enter the cycle through the degradation of starch (imported as the 6-carbon glucose moiety).

Hence, as described by Poolman et al. [2003], the Calvin cycle model has eight EMs (Table C.1 in Appendix C) that connect inputs to outputs:

- three modes (ElMo_1 – ElMo_3) import CO_2 and respectively export one of the three-carbon sugars, see Figure 3.2a,
- three modes (ElMo_4 – ElMo_6) degrade starch to support the fixation of CO_2 and each export one of the three-carbon sugars, see Figure 3.2b,
- one mode (ElMo_7) imports CO_2 and exports starch,
- one mode (ElMo_0) consists of a futile cycle that simultaneously creates and degrades starch, see Figure 3.1.

These pathways can describe the underlying metabolic mechanisms of every potential steady-state flux and identify metabolic functions that can operate in accordance to specific cellular requirements and environmental conditions (such as to understand how the export of a specific product depends on the availability of nutrients and light-derived energy).

For example, since the export of carbon atoms out of the system requires the input of an equivalent amount of carbon, the production of one unit of starch (exported as a six-carbon glucose moiety) requires the fixation of six CO_2 molecules by RuBisCo. Hence, ElMo_7 must be supplied with 19 ATP molecules in order to function (where twelve ATP are required to reduce PGA to GAP and DHAP, six ATP are needed to regenerate the cycle's starting compound, and one ATP is required to transform one glucose moiety into starch).



(a) The export of PGA using CO_2 fixation.

(b) The export of PGA using CO_2 fixation and additional carbon provided by the degradation of starch.

Figure 3.2: The EMs that export PGA in the model of the Calvin cycle (Table C.1) [Poolman et al., 2003].

Similarly, the output of one of the 3-carbon sugar molecules produced by carbon fixation (ElMo_1– ElMo_3) requires the fixation of three CO_2 molecules, and hence the input of eight ATP in the case of PGA, and nine in the case of GAP and DHAP.

Meanwhile, supporting the export of carbon molecules by degrading starch requires less energy. Therefore, the ATP demand of the EMs that support the production of three-carbon sugars via starch degradation (ElMo_4 – ElMo_6) is one-third that of those that do so solely on CO_2 in the cases of GAP and DHAP, and one-fourth in the case of PGA. However, note that it is not possible to export the three-carbon sugars solely through starch degradation since these EMs also include RuBisCo.

ElMo_0 has been hypothesised to be a mechanism that dissipates excess energy produced by the light dependant reactions. Such a mechanism is needed when the cell produces more ATP than it requires, which is a possibility because of the activity of the light-dependent reactions being controlled by the amount of light in the cell's environment.

3.4.1.2 Flux Balance Analysis

The LPEMs algorithm described in this chapter was applied to a set of FBA solutions that simulate how the Calvin cycle responds to variations in the plant's demand for PGA.

The following LP problem was repeatedly solved:

$$\begin{aligned} \text{Find : argmin} \quad & \textstyle\sum_{i=1}^{r} |v_i| \\ \text{subject to} \quad & \begin{cases} \mathbf{Nv} & = \ \mathbf{0}\,, \\ v_{\text{TPT\_PGA}} & = \ t_{\text{PGA}}, \text{ for } t_{\text{PGA}} \in \{0, 5, 10, \ldots, 50\}. \\ v_{\text{LReac}} & = \ 100, \end{cases} \end{aligned} \tag{3.30}$$

The flux carried by the PGA transporter, $v_{\text{TPT\_PGA}}$, was incrementally increased (by a factor of five) from one solution to the next. While, the flux of the light reactions (and therefore the rate of ATP synthesis), v_{LReac}, remained unchanged throughout the analysis, ensuring that the same amount of energy is made available to the cycle in every iteration. The LP was run such that the total sum of fluxes is minimized, and the results may be found in Table C.2 in Appendix C.

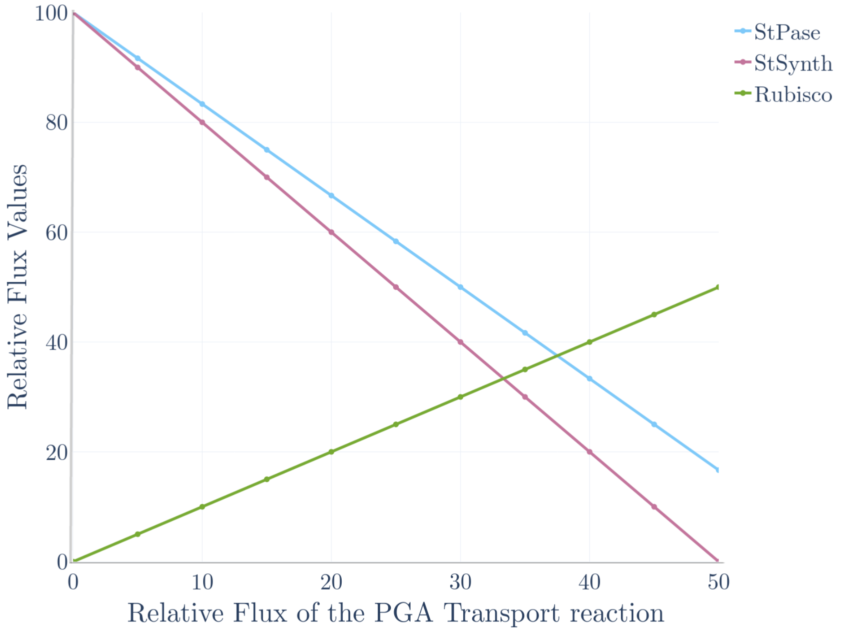


Figure 3.3: Relative fluxes of StPase, StSynth and RuBisCo with respect to changes in PGA export, where StPase is the enzyme responsible for degrading starch, StSynth is responsible to exporting starch, whilst RuBisCo fixes environmental CO_2. If there is no PGA demand, all energy provided by the light reactions is dissipated by the futile cycle. Starch synthesis stops when the cycle's maximum capacity for PGA export is reached.

In the first FBA iteration, the flux of the PGA transporter was constrained to be zero. Correspondingly, the resulting FBA flux portrayed the scenario where no net output of material was produced by the Calvin cycle. Hence, the only active reactions were those leading to starch formation and starch degradation, both operating in such a way that the ATP generated by the light reactions was being degraded at the same rate that it was being produced.

As the flux of the PGA transporter gradually increased in subsequent iterations, the energy produced from the light reactions started being used to power the cycle in order to meet the cell's growing PGA demand. The carbon needed to produce PGA was imported into the system through starch degradation and carbon fixation, such that the rate at which starch was being degraded was greater than the rate at which starch was being produced. This discrepancy between starch degradation and formation was seen to increase with the cell's PGA demand (see Figure 3.3).

The amount of PGA exported was incrementally increased until all of the energy derived from the light reactions was consumed to create PGA. After reaching this point, it is no longer possible to obtain feasible steady-state fluxes that exported a larger amount of PGA.

3.4.1.3 The LPEMs Algorithm

The algorithm described in this chapter was applied to this dataset of FBA solutions (Table C.3 in Appendix C).

The first flux was identified as ElMo_0, i.e. the futile cycle, that makes and simultaneously degrades starch. This outcome is consistent with biological expectations, as in the absence of any output, all energy derived from the light reactions must be dissipated.

In subsequent iterations, a low PGA output was expected to correspond to the scenario where not all of the ATP being produced by the light reactions is being used to export PGA, such that excess ATP must be dissipated using the futile cycle portrayed by ElMo_0. Accordingly, these fluxes were identified by the LPEMs algorithm as the sum of two EMs: ElMo_0 and ElMo_5 (where ElMo_5 exports PGA through the degradation of starch).

As the rate of PGA export continued to increase, the discrepancy between starch degradation and formation grew larger within the set of FBA solutions. This behaviour was identified by the LPEMs algorithm as an inverse relationship between ElMo_0 and ElMo_5, where the higher the PGA export, the lesser the amount of excess energy being dissipated by ElMo_0 (see Figure 3.4).

These results were found to be identical to the results obtained after decomposing the flux vectors into a sum of EMs through the conventional algorithm that requires the calculation of all of the EMs of the system *a priori* (as mentioned in Section 3.1). Both algorithms were also applied on a more complex steady-state flux of the Calvin cycle, and the results obtained were found to be consistent with each other. However, note that the decomposition of flux vectors into EMs is not necessarily unique.

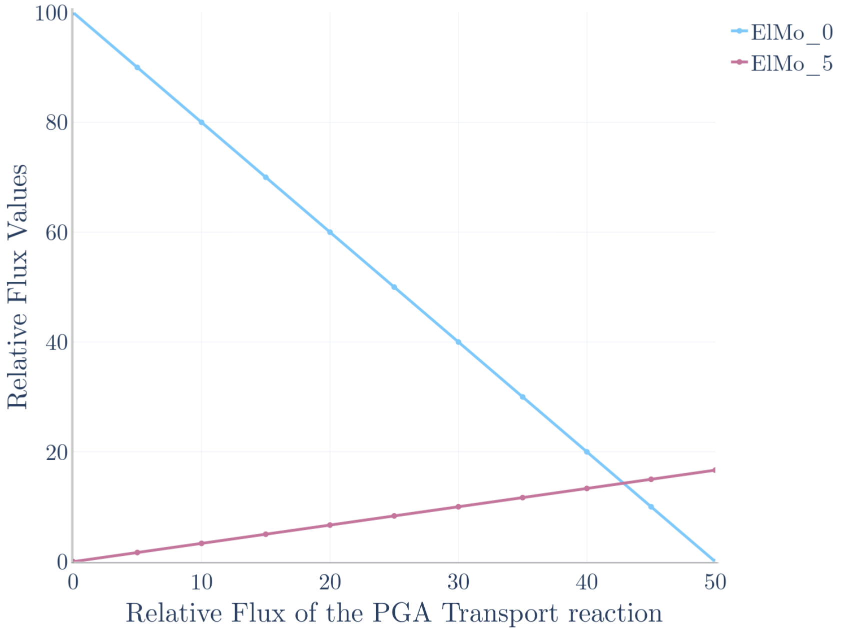


Figure 3.4: Relative fluxes of ElMo_0 (futile cycle, Figure 3.1) and ElMo_5 (PGA formation, Figure 3.2a) with respect to changes in PGA export.

3.4.2 Analysis of *C. jejuni*

Metabolism of *C. jejuni*

The metabolic properties of *C. jejuni* have been extensively studied. However, some aspects are still uncertain (refer to Hofreuter [2014] for a detailed review).

This organism can survive in a variety of hosts and is therefore expected to be tolerant of a wide range of environmental conditions, including varying temperatures, pH values, nutritional sources, and oxygen concentrations. However, some usually important enzymes in glycolysis and the pentose phosphate pathway are missing from its genome [Velayudhan and Kelly, 2002; Wagley et al., 2014]. As a consequence, it is incapable of using common carbohydrates (such as glucose) as a carbon source, and is believed to rely on pyruvate, some amino acids (such as glutamine), and TCA cycle intermediates for energy [Westfall et al., 1986; Guccione et al., 2008].

As discussed in Section 3.1.3, this organism has the capacity for anaerobic respiration (as its ETC has alternative electron acceptors to oxygen) but requires oxygen to generate biomass. Moreover, *C. jejuni* is a microaerophilic organism, meaning that it cannot tolerate high oxygen environments.

C. jejuni's intolerance to high oxygen environments has been speculated to possibly result from the build-up of toxic by-products generated by reactions that involve oxygen, such as hydrogen peroxide, which, if not neutralised, leads to damage in several cellular components [Kim et al., 2015; Rodrigues et al., 2016]. This balance between the presence of reactive oxygen species and the cell's ability to detoxify them is referred to as oxidative stress.

It has been suggested that its requirement for oxygen may be caused by a reliance on an oxygen-dependent ribonucleotide reductase (RNR) enzyme for DNA synthesis [Kelly, 2008; Sellars et al., 2002]. RNR enzymes are a vital part of the DNA synthesis process. They are involved in nucleotide metabolism by converting ribonucleotides (used in RNA) into deoxyribonuclease (used in DNA) [Torrents, 2014]. These enzymes are organised into three classes: I, II and III. The mechanism of class I enzymes requires oxygen to function, whilst class II and III do not.

Sellars et al. [2002] and Alqurashi et al. [2021] reported that *C. jejuni*'s genome encodes for oxygen-dependant class I RNR enzymes only, in contrast with *E. coli*, which encodes for both class I and class III RNR [Boston and Atlung, 2003]. Sellars et al. [2002] supported this hypothesis by showing that the presence of an RNR inhibitor (hydroxyurea) induces the same filamentation behaviour in *C. jejuni* (cells that elongate but do not divide), as that observed when the organism is placed in a strictly anaerobic environment.

Oxygen-dependant heme synthesis enzymes (HemN) have also been discussed within this context [Sellars et al., 2002]. However, the presence of genes that encode several HemN homologues within *C. jejuni*'s genome imply that the lack of anaerobic growth is unlikely to be caused by a heme deficiency [Parkhill et al., 2000; de Vries et al., 2015].

The following sections investigate this dependence in the context of producing biomass precursors whilst growing in a minimal medium.

3.4.2.1 Flux Balance Analysis

The *C. jejuni* model (provided by Tejera et al. [2020]) was initially analysed using the following LP problem:

$$\text{Find : argmin} \quad \textstyle\sum_{i=1}^{r} |w_i v_i|$$
$$\text{subject to} \quad \begin{cases} \mathbf{N}\mathbf{v} &= \mathbf{0}\,, \\ v_b &= t_b, \text{ for one or all } b \in \{1, 2, \ldots, B\}, \\ v_{\mathrm{O}_2} &= 0 \text{ or unconstrained.} \end{cases} \tag{3.31}$$

where $\mathbf{t}$ is a vector that specifies the production ratio of the B biomass components.

The LP's objective was to minimise the weighted sum of fluxes in the solution, $\mathbf{v}$, where the flux, v_i , carried by the i^{th} reaction (where r is the total number of reactions), is multiplied with the corresponding weighting coefficient w_i. These coefficients were all set to 1 (unless otherwise specified), assigning equal importance to the minimisation of every reaction.

This LP was subject to the steady-state condition $\mathbf{N}\mathbf{v} = \mathbf{0}$. Additionally, the production of one or more biomass components was considered by imposing the constraint, $v_b = t_b$, where the suffix b represents the b^{th} out of B exported biomass components whose proportional transport flux, v_b, was defined in accordance with experimental measurements by Metris et al. [2011]

When investigating the O_2 requirement, the flux carried by oxygen transport, v_{O_2}, was either unrestricted or forced to be zero. Similarly, the corresponding penalty weighting, w_{O_2}, was configured as 1 or 10^6 as described below.

Thus five sets of flux vectors were in total obtained, as shown in Table 3.1 and discussed below.

Simultaneous production of all biomass components, with unrestricted oxygen uptake penalty.

When the simultaneous production of all biomass was considered, and the import of oxygen was unrestricted, Equation (3.31) returned an optimal solution, verifying that the model can simulate growth on oxygen.

This flux vector, denoted here by $f_{aerobic}$, consisted of 324 reactions (excluding transport processes), four of which required oxygen. These included the oxygen transporter reaction, ATP synthesis (where oxygen is an electron acceptor within the ETC, R27 in Figure 3.5), pyridoxine (pyridoxamine) 5′-phosphate (PNP) oxidase (a reaction that produces PLP from PNP by using oxygen as a substrate and hydrogen peroxide is a by-product, R30 in Figure 3.5) and a catalase reaction (R31 in Figure 3.5) that degrades the generated hydrogen peroxide into water and oxygen. The ATP synthesis reaction had the highest flux within $f_{aerobic}$, showing the importance of aerobic respiration. Whilst the reaction regarding the synthesis of PLP had a magnitude three times smaller than the median flux.

$f_{aerobic}$ included 28 import reactions (the most dominant being pyruvate, phosphate, and glutamine) and 51 biomass output reactions (whose fluxes were defined when formulating the FBA optimisation problem). The import reactions included glutamine as the most dominant carbon source, followed by pyruvate. Glutamine, methionine, and cysteine provided nitrogen to the organism, with the latter two amino acids also satisfying *C. jejuni*'s demand for sulphur. Excess carbon was primarily exported in the form of carbon dioxide and carbonic acid, whilst excess nitrogen was excreted as NH_4.

Simultaneous production of all biomass components, with no oxygen uptake.

When oxygen import was set to zero, no solution was found, demonstrating that at least one biomass component has a requirement for oxygen.

Simultaneous production of all biomass components, with penalised oxygen uptake.

A solution that accounts for all biomass components while minimising oxygen import was obtained by adjusting the objective of Equation (3.2) such that the weight corresponding to the minimisation of oxygen, w_{O_2}, is dominant (i.e. set to an arbitrarily high value of 10^6). Thus the LP was encouraged to avoid using oxygen-requiring reactions in the resulting solution.

This solution, denoted by $f_{anaerobic}$, also consisted of 324 reactions, of which 320 were in common with $f_{aerobic}$. Three reactions required oxygen: the oxygen transporter reaction, and the reactions that produce PLP and neutralise hydrogen peroxide. In this case, ATP synthesis was not achieved using oxygen but via anaerobic respiration (using nitrate as an electron acceptor). Changes in the inputs/outputs were all associated with the redirection of the ETC from aerobic to anaerobic respiration. For example, the additional import of nitrate.

Individual production of each biomass component, with no oxygen uptake.
To identify oxygen-dependent biomass components, Equation (3.31) was repeatedly solved, once for each biomass component, with the oxygen transport constrained to zero in all cases. This LP was formulated such that a feasible solution is only attainable if the specified biomass component can be produced without oxygen (and vice-versa). In this case, optimal solutions were found for all biomass precursors, except for PLP, the active form of vitamin B6 and a co-substrate for many enzyme-catalysed reactions (see Section 3.5.2.2), thus implying that this precursor is responsible for *C. jejuni*'s lack of anaerobic growth.

Individual production of each biomass component, with unrestricted/penalised oxygen uptake.
Similarly, two sets of solutions that produce each of the biomass precursors individually, where the use of oxygen was (i) unrestricted and (ii) penalised were respectively obtained. As shown by Proposition 3.3.3.1, all of these solutions were EMs.

When oxygen was unrestricted, 43 of the 51 biomass precursors used oxygen to drive aerobic respiration. Aerobic respiration was substituted by anaerobic respiration (using nitrate) when oxygen use was penalised. In this case, only the solution for PLP utilised oxygen.

3.4.2.2 Applying the LPEMs Algorithm to the Aforementioned FBA Results

To further investigate the model's response to varying levels of oxygen, Algorithm 1 was applied to two of the FBA solutions, $f_{aerobic}$ and $f_{anaerobic}$. Both regarded the simultaneous

Table 3.1: LP solutions obtained when solving Equation (3.31) under different conditions.

	Biomass production	**Oxygen Uptake**	**LP**
1.	all components	unrestricted	a solution
2.	all components	blocked	no solution
3.	all components	penalised	a solution
4.	a sequence of LPs each producing one component	unrestricted	51 solutions
5.	a sequence of LPs each producing one component	blocked	50 solutions (none for PLP)
6.	a sequence of LPs each producing one component	penalised	51 solutions

production of all biomass. However, in $f_{aerobic}$, the use of oxygen was unrestricted, whilst, in $f_{anaerobic}$, the use of oxygen was penalised.

Algorithm 1 decomposed $f_{aerobic}$ into 62 EMs, 60 of which required oxygen. All these modes involved oxygen as an electron acceptor in the ETC, whilst one mode used additional oxygen molecules to synthesise PLP via PNP oxidase. Since the toxic hydrogen peroxide is a by-product of PNP oxidase, this EM also contains a reaction that decomposes hydrogen peroxide into water and oxygen.

When Algorithm 1 was applied to $f_{anaerobic}$, this flux was also decomposed into 62 EMs, only one requiring oxygen. As in this scenario, *C. jejuni* obtained energy via anaerobic respiration. In fact, all of the EMs were found to include the electron acceptor nitrate (rather than oxygen) in the ETC. As expected, the only EM that required oxygen synthesised PLP.

Both sets of EMs concurrently produced multiple outputs, however only one EM in both sets included the export of PLP. For example, the EM that produced PLP in $f_{anaerobic}$ produced acetate, succinate, and AMP as bi-products.

3.5 Discussion

3.5.1 Investigations on the Calvin Cycle

Applying the LPEMs algorithm to the results from the Calvin cycle yielded a clearer description than that provided by simply observing the set of varying reaction fluxes in Table C.2, by allowing users to compare the relative importance of the EMs in each biological scenario.

Although the decomposition of flux vectors into EMs is not unique in general, the results obtained here were identical to the results of the similar decomposition algorithm by Poolman et al. [2004b].

3.5.2 Investigations on *C. jejuni*

3.5.2.1 Model Analysis in the Context Of Micro-Aerophilly

The presence of oxygen-dependant reactions within a metabolic system does not necessarily identify which metabolites can and cannot be produced anaerobically. This is as the inherent flexibility of metabolic networks typically results in multiple pathways that can produce the same metabolite via alternative sets of reactions and input requirements. Such advantages allow organisms to adjust their metabolism in response to environmental changes.

In this work, PLP was identified to be the biomass precursor that cannot be exported without oxygen input by simulating the export of each biomass component individually under no oxygen import. However, this analysis did not reveal which specific reactions are responsible for this oxygen requirement.

On the other hand, the LP solutions that produced PLP when the use of oxygen was (i) unrestricted and (ii) penalized allowed for the production of PLP under these two different circumstances to be compared (see Figure 3.5). The model was seen to switch from aerobic to anaerobic respiration, depending on the availability of O_2. However, oxygen was being used by PNP oxidase (R30 in Figure 3.5) to generate PLP in both cases.

This reaction was confirmed to be the reason for oxygen dependency since it forms part of an enzyme subset with the reaction responsible for the transport of PLP out of the network (Section 2.3.3.1). Note that, although the model contains the catalase reaction (R31 in Figure 3.5) that degrades hydrogen peroxide into oxygen, this cannot be used to generate internal oxygen as the generation of hydrogen peroxide itself must ultimately depend on an exogenous oxygen source.

3.5.2.2 PLP Essentially

PLP is the active form of vitamin B6. More than 4% of known enzymes are thought to depend on PLP as a co-factor, including amino-acid metabolism, lipid metabolism, the synthesis of secondary metabolites derived from amino acids, gene expression, and nucleotide synthesis (e.g. thymidylate synthase within the folate cycle). In fact, the loss of PLP synthesis pathway in organisms has been reported to be lethal or lead to severe developmental problems [Parra et al., 2018].

PLP deficiency in *C. jejuni* was previously studied by Asakura et al. [2013], who used a *C. jejuni* mutant that is only capable of synthesising very small quantities of PLP. Lack of PLP prevented *C. jejuni* from building flagella, and resulted in alterations of TCA cycle intermediates. The authors suggested that the flagella impairment was because of PLP being required to build O-linked glycans, whilst alteration in the TCA cycle reflect the impact of PLP on *C. jejuni*'s energy metabolism.

3.5.2.3 Variation in *C. jejuni*'s Respiration

Numerous studies have observed *C. jejuni* shifting its respiration mode in order to account for limited oxygen availability (such as after infecting cells) [Liu et al., 2012; Kim et al., 2015; Reuter et al., 2010; Sellars et al., 2002; Wösten et al., 2017]. For example, Liu et al. [2012] reported that the levels of several proteins, such as those involved in aerobic respiration, are significantly reduced 20 hours after *C. jejuni* infects host cells. Similarly, Woodall et al. [2005] measured the up-regulation of enzymes such as those involved in TCA, electron transport, and dicarboxylate transport, as well as, down regulation of other enzymes such as some involved in amino acid synthesis and membrane lipids.

Whilst *C. jejuni*'s oxygen-limited growth is severely reduced when using standard media, the addition of fumarate, nitrate, nitrite, TMAO, or DMSO has been shown to have a very

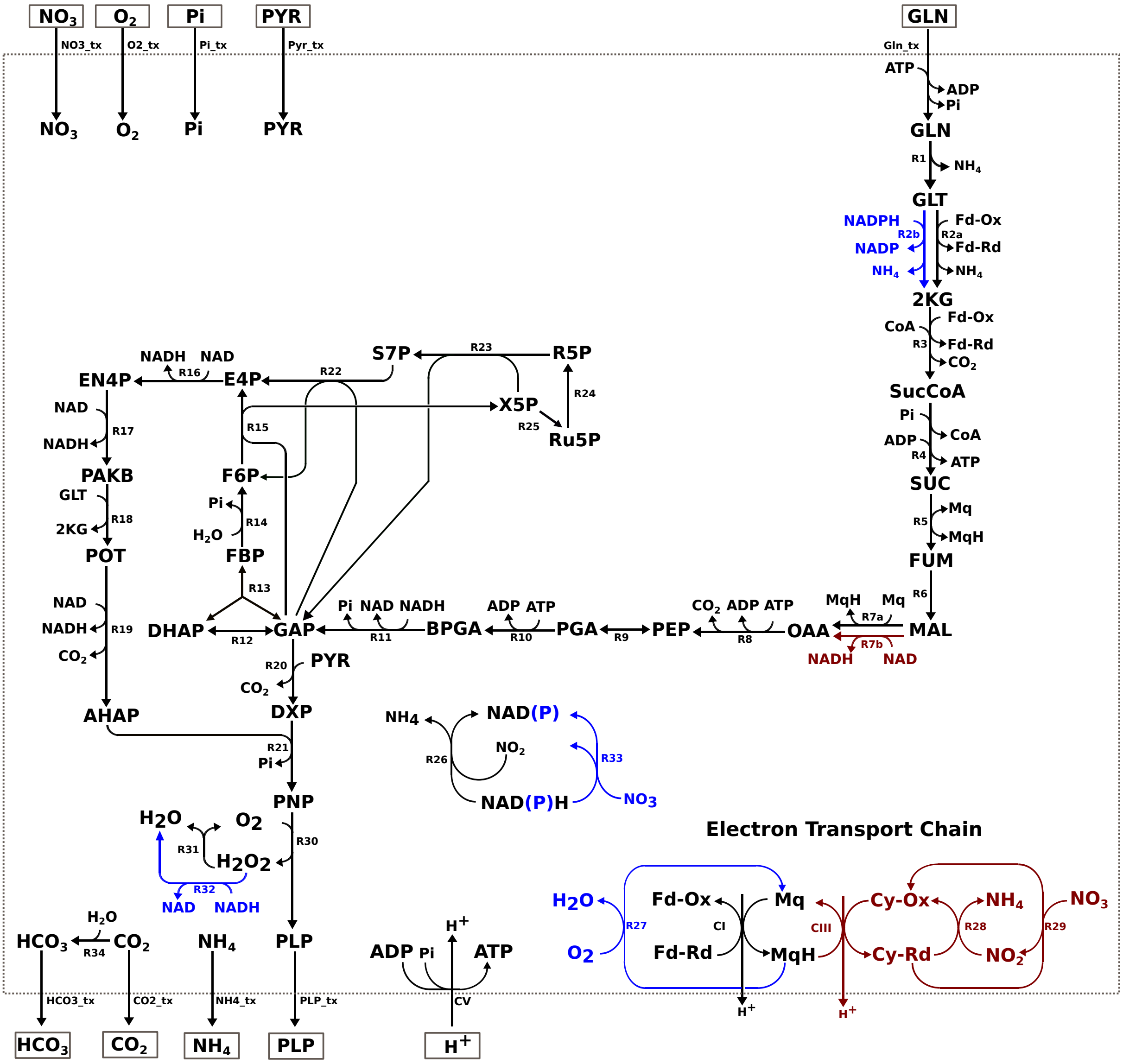


Figure 3.5: PLP synthesis in the *C. jejuni* model with and without out a penalty on O_2 uptake. Reactions in black are active under both the conditions. Reactions in blue are active when there is no penalty on O_2 uptake and reactions in red are active only when the penalty is imposed. See Appendix A.3 for reaction and metabolite abbreviations. This image was generated by Dr Dipali Singh for use in the pre-print within Appendix F.

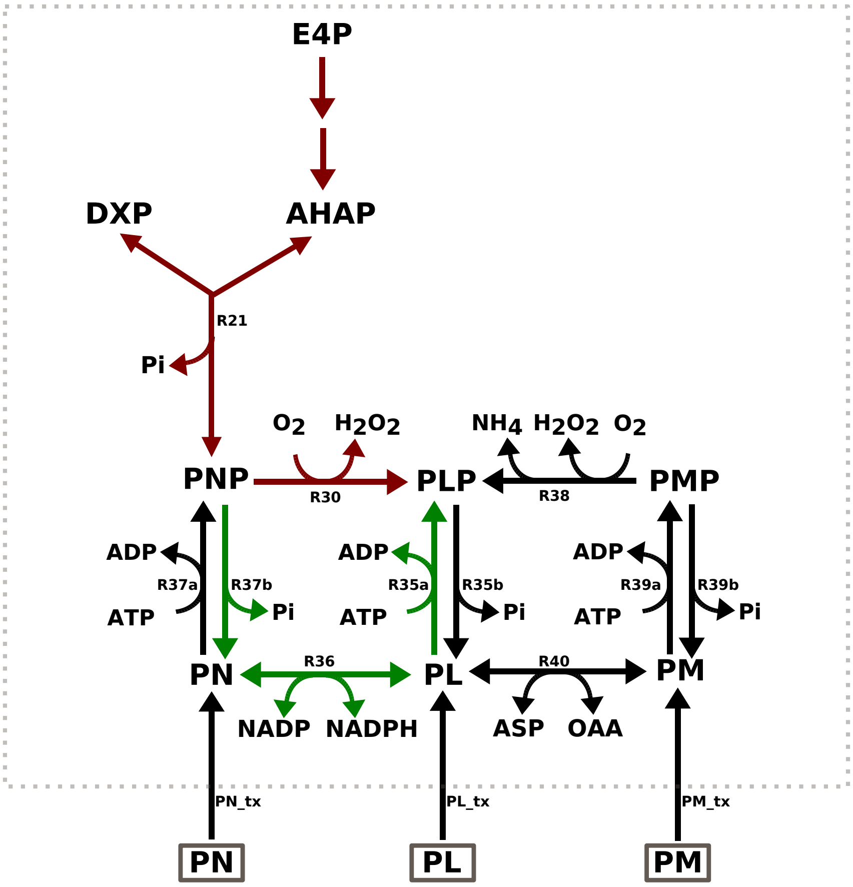


Figure 3.6: Reactions involved in PLP metabolism in *E. coli*. Reactions in red are common to *C. jejuni*, those in green allow for the bypass of the O_2 dependent pyridoxine 5′-phosphate oxidase step (R30). Reactions labelled R35a/R37a/R39a are all catalysed by the same enzyme (similarly for reactions labelled R35b/R37b/R39b and R30/R38). This image was generated by Dr Dipali Singh for use in the pre-print within Appendix F. See Appendix A.3 for reaction and metabolite abbreviations.

positive impact on its growth rate and final cell density [Sellars et al., 2002]. These results indicate that the oxygen-limited growth of *C. jejuni* may be partially reliant on sources like fumarate and nitrate (required as terminal electron acceptors within the ETC), that are likely to be abundantly available within the gut. Although *C. jejuni* may use a wide range of electron acceptors alternative to oxygen, it has been suggested to prefer to use nitrate over fumarate, possibly as fumarate can alternatively be used as a carbon source [Wösten et al., 2017].

3.5.2.4 PLP Synthesis

There are two known routes for the synthesis of PLP: the DXP (deoxy-xylulose 5-phosphate) dependent and DXP independent pathways. This study observed that *C. jejuni* exhibits the DXP-dependent pathway, which can be identified as the 2-branched subset of the pathway in Figure 3.5, starting from E4P and GAP and ending with the export of PLP.

Some organisms that use this pathway have additional reactions that enhance their metabolic flexibility. For example, as shown in Figure 3.6, *E. coli* contains reactions that can bypass the O_2 dependant PNP oxidase reaction (R30 in Figures 3.5 and 3.6), therefore allowing this organism to synthesise PLP without O_2 [Sugimoto et al., 2017; Ito and Downs, 2020]. These aforementioned reactions have not been reported *C. jejuni* M1cam.

3.5.2.5 Elementary Modes

In this study, two distinct sets of EMs were obtained. Using LP to identify pathways that synthesise each biomass precursor in isolation led to the identification of 51 EMs. However, although the overall behaviour generated by the combination of all 51 of these EMs produced all of the biomass precursors needed by *C. jejuni*, the resulting flux vector was not equivalent to that produced by the LP solution that simultaneously produced all biomass. This is as the summation of the 51 individual solutions required more reactions and greater total flux, illustrating how the optimal solution for the synthesis of a single product in isolation is not necessarily optimal in the presence of demand for additional products. Therefore, FBA solutions that maximize the production of a single product must be interpreted with care in the context of a growing organism.

3.5.3 Performance of the Algorithm

3.5.3.1 Efficiency of the Algorithm

A potential bottleneck regarding the efficiency of the LPEMs algorithm is the decomposition of solutions in Equation (3.31) that are not EMs. In such cases, the LP solution is decomposed into constituent EMs by creating a sub-model that only contains the reactions present within the LP solution, $\mathbf{v}'$, denoted by $\mathbf{N}_{\text{S}}$.

Obtaining the entire set of EMs of such a sub-model is efficient if the dimension of the sub-model is small. This was ensured by using an LP objective function that minimizes the l_1 norm (i.e. the sum of the absolute reaction fluxes), an approach that is likely to drive reaction fluxes to zero such that the number of non-zero variables within the LP solution is approximately minimal. In fact, minimization of the l_1 norm is commonly used to find approximate solutions for the problem of 'minimizing the number of non-zero variables in linear systems', which is NP-hard [Amaldi and Kann, 1998]). Therefore, every LP solution obtained from Equation (3.2) is likely to be either an EM or a flux vector with a small number of active reactions.

Furthermore, all reactions are considered to be irreversible (in accordance with the direction of the original flux in $\mathbf{v}$) which further improves efficiency. Indeed, in the results presented in this report (and other tests carried out during the algorithm's development), $\mathbf{N}_\mathrm{S}$ was sufficiently small such that EM enumeration could be achieved very quickly.

3.5.3.2 Accuracy and Reproducibility of the Algorithm

The accuracy of the algorithm is ensured by verifying that the EMs sum to the original flux vector, and that, at each point, flux vectors are at steady state.

Generally, the decomposition of a flux vector into a combination of EMs is not unique (since the number of EMs for a given system typically far exceeds the dimension of the null-space). Therefore, although the algorithm presented here cannot be guaranteed to generate a unique decomposition, as long as at each iteration of Algorithm 1, the value of v_min is unique, then the results generated for a given flux vector are expected to be reproducible (e.g. independent of initial row and column ordering of the stoichiometry matrix). However, the presence of multiple optima when solving Equation (3.2) may lead to repetitions of the algorithm yielding different results, but there was no evidence of this during the course of this study.

3.5.3.3 Size of the Decomposition

The minimal size of the decomposition is 1, if the input flux vector is already an EM, to at most $\binom{2r+m}{2m}$ for a sub-model with m metabolites and r reactions, which is the upper bound for the number of EMs in a model as shown by Theorem 2.4 in Terzer [2009, page 42].

As stated by Corollary 3.3.1.1, for each flux vector there exists a minimal decomposition that has at most S($\mathbf{v}$) EMs. In this study the LPEMs algorithm decomposed the two LP solutions of the *C. jejuni* model, into 62 EMs, which was close to the dimension of their respective null-spaces (51 for both cases).

3.6 Conclusion

This chapter provides a novel method to decompose flux vectors into a set of fundamental flux pathways called EMs, which gives a relatively simple and computationally efficient way to leverage the advantages of EMA with respect to FBA solutions. Results describing the implications of environmental oxygen saturation on a *C. jejuni* bacteria are presented, and a collaboration with the Quadram Institute in Norwich is expected to conclude with the publication of a research paper that is currently under review. Furthermore, the code developed has been released as part of the ScrumPy metabolic modelling software.

Chapter 4

The Right Null-Space and the Steady-State Solution Space

4.1 Introduction

This chapter discusses network reduction strategies, which aim to reduce the size of models by decreasing the number of involved reactions and/or metabolites. Historically, different algorithms based on this concept have been used to achieve a variety of goals, such as to improve the efficiency of modelling algorithms, reduce the occurrence of multiple optima in FBA, and emphasise characteristics of interest.

These techniques can be divided into two categories: lossy and lossless. Lossy techniques permanently discard information (such as by removing reactions/metabolites), potentially changing the overall behaviour of the network whilst preserving some desired attributes. In contrast, lossless techniques reduce the size of networks without altering their behaviour. This is achieved by removing redundancies from the stoichiometry matrix and allows the reduction process to be reversed.

Alternatively, methods that reduce a model's potential behaviour by imposing flux constraints also exist, with a notable example involving the export of biomass precursors in models.

4.1.1 Lossy Model Reduction Techniques

Lossy techniques have been used to extract small sub-networks from GSMs. One of the simplest approaches is to regard the output of FBA as a sub-network on which further analysis can be performed [Hartman et al., 2014]. For example, Minimal Reaction Sets use MILP to identify a minimal number of reactions forming a sub-network capable of supporting growth. This method was first described by Burgard et al. [2001] to demonstrate the high degree of redundancy in a model of *Escherichia coli* (720 reactions) by showing how only a small subset of reactions (referred to as a growth-sustaining core) is essential for supporting growth under a specific set of conditions.

NetworkReducer, developed by Erdrich et al. [2015], expands upon the concept of Minimal Reaction Sets by preserving pre-selected properties, such as protected reactions and phenotypes. This technique uses FVA (Section 2.3.5) to iteratively remove reactions, starting with the reaction whose flux range is closest to zero, until reaching a sub-network from which no more reactions can be removed without violating the pre-selected conditions. Erdrich et al. [2015] applied this method to a model of *E. coli* (2383 reactions). NetworkReducer obtained a sub-network that included a set of pre-selected reactions from central carbon metabolism and achieved a growth yield (on glucose) consistent with that of the larger original model. Similar application on a GSM of the cyanobacteria *Synechocytis sp.* (599 reactions) determined a sub-network that preserved the GSM's phototrophic growth and ethanol yield. This sub-network allowed the relationship between yield and growth to be examined through EMA.

Alternatively, some approaches utilise graph theory: RedGem by Ataman et al. [2017], starts with a set of user-specified sub-networks of interest (such as glycolysis and the citric acid cycle) and identifies the additional reactions needed to connect these modules. Afterwards, optimisation methods restore essential cellular functions (such as biomass production) and ensure thermodynamic consistency [Ataman and Hatzimanikatis, 2017]. This approach aims to retain global features of the network, such as flux variability and gene essentially. Ataman et al. [2017] applied this method to an *E. coli* GSM (1136 reactions). The authors selected six subsystems relating to central carbon metabolism (glycolysis, pentose phosphate pathway, citric acid cycle, glyoxylate), then RedGem extracted more reactions from the original model in order to connect the subsystems and allow for the production of 102 biomass precursors.

A drawback of these techniques is that the analysis of sub-networks is limited to the context in which they were defined.

4.1.2 Lossless Model Reduction Techniques

Lossless techniques take advantage of the row and columns dependencies within the stoichiometry matrix, each of which has been historically used in distinct ways.

As discussed in Chapter 2.3.2, linear dependencies between the rows of the stoichiometry matrix lead to redundancies in the model's dynamic behaviour. Specifically, conservation relations are groups of metabolites whose sum of concentrations must stay constant through time. These relations cause the concentration of some metabolites to be dependent on some others, and such dependent metabolites must be eliminated from the network prior to applying kinetic modelling techniques [Sauro and Ingalls, 2003].

Dependencies between the columns of the stoichiometry matrix provide the means to decrease the number of reactions without sacrificing information about steady-state fluxes and can therefore be used to reduce the size of structural models. Enzyme subsets (see Chapter 2.3.3.1) correspond to groups of coupled reactions that can be combined into one reaction that is defined by the group's net stoichiometry. In addition, dead reactions (see Chapter 2.3.3.2) can be deleted.

In most cases lossy techniques are expected to reduce models more than lossless techniques [Singh and Lercher, 2020].

One advantage of lossless reduction is that since a reduced model retains the same metabolic capacity as its larger original counterpart, any insights obtained from the smaller model can be extrapolated to the larger. This has allowed this reduction technique to be employed as a pre-processing step when implementing computationally costly algorithms [Chindelevitch et al., 2014]. For example, decreasing the size of the stoichiometry matrix by merging enzyme subsets before applying EM enumeration algorithms reduces the computational resources required [Pfeiffer et al., 1999]. After the EMs of the reduced network are calculated, they are then converted to the equivalent EMs of the larger network by reversing the steps taken when merging enzyme subsets.

4.1.3 Biomass Export Constraints

As discussed in Section 2.1.1, biomass precursors consist of the various chemical building blocks that a cell's metabolism must produce for the subsequent synthesis of its required polymers (which are not defined in GSMs). Consequently, a model's steady-state behaviour can be reduced by enforcing these precursors to be produced in the proportions in which they are known to be utilised in the organism. This reduction is likely to direct FBA towards realistic results and decrease the occurrence of multiple optima. It can be achieved in two ways, as discussed below and in Section 4.4.2.

The first technique defines the output of GSMs as a single biomass reaction in which all the components are assumed to combine, in appropriate ratio, to produce one unit (typically one gram) of biomass. Such a definition is convenient when using FBA objective functions that maximise biomass output, since the corresponding LP objective is the maximisation of this biomass reaction [Feist and Palsson, 2010].

The alternative approach, utilised by the CSM group at Oxford Brookes University, assigns individual pseudo-transport reactions to each biomass component. This was originally done as a technical convenience, as it makes it much easier to assess and investigate the effect of changes to biomass composition on the entire network. It is also more realistic as there is no single biomass reaction *in vivo*: in reality all precursors can be produced independently and are combined by a number of different processes to produce the final polymeric compounds that constitute biomass.

When using such models to simulate growth using FBA with an objective to minimise the total amount of flux, a single constraint enforcing the production of one unit of biomass is required for models that have combined biomass export [Al-Saidi, 2020, page 51], whilst the LP for models that contain individual biomass transporters uses one constraint for each precursor, all specified at the experimentally known proportions [Holzhütter, 2004].

The theoretical implications of the contrast between these two approaches have been the cause for discussion. For example, it has been debated whether combining biomass leads to a loss of information that may have detrimental consequences on the modelling results, such as by resulting in less-than-optimal solutions, or losing track of the movement of important metabolites and atoms.

4.1.4 Aims and Objectives

The algorithm presented in this chapter improves upon the reduction achieved through enzyme subsets.

The chemical transformation associated with each enzyme subset is not necessarily unique within the model — meaning that some subsets might have the same net stoichiometry as other subsets or reactions in the system. Therefore, reducing the size of the models by combining enzyme subsets has the potential to create duplicate columns in the stoichiometry matrix, corresponding to groups of reactions with the same net stoichiometry (i.e. identical substrate and products) — defined here as *iso-stoichiometric groups*.

Iso-stoichiometric groups correspond to iso-enzymes or enzyme subsets that perform the same chemical transformation [Schuster et al., 1999; Mavrovouniotis et al., 1990]. Schuster et al. [1999] gave the example of the degradation of adenosine monophosphate (AMP), which can either occur by dephosphorylation followed by deamination or the other way around. The authors recommended that combining each of these groups into a single reaction may decrease the computing burden of EMA.

In this chapter, a strategy that identifies and eliminates all iso-stoichiometric groups from the stoichiometry matrix is presented. This technique, along with combining enzyme subsets, enables the size of metabolic networks to be iteratively reduced in a lossless manner — through a process defined here as *compression* — resulting in a model that cannot be reduced further since all of its reactions have a unique stoichiometry and no enzyme subsets are present.

Apart from achieving a better reduction than when merely combining enzyme subsets, this approach aims to expose structural features in models. Specifically, identifying iso-stoichiometric groups reveals alternative pathways from the same start to end product (also referred to as parallel branches [Schuster et al., 1999; Mavrovouniotis et al., 1990]). This technique facilitates model curation by (i) identifying duplicate processes that are erroneously included in models when automatically constructing them from online databases, and (ii) detecting thermodynamically-infeasible internal cycles, which are known to cause errors during model analysis [Poolman et al., 2007; Kelk et al., 2012].

Once the errors arising from model construction are removed, knowledge of genuine alternative pathways provides insights into the redundancy mechanisms present in metabolic networks, and can therefore be used to pinpoint routes that a network may use to compensate for gene knockouts (therefore aiding the design of multiple-knockouts strategies), and to widen the knowledge gained from FBA solutions by enabling the discovery of alternate optima (as discussed in Chapter 2.3.5).

Furthermore, Section 4.4.2 mathematically defines the difference between the two approaches for defining biomass production in models by showing that (i) the steady-state solution space of a model with combined biomass export is a subset of the solution space of an equivalent model that contains individual transporters for each biomass precursor, and (ii) the solution space of the two aforementioned FBA problems are equivalent to each other.

4.1.5 Chapter Structure Overview

This chapter contains the following sections:

Section 4.2 shows how redundancies in the steady-state behaviour of a network, in the form of enzyme subsets or iso-stoichiometric groups, can be identified through the stoichiometry matrix and removed.

Section 4.3 describes a novel lossless compression technique that sequentially eliminates redundancies from the right null-space of metabolic networks, as well as a technique that uses the information gained from the compression procedure to reveal alternative pathways from the same start and end product.

Section 4.4 provides mathematical results regarding enzyme subsets and the definition of biomass production in models.

Section 4.5 applies these developed methods to various metabolic models.

Section 4.6 discusses the significance of results to model curation and analysis.

4.2 Background for the Novel Algorithm

4.2.1 Enzyme Subsets

As described in Chapter 2.3.3.1, enzyme subsets can be identified and removed from $\mathbf{N}$ as explained below.

Consider the model introduced in Section 2.3 and Figure 2.2, consisting of the following internal stoichiometry matrix

$$\mathbf{N} = \begin{array}{c} \\ \mathrm{A} \\ \mathrm{B} \\ \mathrm{C} \\ \mathrm{D} \end{array} \!\! \begin{array}{c} \begin{array}{ccccc} \mathrm{A\_}tx & r_1 & r_2 & \mathrm{C\_}tx & \mathrm{D\_}tx \end{array} \\ \left(\begin{array}{ccccc} 1 & -2 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{array} \right) \end{array}, \tag{4.1}$$

for which the corresponding external stoichiometry matrix is

$$\mathcal{N} = \begin{array}{c} \\ \mathrm{x\_A} \\ \mathrm{A} \\ \mathrm{B} \\ \mathrm{C} \\ \mathrm{D} \\ \mathrm{x\_C} \\ \mathrm{x\_D} \end{array} \!\! \begin{array}{c} \begin{array}{ccccc} \mathrm{A\_}tx & r_1 & r_2 & \mathrm{C\_}tx & \mathrm{D\_}tx \end{array} \\ \left(\begin{array}{ccccc} -1 & 0 & 0 & 0 & 0 \\ 1 & -2 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{array} \right) \end{array}. \tag{4.2}$$

The right null-space matrix, **K**, of **N** is:

$$\mathbf{K} = \begin{array}{c} \mathrm{A\_}tx \\ r_1 \\ r_2 \\ \mathrm{C\_}tx \\ \mathrm{D\_}tx \end{array} \left(\begin{array}{c} 2 \\ 1 \\ 1 \\ 1 \\ 1 \end{array} \right), \tag{4.3}$$

such that, in any steady-state flux vector, **v**, of this system, the transport reaction for A, A_*tx*, must be operating such that its flux is twice that of all other reactions in the network (which in turn must be equal to each other).

Therefore, the reactions in $\mathbf{v}$ form an enzyme subset, Ess_1, where their fluxes must satisfy:

$$v_A = 2v_1 = 2v_2 = 2v_C = 2v_D, \tag{4.4}$$

where v_A, v_1, v_2, v_C and v_D are the fluxes of A_$tx, r_1, r_2,$ C_tx and D_tx respectively.

Let these reactions correspond to the columns (A_tx), $(r_1), (r_2),$ (C_tx) and (D_tx) in $\mathcal{N}$. Then these columns can be replaced by one column, (Ess_1), corresponding to the net stoichiometry of the subset:

$$Ess_1 : \qquad 2\,\text{x\_A} \rightarrow \text{x\_C} + \text{x\_D}, \tag{4.5}$$

which is calculated as

$$(\text{Ess}_1) = 2(\text{A\_tx}) + (\text{r}_1) + (\text{r}_2) + (\text{C\_tx}) + (\text{D\_tx}), \tag{4.6}$$

such that the external stoichiometry matrix becomes

$$\mathcal{N} = \begin{array}{c} \\ \text{x\_A} \\ \text{x\_C} \\ \text{x\_D} \end{array} \begin{array}{c} Ess_1 \\ \left(\begin{array}{c} -2 \\ 1 \\ 1 \end{array}\right) \end{array}. \tag{4.7}$$

In general, for an $m \times r$ stoichiometry matrix $\mathbf{N}$ with columns $(r_1), (r_2), \ldots, (r_r)$, consider the enzyme subset, Ess_2, that relates the fluxes of $k \leq r$ reactions of $\mathbf{N}$ such that:

$$v_1 = \frac{1}{\lambda_2} v_2 = \cdots = \frac{1}{\lambda_k} v_k, \tag{4.8}$$

where v_1, v_2, and v_k denote fluxes of the reactions whose corresponding columns of $\mathbf{N}$ are (r_1), (r_2), and (r_k) respectively, and the λ_i's are constants. Then, the reactions in Ess_2 can be replaced by a column, (Ess_2), with the net stoichiometry of the subset:

$$(Ess_2) = (r_1) + \lambda_2(r_2) + \cdots + \lambda_k(r_k), \tag{4.9}$$

in **N**, and similarly in $\mathcal{N}$.

The above procedure applies column operations to **N**, which typically alter the null-space and therefore the steady-state solution space. However, this specific procedure leaves the solution space unchanged as shown in by Theorem 4.4.1.1 in Section 4.4.1.

4.2.2 Iso-stoichiometric Groups

Iso-stoichiometric groups are identified as proportional columns of the stoichiometry matrix (where each column corresponds to a vector) and can be removed by deleting all but one of the duplicate reactions (assuming that directionality is respected). For example, consider the two reactions r_1 and r_2:

$$\begin{aligned} r_1 : \quad & \mathrm{A} \rightarrow \mathrm{B}, \\ r_2 : \quad & \mathrm{A} \rightarrow \mathrm{B}, \end{aligned} \tag{4.10}$$

they form an iso-stoichiometric group, and can replaced by one reaction, *iso*;

$$iso : \quad \mathrm{A} \rightarrow \mathrm{B}. \tag{4.11}$$

Iso-stoichiometric groups that contain irreversible reactions that do not have the same directionality as each other should be considered as inconsistent. Their presence gives rise to internal cycles that can cause the results of analysis to violate the first law of thermodynamics (which states that energy can neither be created nor destroyed, but only altered in form [Nelson and Cox, 2004, page 490]) as shown by Figure 4.1. This phenomenon occurs when the directionality of some reactions is wrongly defined during model construction and can be remedied by analysing thermodynamic information relating to all of the reactions in the group. Since this information is not automatically available, an *ad hoc* solution used here is to ensure that the directionality of the retained reaction reflects the joint capabilities of all reactions within its group, for example the following two reactions:

$$\begin{aligned} r_1 : \quad & \mathrm{A} \rightarrow \mathrm{B}, \\ r_2 : \quad & \mathrm{A} \leftarrow \mathrm{B}, \end{aligned} \tag{4.12}$$

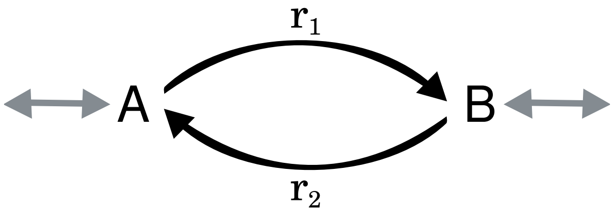


Figure 4.1: An inconsistent iso-stoichiometric group (r_1 and r_2) which forms an internal cycle.

are replaced by

$$iso: \quad \text{A} \leftrightarrow \text{B}. \tag{4.13}$$

It should be noted that some iso-stoichiometric groups considered in this thesis do not consist of individual reactions as described above, but instead contain pathways that achieve the same net-stoichiometry through a sequence of interconnected reactions. These groups are detected by first identifying each pathway as an enzyme subset, which is then combined into a single reaction with the subset's net-stoichiometry (as shown in Section 4.2.1), such that the members of the group are transformed into individual reactions with identical stoichiometry as defined above.

4.3 Methodology

In this section, four algorithms are described:

- Algorithm 2, *Compress*: compresses a model,
- Algorithm 3, *Decompress*: when applied to a compressed model, reverses the steps undertaken during compression,
- Algorithm 4, *Expand_vector*: when applied to a flux vector obtained from the analysis of a compressed model, reverses the steps undertaken during compression, to return the equivalent vector/s of the original model,
- Algorithm 6, *Alternate_vector*: when applied to a flux vector obtained from the analysis of an original model, returns vector/s that achieve the same net stoichiometry through different internal pathways (as identified by the model's iso-stoichiometric groups).

The implementation of these algorithms is designed such that information about the steps undertaken during model compression (Algorithm 2) is accessible to all other associated algorithms (Algorithms 3 - 6).

4.3.1 The Compression Algorithm

Algorithm 2 *Compress*($\mathbf{N}$), which recursively reduces the size of a stoichiometry matrix $\mathbf{N}$, by sequentially combining enzyme subsets and iso-stoichiometric groups until a matrix that cannot be reduced any further is reached.

```
 1: $\mathbf{N_{map}}$ = copy($\mathbf{N}$)
 2: isos = find_isostoichiometric_groups($\mathbf{N}$)
 3: replace_isostoichiometric_groups($\mathbf{N}$, isos)
 4: ess = find_enzyme_subsets($\mathbf{N}$)
 5: replace_enzyme_subsets($\mathbf{N}$, ess)
 6: mapping = [isos, ess, $\mathbf{N_{map}}$]
 7: save_to_mappings(mapping)
 8: if col_size($\mathbf{N_{map}}$) > col_size($\mathbf{N}$) then
 9:     Compress($\mathbf{N}$)
10: end if
```

The compression algorithm (described in Algorithm 2) reduces the size of a stoichiometry matrix, $\mathbf{N}$, by assuming that the following transformations do not alter the steady-state behaviour of the network:

- Enzyme subsets can be replaced by a single reaction (as discussed in Section 4.2.1).
- Iso-stoichiometric groups can be replaced by a single reaction (as discussed in Section 4.2.2).

Before initialising the algorithm, dead reactions are identified and removed.

The algorithm then proceeds as follows: first, reactions that are members of iso-stoichiometric groups are identified and removed (Steps 2 and 3, as discussed in Section 4.2.2). Afterwards, enzyme subsets are calculated (Step 4) and their reactions are combined as discussed in Section 4.2.1 (Step 5). These substitutions result in a reduced stoichiometry matrix, containing fewer reactions and metabolites than the original, and may lead to the formation of new iso-stoichiometric groups (i.e. duplicate columns that were not present in the original matrix), which upon elimination may result in the formation of more enzyme subsets.

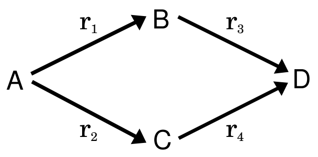
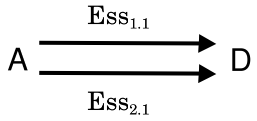
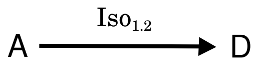
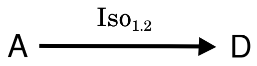
Iteration	**Network**	**Stoichiometry Matrix**	**State**
0		$\begin{array}{ccccc} & r_1 & r_2 & r_3 & r_4 \\ A & -1 & -1 & 0 & 0 \\ B & 1 & 0 & -1 & 0 \\ C & 0 & 1 & 0 & -1 \\ D & 0 & 0 & 1 & 1 \end{array}$	N/A
1		$\begin{array}{ccc} & Ess\_1.1 & Ess\_2.1 \\ A & -1 & -1 \\ D & 1 & 1 \end{array}$	$\mathrm{saved}_1 : [\mathrm{Ess}_{1.1} : [\mathrm{r}_1, \mathrm{r}_3], \mathrm{Ess}_{2.1} : [\mathrm{r}_2, \mathrm{r}_4]\,]$
2		$\begin{array}{cc} & Isos\_1.2 \\ A & -1 \\ D & 1 \end{array}$	$\mathrm{saved}_2 : [\mathrm{Isos}_{1.2} : [\mathrm{Ess}_{1.1}, \mathrm{Ess}_{2.1}]\,]$
3		$\begin{array}{cc} & Isos\_1.2 \\ A & -1 \\ D & 1 \end{array}$	$\mathrm{saved}_3 : [\,]$

Figure 4.2: The process of the *Compress* algorithm applied to a simple model, where the state of the algorithm at the end of each iteration is shown (the zeroth state corresponds to the state of **N** upon initialisation of the algorithm). The algorithm terminates at iteration three when there is no changes in state in comparison to the previous iteration.

Therefore, the previous steps are repeated, allowing for the size of the stoichiometry matrix to be recursively reduced until a stoichiometry matrix that cannot be reduced further, due to lacking iso-stoichiometric groups and enzyme subsets, is obtained (Steps 8 and 9).

At each iteration, information regarding the modifications done to the stoichiometry matrix is saved (Steps 6 and 7), namely:

1. the names of enzyme subsets and the reactions that they contain (weighted by the relative flux ratio that they must carry with respect to each other),
2. the names of iso-stoichiometric groups and the reactions that they contain (weighted by their relative stoichiometry with respect to the reaction retained in **N**),
3. the directionality and stoichiometry of the reactions in the model (in the form of a stoichiometry matrix).

Additionally, the names of inconsistent iso-stoichiometric groups are taken note of. The progress of this algorithm when applied to a simple model is illustrated in Figure 4.2.

4.3.2 The Decompression Algorithm

Algorithm 3 *Decompress*(**N**), which when applied to a reduced stoichiometry matrix, **N**, reverses the steps taken by the compression algorithm to return a stoichiometry matrix, **N′**, that is equivalent to that of the original model.

```
mappings = retrieve_mappings()
$\mathbf{N}'$ = copy($\mathbf{N}$)
for [isos, ess, $\mathbf{N_{map}}$] in mappings do
    $\mathbf{N}'$ = expand_enzyme_subsets($\mathbf{N}'$, ess, $\mathbf{N_{map}}$)
    $\mathbf{N}'$ = expand_isostoichiometric_groups($\mathbf{N}'$, isos, $\mathbf{N_{map}}$)
end for
return $\mathbf{N}'$
```

The decompression algorithm, described in Algorithm 3, reverses the steps undertaken during compression. This algorithm initiates by retrieving the information that had been saved by the compression algorithm, which describes the modifications made to the original model at each step of the compression and stoichiometry of removed reactions.

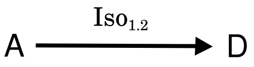
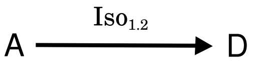
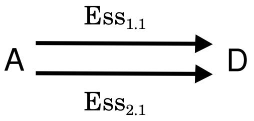
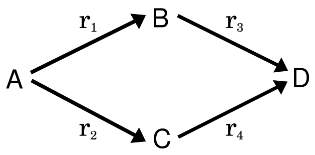
Iteration	Graphical representation	Stoichiometry Matrix	State
0		$\begin{array}{cc} & \mathit{Isos\_1.2} \\ \begin{matrix}A\\D\end{matrix} & \begin{pmatrix}-1\\1\end{pmatrix}\end{array}$	
1		$\begin{array}{cc} & \mathit{Isos\_1.2} \\ \begin{matrix}A\\D\end{matrix} & \begin{pmatrix}-1\\1\end{pmatrix}\end{array}$	$\text{saved}_3 : [\,]$
2		$\begin{array}{cc} & \begin{matrix}\mathit{Ess\_1.1} & \mathit{Ess\_2.1}\end{matrix} \\ \begin{matrix}A\\D\end{matrix} & \begin{pmatrix}-1 & -1\\1 & 1\end{pmatrix}\end{array}$	$\text{saved}_2 : [\text{Isos}_{1.2} : [\text{Ess}_{1.1}, \text{Ess}_{2.1}]\,]$
3		$\begin{array}{cc} & \begin{matrix}r_1 & r_2 & r_3 & r_4\end{matrix} \\ \begin{matrix}A\\B\\C\\D\end{matrix} & \begin{pmatrix}-1 & -1 & 0 & 0\\1 & 0 & -1 & 0\\0 & 1 & 0 & -1\\0 & 0 & 1 & 1\end{pmatrix}\end{array}$	$\text{saved}_1 : [\text{Ess}_{1.1} : [\text{r}_1, \text{r}_3],\ \text{Ess}_{2.1} : [\text{r}_2, \text{r}_4]\,]$

Figure 4.3: The process of the *Decompression* algorithm illustrated on a simple model, where the state of the algorithm at the end of each iteration is shown (the zeroth state corresponds to the state of **N** upon initialisation of the algorithm). The algorithm terminates once there is no more information left to retrieve from the compression algorithm.

Then, an iterative process reverses the modifications made by the compression algorithm, starting from the compression algorithm's most recent iteration. At each step, the columns of the stoichiometry matrix that correspond to enzyme subsets are removed and replaced by columns that represent the reactions within the subset (Step 4), whilst reactions representing iso-stoichiometric groups are duplicated and named according to the original members of the group (Step 5).

Information regarding the directionality and stoichiometry of the reactions to be inserted into the stoichiometry matrix is obtained by retrieving the information that was saved by the compression algorithm (Step 1).

The progress of this algorithm when applied to a simple model is shown in Figure 4.3.

4.3.3 The Vector Expansion Algorithm

Algorithm 4 *Expand_vector*($\mathbf{v}$), which when applied to a vector, $\mathbf{v}$, obtained from the analysis of a compressed model, expands it into the equivalent list of vectors, $\mathbf{V}$, of the model's larger original counterpart. Step 5 is further explained by Algorithm 5.

```
1: v_list = append_to_empty_list($\mathbf{v}$)
2: mappings = retrieve_mappings()
3: for [isos, ess, $\mathbf{N_{map}}$] in mappings do
4:     v_list = expand_enzyme_subsets(v_list, ess, $\mathbf{N_{map}}$)
5:     v_list = expand_v_list_isostoichiometric_groups(v_list, isos, $\mathbf{N_{map}}$)
6: end for
7: return $\mathbf{V}$
```

Flux vectors resulting from the analysis of a compressed model (for example, EMA) can be expanded into the equivalent vectors of the larger original model. This is achieved through the one-to-many transformation shown in Figure 4.4.

This transformation is accomplished by the *Expand_vector* algorithm (Algorithm 4). Similarly to the algorithm described in Section 4.3.2, the vector expansion algorithm decomposes enzyme subsets into their constituent reactions.

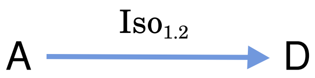
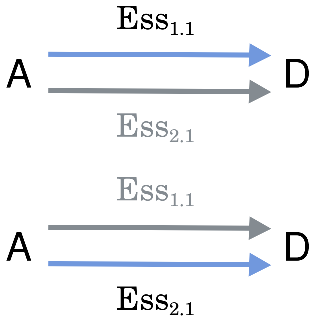
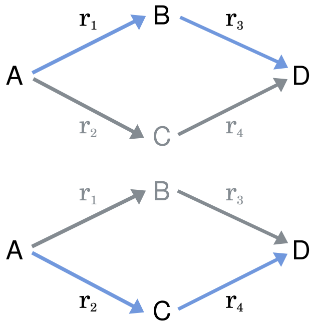
Iteration	Graphical representation	Flux Vectors	State
1		$\begin{array}{rc} & \mathbf{v} \\ \begin{array}{r}\text{A\_in}\\ \text{Iso\_1.2}\\ \text{B\_out}\end{array} & \begin{pmatrix}1\\1\\1\end{pmatrix}\end{array}$	saved$_3$: [ ]
2	 Ess$_{1.1}$ A D Ess$_{2.1}$ Ess$_{1.1}$ A D Ess$_{2.1}$	$\begin{array}{rcc} & \tilde{\mathbf{v}}_1 & \tilde{\mathbf{v}}_2 \\ \begin{array}{r}\text{A\_in}\\ \text{Ess\_1.1}\\ \text{Ess\_2.1}\\ \text{B\_out}\end{array} & \begin{pmatrix}1\\1\\0\\1\end{pmatrix}, & \begin{pmatrix}1\\0\\1\\1\end{pmatrix}\end{array}$	saved$_2$: [Isos$_{1.2}$: [Ess$_{1.1}$, Ess$_{2.1}$]]
3	 r$_1$ B r$_3$ A D r$_2$ C r$_4$ r$_1$ B r$_3$ A D r$_2$ C r$_4$	$\begin{array}{rcc} & \tilde{\mathbf{v}}_1 & \tilde{\mathbf{v}}_2 \\ \begin{array}{r}\text{A\_in}\\ \text{r\_1}\\ \text{r\_2}\\ \text{r\_3}\\ \text{r\_4}\\ \text{B\_out}\end{array} & \begin{pmatrix}1\\1\\0\\1\\0\\1\end{pmatrix}, & \begin{pmatrix}1\\0\\1\\0\\1\\1\end{pmatrix}\end{array}$	saved$_1$: [Ess$_{1.1}$: [r$_1$, r$_3$], Ess$_{2.1}$: [r$_2$, r$_4$]]

Figure 4.4: The process of the *Expand_vector* algorithm, illustrated on a simple vector obtained from the compressed model shown in Table 4.2 (where blue arrows indicate active reactions). The state of the algorithm at the end of each iteration is shown. The algorithm terminates once there is no more information left to retrieve from the compression algorithm.

However, reactions corresponding to iso-stoichiometric groups are treated as follows. Since each member of an iso-stoichiometric group corresponds to a group of alternative pathways from a starting to an end product, flux vectors that contain a representative reaction from such a group must be split into a set of flux vectors, where each vector retains one alternative pathway from the group (reactions that would violate directionality constraints are omitted). For example, consider the vector, **v**, containing the reaction labelled $Iso_{1.2}$ in Figure 4.4. This reaction represents an iso-stoichiometric group with two alternative pathways from metabolites A to D (one via r_1 and r_3, and another via r_2 and r_4). Therefore, **v** is expanded into two vectors, each corresponding to one of the two potential routes in the group.

Algorithm 5 *expand_v_list_isostoichiometric_groups*(*v_list*, *isos*), which when applied to a list of flux vectors, *v_list*, and a set of iso-stoichiometric groups, *isos*, expands the list of vectors with respect to the alternate pathways of the group.

```
for group in isos do
    for reaction in group do
        if directionality not violated by v then
            expanded_list = get_empty_list()
            for v in v_list do
                v′ = copy(v)
                replace_group_with_reaction(v′, group, reaction)
                append(expanded_list, v′)
            end for
            v_list ← expanded_list
        end if
    end for
end for
```

4.3.4 The Alternate Vectors Algorithm

Consider a vector, **v**, derived from the analysis of an uncompressed model. Algorithm 6 uses the knowledge of alternate pathways revealed during model compression to find alternative paths to **v**, resulting in a list of alternative vectors that have the same net stoichiometry as **v** but differ in some internal reactions (corresponding to members of the iso-stoichiometric groups discovered during model compression). For example, consider the vectors $\mathbf{v}_1$ and $\mathbf{v}_2$ shown in iteration 3 of Figure 4.4, that each show an alternate path from metabolites A to D. Applying the *Alternate_vector* algorithm to either one of these vectors, would result in a list that contains both vectors.

This algorithms proceeds as follows. First, the original flux vector is compressed using a process that reverses the steps described in Section 4.3.3. Then, the vector expansion algorithm is applied to the compressed vector to obtain the pathways alternate to it.

Algorithm 6 *Alternate_vector*($\mathbf{v}$), which when applied to a vector, $\mathbf{v}$, obtained from the analysis of an un-compressed model, expands it into a list of vectors, $\mathbf{V}$, that detail the pathways alternative to the reactions in $\mathbf{v}$, according to the iso-stoichiometric groups that were identified during model compression.

```
mappings = retrieve_mappings()
**v_compressed** = compress_vector($\mathbf{v}$, mappings)
**return** *Expand_vector*(**v_compressed**)
```

4.3.5 Implementation Details

The algorithms described within this chapter are implemented in Python and as an add-on to ScrumPy (Chapter 2.5.1). They form part of a class, called *Compression*, that was defined to store information regarding the transformations done to the model during compression and make this information accessible to the other associated algorithms. This is achieved through a stack attribute onto which information is added during each iteration.

The stack enables the transformations to be graphically displayed in the form of a tree where each reaction is shown as the parent of the reactions that it replaced, see Figure 4.5, where reactions replaced by enzyme subsets are shown via black branches, whilst iso-stoichiometric groups are coloured in red.

The *Compression* class initiates by taking a ScrumPy model instance as an input. Some of the functions associated with this class are:

- *Compress()*, an implementation of Algorithm 2,
- *GetTree()*, returns a tree object as described above,
- *OriginalSMX()*, an implementation of Algorithm 3,
- *CompressVector(*$\mathbf{V}$*)*, compresses a vector obtained from the analysis of an original model,

- *DecompressDataSet(***V***)*, an implementation of Algorithm 4.

4.3.5.1 The Compression Algorithm

This algorithm is implemented as described in Section 4.3.1.

Iso-stoichiometric groups are identified as proportional columns of the external stoichiometry matrix and stored as a dictionary that specifies the ratio of the removed reactions with respect to the reaction that is retained in the stoichiometry matrix.

Enzyme subsets are obtained through an existing function of ScrumPy that returns a dictionary detailing the ratio of flux that reactions must carry with respect to each other at steady-state. Their associated net stoichiometries are obtained through a ScrumPy function that multiplies these dictionaries with the external stoichiometry matrix (where each dictionary is transformed into a flux vector before this multiplication).

At each iteration, the dictionaries mentioned above, along with dictionaries specifying the directionality of all removed reactions and stoichiometry of the reactions replaced by enzyme subsets are appended to a list that is pushed to the stack. When compression is complete, the list at the top of the stack will contain the most recent modifications made to the model.

To ensure that all reactions are uniquely defined, a suffix, corresponding to the iteration number, is appended to the names of all reactions during each iteration.

The names of dead reactions and inconsistent iso-stoichiometric groups are noted in list attributes of the *Compression* class (*deadReacs_list* and *inconsistant_Isos_list* respectively).

4.3.5.2 The Decompression Algorithm

This algorithm proceeds by unpacking the stack, one list at a time, and undoing the transformations done by the compression algorithm at each iteration, as discussed in Section 4.3.2.

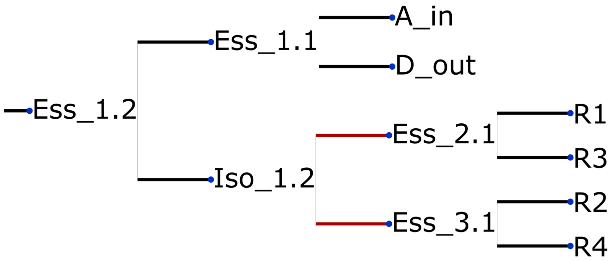


Figure 4.5: A tree showing the compression process on the simple model described by Figure 4.2, where black lines refer to enzyme subsets, red lines refer to iso-stoichiometric groups.

4.3.5.3 The Vector Expansion Algorithm

Similarly to the previous algorithm, the vector expansion algorithm relies on the stack attribute to retrieve information, which is then processed as discussed in Section 4.3.2.

This algorithm requires input in the form of a ScrumPy DataSet object, where the vectors to be expanded are listed as columns. Similarly, the output is a DataSet instance containing the expanded vectors.

4.3.5.4 The Alternate Vectors Algorithm

As in the algorithm described above, this algorithm requires input in the form of a ScrumPy DataSet object and returns a DataSet instance as an output.

4.3.5.5 Visualising The Compression Process as a Tree

The tree provides a graphical interface for exploring and visualising the redundancy characteristics exposed during compression. To illustrate this, the compression algorithm was applied to the simple toy model depicted in Figure 4.2 (with the addition of the external metabolites x_A and x_D and corresponding transporter reactions). The tree associated with its compression, shown in Figure 4.5, groups reactions that are present in the same enzyme subset or iso-stoichiometric group as children of the same node. The stoichiometry of each reaction can be visualised as an attribute of the corresponding node.

4.4 Mathematical Considerations

4.4.1 Combining Enzyme Subsets

Note that for the sake of simplicity, the enzyme subsets in this section are considered to be of size two.

Proposition 4.4.1.1. *Combining enzyme subsets into a single reaction preserves the steady state behaviour of the network*

Proof. Consider an $m \times r$ stoichiometry matrix $\mathbf{N}$, with columns $(r_1), (r_2), \ldots, (r_r)$.

As discussed in Section 4.2.1, enzyme subsets arise as one or more constraints of the form

$$v_i = \frac{1}{\lambda} v_j, \text{ for some } i, j \in \{1, 2, \ldots, r\} \tag{4.14}$$

where v_i and v_j denote the fluxes of reactions whose corresponding columns of $\mathbf{N}$ are (r_i) and (r_j), and λ is a constant. The number of such constraints pertaining to a given enzyme subset corresponds to one less than the number of reactions involved, where the reaction on the left hand side of Equation 4.14 (i.e. v_i) is invariant across all constraints, while the reaction and constant on the right hand side of Equation 4.14 (i.e. v_j and λ) vary according to the relations specified by the subset.

Without loss of generality consider the following constraint

$$v_1 = \frac{1}{\lambda} v_2, \tag{4.15}$$

corresponding to the enzyme subset Ess_3 in Section 4.2.1.

Let the elements of $\mathbf{N}$ be denoted by n_{ij} for metabolite i and reaction j, and let the rows of $\mathbf{N}$, denoted by $\mathrm{X}_1, \mathrm{X}_2, \ldots, \mathrm{X}_\mathrm{m}$, correspond to the constraints $\{\mathrm{X}_1\mathbf{v} = \mathbf{0}, \mathrm{X}_2\mathbf{v} = \mathbf{0}, \ldots, \mathrm{X}_\mathrm{m}\mathbf{v} = \mathbf{0}\}$ that generate the flux cone, C.

Since Ess_3 is an enzyme subset in $\mathbf{N}$, then $v_2 = \lambda v_1$ in all steady-state flux vectors of the system. Therefore the constraint $\mathrm{a}\mathbf{v} = \mathbf{0}$, where

$$\mathrm{a} = \begin{pmatrix} \lambda & -1 & 0 & \dots & 0 \end{pmatrix}, \tag{4.16}$$

must be embedded in the rows of $\mathbf{N}$, since:

$$\mathrm{a}\mathbf{v} = \begin{pmatrix} \lambda & -1 & 0 & \dots & 0 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_n \end{pmatrix} = \lambda v_1 - v_2 = 0 \tag{4.17}$$

Therefore, a can be added to the rows of $\mathbf{N}$ without altering the null-space (since $\mathrm{a}\mathbf{v} = \mathbf{0}$ is a redundant constraint).

Without loss of generality, let a be added as the first row, (a), such that

$$\mathbf{N} = \begin{array}{c} \\ \mathrm{a} \\ \mathrm{X}_1 \\ \mathrm{X}_2 \\ \mathrm{X}_3 \\ \vdots \\ \mathrm{X}_3 \end{array} \begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & \dots & r_r \end{array} \\ \begin{pmatrix} \lambda & -1 & 0 & \dots & 0 \\ n_{11} & n_{12} & n_{13} & \dots & n_{1r} \\ n_{21} & n_{22} & n_{23} & \dots & n_{2r} \\ n_{31} & n_{32} & n_{33} & \dots & n_{3r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ n_{m1} & n_{m2} & n_{m3} & \dots & n_{mr} \end{pmatrix} \end{array}, \tag{4.18}$$

Since row operations on $\mathbf{N}$ do not alter its null-space, the row (a) can be added to every other row of $\mathbf{N}$ such that the column corresponding to reaction r_2 is eliminated:

$$\mathbf{N} = \begin{array}{c} \\ \mathrm{a} \\ \hat{\mathrm{X}}_1 \\ \hat{\mathrm{X}}_2 \\ \hat{\mathrm{X}}_3 \\ \vdots \\ \hat{\mathrm{X}}_\mathrm{m} \end{array} \begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & \dots & r_r \end{array} \\ \begin{pmatrix} \lambda & -1 & 0 & \dots & 0 \\ (n_{11}+\lambda n_{12}) & (n_{12}-n_{12}) & n_{13} & \dots & n_{1r} \\ (n_{21}+\lambda n_{22}) & (n_{22}-n_{22}) & n_{23} & \dots & n_{2r} \\ (n_{31}+\lambda n_{32}) & (n_{32}-n_{32}) & n_{33} & \dots & n_{3r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (n_{m1}+\lambda n_{m2}) & (n_{m2}-n_{m2}) & n_{m3} & \dots & n_{mr} \end{pmatrix} \end{array},$$

$$= \begin{array}{c} \\ \mathrm{a} \\ \hat{\mathrm{X}}_1 \\ \hat{\mathrm{X}}_2 \\ \hat{\mathrm{X}}_3 \\ \vdots \\ \hat{\mathrm{X}}_\mathrm{m} \end{array} \begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & \dots & r_r \end{array} \\ \begin{pmatrix} \lambda & -1 & 0 & \dots & 0 \\ (n_{11}+\lambda n_{12}) & 0 & n_{13} & \dots & n_{1r} \\ (n_{21}+\lambda n_{22}) & 0 & n_{23} & \dots & n_{2r} \\ (n_{31}+\lambda n_{32}) & 0 & n_{33} & \dots & n_{3r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (n_{m1}+\lambda n_{m2}) & 0 & n_{m3} & \dots & n_{mr} \end{pmatrix} \end{array}, \tag{4.19}$$

where $(\hat{\mathrm{X}}_1) = ((\mathrm{X}_1) + n_{12}(\mathrm{a})), (\hat{\mathrm{X}}_2) = ((\mathrm{X}_2) + n_{22}(\mathrm{a})), \dots, (\hat{\mathrm{X}}_\mathrm{m}) = ((\mathrm{X}_\mathrm{m}) + n_{m2}(\mathrm{a}))$.

But as discussed below, a system $\mathbf{Nv} = \mathbf{0}$, where $\mathbf{N}$ is of the form shown in Equation (4.19) can be reduced in size by removing the column corresponding to r_2 in $\mathbf{N}$ and the corresponding flux, v_2, in $\mathbf{v}$, to result in the matrix $\tilde{\mathbf{N}}$ and corresponding flux vector $\tilde{\mathbf{v}}$, where

$$\tilde{\mathbf{N}} = \begin{array}{c} \\ \tilde{\mathrm{X}}_1 \\ \tilde{\mathrm{X}}_2 \\ \tilde{\mathrm{X}}_3 \\ \vdots \\ \tilde{\mathrm{X}}_\mathrm{m} \end{array} \begin{array}{c} \begin{array}{cccc} r_1 & r_3 & \dots & r_r \end{array} \\ \begin{pmatrix} (n_{11}+\lambda n_{12}) & n_{13} & \dots & n_{1r} \\ (n_{21}+\lambda n_{22}) & n_{23} & \dots & n_{2r} \\ (n_{31}+\lambda n_{32}) & n_{33} & \dots & n_{3r} \\ \vdots & \vdots & \ddots & \vdots \\ (n_{m1}+\lambda n_{m2}) & n_{m3} & \dots & n_{mr} \end{pmatrix} \end{array}. \tag{4.20}$$

Any vector, $\tilde{\mathbf{v}}$, obtained from the analysis of $\tilde{\mathbf{N}}\tilde{\mathbf{v}} = \mathbf{0}$ can be transformed into a corresponding vector of $\mathbf{Nv} = \mathbf{0}$ by adding an element v_2 to $\tilde{\mathbf{v}}$ such that $v_2 = \lambda v_1$.

□

Note that enzyme subsets of size greater than two can be similarly eliminated by sequentially removing each of the subsets' constituent constraints from **N** as described above.

The system of equalities generated by Equation (4.19) is equivalent to that of Equation (4.20) because of the following reasons.

Consider the flux cone,

$$C = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{N}\mathbf{v} = \mathbf{0}, \mathbf{v} \geq \mathbf{0}\}, \tag{4.21}$$

where **N** is of the form as in Equation (4.19), then, this system can be regarded as an intersection of two sets of constraints:

$$C = \{\mathbf{v} \in \mathbb{R}^r | \mathrm{a}\mathbf{v} = \mathbf{0} \cap \mathbf{N}_{\mathrm{C}}\mathbf{v} = \mathbf{0}, \mathbf{v} \geq \mathbf{0}\}, \tag{4.22}$$

where:

$$\mathrm{a} = \begin{array}{c} \begin{array}{cccccc} r_1 & r_2 & \dots & & r_r & \end{array} \\ \begin{pmatrix} \lambda & -1 & 0 & 0 & \dots & 0 \end{pmatrix} \end{array}, \tag{4.23}$$

and

$$\mathbf{N}_{\mathrm{C}} = \begin{array}{c} \\ \hat{\mathrm{X}}_1 \\ \hat{\mathrm{X}}_2 \\ \hat{\mathrm{X}}_3 \\ \vdots \\ \hat{\mathrm{X}}_{\mathrm{m}} \end{array} \begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & r_3 & \dots & r_r \end{array} \\ \begin{pmatrix} (n_{11}+\lambda n_{12}) & 0 & n_{13} & \dots & n_{1r} \\ (n_{21}+\lambda n_{22}) & 0 & n_{23} & \dots & n_{2r} \\ (n_{31}+\lambda n_{32}) & 0 & n_{33} & \dots & n_{3r} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ (n_{m1}+\lambda n_{m2}) & 0 & n_{m3} & \dots & n_{mr} \end{pmatrix} \end{array}, \tag{4.24}$$

Now, the system a$\mathbf{v} = 0$ corresponds to the constraint:

$$v_2 = \lambda v_1 \tag{4.25}$$

where v_1 and v_2 are the fluxes of the reactions corresponding to the columns r_1 and r_2 respectively.

When considering $\mathbf{N}_{\mathrm{C}}$, since the column r_2 is zero, then the corresponding dimension is unconstrained within the corresponding flux cone. Hence the column r_2 and flux v_2 can be eliminated from $\mathbf{N_c}\mathbf{v} = \mathbf{0}$, resulting in the matrix $\tilde{\mathbf{N}}$ and vector $\tilde{\mathbf{v}}$, without losing any information from the system.

Let this new system generate the flux cone:

$$\tilde{C} = \{\tilde{\mathbf{v}} \in \mathbb{R}^{(r-1)} | \tilde{\mathbf{N}}\tilde{\mathbf{v}} = \mathbf{0}, \tilde{\mathbf{v}} \geq \mathbf{0}\}, \tag{4.26}$$

where the constraint $v_2 = \lambda v_1$ can be taken into account by the transformation $\mathbf{T}$ that converts any flux vector $\tilde{\mathbf{v}}$ of $\tilde{C}$ into the corresponding vector $\mathbf{v}$ of C:

$$\mathbf{T}\tilde{\mathbf{v}} = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & & \\ \lambda & 0 & 0 & & & & \\ & & & \ddots & & & 0 \\ & & & & 0 & 1 & 0 \\ & & \dots & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_3 \\ \vdots \\ v_r \end{pmatrix} = \begin{pmatrix} v_1 \\ \lambda v_1 \\ v_3 \\ \vdots \\ v_r \end{pmatrix} = \begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_r \end{pmatrix} = \mathbf{v}, \tag{4.27}$$

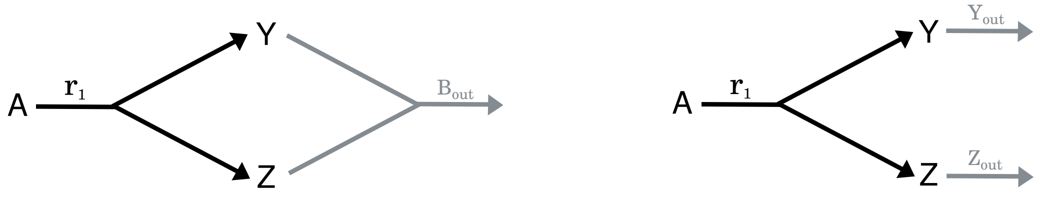
and therefore,

$$\ker(\mathbf{N}) = \mathbf{T}(\ker(\tilde{\mathbf{N}})), \tag{4.28}$$

meaning that there exists a one-to-one transformation between the null-spaces of $\mathbf{N}$ and $\tilde{\mathbf{N}}$, such that $\tilde{C}$ can be regarded as equivalent to C.

4.4.2 Combining Biomass Export

Consider the two simple networks shown below.



The model on the left exports the biomass precursors Y and Z via a single transporter reaction, while the model on the right contains individual transporters for each precursor. Therefore the stoichiometry matrix for each model is

$$\mathbf{N}_{\mathrm{C}} = \begin{array}{c} \\ \mathrm{A} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{cc} r_1 & B_{out} \end{array} \\ \begin{pmatrix} -1 & 0 \\ 1 & -1 \\ 1 & -\lambda \end{pmatrix} \end{array}. \qquad \mathbf{N} = \begin{array}{c} \\ \mathrm{A} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{ccc} r_1 & Y_{out} & Z_{out} \end{array} \\ \begin{pmatrix} -1 & 0 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix} \end{array},$$

where $\mathbf{N}_{\mathrm{C}}$ is the stoichiometry matrix for the model with combined biomass export and $\mathbf{N}$ is the equivalent matrix for the model with individual biomass transport.

When simulating growth using an objective function that minimises the total flux in the system, the LP for the model with combined biomass enforces the export of one unit of biomass via one export constraint. In contrast, the LP for the second model contains individual export constraints for each precursor, such that the FBA problem for each model is defined as

$$\begin{aligned} &\text{Find : argmin} \quad \textstyle\sum_{i=1}^{2} |v_{\mathrm{C}i}| \\ &\text{subject to} \quad \begin{cases} \mathbf{N}_{\mathrm{C}}\mathbf{v}_{\mathrm{C}} & = \mathbf{0}, \\ B_{out} & = 1. \end{cases} \end{aligned} \qquad \begin{aligned} &\text{Find : argmin} \quad \textstyle\sum_{i=1}^{3} |v_i| \\ &\text{subject to} \quad \begin{cases} \mathbf{N}\mathbf{v} & = \mathbf{0}, \\ Y_{out} & = 1, \\ Z_{out} & = \lambda. \end{cases} \end{aligned}$$

Therefore any steady-state flux vector of the model with combined biomass must be within the convex cone, C' defined as follows

$$C' = \{\mathbf{v}_{\mathrm{C}} \in \mathbb{R}^2 : \mathbf{N}_{\mathrm{C}}\mathbf{v}_{\mathrm{C}} = \mathbf{0}, \mathbf{v}_{\mathrm{C}} \geq \mathbf{0}\},$$

while the steady-state solution space, C, for the model with individual biomass is defined as

$$C = \{\mathbf{v} \in \mathbb{R}^3 : \mathbf{N}\mathbf{v} = \mathbf{0}, \mathbf{v} \geq \mathbf{0}\}.$$

Similarly, any FBA solution must be within the convex polyhedra, P' and P, respectively where

$$P' = \{\mathbf{v}_{\mathrm{C}} \in \mathbb{R}^2 : \mathbf{A}_{\mathrm{C}}\mathbf{v}_{\mathrm{C}} = \mathbf{b}_{\mathrm{C}}, \mathbf{v}_{\mathrm{C}} \geq \mathbf{0}\}, \tag{4.29}$$

and

$$P = \{\mathbf{v} \in \mathbb{R}^3 : \mathbf{A}\mathbf{v} = \mathbf{b}, \mathbf{v} \geq \mathbf{0}\}, \tag{4.30}$$

such that

$$\mathbf{A}_{\mathrm{C}} = \begin{array}{c} \\ B_{cons} \\ \mathrm{A} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{cc} r_1 & B_{out} \end{array} \\ \begin{pmatrix} 0 & 1 \\ -1 & 0 \\ 1 & -1 \\ 1 & -\lambda \end{pmatrix} \end{array}, \qquad \mathbf{A} = \begin{array}{c} \\ Y_{cons} \\ Z_{cons} \\ \mathrm{A} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{ccc} r_1 & Y_{out} & Z_{out} \end{array} \\ \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & 0 & 0 \\ 1 & -1 & 0 \\ 1 & 0 & -1 \end{pmatrix} \end{array},$$

and

$$\mathbf{b}_{\mathrm{C}} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \qquad \mathbf{b} = \begin{pmatrix} 1 \\ \lambda \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

where the rows $\mathrm{B}_{\mathrm{cons}}$, $\mathrm{Y}_{\mathrm{cons}}$ and $\mathrm{Z}_{\mathrm{cons}}$ correspond to the biomass export constraints.

The following propositions show that $C' \subseteq C$ and $P' \cong P$.

Proposition 4.4.2.1. *The flux cone, C', of a model with combined biomass is a subspace of the flux cone, C, of an equivalent model with individual biomass transporters.*

Proof. Consider the $m \times r$ stoichiometry matrix, $\mathbf{N}$, and let this denote the model with individual biomass export.

Without loss of generality, suppose that there are two biomass components Y and Z, which correspond to the transporter reactions Y_{out} and Z_{out} respectively. Similarly, let the stoichiometry matrix of the model with combined biomass be $\mathbf{N}_{\mathrm{C}}$, where B_{out} denotes the biomass reaction, such that:

$$\mathbf{N} = \begin{array}{c|cccccc|} & r_1 & r_2 & \cdots & r_{r-2} & Y_{out} & Z_{out} \\ \mathrm{X}_1 & n_{11} & n_{12} & \cdots & n_{1(r-2)} & 0 & 0 \\ \mathrm{X}_2 & n_{21} & n_{22} & \cdots & n_{2(r-2)} & 0 & 0 \\ \mathrm{X}_3 & n_{31} & n_{32} & \cdots & n_{3(r-2)} & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathrm{X}_{\mathrm{m}-2} & n_{(m-2)1} & n_{(m-2)2} & \cdots & n_{(m-2)(r-2)} & 0 & 0 \\ \mathrm{Y} & n_{(m-1)1} & n_{(m-1)2} & \cdots & n_{(m-1)(r-2)} & -1 & 0 \\ \mathrm{Z} & n_{m1} & n_{m2} & \cdots & n_{m(r-2)} & 0 & -1 \end{array}, \tag{4.31}$$

and

$$\mathbf{N}_{\mathrm{C}} = \begin{array}{c|ccccc|} & r_1 & r_2 & \cdots & r_{r-2} & B_{out} \\ \mathrm{X}_1 & n_{11} & n_{12} & \cdots & n_{1(r-2)} & 0 \\ \mathrm{X}_2 & n_{21} & n_{22} & \cdots & n_{2(r-2)} & 0 \\ \mathrm{X}_3 & n_{31} & n_{32} & \cdots & n_{3(r-2)} & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathrm{X}_{\mathrm{m}-2} & n_{(m-2)1} & n_{(m-2)2} & \cdots & n_{(m-2)(r-2)} & 0 \\ \mathrm{Y} & n_{(m-1)1} & n_{(m-1)2} & \cdots & n_{(m-1)(r-2)} & -1 \\ \mathrm{Z} & n_{m1} & n_{m2} & \cdots & n_{m(r-2)} & -\lambda \end{array}. \tag{4.32}$$

The flux cone, C, generated by $\mathbf{N}$ contains the flux vectors, $\mathbf{v}$, that satisfy $\mathbf{Nv} = \mathbf{0}$, where each row of $\mathbf{N}$ is a constraint, i.e.

$$C = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{Nv} = \mathbf{0}, \mathbf{v} \geq \mathbf{0}\}. \tag{4.33}$$

Let the constraint $Z_{out} = \lambda Y_{out}$, denoted by $\mathrm{a}\mathbf{v} = \mathbf{0}$, be added to the rows of $\mathbf{N}$ to create the matrix $\mathbf{A}$ and corresponding convex polyhedron P:

$$P = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{Av} = \mathbf{0}, \mathbf{v} \geq \mathbf{0}\}, \tag{4.34}$$

where

$$\mathbf{A} = \begin{array}{c} \\ \mathrm{a} \\ \mathrm{X_1} \\ \mathrm{X_2} \\ \mathrm{X_3} \\ \vdots \\ \mathrm{X_{m-2}} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{cccccc} r_1 & r_2 & \dots & r_{r-2} & Y_{out} & Z_{out} \end{array} \\ \begin{pmatrix} 0 & 0 & \dots & 0 & \lambda & -1 \\ n_{11} & n_{12} & \dots & n_{1(r-2)} & 0 & 0 \\ n_{21} & n_{22} & \dots & n_{2(r-2)} & 0 & 0 \\ n_{31} & n_{32} & \dots & n_{3(r-2)} & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ n_{(m-2)1} & n_{(m-2)2} & \dots & n_{(m-2)(r-2)} & 0 & 0 \\ n_{(m-1)1} & n_{(m-1)2} & \dots & n_{(m-1)(r-2)} & -1 & 0 \\ n_{m1} & n_{m2} & \dots & n_{m(r-2)} & 0 & -1 \end{pmatrix} \end{array}, \qquad (4.35)$$

and P is a subspace of the original solution-space C.

As shown in Proposition 4.4.1.1, the presence of such a constraint means that the columns corresponding to the reactions Y_{out} and Z_{out} can be combined to create an equivalent matrix, $\tilde{\mathbf{A}}$, such that, such that $\tilde{\mathbf{A}} = \mathbf{N}_\mathrm{C}$.

Therefore, the flux cone of $\mathbf{N}_\mathrm{C}$, C', is equivalent to P and therefore, is a subspace of the flux cone of $\mathbf{N}$, C, i.e. $C' \subseteq C$.

□

Note that in case where Y_{out} and Z_{out} happened to be in an enzyme subset at the ratio $Z_{out} = \lambda Y_{out}$ in $\mathbf{N}$, then the constraint generated by a would be redundant and therefore the spaces C and C' would be identical.

Proposition 4.4.2.2. *The flux polyhedron, P, of a model with combined biomass is equivalent to the flux polyhedron, P' of a corresponding model with individual biomass transporters, given that biomass transporter in P' are constrained at a ratio proportional to that specified in the combined biomass reaction of P.*

Proof. As above, consider the two stoichiometry matrices $\mathbf{N}$ and $\mathbf{N}_\mathrm{C}$, with corresponding optimal solution spaces denoted by P and P' respectively.

Suppose that biomass constraints are added such that $Y_{out} = 1$ and $Z_{out} = \lambda$ in P and $B_{out} = 1$ in P'.

Then P' is defined as:

$$P' = \{\mathbf{v}_{\mathrm{c}} \in \mathbb{R}^{(r-1)} | \mathbf{A}_{\mathrm{c}}\mathbf{v}_{\mathrm{c}} = \mathbf{b}_{\mathrm{c}}, \mathbf{v}_{\mathrm{c}} \geq \mathbf{0}\}, \tag{4.36}$$

where

$$\mathbf{A}_{\mathrm{c}} = \begin{array}{c} \\ B_{cons} \\ \mathrm{X}_1 \\ \mathrm{X}_2 \\ \mathrm{X}_3 \\ \vdots \\ \mathrm{X}_{\mathrm{m-2}} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & \dots & r_{r-2} & B_{out} \end{array} \\ \begin{pmatrix} 0 & 0 & \dots & 0 & 1 \\ n_{11} & n_{12} & \dots & n_{1(r-2)} & 0 \\ n_{21} & n_{22} & \dots & n_{2(r-2)} & 0 \\ n_{31} & n_{32} & \dots & n_{3(r-2)} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ n_{(m-2)1} & n_{(m-2)2} & \dots & n_{(m-2)(r-2)} & 0 \\ n_{(m-1)1} & n_{(m-1)2} & \dots & n_{(m-1)(r-2)} & -1 \\ n_{m1} & n_{m2} & \dots & n_{m(r-2)} & -\lambda \end{pmatrix} \end{array}, \tag{4.37}$$

and

$$\mathbf{b}_C = \begin{pmatrix} 1 \\ \mathbf{0} \end{pmatrix}, \tag{4.38}$$

and the row $\mathrm{B}_{\mathrm{cons}}$ corresponds to the constraint $B_{out} = 1$.

Similarly, P is defined as:

$$P = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{A}\mathbf{v} = \mathbf{b}, \mathbf{v} \geq \mathbf{0}\}, \tag{4.39}$$

where

$$\mathbf{A} = \begin{array}{c} \\ \mathrm{Y}_{\mathrm{cons}} \\ \mathrm{Z}_{\mathrm{cons}} \\ \mathrm{X}_1 \\ \mathrm{X}_2 \\ \mathrm{X}_3 \\ \vdots \\ \mathrm{X}_{\mathrm{m-2}} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{cccccc} r_1 & r_2 & \dots & r_{r-2} & Y_{out} & Z_{out} \end{array} \\ \begin{pmatrix} 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \\ n_{11} & n_{12} & \dots & n_{1(r-2)} & 0 & 0 \\ n_{21} & n_{22} & \dots & n_{2(r-2)} & 0 & 0 \\ n_{31} & n_{32} & \dots & n_{3(r-2)} & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ n_{(m-2)1} & n_{(m-2)2} & \dots & n_{(m-2)(r-2)} & 0 & 0 \\ n_{(m-1)1} & n_{(m-1)2} & \dots & n_{(m-1)(r-2)} & -1 & 0 \\ n_{m1} & n_{m2} & \dots & n_{m(r-2)} & 0 & -1 \end{pmatrix} \end{array}, \tag{4.40}$$

and

$$\mathbf{b} = \begin{pmatrix} 1 \\ \lambda \\ \mathbf{0} \end{pmatrix}, \tag{4.41}$$

and the rows $\mathrm{Y_{cons}}$ and $\mathrm{Z_{cons}}$ correspond to the constraints $Y_{out} = 1$ and $Z_{out} = \lambda$ respectively.

But since $Y_{out} = 1$ and $Z_{out} = \lambda$ then $Z_{out} = \lambda Y_{out}$ must be a redundant constraint in P that can be added as a row to $\mathbf{A}$ without altering the feasible solution space. Let this constraint be denoted by a and be appended as the first row of the system:

$$\mathbf{A} = \begin{array}{c} \\ \mathrm{a} \\ \mathrm{Y_{cons}} \\ \mathrm{Z_{cons}} \\ \mathrm{X_1} \\ \mathrm{X_2} \\ \mathrm{X_3} \\ \vdots \\ \mathrm{X_{m-2}} \\ \mathrm{Y} \\ \mathrm{Z} \end{array} \begin{array}{c} \begin{array}{cccccc} r_1 & r_2 & \dots & r_{r-2} & Y_{out} & Z_{out} \end{array} \\ \begin{pmatrix} 0 & 0 & \dots & 0 & \lambda & -1 \\ 0 & 0 & \dots & 0 & 1 & 0 \\ 0 & 0 & \dots & 0 & 0 & 1 \\ n_{11} & n_{12} & \dots & n_{1(r-2)} & 0 & 0 \\ n_{21} & n_{22} & \dots & n_{2(r-2)} & 0 & 0 \\ n_{31} & n_{32} & \dots & n_{3(r-2)} & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ n_{(m-2)1} & n_{(m-2)2} & \dots & n_{(m-2)(r-2)} & 0 & 0 \\ n_{(m-1)1} & n_{(m-1)2} & \dots & n_{(m-1)(r-2)} & -1 & 0 \\ n_{m1} & n_{m2} & \dots & n_{m(r-2)} & 0 & -1 \end{pmatrix} \end{array}, \tag{4.42}$$

and

$$\mathbf{b} = \begin{pmatrix} 0 \\ 1 \\ \lambda \\ \mathbf{0} \end{pmatrix}. \tag{4.43}$$

As shown in Proposition 4.4.1.1, $\mathbf{N}_\mathrm{C}$ can be transformed through row operations that eliminate the column $\mathrm{Z_{out}}$, leading to an equivalent matrix, $\tilde{\mathbf{A}}$:

$$\tilde{\mathbf{A}} = \begin{array}{c} \\ \tilde{\mathrm{Y}}_{\mathrm{cons}} \\ \tilde{\mathrm{Z}}_{\mathrm{cons}} \\ \tilde{\mathrm{X}}_1 \\ \tilde{\mathrm{X}}_2 \\ \tilde{\mathrm{X}}_3 \\ \vdots \\ \tilde{\mathrm{X}}_{\mathrm{m-2}} \\ \tilde{\mathrm{Y}} \\ \tilde{\mathrm{Z}} \end{array} \begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & \dots & r_{r-2} & Y_{out} \end{array} \\ \begin{pmatrix} 0 & 0 & \dots & 0 & 1 \\ 0 & 0 & \dots & 0 & \lambda \\ n_{11} & n_{12} & \dots & n_{1(r-2)} & 0 \\ n_{21} & n_{22} & \dots & n_{2(r-2)} & 0 \\ n_{31} & n_{32} & \dots & n_{3(r-2)} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ n_{(m-2)1} & n_{(m-2)2} & \dots & n_{(m-2)(r-2)} & 0 \\ n_{(m-1)1} & n_{(m-1)2} & \dots & n_{(m-1)(r-2)} & -1 \\ n_{m1} & n_{m2} & \dots & n_{m(r-2)} & -\lambda \end{pmatrix} \end{array}, \tag{4.44}$$

and corresponding vector $\tilde{\mathbf{b}}$:

$$\tilde{\mathbf{b}} = \begin{pmatrix} 1 \\ \lambda \\ \mathbf{0} \end{pmatrix}. \tag{4.45}$$

Moreover, since the constraints generated by Y_{cons} and Z_{cons} are equivalent ($Y_{out} = 1$ and $\lambda Y_{out} = \lambda$), then one of them is redundant and can be removed. Without loss of generality, let the constraint $\mathrm{Z_{cons}}$ be removed and let the column Y_{out} be renamed to B_{out} and constraint $\mathrm{Y_{cons}}$ be renamed to $\mathrm{B_{cons}}$, then

$$\tilde{\mathbf{A}} = \begin{array}{c} \\ \tilde{\mathrm{B}}_{\mathrm{cons}} \\ \tilde{\mathrm{X}}_1 \\ \tilde{\mathrm{X}}_2 \\ \tilde{\mathrm{X}}_3 \\ \vdots \\ \tilde{\mathrm{X}}_{\mathrm{m-2}} \\ \tilde{\mathrm{Y}} \\ \tilde{\mathrm{Z}} \end{array} \begin{array}{c} \begin{array}{ccccc} r_1 & r_2 & \dots & r_{r-2} & B_{out} \end{array} \\ \begin{pmatrix} 0 & 0 & \dots & 0 & 1 \\ n_{11} & n_{12} & \dots & n_{1(r-2)} & 0 \\ n_{21} & n_{22} & \dots & n_{2(r-2)} & 0 \\ n_{31} & n_{32} & \dots & n_{3(r-2)} & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ n_{(m-2)1} & n_{(m-2)2} & \dots & n_{(m-2)(r-2)} & 0 \\ n_{(m-1)1} & n_{(m-1)2} & \dots & n_{(m-1)(r-2)} & -1 \\ n_{m1} & n_{m2} & \dots & n_{m(r-2)} & -\lambda \end{pmatrix} \end{array}, \tag{4.46}$$

and

$$\tilde{\mathbf{b}} = \begin{pmatrix} 1 \\ \mathbf{0} \end{pmatrix}, \tag{4.47}$$

which is equivalent to the optimal solution space P' (since $\tilde{\mathbf{A}} = \mathbf{A}_{\mathrm{c}}$ and $\tilde{\mathbf{b}} = \mathbf{b}_{\mathrm{c}}$).

Therefore $P' \cong P$ and hence the set of FBA solutions of both models are identical.

□

Models with biomass components of size greater than two can be similarly considered by sequentially adding further redundant constraints to $\mathbf{N}$ as above.

4.5 Results

In this section, both the correctness of the algorithms presented here and their ability to derive biological insights were investigated by applying them to the models listed in Table 2.1 (except for the simplified plant model). Two types of models were analysed: small hand-built models and large GSMs.

The small models are related to the Calvin cycle and photorespiration. The GSMs describe the metabolism of four bacterial species: *C. jejuni*, *C. necator*, *G. thermoglucosidasius*, and *E. coli*. A description of these models is in Section 2.5.2.

In this work, the small models were used to verify the correctness of the compression algorithm, whilst the GSMs were used to demonstrate that the algorithms can identify specific features of large networks, including erroneous internal cycles, duplicate processes arising from database artefacts, and genuine alternative routes from the same start to end products.

4.5.1 Testing of the Algorithms

4.5.1.1 Correctness of the Compression/Decompression Cycle

An essential property of any compression algorithm is that it accurately preserves the original data, ensuring that, in this case, decompressed models and flux vectors are identical to the original.

To verify the correctness of compression and decompression algorithms, the models in Table 2.1 were compressed and then decompressed. The stoichiometry matrix obtained after

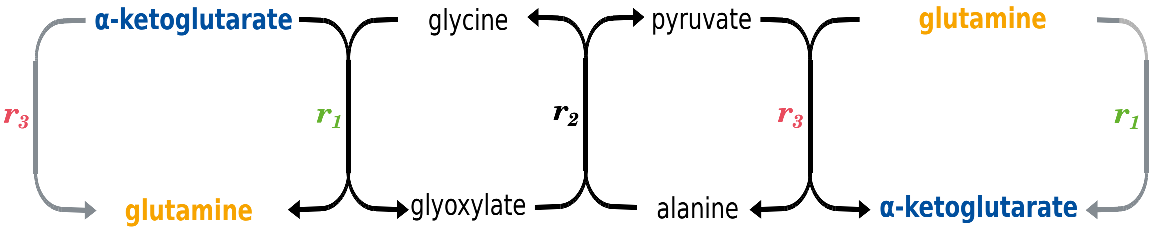


Figure 4.6: The EM of the photorespiration model that is an internal cycle (where r_1: glycine transaminase, r_2: alanine:glyoxilate transaminase, r_3: glutamine:pyruvate transaminase). Identical metabolites/reactions are given the same colour to facilitate identification.

the decompression was confirmed to be identical to the original (i.e. $Decompress(Compress(\mathbf{N})) = \mathbf{N}$)).

The algorithms described within this chapter have the potential to aid EMA as their ability to decrease the number of parameters within a given model is expected to reduce the computational burden associated with this technique. To this end, a model is first compressed using Algorithm 2 and the EMs of the compressed model are calculated. Then, the resultant EMs of the compressed model are converted into the equivalent set of EMs of the original model using Algorithm 4.

To test the output of the vector decompression algorithm, the EMs of the original and compressed models of the Calvin cycle and photorespiration were obtained. The vector decompression algorithm was applied to the EMs of the compressed models as described above and the output was confirmed to be identical to the EMs obtained from the original models.

The above analysis was repeated using models where all constituent reactions had been modified to be reversible. When comparing the EMs obtained for the reversible photorespiration model, one EM identified from the original stoichiometry matrix was not present in the set of EMs obtained from the compressed model. Upon further inspection, it emerged that the mode consisted of an internal cycle and so cannot carry flux at steady-state (Figure 4.6).

Table 4.1: Time to compute EMs in the compressed and uncompressed small models listed in Table 2.1.

	Computing Time		
Model	**Original Model**	**Compressed Model**	**Improvement**
Calvin cycle	0.6s	0.5s	11 %
Photorespiration	123.7s	28.5s	77 %

4.5.1.2 Performance

To determine the extent to which the compression algorithm reduces the size of models, the size of the models listed in Table 2.1 before and after compression were compared. The reductions obtained by only (i) removing dead reactions and (ii) combining enzyme subsets only once (equivalent to one incomplete iteration of the compression algorithm) was also considered.

The results in Table 4.2 showed that the algorithm presented in this chapter consistently achieves a better compression ratio than when combining enzyme subsets only once.

To determine if model compression reduces the time required for EM enumeration, the time taken to calculate the complete set of EMs for the Calvin cycle and photorespiration model, both when using the original and compressed models (including the decompression of the output as discussed in Section 4.5.1.1), was measured. The results in Table 4.1 demonstrate that the compression algorithm provided a slight advantage when calculating the EMs of small models. In addition, attempts to apply EMA to larger models indicated that the network reduction achieved by the compression algorithm is insufficient to make EMA applicable to models on which EMA was previously inapplicable.

The EMA algorithm used is implemented as part of ScrumPy and incompletely compresses models (as described above) before carrying out enumeration. Therefore in effect, the results described in Table 4.1 compare the time taken to carry out EMA of an incompletely compressed *versus* completely compressed model.

Table 4.2: Reduction in the size of the models listed in Table 2.1 after removing dead reactions, one incomplete iteration of the compression algorithm, and complete compression, as well as the iterations needed to achieve complete compression, where Sub-table (a) presents the number of reactions/metabolites in the given models while Sub-table (b) gives the corresponding percentage reduction.

	Original Model		**Removed**		**Combined**		**Complete**		**No. Of**
			Dead Reacs		**Enzyme Subsets**		**Compression**		**Iterations**
Model	**Reacs**	**Mets**	**Reacs**	**Mets**	**Reacs**	**Mets**	**Reacs**	**Mets**	
Calvin Cycle	21	28	21	28	11	18	11	18	2
Photorespiration	90	113	74	100	22	43	20	39	3
C. jejuni	1150	1105	680	561	442	336	421	328	3
C. necator	1358	1454	865	752	431	340	406	330	3
G. thermoglucosidasius	1125	1198	591	453	401	268	386	261	4
E. coli	1659	1714	708	501	497	293	459	281	3

(a)

	Removed		**Combined**		**Complete**	
	Dead Reacs		**Enzyme Subsets**		**Compression**	
Model	**Reacs**	**Mets**	**Reacs**	**Mets**	**Reacs**	**Mets**
Calvin Cycle	0%	0%	48%	36%	48%	36%
Photorespiration	18%	12%	76%	62%	78%	65%
C. jejuni	41%	49%	62%	70%	63%	70%
C. necator	36%	48%	68%	77%	70%	77%
G. thermoglucosidasius	47%	62%	64%	78%	66%	78%
E. coli	57%	71%	70%	83%	72%	84%

(b)

4.5.2 Analysis of Compressed Models

This subsection shows results obtained from the analysis of the models listed in Table 2.1. First, a collective summary of the results is presented, followed by a description of the results for each individual model.

For each model, the algorithms presented throughout this chapter were used to extract iso-stoichiometric groups. Then, the implications of these groups on the results of model analysis, with an emphasise on LP solutions, was investigated.

Recall that, as discussed in Section 4.1, each iso-stoichiometric group represents a set of alternate pathways from the same start to end products, which may exist in models because of:

1. Database artefacts: when models are automatically constructed from databases, some processes or reactions may be included more then once due to the presence of duplicate entries in the database. This occurs, for example, when multi-step reactions are incorporated as both the overall conversion and the individual intermediate steps.

2. Genuine redundancies: reflecting an organism's ability to achieve the same net-conversion via different biochemical routes. These alternate pathways can help organisms fine-tune their metabolism in response to varying environmental conditions, since, for example, redundant enzymes may have different catalytic and regulatory properties.

4.5.2.1 Summary of Results

Following compression, every reaction in the reduced stoichiometry matrix either (i) is equivalent to one reaction from the original model, or (ii) has a net stoichiometry that incorporates a collection of reactions from the original model.

The number of such reactions in the compressed GSMs listed in Table 2.1 are described in Table 4.3.

The total number of enzyme subsets and iso-stoichiometric groups identified throughout

Table 4.3: The number of reactions in the reduced stoichiometry matrix of the models listed in Table 2.1. The *combined reactions* have the net stoichiometry of a combination of reactions from the original model. The number of reactions from the original model that each combined reaction represents was investigated by calculating the median (middle value) number of these constituent original reactions for the set of combined reactions in each each model.

Model	**Retained Original**	**Combined Reactions**	
	Reactions	**Number**	**Median Size**
C. jejuni	342	79	3
C. necator	294	112	2
G. thermoglucosidasius	300	86	2
E. coli	369	90	2

Table 4.4: The number of enzyme subsets and iso-stoichiometric groups identified throughout the course of the compression algorithm, when applied to the GSMs listed in Table 2.1, where the median number of reactions in the subsets/groups is listed.

Model	**Enzyme Subsets**		**Iso-stoichiometric Groups**		
	Number	**Median**	**Number**	**Median**	**Inconsistent**
C. jejuni	92	2	13	2	3
C. necator	119	2	17	2	1
G. thermoglucosidasius	97	2	8	2	3
E. coli	115	2	19	2	1

the compression algorithm and the median number of reactions that each subset/group contained (where the reactions correspond to those of the original stoichiometry matrix during the first iteration, and a partially reduced stoichiometry matrix during subsequent iterations) are listed in Table 4.4. Note that the number of enzyme subsets and iso-stoichiometric groups identified may be more than the number of combined reactions in the completely compressed model since subsets/groups identified during one iteration may be combined to form other subsets/groups in subsequent iterations.

4.5.2.2 *C. jejuni*

The *C. jejuni* model contained four iso-stoichiometric groups at the first iteration. They all related to duplicate reactions of the ETC (such as the reduction of cytochrome and menaquinone).

The remaining eight iso-stoichiometric groups were all represented by enzyme subsets that had identical stoichiometry with one reaction of the original model. Most of these subsets involved database artefacts, involving the erroneous inclusion of multi-step reactions, the

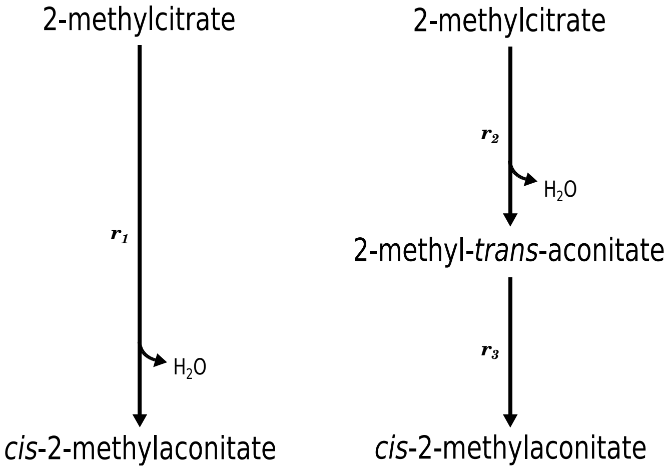


Figure 4.7: The conversion of 2-methylcitrate to cis-2-methylaconitate as part of the methylcitrate cycle. This reaction can occur directly via the enzyme 2-methylcitrate dehydratase (r_1: EC 4.2.1.79). Some organisms do not have this enzyme, and instead carry out the transformation via two reactions (r_2: 2-methylcitrate dehydratase, EC 4.2.1.117; r_3: aconitate isomerase, EC 5.3.3.7) [Brämer and Steinbüchel, 2001]. The two alternatives are distinguished as part of the methylcitrate cycle I and II respectively.

hydrolysis of ATP, or NAD/NADP-dependent transhydrogenase reactions (see Section 4.6.1.3).

Other iso-stoichiometric groups revealed genuine redundancies. For example, the methylcitrate cycle in bacteria is important for the detoxification of propionyl-CoA, a toxic bi-product of amino acid catabolism and odd-chain fatty acid oxidation [Brock et al., 2002]. As part of this cycle, the conversion of 2-methylcitrate to cis-2-methylaconitate can occur directly via the enzyme 2-methylcitrate dehydratase, or by first converting 2-methyl-citrate to 2-methyl-trans-aconitate which is subsequently converted into cis-2-methylaconitate [Brämer and Steinbüchel, 2001]. The model of *C. jejuni* was found to contain both of these pathways, as illustrated by Figure 4.7.

4.5.2.3 *C. necator*

The model of *C. necator* contained six iso-stoichiometric groups at the first iteration. These included duplicates of glucose-6-phosphate isomerase and glycerol-3-phosphate dehydrogenase, the latter group being inconsistent.

Similarly to *C. jejuni*, a large number of groups regarded database artefacts arising from multi-step reactions. This model contained genuine alternative routes for the detoxification of methylglyoxal, a highly reactive compound that can be produced by the

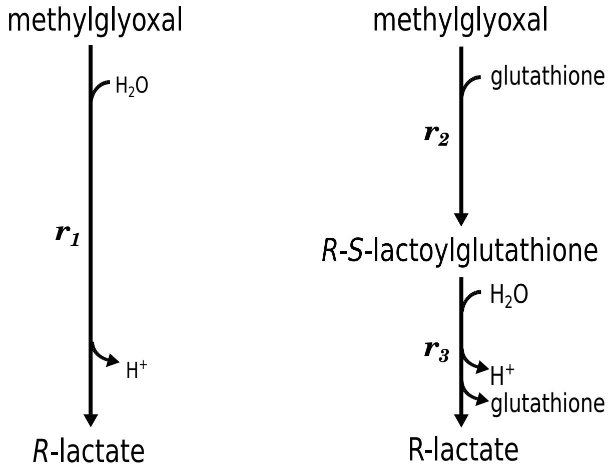


Figure 4.8: Alternative pathways for converting methylglyoxal to R-lactate in *C. necator* (where r_1: D-lactate dehydratase, EC 4.2.1.130; r_2: lactoylglutathione lyase, EC 4.4.1.5 ; r_3: hydroxyacylglutathione hydrolase, EC 3.1.2.6.) .

spontaneous dephosphorylation of triose-phosphates [Kumar et al., 2021]. Methylglyoxal is converted to R-lactate through a two-step glutathione-dependent route (lactoylglutathione lyase and hydroxyacylglutathione hydrolase), or alternatively, a shorter, single, glutathione-independent step (D-lactate dehydratase), as illustrated by Figure 4.8, [Hasim et al., 2014; Zhao et al., 2014].

4.5.2.4 *G. thermoglucosidasius*

The *G. thermoglucosidasius* model contained no iso-stoichiometric groups at the first instance, whilst all of the iso-stoichiometric groups identified throughout the progression of Algorithm 2 regarded database artefacts. Specifically, one iso-stoichiometric group involved NAD/NADP-dependent transhydrogenase reactions, and the others involved multi-step reactions.

4.5.2.5 *E. coli*

The *E. coli* model contained one iso-stoichiometric group at the first iteration, namely, a duplicate of succinate dehydrogenase.

This model contained a comparatively large number of enzyme subsets, 33, that can only act to hydrolyse ATP. Genuine alternative routes involved the conversion of methylglyoxal, as described in *C. necator*, and the conversion of β-D-fructose to fructose-6-phosphate, as shown in Figure 4.9.

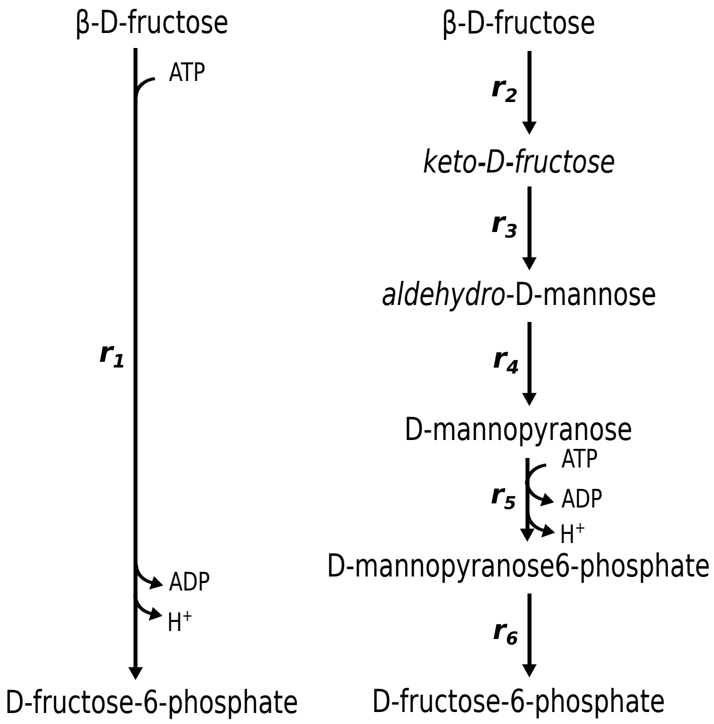


Figure 4.9: The conversion of β-D-fructose to fructose-6-photphate via fructokinase (where r_1: fructokinase, EC 2.7.1.4; r_2: spontaneous; r_3: mannose isomerase, EC 5.3.1.7 ; r_4: mannose isomerase, EC 5.3.1.7; r_5: spontaneous ; r_6: mannokinase, EC 2.7.1.7; r_7: mannose-6-phosphate isomerase, EC 5.3.1.8).

4.5.3 Analysis of Alternate Flux Vectors

FBA solutions of the *C. jejuni* and *G. thermoglycosidasius* were calculated to give insight into the impact of the redundancies identified within this chapter on metabolic modelling results. For example, to investigate whether the *Alternate_vector* algorithm can identify multiple optima when given a single FBA solution.

4.5.3.1 *C. jejuni*

The *C. jejuni* model was investigated using a LP where the production of *C. jejuni*'s 51 biomass precursors was set as a constraint (as detailed in Section 3.4.2.1). The *Alternate_vectors* algorithm was used to expand the LP solution into equivalent vectors that achieve the same net stoichiometry through different internal pathways (as identified by the model's iso-stoichiometric groups). Two equivalent vectors were therefore obtained. The difference between them regarded the conversion of acetyl-CoA to malonyl-CoA, as shown in Figure 4.10.

To investigate the impact of iso-stoichiometric groups on enzyme knock-outs metabolic

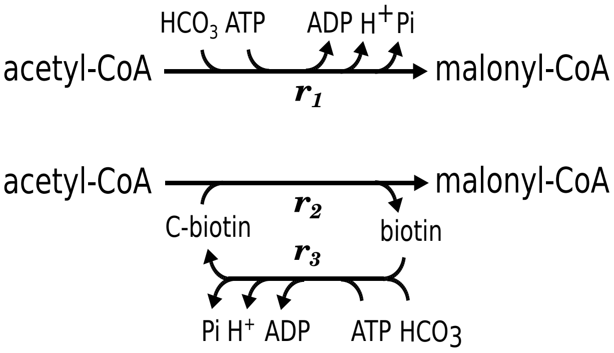


Figure 4.10: Converting acetyl-CoA into malonyl-CoA in *C. jejuni* (where r_1: acetyl-CoA carboxyl transferase (bicarbonate), EC 2.1.3.15; r_2: acetyl-CoA carboxyl transferase (biotin), EC 2.1.3.15; r_3: biotin carboxylase, EC 6.3.4.14).

modelling results, the impact of knocking out individual reactions on *C. jejuni*'s ability to generate biomass (i.e. reaction essentially analysis as introduced in Section 2.4) was studied using both the compressed and original model. This was achieved by repeatedly solving the following LP:

$$\text{Find : argmin} \quad \textstyle\sum_{i=1}^{r} |v_i|$$
$$\text{subject to} \quad \begin{cases} \mathbf{Nv} &= \mathbf{0}\,, \\ v_b &= t_b, \text{ for all } b \in \{1, 2, \dots, B\}, \\ v_j &= 0 \text{ for one } j \in \{1, 2, \dots, R\}. \end{cases} \tag{4.48}$$

where the constraints v_b ensure that the LP solution simultaneously produces all of *C. jejuni*'s B required biomass precursors, while the constraint v_j prevents one of the R internal reactions from carrying flux at each iteration. Failure to obtain an optimal solution indicates that the internal reaction constrained to zero is essential for growth.

Such an analysis of the compressed model revealed that the iso-stoichiometric group described above (conversion acetyl-CoA into malonyl-CoA) is essential for the growth of *C. jejuni*. However, as expected, the individual reactions of this group were not deemed to be essential when Equation (4.48) was solved for the original model. This result indicates that while knocking out one alternate pathway from the group does not prevent *C. jejuni* from generating biomass, the simultaneous knock-out of both alternate pathways is lethal.

4.5.3.2 *G. thermoglycosidasius*

Similarly, an LP algorithm was defined on the model of *G. thermoglycosidasius* with the generation of biomass precursors and fermentation products as constraints and the minimisation of total flux as the objective (as described in [Ahmad et al., 2017]). The LP solution included reactions from five iso-stoichiometric groups (all but one of the groups identified in this model). Every group contained two reactions; therefore, the LP solution was expanded into 32 alternative solutions with the same net stoichiometry (since no reversibility constraints were violated in the iso-stoichiometric groups).

4.5.4 Similarity Between The Compressed GSMs

After compression, the reduced stoichiometry matrices of all GSMs were compared to gain knowledge about the metabolic processes that are common amongst them. Twenty-four reactions with the net stoichiometry of (i) one or (ii) a combination of reaction/s in the original models were found to be shared amongst all compressed GSMs.

Out of the 24 common reactions, 19 were equivalent to reactions retained from the original GSMs, whilst five corresponded to a combination of the GSM's original reactions in at least one case as listed in Table 4.5.

4.6 Discussion

4.6.1 Model Curation

Repeatedly reducing the size of the stoichiometry matrix as described above can reveal hidden systematic errors that often materialise when models are automatically built from databases [Poolman et al., 2006], including the inclusion of duplicate processes and thermodynamically infeasible transformations. The identification and subsequent removal of such errors is important since their presence can compromise the accuracy of model analysis.

Table 4.5: Some of the reactions, r_i, that are common amongst the reduced stoichiometry matrix of all GSMs listed in Table 2.1, where only common reactions that have a combined stoichiometry in at least one of the GSMs are listed. The number of original reactions that each r_i represents in each model is shown.

Model	$\mathbf{r_1}$	$\mathbf{r_2}$	$\mathbf{r_3}$	$\mathbf{r_4}$	$\mathbf{r_5}$
C. jejuni	1	2	3	1	1
C. necator	4	2	6	1	2
G. thermoglucosidasius	1	2	1	1	1
E. coli	1	2	33	2	1

Table 4.6: The net stoichiometry of the reactions listed in Table 4.5.

Identifier	**Net stoichiometry**
r_1	threonine → 2-oxobutanoate + NH_4^+
r_2	L-aspartate + GTP + IMP → fumerate + Pi + AMP + GDP + H^+
r_3	ATP + H_2O → ADP + H^+
r_4	D-threo-isocitrate + NADP → ketoglutarate + NADPH + CO_2
r_5	oxaloacetate + ATP ↔ phospho-enol-pyruvate + ADP + CO_2

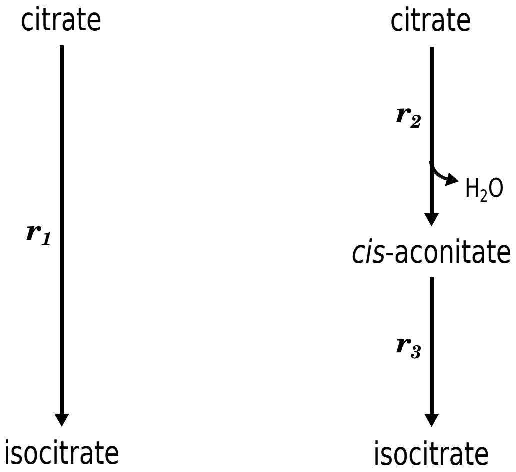


Figure 4.11: The enzyme aconitase converts citrate to isocitrate through the intermediate cis-aconitate (where r_1: aconitase, EC 4.2.1.3; r_2 & r_3: sub-reactions of aconitase). The three metabolites are kept at equilibrium (91% citrate, 3% *cis*-aconitate, and 6% iso-citrate [Siebert, 1965]).

4.6.1.1 Duplicate Processes

All but one of the GSMs investigated included duplicate reactions. Further inspection revealed that these reactions correspond to identical genes and should not have been included as duplicates in the models.

Other duplicate processes involved one reaction of the model having an identical stoichiometry with one or more enzyme subsets. The majority of these involved multi-step reactions, which are catalysed through the formation of one or more intermediate molecules that are then converted into the final product. In some cases, both the net reaction and the individual sub-reactions were found in the GSM. For example, the enzyme aconitase (present in the TCA cycle) converts citrate to isocitrate in two steps: first transforming citrate to cis-aconitate, and later cis-aconitate to isocitrate (Figure 4.11). The *C. jejuni* and *E. coli* GSMs contained both the net-reaction of this process and the two reactions which carry out the individual sub-steps. Similar instances regarded amino acid biosynthesis. For example, isopropylmalate dehydrogenase and isopropylmalate dehydratase (part of the L-leucine biosynthesis pathway, see Figure 4.12) were found similarly duplicated in all of the GSMs listed in Table 2.1.

Other duplications involved the erroneous inclusion of enzyme components as metabolites. An example is the involvement of biotin in the enzyme acetyl-coA carboxylase (an essential part of fatty acid synthesis in bacteria) in the *C. jejuni* GSM. This enzyme catalyses

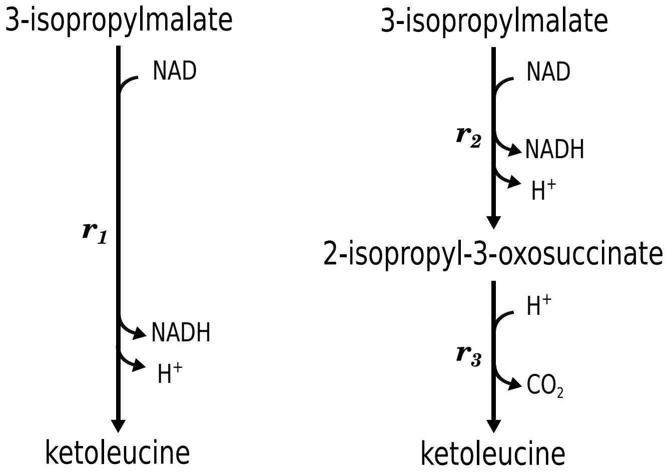


Figure 4.12: The conversion of 3-isopropylmalate to ketoleucine (as part of luecine biosynthesis) occurs through the enzyme isopropylmalate dehydrogenase that catalyses the oxidisation of 3-isopropylmalate to create 2-isopropyl-3-oxosuccinate (r_1/r_2: EC 1.1.1.85). This intermediate product is unstable and spontaneously degrades into ketoleucine.

the carboxylation of acetyl-coA into manonyl-coA in a process that first involves the carboxylation of biotin into C-biotin, and then the transfer of the carboxyl group in C-biotin to acetyl-coA (Figure 4.10). Both the overall stoichiometry of this process and the individual steps were included; however, the latter representation is incorrect as biotin is not an independent metabolite of the system (due to it forming part of the structure of the enzyme).

The novel methods described in this chapter can be used to identify such occurrences, which can then be manually removed from the model (care must be taken to preserve the duplicate reactions that occur when organisms contain different enzymes that catalyse the same net conversion). This is important since such occurrences cause systematic errors, meaning that although the definition of the sub and overall reactions may be biologically correct, including both (as duplicates) in a model might give rise to incorrect behaviour.

For example, their presence makes the identification of essential reactions more difficult. The inclusion of an incorrect duplicate reaction for acetyl-coA carboxylase in *C. jejuni* (Figure 4.10), led to this reaction to be not identified as essential in FBA, a result which was rectified when the duplicates were removed by the compression algorithm (Section 4.5.3). Similarly, duplicate reactions lead to unnecessary multiple optima in LP, such as the 32 alternative pathways identified in the analysis of *G. thermoglycosidasius* (Section 4.5.3). However, it is important to note that an LP objective that minimises flux is likely

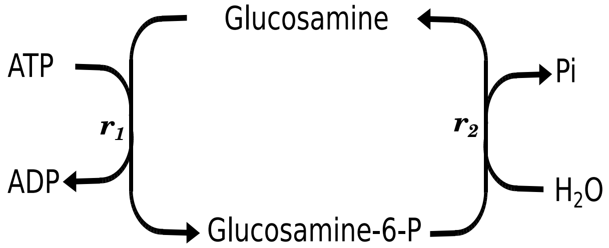


Figure 4.13: An enzyme subset that can only act to hydrolyse ATP. First, glucosamine is converted to glucosamine-6-phosphate in a reaction that uses ATP (where r_1: glucosamine kinase EC 3.1.3). This resultant glucosamine-6-phosphate is then de-phosphorylated (where r_2: glucosamine-6-phosphate hydrolase EC 2.7.1.8.

to avoid including multi-step reactions as optima, since a solution that includes a set of sub-steps of a process would have a higher total flux than one that only contains the overall reaction. This is not necessarily the case for objectives that involve maximisation (such as maximising biomass yield [Schuster et al., 1999]), since the objective value of such programs depends on the net stoichiometry of the solution.

4.6.1.2 Directionality

The methods described in this chapter identify internal cycles (with a zero net stoichiometry) that form inconsistent iso-stoichiometric groups. Such occurrences arise from the direction of reactions being incorrectly defined in the model and prevent the model from obeying the first law of thermodynamics (see Section 4.2.2 and Poolman et al. [2007]).

4.6.1.3 Other Database Artefacts

Other identified iso-stoichiometric groups involved enzyme subsets that hydrolyse ATP, such as, an enzyme subset in the *E. coli* model that consists of a glucosamine kinase reaction. This reaction uses ATP to generate glucosamine 6-phosphate from glucosamine. The resultant glucosamine-6-phosphate is then not used within any other reaction in the model except for a hydrolase reaction that degrades it to release phosphate, see Figure 4.13. The presence of such subsets suggests that the organism requires them to deal with a surplus of ATP [Poolman et al., 2003], or that some reactions have been incorrectly included in the model (likely to be caused by the presence of genes that correspond to the catalysis of multiple reactions [Poolman et al., 2006]).

Similarly, pairs of reactions that are identical except for the use of NADH/NADPH result in an enzyme subset with the net stoichiometry of one co-factor reducing the other (i.e. the net stoichiometry: NADP + NADH → NADPH + NAD). These pairs corresponded to enzymes that have an affinity for both NADH and NADPH as electron donors.

4.6.2 Model Analysis

After identifying and removing the model inconsistencies discussed above, the remaining reactions provide insights into genuine model redundancies and allow for net-processes to be compared amongst organisms.

4.6.2.1 Comparison of Compressed Models

Comparison of the reduced stoichiometry matrix of the GSMs listed in Table 2.1 revealed similarities in the metabolism of the different bacteria, shedding light on net processes important for all the species discussed but which might occur through different intermediate reactions.

Many shared reactions involved the synthesis and degradation of amino acids, such as the conversion of serine and malate into pyruvate, and the conversion of isocitrate to glutamate. Reactions from the non-oxidative branch of the pentose phosphate pathway (ribulose-5-phosphate isomerase and transketolase) were also present.

Most of the net-conversions that varied across models were caused by multi-step reactions, as discussed above. However, as part of the gluconeogenesis pathway, all GSMs except for *C. necator* convert oxaloacetate to phosphoenolpyruvate via a phosphoenolpyruvate carboxykinase enzyme that uses ATP as a substrate. *C. necator* uses ITP as a substrate, which is then regenerated using a reaction that requires ATP (Figure 4.14).

Such comparisons were possible since all of the GSMs analysed in this chapter use identical metabolite identifiers (since they were constructed directly from the MetaCyc database using ScrumPy). It is important to keep in mind that the comparing models whose metabolites are defined using different names/abbreviations would be difficult.

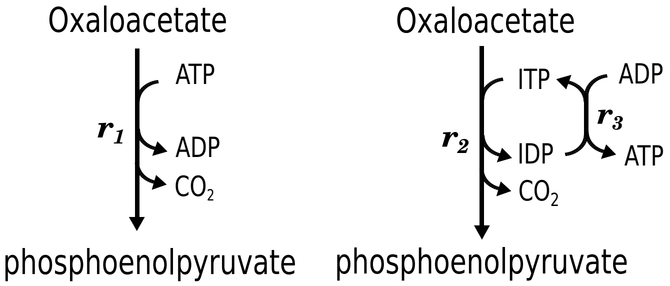


Figure 4.14: The conversion of oxaloacetate to phosphoenolpyruvate occurs via a phosphoenolpyruvate carboxikinase enzyme that uses ATP as a substrate (r_1: EC 4.1.1.49) in all models except for *C. necator*, where ITP is first used as a substrate, and is afterwards regenerated using a reaction that requires ATP (r_2: EC 4.1.1.32 and r_3: EC 2.7.4.6)

4.6.2.2 Genuine Alternative Routes

Iso-stoichiometric groups that arise from genuine alternate routes from a start to end product were also identified. These occurrences may correspond to enzymes which are active only under specific circumstances, the presence of which can be verified experimentally.

This information can enhance the insights attainable from current metabolic modelling techniques. For example, when FBA is applied to compressed models, the resultant pathway can be seen as a sub-module of the original model, which can subsequently be decomposed into some alternative flux vectors. Such a decomposition can help alleviate the problem of multiple optima (Section 2.3.5) when using LP by providing some alternative solutions.

However, the relevance of the optima provided by the algorithm is dependent on the optimisation criteria used. For example, the sum of fluxes of the alternate solutions for the *C. jejuni* model were slightly different (135.0 and 136.0 units of flux respectively), such that the two solutions would not be identified as alternate optima when the LP objective function is the minimisation of the sum of fluxes.

EMA can also benefit from the techniques described. Although compression reduces the number of EMs, their unique net stoichiometries remain unaffected. Grouping iso-stoichiometric EMs is a known technique for simplifying the analysis of EMs. Standard methods achieve this by first calculating the complete set of EMs. In contrast, the methods described in this chapter eliminate the need for such an enumeration whilst also facilitating the exploration of the simplifications that are made (such as through the tree in Figure

4.5). Compression also eliminates some unfeasible EMs that may result from traditional algorithms (as shown in Section 4.5.1).

4.6.3 The Combining of Biomass Export Transport Reactions

A theoretical result that followed from this chapter's analysis of enzyme subsets regards the export of biomass in GSMs. This work clarifies the distinction between the two distinct methods for defining biomass export by showing that, while models with individual biomass export offer greater flexibility than their combined counterparts, results obtained from (minimization) FBA when utilising either method are equivalent.

4.7 Conclusion

This chapter describes a method that iteratively reduces the size of metabolic networks. This approach facilitates the understanding of large metabolic models, by providing a framework for visualising and exploring enzyme subsets and redundant pathways. These methods can also aid model curation by discovering systematic inconsistencies in GSMs such as the inclusion of duplicate reactions and internal cycles with a net stoichiometry of zero.

The associated algorithms will be incorporated into the ScrumPy source tree in due course.

Chapter 5

The Left Null-Space and Metabolite Similarity

5.1 Introduction

As discussed in Chapter 2.3.2, the left null-space reflects linear dependencies between the rows of the stoichiometry matrix and reveals relationships between metabolite concentrations and their chemical composition.

5.1.1 Metabolite Concentrations

Relationships between metabolite concentrations are established through conservation relations: groups of metabolites whose linear combination of molar amounts stays constant over time. Each relation corresponds to a vector of the left null-space basis [Heinrich and Schuster, 1996, pages 78-87], and allowed Reder [1988] to partition metabolites into *dependent* and *independent* groups such that the concentrations of the dependent metabolites can be calculated from those of the independent metabolites (via the link matrix as described in Appendix D.1). This method is helpful to facilitate dynamic simulations. Indeed, dependent metabolites must be eliminated from **N** prior to most kinetic modelling techniques, as the calculation of a non-singular Jacobian matrix from **N** is only achievable if **N** is full row rank [Stelling and Klamt, 2006; Sauro and Ingalls, 2003]. Nevertheless, a network's set of conservation relations cannot be uniquely identified, as various sets of basis vectors can be used to define the left null-space of a matrix, and to date, no efficient method for identifying all potential conservation relations of a system has been described.

Such a description would be helpful to reveal the potential relationships between metabolite concentrations in the model and to show how structural changes (such as the removal of a reaction) can impact these relationships.

5.1.2 Metabolite Composition

Since reactions comprise of molecules being chemically transformed through the breakage and formation of bonds, metabolic networks can be characterised by the chemical building blocks that combine in different arrangements to form the networks' metabolites. Groups of atoms that form an identifiable unit within a molecule are referred to as *moieties*. Among these, *conserved moieties* remain intact throughout all transformations in the network (the simplest of which are individual atoms). The identification of conserved moieties within networks establishes relationships between metabolites by recognising the chemical species that they share [Schuster and Hilgetag, 1995; Famili and Palsson, 2003]. The conservation of each conserved moiety corresponds to a non-negative vector (i.e. containing no negative elements) in the left null-space (to be discussed further in Section 5.2.2). However, although algorithms that can calculate a network's entire set of such (convex) basis vectors exist [Schuster and Hilgetag, 1995], assigning moieties to the vectors is only possible if the chemical structure of each metabolite is available [Haraldsdóttir and Fleming, 2016] — a requirement that is difficult to satisfy, especially when considering large models.

Knowledge of metabolite composition is essential to ensure mass conservation when building models. Furthermore, once the structure of a group of metabolites is determined, it can be used to compare similarities between them. For example, molecular-fingerprint analysis compares metabolites on the basis of the molecular substructures that they have in common, as described in Section 5.3.2.4. This technique has applications in fields such as pharmacology, where the biological function of a novel compound is inferred by comparing its structure with those of metabolites whose functions are already known. This reasoning is based on the assumption that metabolites with similar chemical structures play similar physiological roles.

5.1.3 Aims and Objectives

The work described in this chapter shows that although the left null-space can be represented by different bases, the angles between the rows of an orthogonal left null-space are invariant. This allows for the introduction of a similarity measure that can describe the relationships between metabolite concentrations independently of the choice of basis, and with minimal additional information.

This measure was previously applied to the right null-space by Poolman et al. [2007], where it was shown to be equivalent to Pearson's correlation coefficient between all possible steady-state vectors. The authors used this result to relate steady-state fluxes such as by identifying disconnected sub-networks (as discussed in Chapter 2.3.3.2).

In addition, since vectors representing chemical moieties are located in the left null-space, this similarity measure can be used to gain information about the chemical similarity between metabolites. Therefore, since for a given model, it is often the case that some (but not all) metabolite elemental composition data is available, a method that enhances existing metabolite composition information by incorporating the information gained from the left null-space was developed. This method can be used to calculate the composition of metabolites that were previously unknown, and identify models that violate the law of conservation of mass.

Furthermore, two algorithms, referred to as the network pruning and pathway finding tools, that aim to follow the transfer of material in a network based on the similarity between the substrates and products were also developed.

5.1.4 Chapter Structure Overview

This chapter contains the following sections:

Section 5.2 introduces the left null-space in the context of metabolite concentration and composition.

Section 5.3 defines the similarity measure in the context of comparing metabolites by their concentration changes and chemical composition. Additionally, novel algorithms that can (i) calculate the elemental composition of unknown metabolites by using the information gained from the left null-space, (ii) reduce the size of models by discarding metabolites that have a low similarity to metabolites of interest (lossy compression) and (iii) extract pathways leading from a user-selected starting metabolite, where each metabolites in the pathway is chosen to be most similar to to the one preceding it, are developed.

Section 5.4 applies these methods to various metabolic models.

Section 5.5 discusses their implications on model curation and analysis.

Appendix D presents mathematical proofs regarding the similarity measure.

5.2 Background

5.2.1 The Left Null-Space and Conservation Relations

In this thesis, the left null-space matrix is denoted by the matrix $\mathbf{G}$, such that

$$\mathbf{N}^{\top}\mathbf{G} = \mathbf{O_{r\times q}} \left(\text{or } \mathbf{G}^{\top}\mathbf{N} = \mathbf{O_{q\times r}}\right), \tag{5.1}$$

where $\mathbf{N}$ is an $m \times r$ stoichiometry matrix and $\mathbf{G}$ is $m \times q$. Note that this method can be equally applied to the external stoichiometry matrix, $\mathcal{N}_{l\times r}$, to generate an external left null-space matrix, $\mathcal{G}_{l\times t}$, that has a different interpretation from $\mathbf{G}$ (relating to the chemical composition of metabolites, to be discussed further in Section 5.2.2).

Each row of $\mathbf{G}$ corresponds to a metabolite, and each column to a conservation relation, $\mathbf{g}$, derived as follows. Recall that the stoichiometry of a metabolic system can be represented as a set of ordinary differential equations describing changes in metabolite concentrations, $\mathbf{s}$, with time:

$$\frac{\mathrm{d}\mathbf{s}}{\mathrm{d}t} = \mathbf{N}\mathbf{v}, \tag{5.2}$$

where $\mathbf{v}$ is a vector of fluxes. Multiplying this system by the transpose of any left null-space vector, $\mathbf{g}$, on the left, yields

$$\mathbf{g}^{\top}\frac{\mathrm{d}\mathbf{s}}{\mathrm{d}t} = \mathbf{g}^{\top}\mathbf{N}\mathbf{v}, \tag{5.3}$$

which results in

$$\mathbf{g}^{\top}\frac{\mathrm{d}\mathbf{s}}{\mathrm{d}t} = 0. \tag{5.4}$$

Therefore identifying metabolites whose rate of change of concentrations must sum to zero. Moreover, integrating Equation (5.4) with respect to time reveals:

$$\mathbf{g}^{\top}\mathbf{s}(\mathbf{t}) = \mathbf{g}^{\top}\mathbf{s}(\mathbf{0}) = \lambda, \tag{5.5}$$

where $\mathbf{s}(\mathbf{0})$ is a vector of constants corresponding to initial metabolite concentrations such that $\mathbf{g}^{\top}\mathbf{s}(\mathbf{0})$ is a real constant, λ. Therefore, each $\mathbf{g}$ identifies a group of metabolites whose rate of change of concentrations must sum to zero (Equation (5.4)), or equivalently, whose

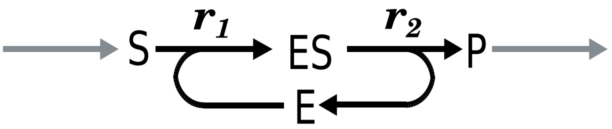


Figure 5.1: An enzyme catalysed reaction, where S: substrate; ES: enzyme-substrate complex; E: free enzyme; P: product.

combination of total concentrations must stay constant through time (Equation (5.5)).

λ represents a constant sum of concentrations in models of single compartments and a constant sum of metabolite molar amounts in models of multiple compartments. In the latter instance, **g** can be converted into metabolite concentrations by taking the relative volume of the different compartments into account (as described by Hofmeyr [2020]).

A simple example of a conservation relation is that described by the amount of enzyme in enzyme-catalysed reactions as shown in Figure 5.1. At any point in time, as the enzyme-substrate complex is produced, an equal amount of free enzyme must be consumed, such that the total amount $E + ES$ remains unchanging. This behaviour corresponds to the conservation relation

$$\frac{\mathrm{d}E(t)}{\mathrm{d}t} + \frac{\mathrm{d}ES(t)}{\mathrm{d}t} = 0, \tag{5.6}$$

that is equivalent to:

$$E(t) + ES(t) = E(0), \tag{5.7}$$

where $E(t)$ and $ES(t)$ represent the amounts of free enzyme and enzyme-substrate intermediate, while $E(0)$ is a real positive constant that represents the total amount of free enzyme at time $t = 0$ (when $ES(0) = 0$).

More complex examples will be considered in Section 5.2.3.

5.2.2 Elemental Conservation

The elemental composition (empirical formula) of a metabolite describes the type and number of atoms of which it is comprised. This information is associated with the left null-space of the external stoichiometry matrix, $\mathcal{G}$, since, in a closed system, the total number of atoms of each element on both sides of a reaction must be equal, such that vectors representing the proportion of elements in every metabolite of the network are in

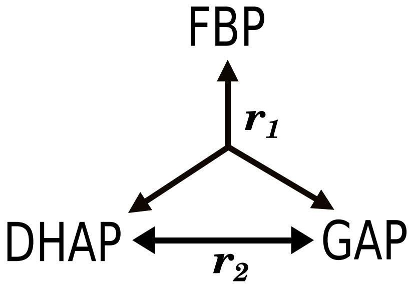


Figure 5.2: The two reversible reactions aldolase (r_1) and triose-phosphate isomerase (r_2), where FBP: fructose-1,6-bisphosphate; GAP: glyceraldehyde-3-phosphate; DHAP: dihydroxyacetone phosphate.

$\mathcal{G}$.

For example, consider the number of carbon atoms in the two reaction scheme shown in Figure 5.2 which is comprised of the reactions r_1 (aldolase) and r_2 (triose-phosphate isomerase), given by

$$\begin{aligned} r_1 : \mathrm{C_6H_{10}O_{12}P_2} &\rightarrow \mathrm{C_3H_5O_6P} + \mathrm{C_3H_5O_6P}, \\ r_2 : \mathrm{C_3H_5O_6P} &\leftrightarrow \mathrm{C_3H_5O_6P}, \end{aligned} \tag{5.8}$$

or equivalently, by the set of ordinary differential equations:

$$\begin{pmatrix} \frac{\mathrm{dFBP}}{\mathrm{d}t} \\ \frac{\mathrm{dGAP}}{\mathrm{d}t} \\ \frac{\mathrm{dDHAP}}{\mathrm{d}t} \end{pmatrix} = \begin{pmatrix} -1 & 0 \\ 1 & -1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} v_1 \\ v_2 \end{pmatrix}. \tag{5.9}$$

A vector of positive integers, $\mathbf{a}$, can be constructed such that every entry a_i corresponds to the number of carbon atoms in metabolite i, namely

$$\mathbf{a} = \begin{array}{r} \text{FBP} \\ \text{GAP} \\ \text{DHAP} \end{array} \begin{pmatrix} 6 \\ 3 \\ 3 \end{pmatrix}. \tag{5.10}$$

The result of multiplying the stoichiometry of any reaction within the network (corresponding to a column of $\mathcal{N}$) with the vector $\mathbf{a}$ should be zero. For example, consider the first column of $\mathcal{N}$, $\mathbf{n}_1$, then

$$\mathbf{n}_1^\top \mathbf{a} = \begin{pmatrix} -1 & 1 & 1 \end{pmatrix} \begin{pmatrix} 6 \\ 3 \\ 3 \end{pmatrix} = 0, \tag{5.11}$$

confirming that the number of carbon atoms being consumed (6) is equal to those being produced (3+3).

Such relations hold for all of the reactions in the network and therefore $\mathbf{a}$ must be an element of the external left null-space (i.e. $\mathcal{N}^\top \mathbf{a} = \mathbf{0}$). This can be extended to account for every chemical element in the network to generate the atomic matrix, $\mathbf{A}$, whose entries, a_{ij}, correspond to the number of atoms of element j in metabolite i. The matrix $\mathbf{A}$ for the system in Figure 5.2 is therefore

$$\mathbf{A} = \begin{array}{c} \\ \text{FBP} \\ \text{GAP} \\ \text{DHAP} \end{array} \begin{array}{c} \begin{array}{cccc} C & H & O & P \end{array} \\ \begin{pmatrix} 6 & 10 & 12 & 2 \\ 3 & 5 & 6 & 1 \\ 3 & 5 & 6 & 1 \end{pmatrix} \end{array}. \tag{5.12}$$

Since all columns of $\mathbf{A}$ are in the external left null-space, then by the Steinitz Exchange Lemma (a well-known result in linear algebra which states that for a given vector space, any set of linearly independent vectors within the space can be expanded, by adding more vectors, to form a basis for the space [Stiefel, 1963]), it is possible to construct a basis for the external left null-space that includes all of the linearly independent columns of $\mathbf{A}$ (as shown by Proposition D.2.0.1 in Appendix D.2.0.1), such that these columns of $\mathbf{A}$ (along with some additional columns if $\text{rank}(\mathbf{A}) < \text{rank}(\ker(\mathcal{N}^\top))$) can combine to form all possible external left null-space vectors of the network.

Note that in the above example, all columns of the atomic matrix are linearly dependent (i.e. $\text{rank}(\mathbf{A}) = 1$). This means that all elements appear at the same atomic ratio in all three species in the system ($C_3H_5O_6P$ or its multiple), such that the chemical composition of the metabolites can be alternatively represented by

$$\mathbf{A_{moie}} = \begin{array}{c|c|} & \mathrm{C_3H_5O_6P} \\ \mathrm{FBP} & 2 \\ \mathrm{GAP} & 1 \\ \mathrm{DHAP} & 1 \end{array}. \tag{5.13}$$

Atomic vectors are not necessarily in the left null-space, $\mathbf{G}$, of the internal stoichiometry matrix, $\mathbf{N}$, since $\mathbf{N}$ represents an open system in which not all matter is conserved. For instance, multiplying $\mathbf{A}$ by a vector of $\mathbf{N}$ corresponding to a transporter reaction will yield the number of atoms that the external metabolites are comprised of (rather than $\mathbf{0}$ for the corresponding vector of $\mathcal{N}$). However, multiplying $\mathbf{A}$ with a closed reaction (i.e. any reaction that does not interact with external metabolites) yields $\mathbf{0}$. As a consequence, the vectors of internal left null-space concern the conservation of elements/quantities within a closed loop inside the network [Sauro and Ingalls, 2003], a concept that is expanded upon below.

5.2.3 Moiety Conservation

Apart from atomic conservation, charge conservation (the flow of electrons) and moiety conservation (the flow of intact atomic groups) have also been historically considered.

As discussed by Sauro and Ingalls [2003], conservation relations in the left null-space of the internal stoichiometry matrix represent the flow of conserved moieties within closed loops. For example, the conservation of the amount of enzyme as shown by the relation described by Equation (5.7). While vectors of the external left null-space also include quantities that are conserved as part of pathways leading from external inputs to outputs (such as atoms, as discussed previously). Therefore, the vectors in $\mathbf{G}$ reveal a subset of the information in $\mathcal{G}$ [Haraldsdóttir and Fleming, 2016].

Attempts to infer biological significance from the left null-space have been aimed at understanding the extent of the contribution that these different types of conserved quantities have on $\mathbf{G}$ [Famili and Palsson, 2003; Haraldsdóttir and Fleming, 2016]. For example, consider an extract of glycolysis depicted in Figure 5.3. The left null-space of its internal stoichiometry matrix is listed in Table 5.1. This matrix has three conservation relations that reflect the conservation of the following moieties:

c_1: phosphate,

c_2: ADP,

c_3: NAD,

where phosphate, ADP, and NAD, are each conserved via closed loops inside the network. ADP is inter-converted between its phosphorylated and dephosphorylated forms and NAD is inter-converted between its oxidised and reduced forms, whilst phosphate is first transferred from ATP to the pathway's 6-carbon sugars (by hexokinase and phosphofructokinase, denoted by r_1 and r_3), and then incorporated back into ADP (by pyruvate kinase, denoted by r_{10}) during the pathway's final step, to regenerate ATP.

Note that glucose and pyruvate are not part of any conservation relation in $\mathbf{G}$ since they do not contain moieties that are internally conserved in the system. Such an observation would not be possible in case of $\mathcal{G}$, as due to the principle of conservation of mass all metabolites of the system must be present in at least one conservation vector.

If present, conserved moieties can be added to the columns of $\mathbf{A}$. For example, a column vector corresponding to the *ADP* moiety can be added to $\mathbf{A}$, such that the metabolite ADP is listed to contain one such moiety and the metabolite ATP is listed to contain one molecule of *ADP* and one phosphate.

Although the left null-space basis shown above is simple to understand, bases obtained from more complex networks can be challenging to interpret. Indeed, the calculation of one basis does not necessarily lead to the identification of all conserved moieties. For example, some vectors may not correspond to a single conserved quantity but a linear combination of quantities (such the number of carbon atoms subtracted by the number of hydrogen atoms), leading to entries with negative coefficients that cannot be assigned to physical quantities. For example, consider the simple system in Figure 5.4, that only

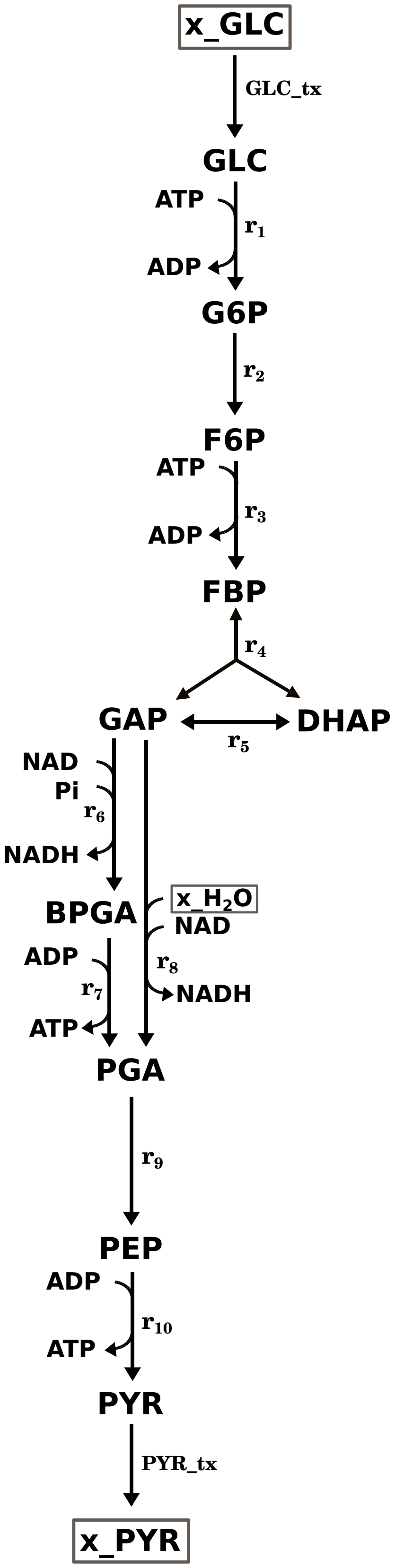


Figure 5.3: An extract of glycolysis. See Appendix A.2 for reaction and metabolite abbreviations.

Table 5.1: A left null-space basis of the internal stoichiometry matrix for the model of glycolysis in Figure 5.3. The columns reflect the conservation of the phosphate, ADP, and NAD moieties. See Appendix A.2 for reaction and metabolite abbreviations.

	c_1	**c_2**	**c_3**
GLC	0	0	0
G6P	1	0	0
ATP	1	1	0
ADP	0	1	0
F6P	1	0	0
FBP	2	0	0
GAP	1	0	0
DHAP	1	0	0
NAD	0	0	1
Pi	1	0	0
NADH	0	0	1
BPGA	2	0	0
PGA	1	0	0
PEP	1	0	0
PYR	0	0	0

contains the following conservation relation, **g**,

$$\mathbf{g} = \begin{array}{c} \mathrm{A} \\ \mathrm{B} \\ \mathrm{C} \\ \mathrm{D} \end{array} \left(\begin{array}{c} 0 \\ 1 \\ -1 \\ 0 \end{array} \right), \tag{5.14}$$

This conservation relation shows that an increase in the concentration of B must be accompanied by an equal increase in the concentration of C, but this does not correspond to the presence of an identical physical conserved moiety in both metabolites.

5.2.4 Maximal Conserved Moieties

The largest possible sub-parts of metabolites that never get broken down are referred to as maximal conserved moieties. Each maximal moiety was shown to correspond to a

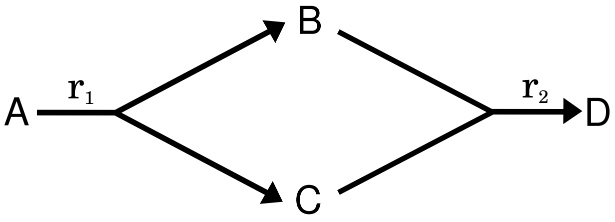


Figure 5.4: A simple system where an increase in the concentration of metabolite B must be accompanied by an equal increase in the concentration of metabolite C.

fundamental solution of the system

$$\{\mathbf{z} \in \mathbb{Z}^m | \quad \mathcal{N}^\top \mathbf{z} = \mathbf{0}\,, \mathbf{z} \geq 0\}, \tag{5.15}$$

where m is the number of metabolites in the external stoichiometry matrix $\mathcal{N}$ and $\mathbf{z}$ denotes a maximal conserved moiety vector Schuster and Hilgetag [1995]. The set of all such vectors (i.e. all fundamental solutions of Equation 5.15) is denoted by $\mathbf{Z}$.

An algorithm that can characterise $\mathbf{Z}$ was proposed by Schuster and Hilgetag [1995]. This algorithm is based on the fact that conserved moiety vectors must always be non-negative and integer, and is similar to that employed for the calculation of EMs [Schuster and Hilgetag, 1994].

This method can relate metabolites by identifying potential conservation of moieties among them (in terms of conservation in the empirical formula). However, does not reveal the chemical composition of these conserved moieties. If the moieties' composition is of interest, additional information regarding the chemical composition/structure of the metabolites of the network can be used to assign atomic groups to some of the vectors in $\mathbf{Z}$. For example, [Schuster and Hilgetag, 1995] calculated the complete set of maximal conserved moiety vectors for a small network and manually identified that two out of the network's four maximal conserved moiety vectors correspond to conserved atomic groups, one reveals the transfer of a hydrogen atom between metabolites, and the other is not physically meaningful. Alternatively, Haraldsdóttir and Fleming [2016] designed an algorithm that, when given information about metabolite structure, follows the position of atoms as they move from substrate to product, such that atomic groups that remain intact during the transformations are identified. Similarly to Schuster and Hilgetag [1995], the authors noted that when the algorithm was applied to networks, not all of the calculated maximal conserved moiety vectors were found to be physically meaningful.

5.2.5 Calculating the Left Null-Space Basis.

The left null-space basis can be calculated through the Gauss-Jordan Method or Singular Value Decomposition (SVD). The Gauss-Jordan method generates a sparse basis, whilst SVD yields an orthogonal set of basis vectors. Although SVD is more efficient, the Gauss-Jordan method has been historically favoured since its basis vectors are easier to interpret in a biological context (since each vector is likely to involve a smaller number of metabolites, for example, the basis in Table 5.1).

The methods presented here extract biologically relevant information from normalized bases that are calculated using SVD.

5.3 Methodology

This section concerns the development of a similarity measure that can relate metabolites both by the conservation relations that they are in, as well as by their chemical similarity. Mathematical results relating to the similarity measure can be found in Appendix D.3.

The techniques and proofs described here were adapted from prior research by Poolman et al. [2007], wherein the aforementioned similarity measure was applied to the right null-space to establish relationships between steady-state fluxes.

5.3.1 Comparing Metabolites Through Conservation Relations

In this chapter, metabolites are compared by measuring the cosine of the angle between the corresponding rows of an orthonormal basis for the left null-space matrix of the internal stoichiometry matrix $\mathbf{G}$, as defined by Equation (5.16):

$$\phi_{ij}^{G} = \cos(\theta_{ij}^{G}) = \frac{\boldsymbol{G_i}\boldsymbol{G_j}^{\top}}{\sqrt{\boldsymbol{G_i}\boldsymbol{G_i}^{\top}}\sqrt{\boldsymbol{G_j}\boldsymbol{G_j}^{\top}}}, \tag{5.16}$$

where ϕ_{ij}^{G} denotes the similarity between metabolites i and j, and $\mathbf{G_i}$ is a row of $\mathbf{G}$.

This measure can alternatively be applied to the left null-space of the external stoichiometry matrix, $\mathcal{G}$, resulting in a different interpretation to when applied on $\mathbf{G}$ (to be discussed later).

Although the basis of the left null-space can be expressed by different sets of generating vectors, ϕ_{ij}^{G} is unique (as shown by Theorem D.3.1.1 in Appendix D.3, adapted from Lemma 1 in [Poolman et al., 2007]) and corresponds to Pearson's correlation coefficient between the set of all possible conservation relations (see Proposition D.3.2.1 in Appendix D.3.2.1).

Therefore, ϕ_{ij}^{G} relates metabolites based on how often they appear within the same conservation relation when considering the set of all possible conservation relations in the network. It reflects the connectivity between the concentration values of metabolites within the same compartment, and molar amounts of metabolites in different compartments.

A correlation matrix, $\mathbf{R}$, can be calculated using this measure such that every entry, r_{ij}, corresponds to the similarity between metabolites i and j (i.e. $r_{ij} = \phi_{ij}^{G}$). This matrix can then be used to cluster metabolites by their similarity. In this thesis, this clustering is achieved through the Weighted Pair Group Method Using Arithmetic Averaging (WPGMA) algorithm, as described by Poolman et al. [2007].

The values of ϕ^{G}range from 0 to ±1 as discussed below.

Maximal similarity (proportional rows).

A value of $\phi_{ij}^{G} = \pm 1$ implies that metabolites i and j must always be present together in the same conservation relations, since every possible conservation relation, $\mathbf{g}$, that contains i must also contain j and vice versa (as discussed in Appendix D.3.4). A positive value signifies that the concentration changes of metabolites i and j must have the same sign in all conservation relations, whilst a negative value implies that they must have opposing signs.

Furthermore, it is not possible to construct a link matrix in which i and j are both independent metabolites, meaning that the system of differential equations, $\frac{\mathrm{d}\mathbf{S}}{\mathrm{d}t} = \mathbf{N}\mathbf{v}$, cannot be modified such that the concentration of parallel metabolites is used to calculate the concentration of some other metabolites (as shown in Appendix D.1).

Maximal difference (orthogonal rows).

If there is no conservation relation vector, $\mathbf{g}$, that contains non-zero elements for both metabolite i and j then $\phi^G_{ij} = 0$ (as shown in Appendix D.3.3). However, the reverse is not necessarily true.

Hence some cases of $\phi^G_{ij} = 0$ are expected to be caused by metabolites that have independent concentrations, since no conservation relation vector, $\mathbf{g}$, has non-zero elements for both metabolites. In other words, the concentration of metabolite i is not influenced by (and therefore cannot reveal information about) the concentration of metabolite j, for example if the metabolites are in stoichiometrically disconnected sub-networks.

5.3.2 Comparing Metabolites Through Their Elemental Composition

This subsection describes the methods developed in this thesis for relating the elemental composition of metabolites through:

1. inferring knowledge from the left null-space, by applying the similarity measure to the orthonormal basis of the external left null-space,
2. retrieving metabolite composition information from online databases,
3. augmenting the information gained from the databases using the knowledge inferred from the left null-space, by, when given a metabolite whose elemental composition could not be retrieved from the database, predicting its composition using information from the left null-space and the known composition of other metabolites.

5.3.2.1 The External Left Null-Space

Since every conserved moiety vector (including atomic vectors as discussed in Section 5.2.2) must be in the left null-space, the similarity measure described above can be applied to the left null-space of the external stoichiometry matrix, $\mathcal{G}$, to obtain information about elemental composition as described below. Note that in this case, the absolute value of the measure was considered in order to facilitate the clustering of metabolites.

Maximal similarity (proportional rows).

A value of $\phi_{ij}^{\mathcal{G}} = 1$ implies that metabolites i and j must have an identical elemental composition ratio, since every maximal conserved moiety vector, $\mathbf{z}$, and atomic composition vector, $\mathbf{a}$, must contain the same elements in both metabolites (as discussed in Appendix D.3.4). The ratio of the rows pertaining to metabolites i and j in the left null-space indicates whether they are isomers (1:1) or whether the empirical formula of one is a multiple of the other.

Note that since $\phi_{ij}^{\mathcal{G}}$ is dependent on the structural connectivity between the metabolites, not all metabolites that have an identical elemental composition ratio are identified by $\phi_{ij}^{\mathcal{G}}$ (although all metabolites $\phi_{ij}^{\mathcal{G}} = 1$ must be so).

Maximal difference (orthogonal rows).

If there is no external left null-space vector, $\mathbf{g}$, that contains non-zero elements for both metabolite i and j then $\phi_{ij}^{\mathcal{G}} = 0$ (as discussed in Appendix D.3.3). However, the reverse is not necessarily true.

Since, maximal conserved moiety vectors, $\mathbf{z}$, and atomic composition vectors, $\mathbf{a}$, are in the external left null-space, then some cases where $\phi_{ij}^{\mathcal{G}} = 0$ will be caused by metabolites i and j having an unrelated elemental composition.

As above, metabolites with an unrelated elemental composition may still contain identical physical moieties (such as similar atomic groups), that are, however, considered to be separate quantities in $\mathbf{Z}$ because no material transfer between the two metabolites (either directly or through intermediates) can occur, for example, because the metabolites are in unconnected compartments (to be further discussed in Section 5.4).

5.3.2.2 Information from Online Databases

Atomic matrices can be constructed by retrieving information from databases such as MetaCyc and KEGG. However, retrieving all of the required information is difficult, often caused by (i) human annotation of metabolites being different from the identifiers listed in the database or (ii) metabolites (such as polymers) not having an associated empirical formula.

In this work, information was obtained from MetaCyc and the resultant matrix is denoted by $\mathbf{A_{db}}$.

5.3.2.3 Supplementing the Information Gained from Online Databases using the Left Null-space

When given an incomplete atomic matrix, a novel algorithm that can calculate the composition of unknown metabolites by determining the values that can be assigned to them when ensuring conservation of matter in the system was devised.

This approach also identifies reactions that violate conservation of mass (labelled here as *inconsistent* reactions) when given $\mathbf{A_{db}}$. This is accomplished by finding the reactions that satisfy one of the following two conditions

1. if the composition of all of the metabolites in a reaction is known in $\mathbf{A_{db}}$, then the atoms in the left and right-hand side of the reaction are not balanced, or

2. if the composition of some of the metabolite in a reaction is unknown, then at least one of the metabolites cannot contain a positive number of moieties without violating the conservation of mass in the system.

The algorithm assumes that all metabolite composition information in $\mathbf{A_{db}}$ is correct and proceeds as follows.

Consider an incomplete atomic matrix, $\mathbf{A_{db}}$, specifying the elemental composition of a subset of metabolites in a given system. Then, the atomic matrix, $\mathbf{A}$, can be split into two parts:

$$\mathbf{A} = \begin{pmatrix} \mathbf{A_{known}} \\ \mathbf{A_{unknown}} \end{pmatrix}, \tag{5.17}$$

where

1. $\mathbf{A_{known}}$ is the set of metabolites whose composition is known (i.e. the metabolites in $\mathbf{A_{db}}$), and

2. $\mathbf{A_{unknown}}$ is the set of metabolites whose composition is unknown (i.e. the metabolites in $\mathcal{N}$ but not in $\mathbf{A_{db}}$).

Algorithm 7 *Extend(*$\mathbf{A_{db}}$*,* $\mathcal{N}$*)*, which when applied to an incomplete atomic matrix returns a matrix, $\mathbf{A_{extended}}$, that extends upon the rows of $\mathbf{A_{db}}$ to include information about more metabolites, and a matrix $\mathbf{A_{ranges}}$ that specifies the range of values that the metabolites which are still unknown must have.

```
$\mathbf{A_{extended}}$ = empty_matrix()
$\mathbf{A_{ranges}}$ = empty_matrix()
for element in col_names($\mathbf{A_{db}}$) do
    $\mathbf{a_{known}}$ = get_col($\mathbf{A_{db}}$, element)
    ilp = $\mathcal{F}(\mathbf{a_{known}}, \mathcal{N})$
    extended_el_dict = empty_dictionary()
    ranges_el_dict = empty_dictionary()
    $\mathbf{a_{unknown}}$ = get_difference_rnames($\mathbf{A_{db}}$, $\mathcal{N}$)
    for met in $\mathbf{a_{unknown}}$ do
        min_val = get_sol_minimize(ilp, met)
        max_val = get_sol_maximize(ilp, met)
        if min_val == max_val then
            extended_el_dict[met] = min_val
        else
            ranges_el_dict[met] = (min_val, max_val)
        end if
    end for
    add_col($\mathbf{A_{extended}}$, extended_el_dict, name = element)
    add_col($\mathbf{A_{ranges}}$, ranges_el_dict, name = element)
end for
return $\mathbf{A_{extended}}$, $\mathbf{A_{ranges}}$.
```

The composition of the metabolites in $\mathbf{A_{unknown}}$ are inferred, one element at a time, by Algorithm 7. Specifically, each metabolite is identified as:

- having a particular elemental composition, or
- having a composition from a known range.

This is accomplished by repeatedly solving the following Integer Linear Optimisation Problem (ILP):

$$\mathcal{F}(\mathbf{a_{known}}, \mathcal{N}) = \min/\max \quad a_k \,, \ \text{for one} \ a_k \in \mathbf{a_{unknown}}$$
$$\text{subject to} \quad \begin{cases} \mathcal{N}^\top \mathbf{a} & = & \mathbf{0}\,, \\ a_i & = & val_i, \ \text{for all} \ a_i \in \mathbf{a_{known}}\,, val_i \in \mathbf{A_{db}}, \\ a_i & \geq & 0, \ \text{for all} \ a_i \in \mathbf{a_{unknown}}. \end{cases} \tag{5.18}$$

The constraints are based on the fact that all atomic vectors, $\mathbf{a}$, must have the following three properties:

1. $\mathbf{a}$ must be in the left null-space (i.e. $\mathcal{N}^\top \mathbf{a} = \mathbf{0}$),
2. the metabolites in $\mathbf{a_{known}}$ must be equal to their values in $\mathbf{A_{db}}$ ($a_i = val_i$, where val_i is the corresponding value in $\mathbf{A_{db}}$), and
3. every element of $\mathbf{a}$ must be a positive integer (i.e. $\mathbf{a}_i \geq \mathbf{0}$ and $\mathbf{a} \in \mathbb{Z}^m$).

The algorithm iterates through all of the metabolites in $\mathbf{a_{unknown}}$ and Equation (5.18) is solved twice for each metabolite. The objective function is defined such that the first solution obtains the minimum possible value that can be assigned to the metabolite whilst satisfying the pre-defined constraints and the second solution obtains the maximum value (Steps 10 and 11 in Algorithm 7). If the two values are equal, then only one elemental value can allow conservation of matter in the system, and that value is saved for addition into the extended atomic matrix $\mathbf{A_{extended}}$ (Step 13). If this is not the case, the metabolite's potential range of values is saved to be added to $\mathbf{A_{ranges}}$ (Step 15).

The algorithm repeats the previous steps until all columns in $\mathbf{A_{db}}$ are extended.

Figure 5.5 shows examples of the output of the algorithm when applied to simple small models.

Inconsistent reactions.

When constructing the ILP described by Equation (5.18), the rows of $\mathcal{N}^\top$ (each of which corresponds to a reaction) are added as constraints one row at a time. At each step, the reaction to be added is tested for whether it satisfies the law of conservation of matter as specified by $\mathbf{A_{db}}$ and the constraints generated by rows of $\mathcal{N}^\top$ that had been already added to the ILP. If this is found not to be the case, the reaction is assumed to not be correctly balanced, recorded, and removed from the constraints of the ILP (i.e. deleted from the rows of $\mathcal{N}^\top$). A more comprehensive solution for this problem would be to redefine unbalanced reactions to ensure the matter is conserved, but this has not been considered in this thesis.

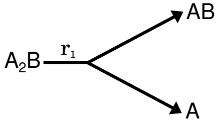
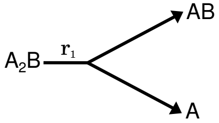
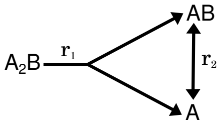
Network	**Input**	**Outputs**
	$\mathbf{A_{db}} = \begin{array}{c} \\ \text{AB} \\ \text{A} \end{array} \begin{array}{cc} A & B \\ \left(\begin{array}{cc} 1 & 1 \\ 1 & 0 \end{array}\right) \end{array}$	$\mathbf{A_{extend}} = \begin{array}{c} \\ \text{A}_2\text{B} \\ \text{AB} \\ \text{A} \end{array} \begin{array}{c} A \quad B \\ \left(\begin{array}{cc} 2 & 1 \\ 1 & 1 \\ 1 & 0 \end{array}\right) \end{array}$
	$\mathbf{A_{db}} = \begin{array}{c} \\ \text{A}_2\text{B} \end{array} \begin{array}{c} A \quad B \\ \left(\begin{array}{cc} 2 & 1 \end{array}\right) \end{array}$	$\mathbf{A_{extend}} = \begin{array}{c} \\ \text{A}_2\text{B} \end{array} \begin{array}{c} A \quad B \\ \left(\begin{array}{cc} 2 & 1 \end{array}\right) \end{array}$ $\mathbf{A_{ranges}} = \begin{array}{c} \\ \text{AB} \\ \text{A} \end{array} \begin{array}{c} A \qquad B \\ \left(\begin{array}{cc} (0,2) & (0,1) \\ (0,2) & (0,1) \end{array}\right) \end{array}$
	$\mathbf{A_{db}} = \begin{array}{c} \\ \text{AB} \\ \text{A} \end{array} \begin{array}{c} A \quad B \\ \left(\begin{array}{cc} 1 & 1 \\ 1 & 0 \end{array}\right) \end{array}$	$\mathbf{A_{ranges}} = \begin{array}{c} \\ \text{A}_2\text{B} \\ \text{AB} \\ \text{A} \end{array} \begin{array}{c} A \quad B \\ \left(\begin{array}{cc} 2 & 1 \\ 1 & 1 \\ 1 & 0 \end{array}\right) \end{array}$ inconsistant_reacs : [r$_2$]

Figure 5.5: The results of Algorithm 7 when applied to simple models, where the output of the algorithm is shown. In the first example, the composition of both products of r_1 is known, such that the composition of the substrate must be the sum of the composition of the products. In the second example, only the composition of the substrate is known, such that the products can be assigned a range of values reflecting the various ways through which the substrate can be divided in two. In the third example, r_2 violates conservation of matter (since its substrates and products have a different elemental composition) and is therefore labelled as inconsistent.

Furthermore, note that, due to the sequential nature of the algorithm, reactions identified as inconsistent do not necessarily have incorrect stoichiometry, but may be classified as so because of incorrect reactions that were added to the ILP in earlier steps. For example, consider the third panel in Figure 4.4, if the order of reactions in $\mathcal{N}$ was alternatively $\{r_2, r_1\}$, then r_2 would be the first reaction added to the ILP. Following this, r_1 would be determined to be inconsistent.

5.3.2.4 Molecular-Fingerprint Encoding

In this chapter, molecular-fingerprint encoding was used to compare the results obtained from the similarity measure with those of more established techniques.

Molecular-fingerprint encoding is a widely used method for comparing the structure of metabolites [Brown et al., 2005]. It was developed to facilitate the similarity-based virtual screening aspect of the drug-discovery process, where large databases of small molecules are evaluated for their likely-hood to bind to a drug target (by comparing their structure with that of known ligand) [Willett, 2006].

This method assigns each metabolite a binary sequence (referred to as a *fingerprint*) that details whether the metabolite contains occurrences from a set of pre-defined chemically interesting substructures. Specifically, each bit in the sequence corresponds to a specific substructure that is set to one or zero depending on whether the substructure is present in the molecule. Subsequently, the fingerprints are compared with each other using methods such as the *Tanimoto coefficient*, which compares two fingerprints by calculating the ratio of substructures that are common to both fingerprints in relation to the substructures that are present in only one of the fingerprints [Bajusz et al., 2015].

5.3.3 The Network Pruning and Pathway Finding Algorithms

As discussed in Section 1, one of the challenges in understanding metabolic networks and the results of their analysis stems from their large size. Therefore, a use of metabolite chemical composition data is to reduce the size of models by selectively retaining metabolites of interest, in a process that can be referred to as *lossy* compression. For example, removing co-factors (by considering them as external) was proposed as a method for alleviating the combinatorial explosion associated with the calculation of EMs by Schuster et al. [1999], while Huang et al. [2017] and Ghaderi et al. [2020] used atom mapping information to extract pathways that follow the movement of specific moieties through pathways.

A method that uses the similarity measure to simplify networks was developed as part of this thesis, referred to here as *network pruning.* This method aims to linearise networks as much as possible, resulting in mono-molecular reactions (only one substrate and only one product) that retain the substrates and products that have a maximal amount of similarity with each other.

Additionally, a novel method that uses the similarity measure to generate a path from a start product, by connecting metabolites that have the highest similarity between them was also designed and referred to here as the *pathway finding* algorithm.

The network pruning algorithm.

In order to simplify a metabolic network, metabolites are here classified as *leading* or *supporting*, where in any reaction, the leading metabolites are assumed to have a maximal similarity with the other leading metabolites in the network. For example, if a researcher was interested in the paths leading to the generation of PGA by the enzyme RuBisCo as shown by reaction r_1:

$$r_1 : RuBP + CO_2 + H_2O \rightarrow 2\,PGA + 2H^+, \tag{5.19}$$

Then RuBP and PGA would be classified as leading, whilst the supporting metabolites would be carbon dioxide, water, and protons. Removing the supporting metabolites results in an approximate reaction for visualising the transformation of PGA:

$$\tilde{r}_1 : RuBP \rightarrow 2\,PGA. \tag{5.20}$$

This classification is achieved whilst ensuring that:

1. every reaction has at least one leading substrate and one product,

2. the similarity ϕ between each substrate and product is maximal,

where the similarity measure can be taken to be:

1. the angle between the rows of the orthonormal external left null-space, $\phi^{\mathcal{G}}$, or
2. the angle between the rows of the atomic matrix, ϕ^{A}.

Given these definitions, this problem was tackled through the development of Algorithm 8 that iteratively removes metabolites from the network, in a process that identifies leading metabolites until all reactions in the network have at least one leading substrate and product.

First, an initial subset of labelled metabolites is obtained directly from $\mathcal{N}$ where

1. metabolites identified as proportional rows of $\mathcal{N}$ (corresponding to conservation relations with two non-zero elements, which identify conserved moieties such as ADP and NADP) are classified as supporting and removed from $\mathcal{N}$ (Steps 2 and 3 in Algorithm 8),
2. metabolites that are the sole substrate/product of a reaction are classified as leading (Step 4),
3. metabolites with identical elemental composition ratios to any of the metabolites labelled in points 1 and 2 above (identified as parallel groups by the similarity measure) are assigned the same label as their parallel counterparts (Step 6).

Then, every unclassified metabolite in $\mathcal{N}$ is assigned a cost value (provided that it is on the other side of (i.e. opposite to) a leading metabolite in at least one reaction):

$$cost_i = av\_dist\_from\_leading_i + \lambda\, met\_degree_i, \tag{5.21}$$

where *av_dist_from_leading* is one minus the average similarity value between the considered metabolite and any leading metabolites that it is opposite to in a reaction (where the smaller the value, the more similar), whist *met_degree* is the proportion of reactions that the metabolite is involved in (standardised such that the metabolite involved in the largest number of reactions has a value of one), therefore taking into account that metabolites involved in a relatively large number of reactions are likely to be co-factors. λ is a constant chosen by the user (chosen to be one during the work presented in this chapter).

The metabolite with the largest cost value is classified as supporting and removed from $\mathcal{N}$ (Steps 8 and 9). Any metabolite, which, after this removal, becomes the sole substrate/product of a reaction, is classified as leading (Steps 10). In addition, unlabelled metabolites with identical elemental composition ratios to metabolites that were labelled in the current iteration are assigned the same label (Step 12).

This process is iteratively repeated until all of the reactions in the model have at least one leading substrate and product. Once this is achieved, all of the remaining unlabelled metabolites are classified as supporting.

Algorithm 8 *Network_pruning(*$\mathbf{R}$*,* $\mathcal{N}$*)*, which when applied to a correlation matrix, $\mathbf{R}$, and stoichiometry matrix, $\mathcal{N}$, prunes metabolites from $\mathcal{N}$, on the basis of maximising the similarity between the substrates and products of reactions.

```
 1: labels_dict = empty_dictionary()
 2: cofacs = get_cofacs($\mathcal{N}$)
 3: del_mets_from_N($\mathcal{N}$, cofacs)
 4: leading_mets = get_leading($\mathcal{N}$)
 5: save_labels(labels_dict, cofacs, leading_mets)
 6: extend_labels_to_isomers($\mathbf{R}$)
 7: while not every reaction has leading metabolites in left and right do
 8:     next_supporting = get_met_highest_cost($\mathcal{N}$, $\mathbf{R}$)
 9:     del_mets_from_N($\mathcal{N}$, next_supporting)
10:     leading_mets = get_leading($\mathcal{N}$)
11:     save_labels(labels_dict, cofacs, leadin_mets)
12:     extend_labels_to_isomers($\mathbf{R}$)
13: end while.
```

The pathway finding algorithm.

Similarly to Algorithm 8 described above, Algorithm 9 maximises the similarity between metabolites. Starting from a user-selected substrate metabolite, the algorithm generates a linear pathway, where, at each step, the product metabolite with the highest similarity to the substrate is selected to be added to the pathway (or vice-versa in the case that a product is chosen) (Steps 6 and 7 in Algorithm 9). This algorithm terminates when an external metabolite is reached, or the length of the path reaches a user-specified maximum.

Note that this algorithm does not take the directionality of reactions into account. Furthermore, groups of connected metabolites that are identified to have an identical elemental composition ratio by the similarity measure will be connected to each other in the pathway produced by the algorithm.

Algorithm 9 *Pathway_finding(met,* **R***, $\mathcal{N}$, max_iters = 10)*, which when applied to a starting metabolite, a correlation matrix, **R**, and stoichiometry matrix, $\mathcal{N}$, finds a linear path starting from *met* where at each step, the next metabolite in the pathway is chosen on the basis of maximising similarity.

```
path = [met]
while (path[-1] is not external_met) and (path[-1] is not in path[:-1]) and (len(path) < max_iters) do
    prev_met = path[-1]
    opposite_mets_list = get_opposite_to_met_in_reacs($\mathcal{N}$, prev_met)
    next_met = find_most_similar_to(opposite_met_list, prev_met)
    path.append(next_met)
end while
return path
```

5.3.4 Implementation

The algorithms described within this chapter are implemented as part of three modules.

The first module, contains a class called *LeftNS* that contains a left null-space basis and associated correlation matrix. The second, *AtomicMatrix* retrieves atomic matrices from the metacyc database and implements the *Extend* algorithm. Finally the *ClassifyMets* module implements the network pruning and pathway finding algorithms.

New instances of *LeftNS* are initiated by giving a stoichiometry matrix as an input. A left null-space basis is automatically calculated using an existing ScrumPy SVD algorithm that forms part of the Scipy[1] library. Since the work presented in this chapter is derived from previous methods applied to the right null-space by Poolman et al. [2007], a function that calculates the similarity measure, associated tree, and correlation matrix were existing in ScrumPy. As part of this work, these functions were combined into a class called *diffMtx* that contains a correlation matrix with the following additional attributes:

- *GetIndep():* returns the metabolites that are orthogonal to each other,
- *GetParallel():* returns that metabolites that are parallel to each other,

[1] scipy.org

- *GetTree():* returns a tree object that displays the correlation between metabolites using the WPGMA algorithm as implemented in ScrumPy and described by Poolman et al. [2007] and Morgan and Ray [1995], such that the greater the correlation between any two metabolites, the closer they are found within the tree.

The atomic matrix was initially constructed as a ScrumPy DataSet instance, which is then populated using existing ScrumPy functions that retrieve information from the MetaCyc database. This matrix was then extended using an ILP, defined using the cvxopt[2] Python library (which is itself an interface for the GNU Linear Programming Toolkit, glpk[3]). The ILP is implemented as part of a class that contains internal functions for defining the ILP problem and finding unbalanced reactions. This class, called *ExtendAtomic*, is a child class of *LeftNS* and requires an incomplete atomic matrix and an external stoichiometry matrix instance as an input.

The *classifyMets* module contains a class that includes the following attributes:

- *smx*: a stoichiometry matrix instance, $\mathcal{N}$, that is simplified by the network pruning algorithm,
- *diffMtx*: a *diffMtx* instance that specifies the similarity between metabolites,
- *ClassifyMets()*: an implementation of the network pruning algorithm that classifies all metabolites in $\mathcal{N}$ as leading or supporting and accordingly simplifies $\mathcal{N}$.
- *FindPathway(met)*: an implementation of the pathway finding algorithm that obtains a linear pathway starting from the metabolite *met*.

In addition, molecular-fingerprint encodings for metabolites were calculated from SMILES data using the Chem module of the RDkit[4] chem-informatics Python library, and correlated using the Tanimoto similarity metric.

[2] cvxopt.org

[3] gnu.org/software/glpk/

[4] rdkit.org/docs/source/rdkit.Chem

Table 5.2: The sub-models of *C. jejuni* used as described in this chapter.

Model	Reactions	Metabolites	Inputs	Outputs
Biomass Sub-Model	319	365	30	61
PLP Sub-Model	47	63	6	5

5.4 Results

The methods described in this chapter were applied to the Calvin cycle, simplified plant, photorespiration and *C. jejuni* models listed in Table 2.1 and described in Chapter 2.5.2.

The models of the Calvin cycle, simplified plant, and photorespiration are small, whilst the model of *C. jejuni* is genome scale. Additionally, two sub-models were extracted from the model of *C. jejuni*, as listed in Table 5.2, and analysed. One of these models contains the reactions present in an FBA solution that produces all biomass components, and *the* other, smaller, model contains the reactions in an FBA solution that produces only PLP (obtained as described in Chapter 3.4.2.1).

All metabolites of the Calvin cycle and *C. jejuni* models are within one compartment (the stroma in the case of the Calvin cycle). The photosynthesis model is compartmentalised into the cytosol, chloroplast, peroxisome, and mitochondria, where chemically identical metabolites are assigned different identifiers based on the compartment in which they are located, and some are allowed to travel between compartments via transporter reactions. Similarly, the simplified plant model includes two chloroplast compartments that each contain identical copies of the Calvin cycle model (but whose internal metabolites are labelled differently to be distinguished), the cytosol, and an amyloplast (site of starch storage and synthesis within the roots), as shown by Figure 5.6. The only metabolites exchanged between the chloroplasts and the cytosolic environment are PGA and phosphate.

5.4.1 Summary of Results

This section summarises the analysis of all the models in Table 5.2. The results are then expanded upon within the context of each specific model in the following subsections.

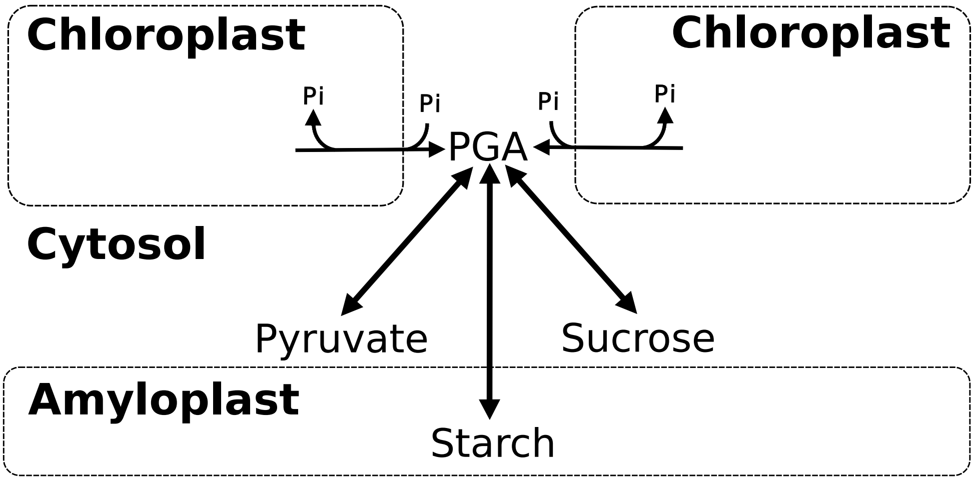


Figure 5.6: The simplified plant model. The chloroplast and amyloplast are organelles (subcellular structures enclosed by their own membrane) that perform specific functions within a cell, while the cytosol refers to the fluid between the cell membrane and the organelles. This model contains the Calvin cycle in the chloroplasts, glycolysis and sucrose synthesis in the cytosol, and starch synthesis in the amyloplast. Derived with permission from Poolman et al. [2007].

The similarity measure.

To investigate the relation between the external left null-space and chemical similarity, the models' metabolites were compared on the bases of their correlation in the:

1. orthonormal internal and external left null-space bases,
2. atomic matrix, and
3. molecular-fingerprint encoding (Calvin cycle only).

The similarity measure applied to the rows of **A**, ϕ^A, compared metabolites according to their empirical formula, while molecular-fingerprint encoding (Section 5.3.2.4) related metabolites by the structural motifs that they have in common [Brown et al., 2005].

These relationships were shown as a tree having metabolites as leaves, see Section 5.3.1, where the more similar two metabolites are to one another, the closer they are found together in the tree.

The information gained from the left null-space of the internal stoichiometry matrix was also investigated. However, the analysis that follows uses $\mathcal{G}$ (unless stated otherwise).

The rank of the matrices **G**, $\mathcal{G}$, and **A** are listed in Table 5.3, while the number of

Table 5.3: The rank of the left null-space and atomic matrices of the models used within this chapter, where Algorithm 7 was used to calculate **A** for all models except for the complete model of *C. jejuni*, whose **A** is comprised of only database records.

Model	Rank(G)	Rank($\mathcal{G}$)	Rank(A)
Calvin cycle	2	8	5
Simplified plant	6	13	5
Photorespiration	20	26	6
C. jejuni	129	146	13
C. jejuni biomass sub-model	6	46	7
C. jejuni PLP sub-model	6	16	6

Table 5.4: The number of groups of parallel metabolites and pairs of orthogonal metabolites identified from **G** and **A** of the models listed.

	Parallel Groups		Orthogonal Pairs	
Model	$\mathcal{G}$	A	$\mathcal{G}$	A
Calvin cycle	4	5	0	1
Simplified plant	5	7	64	0
Photorespiration	21	27	630	127
C. jejuni	144	131	2317	5660
C. jejuni biomass sub-model	74	78	1086	674
C. jejuni PLP sub-model	10	14	0	24

parallel/independent metabolites identified by the similarity measure applied to $\mathcal{G}$ and **A** are in Table 5.4.

The *Extend* algorithm.

Algorithm 7 was applied to all models except for the complete model of *C. jejuni* (due to efficiency limitations of the algorithm). The results are in Table 5.5.

The network pruning and pathway finding algorithms.

The network pruning algorithm, described in Section 5.3.3 was applied to all models listed in Table 5.2. The number of metabolites classified as supporting or leading when using $\mathcal{G}$ and **A** are listed in Table 5.6.

In addition, to determine whether the pathway finding algorithm can successfully extract chemically similar connected metabolites from models, this algorithm was used to generate linear pathways starting from specific external metabolites. The biological relevance of

Table 5.5: The number of metabolites in: (i) the external stoichiometry matrix, $\mathcal{N}$, (ii) the atomic matrix obtained from the metacyc database, $\mathbf{A_{db}}$, and (iii) the atomic matrix obtained after applying Algorithm 7 to $\mathbf{A_{db}}$, $\mathbf{A} = \mathit{Extend}(\mathbf{A_{db}})$, for the models listed.

model	**Number Of Metabolites**		
	$\mathcal{N}$	$\mathbf{A_{db}}$	$\mathbf{A}$
Calvin	28	28	N/A
Simplified plant	77	60	73
Photorespiration	113	103	107
C. jejuni	1105	881	N/A
C. jejuni biomass sub-model	365	253	325
C. jejuni PLP sub-model	63	50	57

Table 5.6: The number of leading and supporting metabolites identified from $\mathcal{G}$ and $\mathbf{A}$ of the models listed.

	Leading Metabolites		**Supporting Metabolites**	
Model	$\mathcal{G}$	$\mathbf{A}$	$\mathcal{G}$	$\mathbf{A}$
Calvin cycle	18	18	10	10
Simplified plant	56	53	21	24
Photorespiration	60	53	53	60
C. jejuni biomass sub-model	173	142	192	223
C. jejuni PLP sub-model	30	33	33	30

these pathways was evaluated using criteria such as the presence of chemical moieties that are shared amongst the constituent metabolites and the pathways' resemblance to known synthesis routes of metabolic products.

5.4.2 The Calvin Cycle Model

The Calvin cycle model contains two moieties that are conserved within internal loops: ADP and phosphate (see Section 5.5.1). Therefore, there exists an internal left null-space basis that contains two atomic vectors as columns: one detailing the presence of ADP in every metabolite (i.e. a coefficient of one for ADP and ATP and zero elsewhere), and another showing the presence of phosphate (i.e. a coefficient of one for phosphates, two for bi-phosphates, and zero elsewhere). As a result, the phosphate moiety and the cycle's sugar-phosphates were all parallel with each other in $\mathbf{G}$.

The similarity between the metabolites of the external left null-space are presented as trees in Figures 5.7, 5.8, and 5.9, where each figure shows the similarity in $\mathcal{G}$, $\mathbf{A}$, and molecular-fingerprints respectively. All of the isomers (and compounds with an identical elemental ratio) were perfectly correlated both in the external left null-space and atomic matrix. However, the internal trios-phosphate sugars were weakly correlated with their external counterparts in the left null-space. A potential reason for this observation is that the impact of internal phosphate conservation led to the internal metabolites being more correlated with each other than with the external metabolites of the network.

The *Extend* algorithm.
To verify the output of the *Extend* algorithm (Algorithm 7), a complete atomic matrix, $\mathbf{A}$, was manually built for the Calvin cycle model.

A number of metabolites were randomly selected and removed from $\mathbf{A}$ to create an incomplete atomic matrix, $\mathbf{A}$. Then, the *Extend* algorithm, along with a traditional algorithm that iteratively calculates the composition of unknown metabolites by balancing the reactions that have only one unknown (terminating when all reaction of the model that contain unknown metabolites have at least two unknowns) were applied to $\mathbf{A}$. The results of both algorithms, shown in Table 5.7, were confirmed to be consistent with the original $\mathbf{A}$. Furthermore, the *Extend* algorithm consistently identified a greater/equal

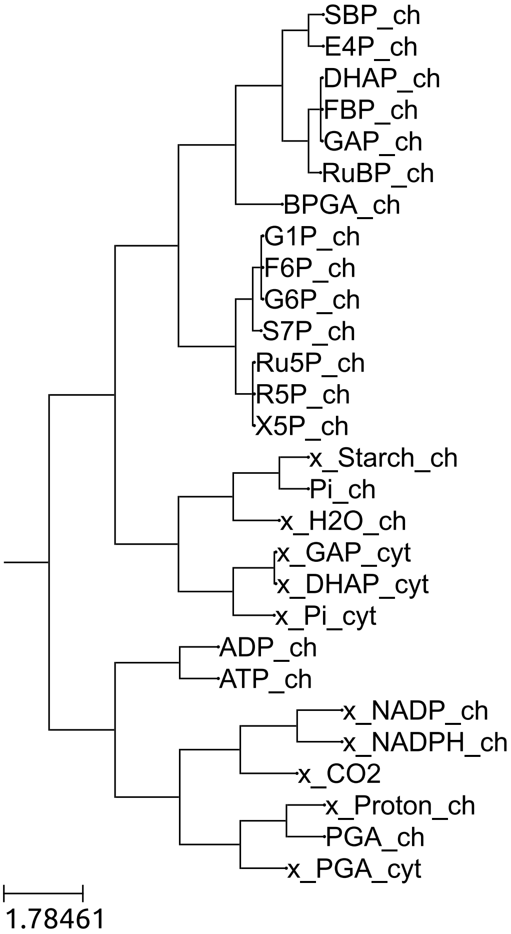


Figure 5.7: The tree obtained from the left null-space of the Calvin cycle Model, where the greater the similarity between two metabolites, the smaller the distance between them in the tree. The external sugar metabolites are clustered together, as are the co-factor pairs. The internal sugar metabolites of the cycle (except for PGA) can be found at the centre of the tree where isomers are perfectly correlated. See Figure 2.7 for the original model.

number of metabolites then the traditional algorithm.

Furthermore, to assess Algorithm 7's ability to identify unbalanced reactions, the model of the Calvin cycle was modified such that some reactions were unbalanced. Then, the algorithm was applied to incomplete atomic matrices in a similar process to that described above. This algorithm identified the presence of unbalanced stoichiometry in all cases, however, the specific reactions that were unbalanced were not always correctly identified (seen here in iterations that contained a large number of unknown metabolites). This result occurred since the identification of unbalanced reactions is dependent on the order in which reactions are added to Equation (5.18). As a consequence, although this algorithm can detect the presence of unbalanced reactions in models, it is not guaranteed to find

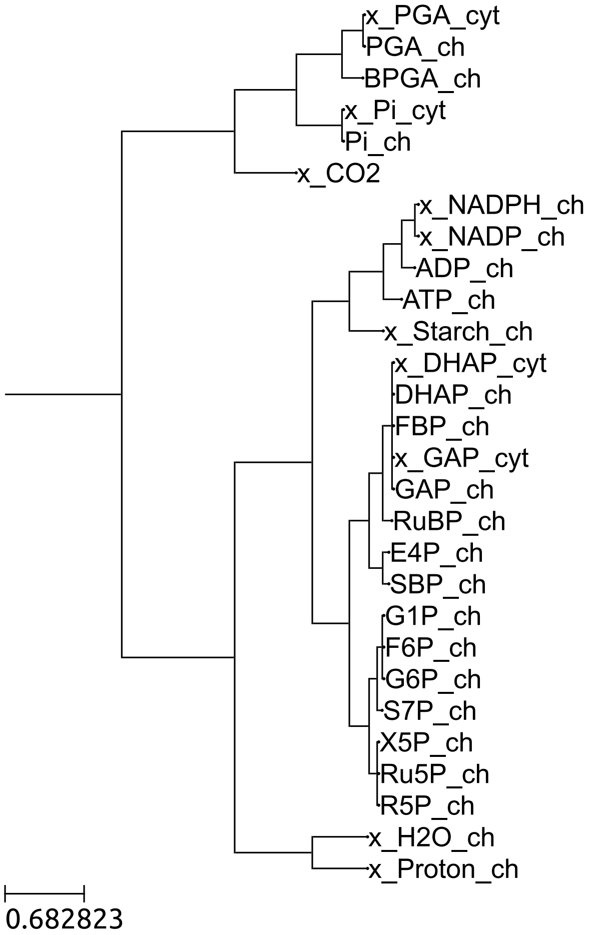


Figure 5.8: The tree obtained from the atomic matrix of the Calvin cycle Model. In contrast to the tree obtained from the left null-space, isomers are perfectly correlated with each other irregardless of whether they are internal or external in the model. See Figure 2.7 for the original model.

the reactions that cause this unbalance.

The network pruning and pathway finding algorithms.

The results of the network pruning algorithm were identical both when using $\mathcal{G}$ and $\mathbf{A}$, where the metabolites retained within the reduced model were all of the carbon sugars with the exception of E4P, as shown by Figures 5.10 and 5.11.

As described in Section 2.5.2, the Calvin cycle synthesises and stores starch during periods of high light intensity. When less light is available, the rate of CO_2 assimilation via RuBisCo is reduced, which prompts starch to be degraded in order to supplement the cycle's production of triose-phosphate sugars with additional carbon.

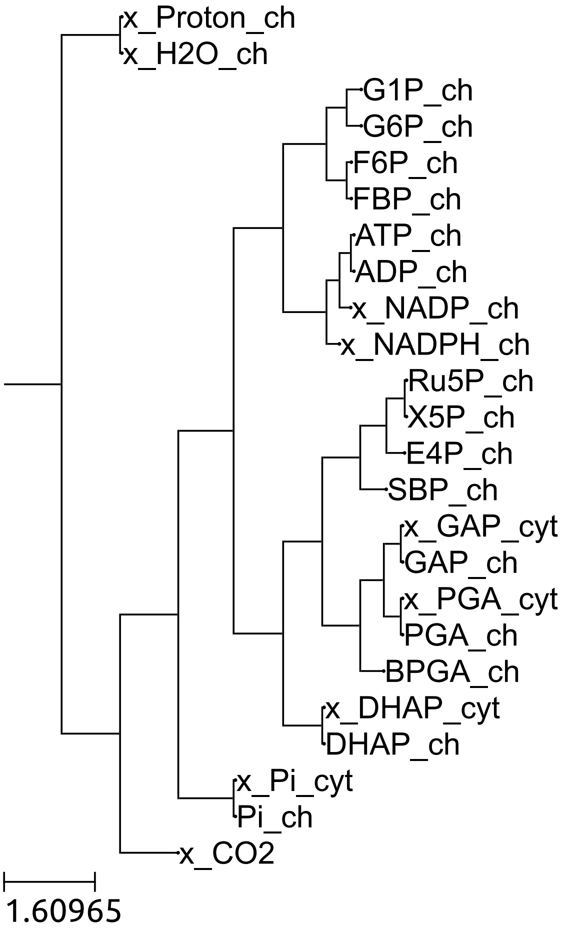


Figure 5.9: The tree obtained from the fingerprint similarity of the metabolites in the Calvin cycle Model, illustrating the similarity between the chemical structure of the metabolites. See Figure 2.7 for the original model.

This was investigated by using the pathway finding algorithm to generate pathways that start from PGA, using both $\mathcal{G}$ and $\mathbf{A}$. This resulted in different paths that linked PGA to external starch (see Figures 5.12 and 5.13, respectively), where the external left null-space extracted sugars from the reductive branch of the Calvin cycle, whilst the atomic matrix extracted sugars from the regenerative branch of the cycle. The reason for this difference is that PGA was most similar to BPGA in the left null-space, and RuBP in the atomic matrix.

5.4.3 The Simplified Plant Model

Similarly to the results described above, the internal left null-space contained basis vectors that show the conservation of phosphate and ADP/NAD moieties. Phosphate was

Table 5.7: The number of metabolites in (i) an incomplete atomic matrix, (ii) the output of a traditional *Extend* algorithm applied to **A**, and (iii) the output of the novel *Extend* algorithm applied to **A**. The incomplete atomic matrices were created by randomly removing five, ten, fifteen, and, twenty metabolites for each case. This process was repeated three times and the average output of the algorithms is shown.

Number Of Metabolites		
Incomplete A	***Traditional*(A)**	***Extend*(A)**
23	28	28
18	28	28
13	26	27
8	17	20

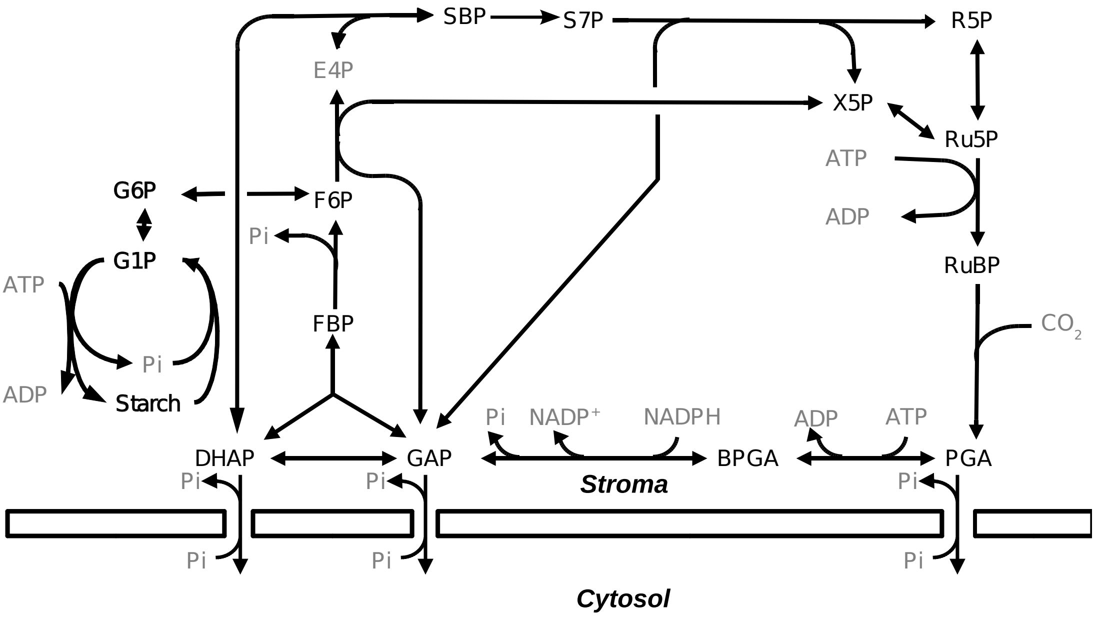


Figure 5.10: The Calvin cycle model, where metabolites identified as supporting are coloured in grey (including all currency metabolites and erythrose-4-phosphate). See Figure 2.7 for the original model.

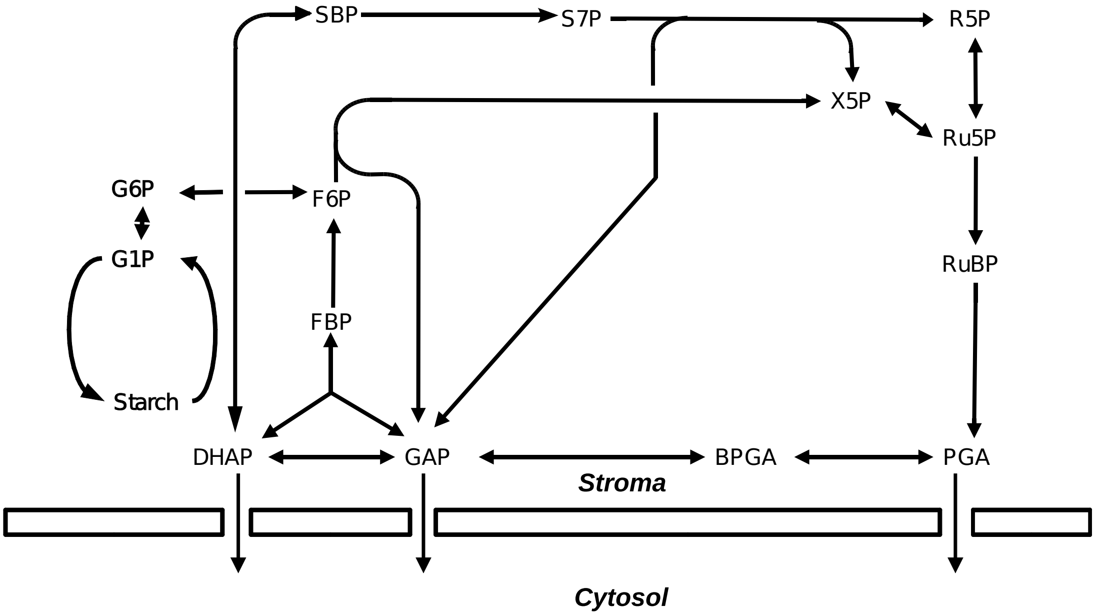


Figure 5.11: The simplified Calvin cycle model, obtained after supporting metabolites are removed, leading to the linearisation of some reactions. See Figure 2.7 for the original model.

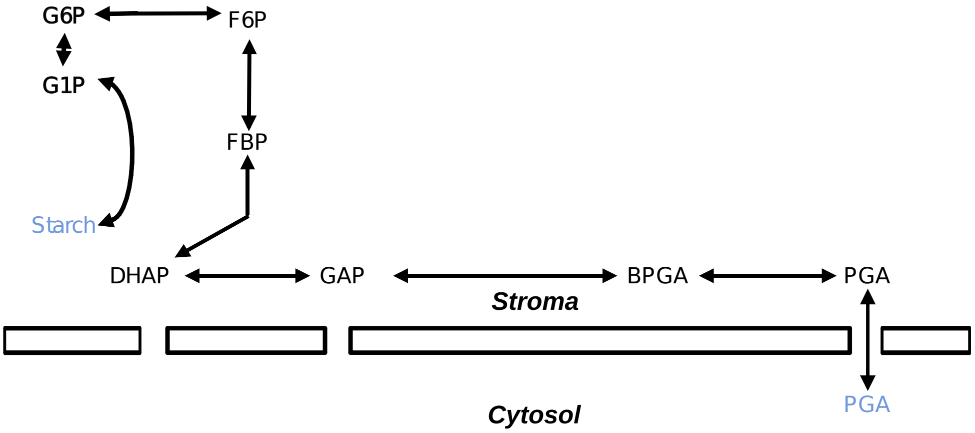


Figure 5.12: A pathway linking PGA to starch in the Calvin cycle model, obtained by from the pathway finding algorithm and $\mathcal{G}$. See Figure 2.7 for the original model.

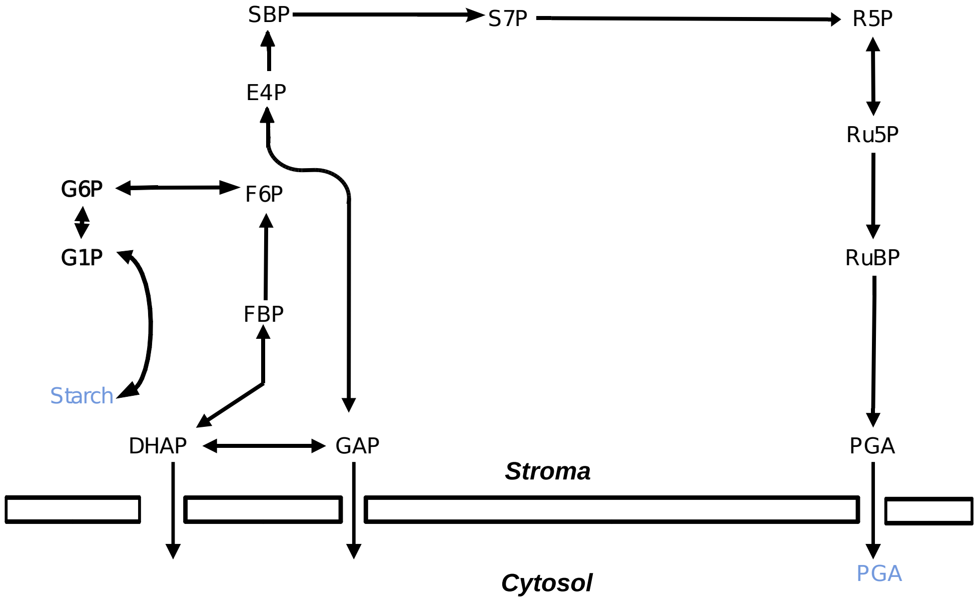


Figure 5.13: A pathway linking PGA to starch in the Calvin cycle model, obtained from the pathway finding algorithm and **A**. See Figure 2.7 for the original model.

separately conserved as part of the Calvin cycle in the two respective chloroplasts, and glycolysis in the cytosol and amyloplast. This led to three groups of parallel sugar-phosphate metabolites.

Comparison of the trees obtained from **A** and $\mathcal{G}$ showed that the external left null-space was highly influenced by compartmentalisation, where identical carbohydrates present in three different compartments (a Calvin cycle in each the two chloroplasts and glycolysis in the cytosol), were more correlated with similar carbohydrates in their own compartment, than their intra-compartment counterparts in $\mathcal{G}$. As with the Calvin cycle, this result is likely to have been caused by the conservation of internal phosphate (see Section 5.5.1).

All isomers were perfectly correlated within compartments; however, pyruvate in the cytosol was parallel to fructose, maltose, sucrose, glucose, and starch in $\mathcal{G}$, despite the fact it does not have the same elemental composition ratio as the latter metabolites. This relation was caused by a pyruvate kinase reaction that was not balanced for protons in the model.

Upon identification of the orthogonal metabolites, NAD and NADH in the cytosol were orthogonal to all metabolites in the model except for each other. In the model, these metabolites were only involved in an internal cycle where NADH is oxidised by GAP

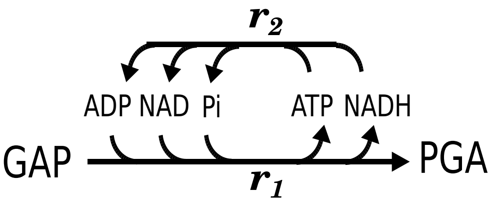


Figure 5.14: An internal cycle involving the oxidization of NADH (where r_1: merger of phosphoglycerate kinase & GAP dehydrogenase; r_2: oxidative phosphorylation). Note that this cycle emerged since the stoichiometry of r_1 is incorrect.

dehydrogenase and subsequently regenerated by the ETC. This cycle emerged since the stoichiometry for GAP dehydrogenase was incorrectly merged with that of PGA kinase as shown in Figure 5.14.

The *Extend* algorithm.

23 reactions were identified as unbalanced within this model. Most of these involved water and protons, which was to be expected since the conservation of these metabolites was not taken into account when the model was constructed (note that, since these metabolites are external, their omission did not impact the results of analysis derived from the internal stoichiometry matrix, for which the models were originally developed). Other reactions regarded the synthesis/degradation of ATP or starch.

As a consequence of these unbalanced reactions, the majority of the metabolite compositions calculated by the *Extend* algorithm were incorrect.

The network pruning and pathway finding algorithms.

Applying the network pruning algorithm returned a simplified model containing the carbohydrates of the Calvin cycle as shown in Figure 5.11, the sugars of glycolysis, and the synthesis/degradation of starch in the amyloplast.

The results for $\mathcal{G}$ and $\mathbf{A}$ were identical except for ATP/ADP and phosphate in the amyloplast, which were retained in the simplified model when using $\mathcal{G}$.

The pathway finding algorithm was used to investigate the formation of starch in the chloroplasts and amyloplast. Seeking a pathway starting from external starch in the amyloplast extracted the non-phosphorylated hexose sugars within this compartment, by creating a cycle from starch to maltose, glucose, and finally starch once more, whereas, pathways starting from starch in the chloroplasts grouped the sugar-phosphates of the cycle (excluding DHAP) and phosphate.

The results of Algorithm 9 were identical for both $\mathcal{G}$ and $\mathbf{A}$.

5.4.4 The Photorespiration Model

As in the previous models, the internal left null-space exposed the internal conservation of moieties in the model. This included ADP in the Calvin cycle, photorespiration, and TCA cycle.

In addition, the metabolite similarity associated with the external left null-space was clustered in a way that reflects the compartmentalised nature of the model (see Figure 5.15). For example, isomers in the same compartment were parallel in $\mathcal{G}$, while there were several occurrences of identical metabolites present in different compartments that were unrelated in $\mathcal{G}$, including NAD/NADH and ADP/ATP.

This model contained 630 pairs of metabolites that were orthogonal to each other within the left null-space. All pairs involved phosphate and ADP/ATP within the cytosol and mitochondria, which form part of an internal cycle that synthesises ATP within the TCA cycle and consumes it within the cytosol, see Figure 5.16.

The *Extend* algorithm.
Eighteen reactions were identified as unbalanced in this model. Similarly to the simplified plant model, the majority of these were caused by the omission of protons and water. Others were caused by the inclusion of reactions that include *phantom* metabolites to simulate the cell's ATP requirements. Finally, three reactions were falsely identified as unbalanced due to the metabolite glycine being incorrectly defined in $\mathbf{A_{db}}$.

Four additional metabolites where assigned a composition by Algorithm 7. These were S7P, RuBP, ubiquinol, and a carbohydrate polymer. The calculated composition of S7P was verified to be correct. However, proton unbalancing in the model lead to the composition of RuBP to be incorrect. Furthermore, the composition of ubiquinol was incorrect as its oxidised form, ubiquinone, was incorrectly defined in $\mathbf{A_{db}}$.

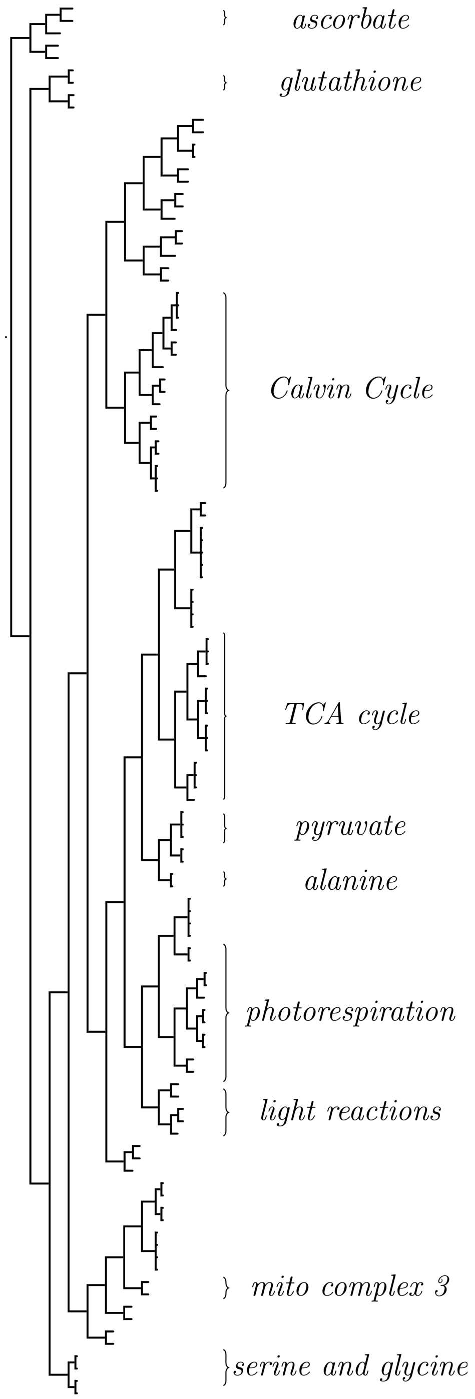


Figure 5.15: The tree obtained from the left null-space of the photorespiration model. The modular nature of this tree is emphasised by labelling groups of metabolites that form part of particular metabolic processes. See Figure 2.8 for the original model.

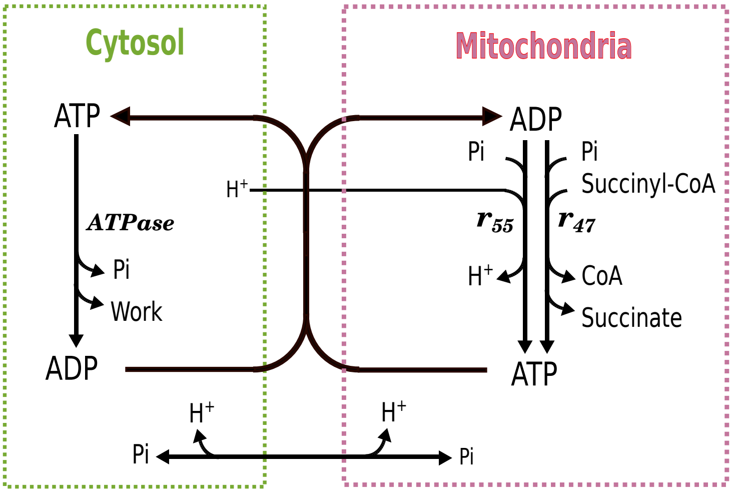


Figure 5.16: An internal cycle where ATP hydrolysed in the cytosol is replenished by ATP synthesis in the mitochondria. As a consequence of this cycle, the molar amounts of ADP, ATP, and phosphate are independent from the other metabolites in the model. See Figure 2.8 for the original model. Note that EM analysis of this model by Huma et al. [2018] showed that ATP hydrolysis is actually driven by the creation of NADH from the conversion of glycine to serine (r_{46} in Figure 2.8) and not by r_{47}, which is inactive.

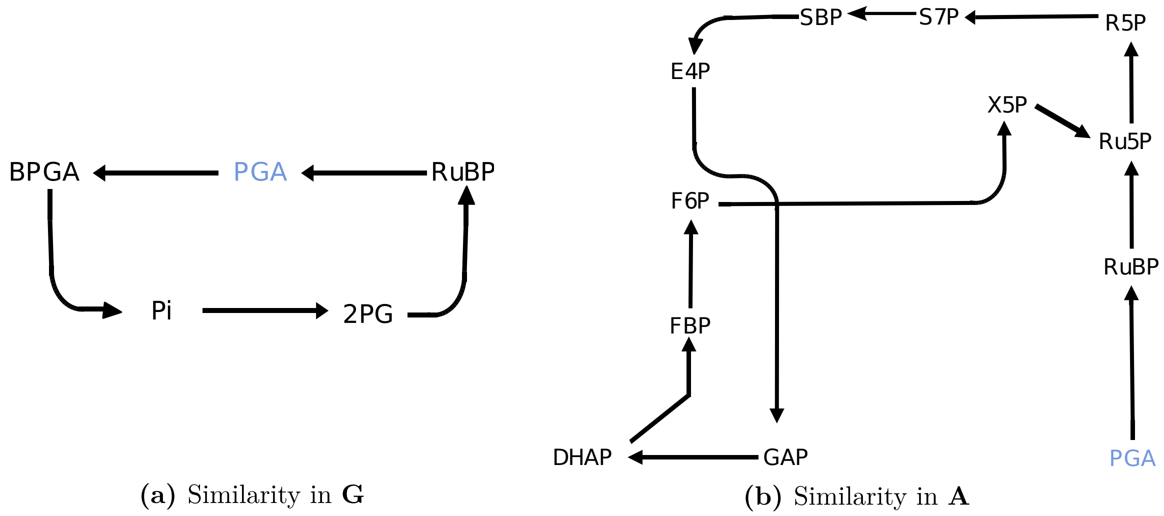
The network pruning and pathway finding algorithms.

When applying the network pruning algorithm, the results from **G** and **A** differed. Metabolites identified as leading in both cases included most carbon sugars of the Calvin cycle, metabolites involved in nitrogen metabolism, ATP in the chloroplasts, and oxygen moieties in some compartments. Common discarded metabolites included NAD and ATP in the mitochondria, protons, and CoA. Some amino acids were not retained in the simplified model obtained from **G**, but were so when **A** was used. While **G** identified ammonia and phosphate as supporting, which was not the case in when **A** was used to establish similarity.

In this case, **A** was incomplete (as listed in Table 5.5), and so the total set of metabolites classified was different for both similarity measures.

Similarly to the model of the Calvin cycle, the pathway finding algorithm was used to obtain pathways starting from PGA. As shown in Figure 5.17, $\mathcal{G}$ led to a short cycle of metabolites that share a phosphate moiety, whilst **A** produced a cycle that grouped similar sugar-phosphates within the Calvin cycle, and similarly shows the conservation of phosphate.

Note that the pathway obtained from $\mathcal{G}$ included a free phosphate since this molecule had



(a) Similarity in **G**

(b) Similarity in **A**

Figure 5.17: The paths identified by the pathway finding algorithm, when starting with PGA in the photorespiration model, both of which highlight the conservation of phosphate. See Figure 2.8 for the original model.

a higher similarity with BPGA than with GAP within $\mathcal{G}$. This is chemically accurate since the elemental composition ratio of BPGA is also more similar to that of free phosphate within **A**.

5.4.5 The *C. jejuni* Model

This model had 131 groups of parallel metabolites in the internal left null-space, reflecting the extensive scale of this model. The biomass and PLP sub-models both contained six such parallel groups, including the conservation of electron carrier moieties, such as cytochrome, menaquinone, and ferrodoxin. In addition, the moieties ADP, NAD, and Co-A were conserved in the PLP sub-model. While, acyl-carrier-proteins, a co-factor of fatty acid biosynthesis, were conserved in the biomass sub-model.

All metabolites involved in isomerizations were identified as parallel in the external left null-space. Additionally, groups of metabolites, which although not isomers, must have an identical composition ratio because of the rules of conservation of mass, were also obtained from **G** including metabolites that are co-transported with ions as well as the ions themselves (such as internal and external alanine, internal and external phosphate), and methyl-citrate and methyl-isocitrate which are connected together in the pathway shown by Figure 5.18.

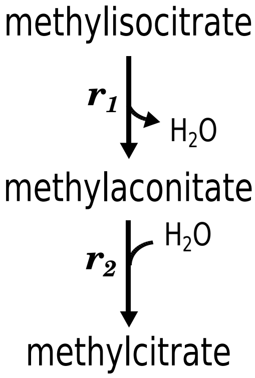


Figure 5.18: A pathway connecting methylisocitrate to methylcitrate, where the two metabolites are identified as having an identical elemental composition ratio in the left null-space of the *C. jejuni* model, where r_1: methylisocitrate dehydrogenase; r_2 methylcitrate dehydratase.

As discussed previously, not all metabolites with an identical composition ratio had proportional rows in $\mathcal{G}$. In fact, 31 groups of metabolites were identified as parallel in **A** but not in $\mathcal{G}$ (taking note that **A** was incomplete). An example is lysine and its external counterpart, which is co-transported into the cell by a reaction that requires the hydrolysis of ATP. Similarly, a group containing hydrogen peroxide, a hydroxide ion, and a hydroxide radical was not identified as parallel in $\mathcal{G}$. This may be since, unlike **A**, $\mathcal{N}$ is also influence by the conservation of electrons (which the above metabolites have a different number of).

Sodium ions (both internal and external) were independent of all other metabolites in the model (both in $\mathcal{N}$ and **A**). These ions are co-transported with metabolites such as protons, proline, and $\alpha-$alanine, to ensure charge conservation in the model. Additionally, the left null-space contained a number of metabolites that were independent of a subset of other metabolites. The most significant was, palmitoleoyl-CoA, a metabolite that acts in reactions that incorporate palmitoleoyl moiety into lipids. This metabolite was independent of 34 metabolites (mostly involving metabolites that contain cysteine, methyl-sulfanyl, or thiohydroximate residues).

The *Extend* algorithm.

Applying the *Extend* algorithm to the complete model of *C. jejuni* was challenging because of the computational requirements of the algorithm. Hence it was only applied to the sub-models obtained from LP solutions.

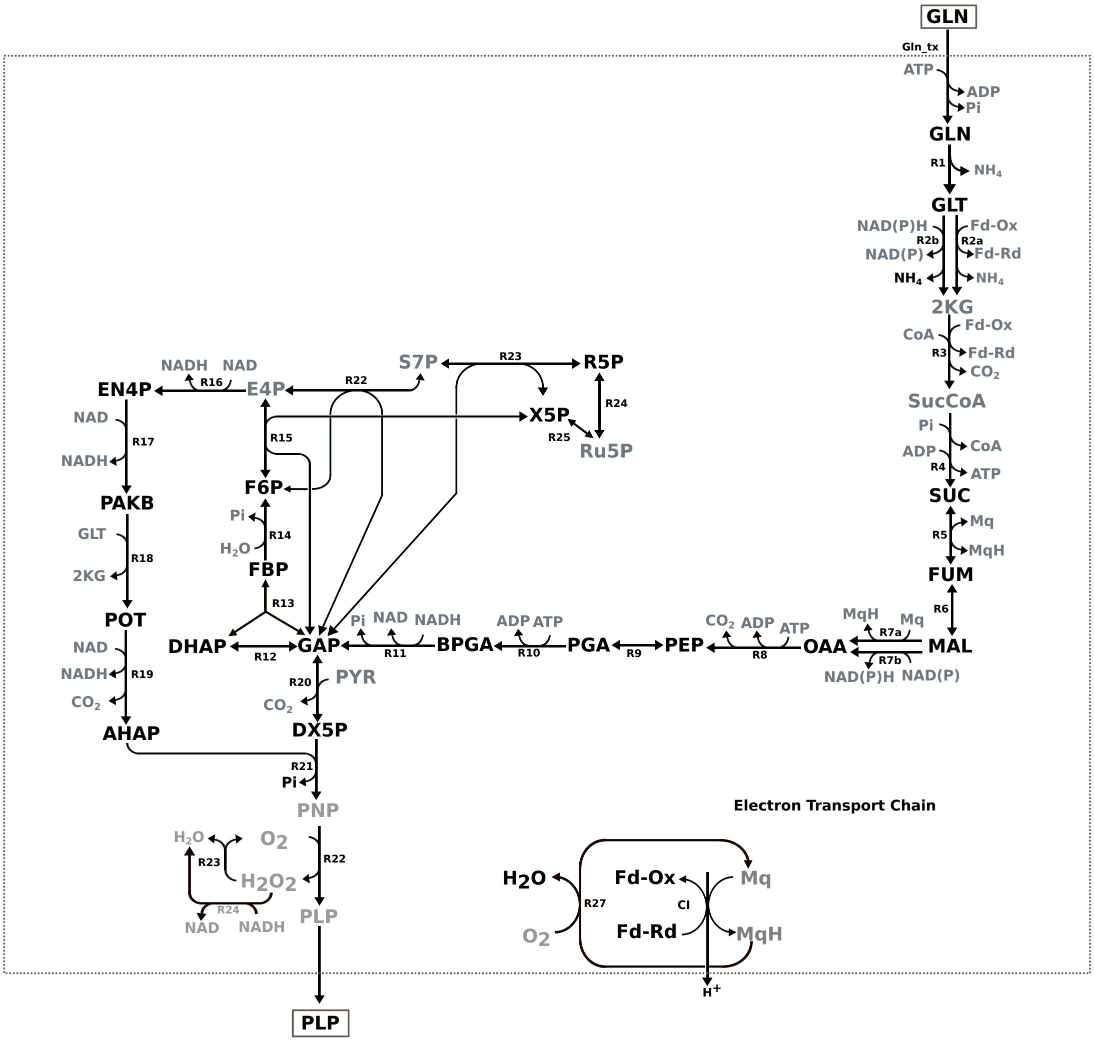


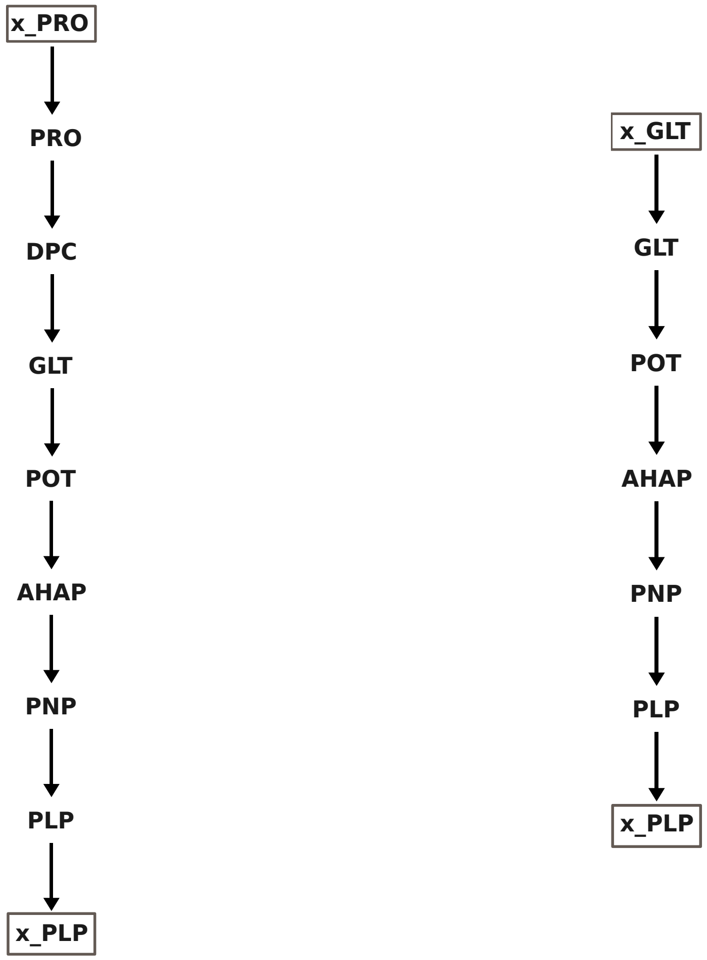
Figure 5.19: The *Network_pruning* algorithm applied to an FBA solution that produces PLP in *C. jejuni*, where supporting metabolites are coloured in grey. See Figure 3.5 for the original model.

The network pruning and pathway finding algorithms.

The network pruning algorithm was applied to the *C. jejuni* sub-models, where different results were obtained for $\mathcal{G}$ and $\mathbf{A}$. In the case of the biomass sub-model, 110 of the leading metabolites were given the same label when using $\mathcal{G}$ and $\mathbf{A}$, and 26 for the PLP sub-model.

The results of this algorithm when applied on the PLP FBA solution using $\mathcal{G}$ is shown in Figure 5.19, where metabolites such as NAD, ATP, CO_2, CoA, and H_2O, pyruvate and E4P were labelled as supporting.

Furthermore, the pathway finding algorithm was applied to the *C. jejuni* model to gener-



(a) PLP submodel

(b) Biomass submodel

Figure 5.20: The paths identified by the pathway finding algorithm, when starting from external PLP, applied to a FBA solutions that produce PLP and Biomass in *C. jejuni.* See Figure 3.5 for the original model.

ate a path that produces PLP. In this case, only $\mathcal{G}$ could be used since the atomic matrix generated for the model did not include PLP as an entry.

This algorithm was similarly applied to the PLP and Biomass sub-models, where the left null-space resulted in the similar pathways shown in Figure 5.20. On the other hand, the atomic matrix led to a pathway that contained sugars of the pentose phosphate pathway for the PLP sub-model, while the results from the biomass sub-model contained additional metabolites, such as some from methylerythritol phosphate biosynthesis pathway.

5.5 Discussion

5.5.1 The Similarity Measure

Conserved moieties in the internal left null-space.

As discussed in Section 5.1, conservation relations in the internal left null-space regard moieties that are conserved within closed loops inside the network. However, these moieties may be difficult to identify when given a set of basis vectors for **G**, since each such vector does not necessarily correspond to the presence of a single moiety but rather a combination of them. The similarity measure can address this challenge since groups of metabolites that are parallel in **G** must contain the same conserved moiety. For example, one such characteristic of the Calvin cycle is that the amount of phosphate inside the cycle remains constant through time. This is as the amount of phosphate entering and leaving the chloroplasts must always be equal (as shown in Figure 5.21). As a consequence, all phosphate-containing metabolites of the cycle were parallel to each other in **G** when investigated above.

Chemical similarity in the external left null-space.

The external left null-space was able to recover some (but not all) of the models' known chemical similarities. In fact, within single compartments, the relationships from the external left null-space were found to be similar to those obtained from the atomic matrix. This is to be expected since, as shown in Section 5.2.2, chemical moiety vectors have an influence on the similarity measure applied to the external left null-space, $\mathcal{G}$. However, the extent of this influence is not guaranteed and also depends on whether reactions in models are correctly balanced. Unbalanced reactions can distort results, as seen by the relation of pyruvate to glucose in the simplified plant model. Moreover, similarities in the empirical formula of metabolites (as measured by **A**) do not necessarily correspond to similar metabolite structures. Indeed, the tree showing fingerprint similarity for the Calvin cycle was different from those obtained from $\mathcal{G}$ and **A**. For example, fructose-6-phosphate and fructose-1,6-biphosphate were loosely correlated in **A** and $\mathcal{G}$, despite the substantial similarity between their structures.

Both the identification of metabolites that are erroneously parallel to others, as well as, metabolites which are orthogonal to others were useful to identify errors in models as shown in Section 5.4.3. Furthermore, the identification of pairs of orthogonal metabolites exposed the presence of internal cycles and metabolites containing unique elements. For example, sodium ions in the *C. jejuni* model are used for co-transport and not incorporated into any of the other metabolites in the model, such that no moiety conservation vector that relates the sodium ions with other metabolites of the network can exist.

The influence of compartmentalisation.

Similarly to when the similarity measure was applied to the right null-space by Poolman et al. [2007], compartmentalisation was found to have a significant impact on the way that metabolites are clustered, which reflects that a matrix's null-space is directly impacted by the connectivity between the elements of the network that it generates, see Section 2.3.2.

For example, some chemically identical metabolites that were present in different compartments were not identified as parallel in the left null-space. These included metabolites that are conserved within compartments or metabolites trapped within one or more compartments due to a lack of a transporter, such as, the non-decomposable ADP moiety in the chloroplast of the photorespiration model which was unrelated to the ADP in other compartments due to a lack of transport between the different compartments. In such cases, although the same metabolite may be present in more than one compartment, lack of material transfer between the compartments would prevent the elemental relationship from being embedded into the stoichiometry matrix and thus in $\mathbf{G}$.

Another example is the lack of a strong similarity between identical sugars in the two chloroplasts of the simplified plant model. This observation is likely to be caused by the conservation of phosphate inside the chloroplasts, a hypothesis that was tested by removing the phosphate conservation from the model (achieved by allowing phosphate to flow freely between the chloroplasts and the cytosol), which led to the identical sugars emerging as parallel in the left null-space.

It was noted that although $\mathbf{A}$ is not constructed from the stoichiometry matrix, the tree associated with it displays some modularity similar to that found in $\mathcal{G}$ since, due to conservation rules, metabolites are more likely to share reactions (and thus be within the same pathway in $\mathcal{N}$) with similar compounds rather than with ones that are highly

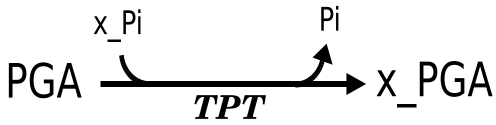


Figure 5.21: The export of PGA balanced by the import of phosphate, where TPT: triose-phosphate translocator.

different.

5.5.2 The Atomic Matrix

The tools provided here can facilitate the determination of the approximate elemental composition of metabolites that are unknown. Furthermore, apart from calculating the atomic composition of metabolites, the *Extend* algorithm was able to identify models that are not correctly balanced, therefore facilitating model curation. This method improves upon a technique previously developed by Gevorgyan [2009, pages 89-93], who used the left null-space to determine whether all metabolites in a network can be assigned positive mass. In the work presented in this thesis, available information on the elemental composition of metabolites is also considered to further restrict the criteria for model correctness.

5.5.3 The Network Pruning Algorithm

Network pruning algorithms have been historically used to facilitate the application of techniques from graph theory to models [Gerlee et al., 2009], and to aid the graphical visualisation of pathways [Zhou and Nakhleh, 2011]. A common strategy is to eliminate currency metabolites, identified by finding metabolites that are involved in a large number of reactions.

When the pruning algorithm was used to simplify the Calvin cycle and simplified plant models, the metabolites removed were all currency metabolites, therefore in agreement with historical applications of network pruning. This observation was challenged when applying the algorithm to the photorespiration model where some currency metabolites were classified as leading. For example, such a labelling of ATP in the chloroplasts was caused by this metabolite pertaining to a reaction that represents the light reactions of photosynthesis, which contains ATP as the sole product. Similarly, protons in the mitochondria were classified as leading directly from the stoichiometry matrix. This was because of being involved in ETC reactions that consisted of protons and metabolites (such as ATP) that could be classified as supporting from the stoichiometry matrix. This led to Succinyl-CoA in the mitochondria being classified as supporting, which simplified an $\alpha-$ketoglutarate dehydrogenase reaction into one that converts $\alpha-$ketoglutarate into protons, which is not biologically relevant. Similarly, a reaction involving the degradation of hydrogen peroxide led to this metabolite being classified as leading, which in turn led to an ascorbate oxidase reaction being simplified into one that converts hydrogen peroxide into dehydroascorbate (rather than ascorbate into dehydroascrobate).

Moreover, applying the algorithm to FBA solutions facilitated their comprehension, leading to the simple illustration of PLP synthesis in Figure 5.19. However, direct comparisons between the results obtained from $\mathcal{G}$ and $\mathbf{A}$ is difficult because of the use of incomplete atomic matrices. This meant that some metabolites were considered to be missing from the models when using $\mathbf{A}$. For example, pyridoxine, the precursor of PLP was unaccounted for in the *C. jejuni* model, hence the results of the pathway pruning algorithm using $\mathbf{A}$ were not relevant.

5.5.4 The Pathway Finding Algorithm

The results obtained from the pathway finding algorithm were inconsistent. Application to the Calvin cycle and simplified plant model highlighted the plants' ability to interconvert similar sugars in processes that synthesise, store, and degrade starch. However, the pathway obtained from the external left null-space of the photorespiration model was less biologically relevant as it documented the exchange of free phosphate between four metabolites of the model (Figure 5.17).

Furthermore, while the left null-space of the sub-models of *C. jejuni* retrieved some of the initial steps involved known PLP synthesis pathways (as illustrated in 3.5.2.4), the insights that could be gained the $Network\_pruning$ algorithm revealed more information and therefore were seen to be more suitable for analysis.

5.6 Conclusion

The theoretical results of this chapter show that the left null-space $\mathbf{G}$ is useful for identifying invariant relationships between concentration changes of metabolites. Furthermore, a novel method that can use the external left null-space to relate the elemental composition of metabolites is presented, which also facilitates model curation by identifying errors such as reactions that do not satisfy the law of conservation of mass. Finally, two algorithms that can simplify models and FBA solutions by extracting metabolites that are most similar.

The associated algorithms will be incorporated into the ScrumPy source tree in due course.

Chapter 6

Conclusion

As, over the past decades, the understanding of cellular metabolism has been steadily increasing, so too has the size and complexity of metabolic models. Although this scale permits a more realistic portrayal of metabolic behaviour, the size itself makes the analysis of these models more difficult, a problem that motivated the work presented in this thesis. The primary aim of this research is to explore novel methods to extract information from the stoichiometric structure of networks (structural modelling), which along with aspects from more established methods can generate useful insights whilst still being applicable to larger models.

This aim is achieved using three different approaches: extracting fundamental flux pathways from the network's architecture (Chapter 3), exploring methods that reduce the size of models (Chapter 4), and establishing relationships between the concentration and composition of metabolites (Chapter 5). All of these techniques rely on the analyses of the right and left null-spaces of the stoichiometry matrix. They have been further shown to be useful in model curation as well as the subsequent analysis.

An overview of the research work presented in this thesis is given below, including the primary results and findings, their novelty, and their relevance to the research field.

6.1 Summary of Methodology and Results

Chapter 3 describes a method that decomposes flux measurements into a set of EMs with desirable properties. This task was first achieved by Poolman et al. [2004b], who designed an algorithm that requires the entire set of EMs of a network to be calculated *a priori* and, therefore, cannot be applied to GSMs. In contrast, the algorithm presented in this thesis has a much smaller computational burden, making it suitable for large-scale models (hundreds of reactions).

The algorithm decomposes a steady-state flux vector, $\mathbf{v}$, into constituent EMs, via a sequence of iterations that produce EMs. Every iteration solves an LP that extracts a sub-vector, $\mathbf{v}'$, from $\mathbf{v}$. This LP is defined such that (i) the reaction with the smallest flux in $\mathbf{v}$, $v_{\min}$, has the same flux in $\mathbf{v}'$ (i.e. $v_{\min} = v'_{\min}$), and (ii) $\mathbf{v}'$ contains a much smaller number of reactions than $\mathbf{v}$. The second requirement increases the likelihood that $\mathbf{v}'$ is an EM. If this is not true, then $\mathbf{v}'$ is efficiently decomposed into EMs by calculating the EMs of a small sub-model that only includes the reactions of $\mathbf{v}'$ (as discussed in Theorem 3.3.1.1). After saving the calculated EMs, the loop restarts using a modified flux vector that is obtained by subtracting $\mathbf{v}'$ from $\mathbf{v}$ (such that v_{min} is eliminated).

The findings were consistent with those of the less efficient algorithm previously described by Poolman et al. [2004b]. Furthermore, as discussed in Section 3.1.2, although similar approaches to those presented here (that also use optimisation to decompose flux vectors) exist [Oddsdóttir et al., 2015; Hung et al., 2011; Jungers et al., 2011], they were designed with different aims and so are suitable for different problems, such as finding a minimal decomposition. The distinguishing property of the decomposition provided here is that, generally, reactions with a high flux value in $\mathbf{v}$ are likely to appear in many EMs within the decomposition and vice-versa. Another significant advantage is that the implementation of this algorithm is easily accessible as part of the open-source metabolic modelling software ScrumPy.

Chapter 3 also presents a theorem stating that for any flux vector, $\mathbf{v}$, there exist a minimal EM decomposition whose upper-bound is the dimension of the right null-space of a sub-matrix of the stoichiometry matrix that contains only the reactions that carry non-zero flux in $\mathbf{v}$ (Corollary 3.3.1.1). Although Müller and Regensburger [2016] and Jungers et al. [2011] have previously made similar claims, they only considered the original stoichiometry matrix (rather than the smaller sub-matrix), and therefore this thesis defines a tighter upper-bound.

Chapter 4 investigates methods that reduce the size of models by eliminating redundancies in the right null-space of the stoichiometry matrix. This concept was first introduced by Pfeiffer et al. [1999], who demonstrated that the stoichiometry matrix, $\mathbf{N}$, can be reduced by combining the reactions in enzyme subsets with a single reaction that embodies the net stoichiometry of the subset. Combining enzyme subsets does not change the steady-state behaviour of the network (as shown by Theorem 4.4.1.1) and is used as a pre-processing step for EMA in software such as ScrumPy and MetaTool.

The algorithm described in Chapter 4 expands upon this idea by introducing the notion of iso-stoichiometric groups. Each such group specifies a set of reactions that have identical stoichiometry, such that all but one of these duplicate reactions can be removed from **N** without affecting the net conversions of metabolites that can be performed by the network. As a consequence, models that have been reduced through the elimination of enzyme subsets can be further reduced by eliminating iso-stoichiometric groups. This reduction may cause new enzyme subsets to emerge in **N**, and therefore, **N** can be recursively reduced (by eliminating more enzyme subsets and iso-stoichiometric groups at each iteration) until a stage is reached where no enzyme subsets or iso-stoichiometric groups are present. Such a reduction is lossless: results obtained from the analysis of a reduced model can be extrapolated to equivalent results of the original model.

Apart from leading to a better compression ratio than when merely combining enzyme subsets, the identification of iso-stoichiometric groups reveals important network characteristics useful in model curation and analysis, as discussed in the following section.

In addition, Chapter 4 investigates two alternate approaches for defining biomass output in models. This can be accomplished in one of two ways: (i) a collective transporter that exports 1 unit of biomass (at the required proportions), or (ii) individual transporters that export each component by itself. When simulating growth using FBA (with a minimisation objective), one constraint that generates flux in the reaction with combined biomass is used for the first type of model, whilst the latter models require individual constraints for each biomass transporter. This thesis establishes that FBA solutions derived from either approach are equivalent.

Chapter 5 exploits one of the defining features of metabolic interaction: the existence of conservation laws which establish relationships between the network's metabolites. Although it has always been apparent that these laws impact on the left null-space of the stoichiometry matrix, few tools have been described for extracting information from this space.

Hence, Chapter 5 develops new methods for extracting information about metabolite properties and relationships using left null-space analysis. This approach builds upon methods applied previously to the right null-space in Poolman et al. [2007], which introduced a similarity measure that when applied to the rows of the right null-space clustered reactions according to the similarity in their steady-state flux. The work described extends upon this method by showing that applying this measure to the internal left null-space clusters metabolites with closely linked concentration values. This technique has many advantages, for example, including the identification of conserved moieties.

In addition, since information about the elemental composition of metabolites is embedded in the external left null-space, this similarity measure is used to cluster metabolites based on their chemical similarity and extract pathways that contain chemically similar metabolites. Notably, this technique relies solely on the network's structure and, therefore, unlike previous methods such as Haraldsdóttir and Fleming [2016], does not require external information describing the elemental composition of metabolites. This is useful to cater for cases where not all metabolites' chemical formulae can be defined.

Furthermore, when the composition of some (but not all) metabolites is known, a method that integrates this known information with the left null-space to calculate the composition of unknown metabolites and identify the presence of unbalanced reactions is also developed. As shown in Chapter 5, this method performed better than standard techniques that were previously utilised by the CSM group for calculating the composition of unknown metabolites, and is able to identify models that were not mass balanced correctly.

6.2 Further Potential Impact of Findings

The methods described above uncover features from the underlying structure of metabolic networks. In addition to providing novel insights, these methods can be incorporated into established modelling procedures by giving alternatives to time-consuming existing techniques, or by providing ways to enhance their output. To this end, this thesis proposes methods that facilitate model curation and that address the limitations of EMA and FBA, as discussed below.

6.2.1 Model Curation

Chapters 4 and 5 both offer different techniques that can facilitate model curation by identifying errors in models that can subsequently be manually corrected by the modeller.

For example, Algorithm 2 in Chapter 4 can identify duplicate processes that arise when automatically constructing models from databases, the most common being caused by multi-step enzymes being included as both the net stoichiometry and individual substeps. These duplications may cause errors in the results of model analysis, for example, by making it more difficult to identify essential reactions, and therefore should be removed from models. This algorithm also identifies iso-stoichiometric groups whose constituent reactions have inconsistent reversibility criteria. Such groups give rise to internal cycles that violate Hess's law (Chapter 2.1.2). Furthermore, reactions with incorrect reversibility may lead to energy being generated from no input, which is a known problem in metabolic models [Adhikari, 2017, page 67]. Previous methods for identifying such cycles include EMA (as EMs with a net stoichiometry of zero), which is less convenient than the methods proposed here due to its computational burden.

In addition, Chapter 5 provides a technique that can identify elemental imbalances in models by using the left null-space to calculate the chemical composition of metabolites. Previously established methods that test mass balancing in models do so by either (i) only considering reactions for which all metabolite compositions are known (as is commonly done at the CSM group and elsewhere [informal communication]), or (ii) identifying whether every metabolite can be assigned positive mass [Gevorgyan et al., 2008]. The method proposed here, however, combines and improves upon these two approaches by identifying whether the metabolites with unknown composition in the model can be assigned a positive composition when given the known composition of the other metabolites. In addition, further errors in the definitions of reactions were identified when analysing and comparing metabolite similarities in the left null-space and atomic matrix of models.

6.2.2 Model Analysis

EMA decomposes metabolic networks into a collection of minimal independent pathways called EMs [Schuster et al., 1999; Ullah et al., 2019]. This technique facilitates model analysis by converting intractable networks into tractable pathways that can be manually analysed, understood, and whose function can be identified. However, calculating a network's set of EMs is computationally intensive and, therefore, impractical for large models. Chapter 4 addresses this limitation by allowing aspects of EMA to be applied to the analysis of larger models. Another advantage of EMA reflected within this thesis is the analysis of redundancies shown in Chapter 4, which can alternatively be identified as groups of EMs with the same net stoichiometry.

Similarly, the technique of FBA uses Linear Programming (LP) to obtain a steady-state pathway (which may or may not be minimal) that best achieves an objective whilst satisfying a set of user-defined constraints [Orth and Palsson, 2010]. As discussed in Section 2.3.5, the redundancy in metabolic networks mean that any given FBA solution is likely to be one of many alternate pathways that achieve the same objective using different routes (a problem known as multiple optima). Since the calculation of the entire set of alternate FBA solutions is challenging, the most widely used method for studying multiple optima is flux variability analysis, a technique introduced by Mahadevan and Schilling [2003], that uses LP to obtain the range of values that each reaction can take whilst preserving the objective. Chapter 3 provides an additional method through which these alternate solutions can be explored, specifically uncovering redundancies that cause alternate solutions, which led to the design of Algorithm 6 that calculates alternate pathways to a given FBA solution.

Another limitation of FBA is that solutions that simultaneously produce multiple products typically contain many reactions, making them difficult to understand. A method to obtain more straightforward solutions is to reduce the number of FBA output constraints. However, as shown by the application of Algorithm 1 to *C. jejuni* in Chapter 3, such FBA solutions are not necessarily realistic since the pathways used by living beings must simultaneously meet many demands. This thesis tackles this problem by offering two methods that simplify FBA solutions by (i) decomposing them into EMs as described above and (ii) simplifying their visualisation using the pathway finding algorithms in Chapter 5.

6.2.3 The Oxygen Requirement of *C. jejuni*

An additional aim of Chapter 3 is to investigate the oxygen requirements of *C. jejuni*, a gram-negative microaerophilic bacteria that is recognised as one of the primary causes of food poisoning worldwide. This organism has the potential capacity for anaerobic respiration but requires small amounts of oxygen to grow. Previous authors have suggested that this requirement is because of oxygen-dependent heme synthesis or DNA synthesis enzymes [Kelly, 2008; Sellars et al., 2002]. However, the reason for *C. jejuni*'s dependence on oxygen is still unclear, and until the analysis presented in this thesis, there had been no study to investigate oxygen dependency with regard to biomass precursors. Therefore, in Chapter 3, a genome-scale metabolic model of *C. jejuni* is used to calculate this organism's metabolic response to changes in the rate of oxygen uptake, which raised the novel hypothesis that oxygen is required for synthesising PLP, which is essential for growth.

6.3 Future Work

The methods developed in this thesis are expected to become standard procedures in the model development and analysis pipelines at the CSM group at Oxford Brookes University and its collaborators. To this end, the algorithm discussed in Chapter 4 has been integrated into the 3.0 release of the ScrumPy modelling software, while an implementation of the algorithms discussed in Chapters 4 and 5 will be similarly integrated in due course.

The research presented in Chapter 4 is expected to result in the publication of a research paper (currently under review). Future collaboration with the Quadram Institute in Norwich will involve validating the theoretical result regarding *C. jejuni*'s oxygen requirements *in vivo*.

The work presented in Chapter 4 can be developed further by applying Algorithm 1 to additional GSM models and investigating the results. A potential plan for this is the GSM of *E. coli*, which the CSM group is currently investigating. Furthermore, since the output of Algorithm 1 was only compared with the algorithm by Poolman et al. [2004b], it would be interesting to compare Algorithm 1 with the other similar algorithms discussed in Section 3.1.2, therefore establishing the difference between the biological relevance of the different approaches.

A further development of the analysis in Chapter 4 regards the investigation of how the identified redundancies impact FBA solutions, for example, to facilitate the design of multiple gene knock-out strategies, and gather insight on the proportion of multiple optima uncovered by Algorithm 6.

In addition, the analysis of two different approaches to defining biomass production in GSMs is relevant to FBA solutions obtained using a minimisation objective function (which is the strategy most commonly used by the CSM group); it would be helpful to expand this work to include other objective criteria currently in use by the metabolic modelling community.

Chapter 5 provides methods that relate the concentration of metabolites in models. Although this is based on established theoretical concepts, the implications on metabolomic data were not assessed. In relation to this topic, an interesting future investigation includes the development of a framework for understanding how the changes in the structure of a network affect the relationship between metabolite concentrations as identified by the similarity measure in Chapter 5. Once such a theory is established, it can be used to infer which structural changes may have occurred to cause a change in concentration that is observed via metabolomic data.

Moreover, the efficiency of the method that calculates missing atomic composition in Chapter 5 can be improved further. Currently, a new MILP instance is created in each iteration of the algorithm. Future work involves implementing the glpk[1] MILP solver as part of ScrumPy (in the same manner that the LP module), which would improve the algorithm by allowing a single MILP instance to be used during all iterations (by sequentially updating the constraints). Furthermore, although the method presented in Chapter 5 can detect models that are not mass balanced, the algorithm depends on the initial ordering of the stoichiometry matrix such that the reactions identified as incorrect and therefore removed from the stoichiometry matrix are not necessarily the cause of this mass balance. As seen in Chapter 5, this can lead to the composition of some metabolites being incorrectly calculated. Hence, at this stage, the results of Algorithm 7 for models that are not correctly balanced must be carefully analysed. A potential future remedy for this is to design a strategy in which the reactions detected to be incorrect are chosen to be minimal (rather than depending on their order in $\mathbf{N}$). Another potential improvement

[1] gnu.org/software/glpk/

is to take into account the fact that imbalances are more likely to be caused by missing protons and water compounds.

In summary, this thesis describes new methods through which metabolic networks can be investigated. It demonstrates the pivotal role that modelling plays in advancing future breakthroughs within the field of metabolism research. Furthermore, the author sincerely hopes that this work will serve as evidence of the importance of interdisciplinary collaboration and inspire further research in this fascinating field.

References

V. Acuña, F. Chierichetti, V. Lacroix, A. Marchetti-Spaccamela, M. F. Sagot, and L. Stougie. Modes and cuts in metabolic networks: Complexity and algorithms. *BioSystems*, 95(1):51–60, 2009. ISSN 03032647. doi: 10.1016/j.biosystems.2008.06.015.

K. Adhikari. *Genome scale metabolic modelling of Arabidopsis thaliana and Chlamydomonas reinhardtii* . PhD thesis, Oxford Brookes University, 2017.

A. Ahmad, H. B. Hartman, S. Krishnakumar, D. A. Fell, M. G. Poolman, and S. Srivastava. A Genome Scale Model of *Geobacillus thermoglucosidasius* (C56-YS93) reveals its biotechnological potential on rice straw hydrolysate. *Journal of Biotechnology*, 251 (April):30–37, 2017. ISSN 18734863. doi: 10.1016/j.jbiotec.2017.03.031.

E. Al-Saidi. *Computational modelling of the glycinergic synapse.* PhD thesis, Oxford Brookes University, 2020.

A. Alqurashi, L. Alfs, J. Swann, J. N. Butt, and D. J. Kelly. The flavodoxin FldA activates the class Ia ribonucleotide reductase of *Campylobacter jejuni*. *Molecular Microbiology*, 116(1):343–358, 2021. ISSN 13652958. doi: 10.1111/mmi.14715.

E. Amaldi and V. Kann. On the approximability of minimizing nonzero variables or unsatisfied relations in linear systems. *Theoretical Computer Science*, 209(1-2):237–260, 1998. ISSN 03043975. doi: 10.1016/S0304-3975(97)00115-1.

H. Asakura, N. Hashii, M. Uema, N. Kawasaki, Y. Sugita-Konishi, S. Igimi, and S. Yamamoto. *Campylobacter jejuni* pdxA Affects Flagellum-Mediated Motility to Alter Host Colonization. *PLoS ONE*, 8(8), 2013. ISSN 19326203. doi: 10.1371/journal.pone.0070418.

M. Ataman and V. Hatzimanikatis. lumpGEM: Systematic generation of subnetworks and elementally balanced lumped reactions for the biosynthesis of target metabolites. *PLoS Computational Biology*, 13(7):1–21, 2017. ISSN 15537358. doi: 10.1371/journal.pcbi.1005513.

M. Ataman, D. F. Hernandez Gardiol, G. Fengos, and V. Hatzimanikatis. redGEM: Systematic reduction and analysis of genome-scale metabolic reconstructions for development of consistent core metabolic models. *PLoS Computational Biology*, 13(7):1–22, 2017. ISSN 15537358. doi: 10.1371/journal.pcbi.1005444.

D. Bajusz, A. Rácz, and K. Héberger. Why is Tanimoto index an appropriate choice for fingerprint-based similarity calculations? *Journal of Cheminformatics*, 7(1):1–13, 2015.

ISSN 17582946. doi: 10.1186/s13321-015-0069-3.

A. Biz, S. Proulx, Z. Xu, K. Siddartha, A. Mulet Indrayanti, and R. Mahadevan. Systems biology based metabolic engineering for non-natural chemicals. *Biotechnology Advances*, 37(6):107379, 2019. ISSN 07349750. doi: 10.1016/j.biotechadv.2019.04.001.

T. Boston and T. Atlung. FNR-mediated oxygen-responsive regulation of the nrdDG operon of Escherichia coli. *Journal of Bacteriology*, 185(17):5310–5313, 2003. ISSN 00219193. doi: 10.1128/JB.185.17.5310-5313.2003.

C. O. Brämer and A. Steinbüchel. The methylcitric acid pathway in *Ralstonia eutropha*: New genes identified involved in propionate metabolism. *Microbiology*, 147(8):2203–2214, 2001. ISSN 13500872. doi: 10.1099/00221287-147-8-2203.

M. Brock, C. Maerker, A. Schütz, U. Völker, and W. Buckel. Oxidation of propionate to pyruvate in *Escherichia coli*: Involvement of methylcitrate dehydratase and aconitase. *European Journal of Biochemistry*, 269(24):6184–6194, 2002. ISSN 00142956. doi: 10.1046/j.1432-1033.2002.03336.x.

N. Brown, B. McKay, and J. Gasteiger. Fingal: A novel approach to geometric fingerprinting and a comparative study of its application to 3D-QSAR modelling. *QSAR and Combinatorial Science*, 24(4):480–484, 2005. ISSN 1611020X. doi: 10.1002/qsar.200430923.

A. P. Burgard, S. Vaidyaraman, and C. D. Maranas. Minimal reaction sets for *Escherichia coli* metabolism under different growth requirements and uptake environments. *Biotechnology Progress*, 17(5):791–797, 2001. ISSN 87567938. doi: 10.1021/bp0100880.

A. P. Burgard, E. V. Nikolaev, C. H. Schilling, and C. D. Maranas. Flux coupling analysis of genome-scale metabolic network reconstructions. *Genome Research*, 14(2):301–312, 2004. ISSN 10889051. doi: 10.1101/gr.1926504.

M. K. Campbell and S. O. Farrell. *Biochemistry.* Thomson Brooks/Cole, 6th edition, 2009.

S. Cardoso Diniz, I. Voss, and A. Steinbuchel. Optimization of cyanophycin production in recombinant strains of *Pseudomonas putida* and *Ralstonia eutropha* Employing Elementary Mode Analysis and Statistical Experimental Design. *Biotechnology and Bioengineering*, 93(4):698–717, 2006. ISSN 0021-8782. doi: 10.1002/bit.

L. Chindelevitch, J. Trigg, A. Regev, and B. Berger. An exact arithmetic toolbox for a consistent and reproducible structural analysis of metabolic network models. *Nature Communications*, 5, 2014. ISSN 20411723. doi: 10.1038/ncomms5893.

W. D. Cook and R. J. Webster. Caratheodory's Theorem. *Canadian Math Bullitin*, 15 (2), 1972.

C. Culley, S. Vijayakumar, G. Zampieri, and C. Angione. A mechanism-aware and multiomic machine-learning pipeline characterizes yeast cell growth. *Proceedings of the National Academy of Sciences of the United States of America*, 117(31):18869–18879, 2020. ISSN 10916490. doi: 10.1073/pnas.2002959117.

L. F. de Figueiredo, A. Podhorski, A. Rubio, C. Kaleta, J. E. Beasley, S. Schuster, and F. J. Planes. Computing the shortest elementary flux modes in genome-scale metabolic networks. *Bioinformatics*, 25(23):3158–3165, 2009. ISSN 13674803. doi: 10.1093/bioinformatics/btp564.

S. P. W. de Vries, S. Gupta, A. Baig, J. Lapos Heureux, E. Pont, D. P. Wolanska, D. J. Maskell, and A. J. Grant. Motility defects in *Campylobacter jejuni* defined gene deletion mutants caused by second-site mutations. *Microbiology*, 161(12):2316–2327, 2015. ISSN 1465-2080. doi: 10.1099/mic.0.000184.

P. Erdrich, R. Steuer, and S. Klamt. An algorithm for the reduction of genome-scale metabolic network models to meaningful core models. *BMC Systems Biology*, 9(1): 1–12, 2015. ISSN 17520509. doi: 10.1186/s12918-015-0191-x.

I. Famili and B. O. Palsson. The convex basis of the left null space of the stoichiometric matrix leads to the definition of metabolically meaningful pools. *Biophysical Journal*, 85(1):16–26, 2003. ISSN 00063495. doi: 10.1016/S0006-3495(03)74450-6.

X. Fang, C. J. Lloyd, and B. O. Palsson. Reconstructing organisms in silico: genome-scale models and their emerging applications. *Nature Reviews Microbiology*, 18(December): 23–26, 2020. ISSN 17401534. doi: 10.1038/s41579-020-00440-4.

Z. Fatma, H. Hartman, M. G. Poolman, D. A. Fell, S. Srivastava, T. Shakeel, and S. S. Yazdani. Model-assisted metabolic engineering of Escherichia coli for long chain alkane and alcohol production. *Metabolic Engineering*, 46:1–12, 2018. ISSN 10967184. doi: 10.1016/j.ymben.2018.01.002.

A. M. Feist and B. O. Palsson. The biomass objective function. *Current Opinion in Microbiology*, 13(3):344–349, 2010. ISSN 13695274. doi: 10.1016/j.mib.2010.03.003.

D. Fell. *Understanding the Control of Metabolism.* Portland Press, London, 1997.

D. a. Fell and J. R. Small. Fat synthesis in adipose tissue. *The Biochemical journal*, 238 (3):781–786, 1986. ISSN 02646021.

W. Fenchel. Convex cones, sets, and functions. Technical Report September, Princeton University, 1953.

Z. Field, A. Field, and J. Miles. *Discovering statistics using R.* SAGE, London, 2012. ISBN 9781446200452. doi: 10.5860/choice.50-2114.

C. J. Fritzemeier, D. Hartleb, B. Szappanos, B. Papp, and M. J. Lercher. Erroneous energy-generating cycles in published genome scale metabolic networks: Identification and removal. *PLoS Computational Biology*, 13(4):1–14, 2017. ISSN 15537358. doi: 10.1371/journal.pcbi.1005494.

K. Fukuda and A. Prodon. Double description method revisited. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1120:91–111, 1996. ISSN 16113349. doi: 10.1007/3-540-61576-8_77.

J. Gagneur and S. Klamt. Computation of elementary modes: A unifying framework and the new binary approach. *BMC Bioinformatics*, 5:1–21, 2004. ISSN 14712105. doi: 10.1186/1471-2105-5-175.

P. Gerlee, L. Lizana, and K. Sneppen. Pathway identification by network pruning in the metabolic network of *Escherichia coli*. *Bioinformatics*, 25(24):3282–3288, 2009. ISSN 13674803. doi: 10.1093/bioinformatics/btp575.

A. Gevorgyan. *Analytical methods for genome-scale metabolic networks applied to Streptococcus agalactiae*. PhD thesis, Oxford Brookes University, 2009.

A. Gevorgyan, M. G. Poolman, and D. A. Fell. Detection of stoichiometric inconsistencies in biomolecular models. *Bioinformatics*, 24(19):2245–2251, 2008. ISSN 13674803. doi: 10.1093/bioinformatics/btn425.

S. Ghaderi, H. S. Haraldsdóttir, M. Ahookhosh, S. Arreckx, and R. M. Fleming. Structural conserved moiety splitting of a stoichiometric matrix. *Journal of Theoretical Biology*, 499, 2020. ISSN 10958541. doi: 10.1016/j.jtbi.2020.110276.

M. Grotschel, L. Lovasz, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer, Berlin, 1988. doi: 10.2307/2583689.

C. Gu, G. B. Kim, W. J. Kim, H. U. Kim, and S. Y. Lee. Current status and applications of genome-scale metabolic models. *Genome Biology*, 20(1):1–18, 2019. ISSN 1474760X. doi: 10.1186/s13059-019-1730-3.

E. Guccione, M. Del Rocio Leon-Kempis, B. M. Pearson, E. Hitchin, F. Mulholland, P. M. Van Diemen, M. P. Stevens, and D. J. Kelly. Amino acid-dependent growth of *Campylobacter jejuni*: Key roles for aspartase (AspA) under microaerobic and oxygen-limited conditions and identification of AspB (Cj0762), essential for growth on glutamate. *Molecular Microbiology*, 69(1):77–93, 2008. ISSN 0950382X. doi: 10.1111/j.1365-2958.2008.06263.x.

M. B. Guebila. VFFVA: Dynamic load balancing enables large-scale flux variability analysis. *BMC Bioinformatics*, 21(1):1–13, 2020. ISSN 14712105. doi: 10.1186/s12859-020-03711-2.

F. Guil, J. F. Hidalgo, and J. M. García. Boosting the extraction of elementary flux modes in genome-scale metabolic networks using the linear programming approach. *Bioinformatics*, pages 1–8, 2020. doi: 10.1093/bioinformatics/btaa280.

O. Hädicke and S. Klamt. Computing complex metabolic intervention strategies using constrained minimal cut sets. *Metabolic Engineering*, 13(2):204–213, 2011. ISSN 10967176. doi: 10.1016/j.ymben.2010.12.004.

N. Halbwachs, D. Merchat, and L. Gonnord. Some ways to reduce the space dimension in polyhedra computations. *Formal Methods in System Design*, 29(1):79–95, 2006. ISSN 09259856. doi: 10.1007/s10703-006-0013-2.

H. S. Haraldsdóttir and R. M. Fleming. Identification of conserved moieties in metabolic networks by graph theoretical analysis of atom transition networks. *PLoS Computa-*

tional Biology, 12(11):1–27, 2016. ISSN 15537358. doi: 10.1371/journal.pcbi.1004999.

H. B. Hartman, D. A. Fell, S. Rossell, P. R. Jensen, M. J. Woodward, L. Thorndahl, L. Jelsbak, J. E. Olsen, A. Raghunathan, S. Daefler, and M. G. Poolman. Identification of potential drug targets in *Salmonella enterica* sv. *Typhimurium* using metabolic modelling and experimental validation. *Microbiology (United Kingdom)*, 160(PART 6): 1252–1266, 2014. ISSN 14652080. doi: 10.1099/mic.0.076091-0.

S. Hasim, N. A. Hussin, F. Alomar, K. R. Bidasee, K. W. Nickerson, and M. A. Wilson. A glutathione-independent glyoxalase of the DJ-1 superfamily plays an important role in managing metabolically generated methylglyoxal in *Candida albicans*. *Journal of Biological Chemistry*, 289(3):1662–1674, 2014. ISSN 00219258. doi: 10.1074/jbc.M113.505784.

R. Heinrich and S. Schuster. *The Regulation of Cellular Systems*. Springer, New York, 1996. ISBN 9781461284925. doi: 10.1007/978-1-4613-1161-4.

H. A. Herrmann, B. C. Dyson, L. Vass, G. N. Johnson, and J. M. Schwartz. Flux sampling is a powerful tool to study metabolism under changing environmental conditions. *npj Systems Biology and Applications*, 5(1):1–8, 2019. ISSN 20567189. doi: 10.1038/s41540-019-0109-0.

J. H. S. Hofmeyr. Kinetic modelling of compartmentalised reaction networks. *BioSystems*, 187:319–328, 2020. doi: 10.1016/j.biosystems.2020.104203.

D. Hofreuter. Defining the metabolic requirements for the growth and colonization apacity of *Campylobacter jejuni*. *Frontiers in Cellular and Infection Microbiology*, 4(SEP):1–19, 2014. ISSN 22352988. doi: 10.3389/fcimb.2014.00137.

H. G. Holzhütter. The principle of flux minimization and its application to estimate stationary fluxes in metabolic networks. *European Journal of Biochemistry*, 271(14): 2905–2922, 2004. ISSN 00142956. doi: 10.1111/j.1432-1033.2004.04213.x.

M. K. Hon, M. S. Mohamad, A. H. Mohamed Salleh, Y. W. Choon, K. Mohd Daud, M. A. Remli, M. A. Ismail, S. Omatu, R. O. Sinnott, and J. M. Corchado. Identifying a gene knockout strategy using a hybrid of simple constrained artificial bee colony algorithm and flux balance analysis to enhance the production of succinate and lactate in *Escherichia Coli*. *Interdisciplinary Sciences: Computational Life Sciences*, 11(1): 33–44, 2019. ISSN 18671462. doi: 10.1007/s12539-019-00324-z.

Y. Huang, C. Zhong, H. X. Lin, and J. Wang. A method for finding metabolic pathways using atomic group tracking. *PLoS ONE*, 12(1):1–26, 2017. ISSN 19326203. doi: 10.1371/journal.pone.0168725.

B. Huma, S. Kundu, M. G. Poolman, N. J. Kruger, and D. A. Fell. Stoichiometric analysis of the energetics and metabolic impact of photorespiration in C3 plants. *Plant Journal*, 96(6):1228–1241, 2018. ISSN 1365313X. doi: 10.1111/tpj.14105.

S. Hung, J. Chan, and P. Ji. Decomposing flux distributions into elementary flux modes in genome-scale metabolic networks. *Bioinformatics*, 27(16):2256–2262, 2011. ISSN

14602059. doi: 10.1093/bioinformatics/btr367.

K. Ip, C. Colijn, and D. S. Lun. Analysis of complex metabolic behavior through pathway decomposition. *BMC Systems Biology*, 5, 2011. ISSN 17520509. doi: 10.1186/1752-0509-5-91.

T. Ito and D. M. Downs. Pyridoxal reductase, pdxi, is critical for salvage of pyridoxal in *Escherichia coli*. *Journal of Bacteriology*, 202(12):e00056–20, 2020. doi: 10.1128/JB.00056-20.

S. Jagadevan, A. Banerjee, C. Banerjee, C. Guria, R. Tiwari, M. Baweja, and P. Shukla. Recent developments in synthetic biology and metabolic engineering in microalgae towards biofuel production. *Biotechnology for Biofuels*, 11(1):1–21, 2018. ISSN 17546834. doi: 10.1186/s13068-018-1181-1.

D. Jevremovic, C. T. Trinh, F. Srienc, C. P. Sosa, and D. Boley. Parallelization of Nullspace Algorithm for the computation of metabolic pathways. *Parallel Computing*, 23(1):1–7, 2011. ISSN 15378276. doi: 10.1038/jid.2014.371.

R. M. Jungers, F. Zamorano, V. D. Blondel, A. V. Wouwer, and G. Bastin. Fast computation of minimal elementary decompositions of metabolic flux vectors. *Automatica*, 47 (6):1255–1259, 2011. ISSN 00051098. doi: 10.1016/j.automatica.2011.01.011.

M. Kanehisa and S. Goto. KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*, 28(1):27–30, 2000. ISSN 17921082. doi: 10.3892/ol.2020.11439.

P. D. Karp, R. Billington, R. Caspi, C. A. Fulcher, M. Latendresse, A. Kothari, I. M. Keseler, M. Krummenacker, P. E. Midford, Q. Ong, W. K. Ong, S. M. Paley, and P. Subhraveti. The BioCyc collection of microbial genomes and metabolic pathways. *Briefings in Bioinformatics*, 20(4):1085–1093, 2018. ISSN 14774054. doi: 10.1093/bib/bbx085.

S. M. Kelk, B. G. Olivier, L. Stougie, and F. J. Bruggeman. Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. *Scientific Reports*, 2:44–46, 2012. ISSN 20452322. doi: 10.1038/srep00580.

D. J. Kelly. Complexity and Versatility in the Physiology and Metabolism of *Campylobacter jejuni*. In I. Nachamkin, C. M. Szymanski, and M. J. Blasr, editors, *Campylobacter*, chapter 3. ASM Press, Washington, DC, 3rd edition, 2008.

L. Khachiyan, E. Boros, K. Borys, K. Elbassioni, and V. Gurvich. Generating all vertices of a polyhedron is hard. *Discrete and Computational Geometry*, 39(1-3):174–190, 2008. ISSN 14320444. doi: 10.1007/s00454-008-9050-5.

A. Z. Khan, M. Bilal, S. Mehmood, A. Sharma, and H. M. Iqbal. State-of-the-art genetic modalities to engineer cyanobacteria for sustainable biosynthesis of biofuel and fine-chemicals to meet bio–economy challenges. *Life*, 9(3):1–22, 2019. ISSN 20751729. doi: 10.3390/life9030054.

S. Khan, P. Somvanshi, T. Bhardwaj, R. K. Mandal, S. A. Dar, M. Wahid, A. Jawed, M. Lohani, M. Khan, M. Y. Areeshi, and S. Haque. Aspartate-β-semialdeyhyde dehydrogenase as a potential therapeutic target of *Mycobacterium tuberculosis* H37Rv: Evidence from in silico elementary mode analysis of biological network model. *Journal of Cellular Biochemistry*, 119(3):2832–2842, 2018. ISSN 10974644. doi: 10.1002/jcb.26458.

J. C. Kim, E. Oh, J. Kim, and B. Jeon. Regulation of oxidative stress resistance in *Campylobacter jejuni*, a microaerophilic foodborne pathogen. *Frontiers in Microbiology*, 6(JUL):1–12, 2015. ISSN 1664302X. doi: 10.3389/fmicb.2015.00751.

S. Klamt and J. Stelling. Combinatorial complexity of pathway analysis in metabolic networks. *Molecular Biology Reports*, 29(1-2):233–236, 2002. ISSN 03014851. doi: 10.1023/A:1020390132244.

B. Kumar, C. Kaur, A. Pareek, S. K. Sopory, and S. L. Singla-Pareek. Tracing the evolution of plant glyoxalase iii enzymes for structural and functional divergence. *Antioxidants*, 10(5):1–19, 2021. ISSN 20763921. doi: 10.3390/antiox10050648.

H. M. Leicester. Germain Henri Hess and the foundations of thermochemistry. *Journal of Chemical Education*, pages 581–583, 1951. ISSN 00219584. doi: 10.1021/ed028p581.

X. Liu, B. Gao, V. Novik, and J. E. Galán. Quantitative proteomics of intracellular *Campylobacter jejuni* reveals metabolic reprogramming. *PLoS Pathogens*, 8(3), 2012. ISSN 15537366. doi: 10.1371/journal.ppat.1002562.

T. Maarleveld. *Fluxes and Fluctuations in Biochemical Models*. PhD thesis, Vrije Universiteit Amsterdam, 2015.

T. R. Maarleveld, M. T. Wortel, B. G. Olivier, B. Teusink, and F. J. Bruggeman. Interplay between constraints, objectives, and optimality for genome-scale stoichiometric models. *PLoS Computational Biology*, 11(4):1–21, 2015. ISSN 15537358. doi: 10.1371/journal.pcbi.1004166.

R. Mahadevan and C. H. Schilling. The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic Engineering*, 5(4):264–276, 2003. ISSN 10967176. doi: 10.1016/j.ymben.2003.09.002.

M. Masid, M. Ataman, and V. Hatzimanikatis. Analysis of human metabolism by reducing the complexity of the genome-scale models using redHUMAN. *Nature Communications*, 11(1):1–12, 2020. ISSN 20411723. doi: 10.1038/s41467-020-16549-2.

M. L. Mavrovouniotis, G. Stephanopoulos, and G. Stephanopoulos. Computer-aided synthesis of biochemical pathways. *Biotechnology and Bioengineering*, 36(11):1119–1132, 1990. ISSN 10970290. doi: 10.1002/bit.260361107.

N. Mesfin and D. Fell. Using a metabolic model of *Acetobacterium woodii* for insights into its utility for biotechnological purposes. *Access Microbiology*, 1(1A), 2019.

A. Metris, M. Reuter, D. J. Gaskin, J. Baranyi, and A. H. van Vliet. In vivo and in silico determination of essential genes of *Campylobacter jejuni*. *BMC Genomics*, 12(1):535, 2011. ISSN 14712164. doi: 10.1186/1471-2164-12-535.

O. Meyerhof and R. Junowicz-Kocholaty. The equilibria of isomerase and aldolase, and the problem of the phosphorylation of glyceraldehyde phosphate. *Journal of Biological Chemistry*, 149(1):71–92, 1943. ISSN 00219258. doi: 10.1016/s0021-9258(18)72218-7.

R. Moreno-Sánchez, E. Saavedra, S. Rodríguez-Enríquez, and V. Olín-Sandoval. Metabolic Control Analysis: A tool for designing strategies to manipulate metabolic pathways. *Journal of Biomedicine and Biotechnology*, 2008(1), 2008. ISSN 11107243. doi: 10.1155/2008/597913.

J. T. B. Morgan and P. G. A. Ray. Non-uniqueness and inversions in cluster analysis. *Journal of the Royal Statistical Society. Series C*, 44(1):117–134, 1995.

S. Müller and G. Regensburger. Elementary vectors and conformal sums in polyhedral geometry and their relevance for metabolic pathway analysis. *Frontiers in Genetics*, 7 (MAY):1–11, 2016. ISSN 16648021. doi: 10.3389/fgene.2016.00090.

D. L. Nelson and M. M. Cox. *Principles of Biochemistry.* Macmillan Learning, New York, NY, 7th edition, 2004.

H. Æ. Oddsdóttir, E. Hagrot, V. Chotteau, and A. Forsgren. On dynamically generating relevant elementary flux modes in a metabolic network using optimization. *Journal of Mathematical Biology*, 71(4):903–920, 2015. ISSN 14321416. doi: 10.1007/s00285-014-0844-1.

J. D. Orth and B. Palsson. Systematizing the generation of missing metabolic knowledge. *Biotechnology and Bioengineering*, 107(3):403–412, 2010. ISSN 00063592. doi: 10.1002/bit.22844.

J. Parkhill, B. Wren, K. Mungall, J. Ketley, C. Churcher, D. Basham, T. Chillingworth, R. Davies, T. Feltwell, S. Holroyd, K. Jagels, A. Karlyshev, S. Moule, M. Pallen, C. Penn, M. Quail, M. Rajandream, K. Rutherford, A. van Vliet, and B. Barrell. The genome sequence of the food-borne pathogen *Campylobacter jejuni* reveals hypervariable sequences. *Nature*, 403:665–8, 03 2000. doi: 10.1038/35001088.

M. Parra, S. Stahl, and H. Hellmann. Vitamin B6 and its role in cell metabolism and physiology. *Cells*, 7(7), 2018. ISSN 20734409. doi: 10.3390/cells7070084.

N. Pearcy, M. Garavaglia, T. Millat, J. P. Gilbert, Y. Song, H. Hartman, C. Woods, C. Tomi-Andrino, R. R. Bommareddy, B. K. Cho, D. A. Fell, M. Poolman, J. R. King, K. Winzer, J. Twycross, and N. P. Minton. A genome-scale metabolic model of *Cupriavidus necator* H16 integrated with TraDIS and transcriptomic data reveals metabolic insights for biotechnological applications. *PLoS Computational Biology*, 18 (5):1–35, 2022. ISSN 15537358. doi: 10.1371/journal.pcbi.1010106.

T. Pfeiffer, I. Sánchez-Valdenebro, J. C. Nuño, F. Montero, S. Schuster, I. Sánchezćvaldenebro, J. C. Nuño, F. Montero, and S. Schuster. METATOOL: for studying metabolic networks. *Bioinformatics*, 15(3):251–257, 1999. ISSN 13674803. doi: 10.1093/bioinformatics/15.3.251.

E. Pitkänen, J. Rousu, and E. Ukkonen. Computational methods for metabolic reconstruction. *Current Opinion in Biotechnology*, 21(1):70–77, 2010. ISSN 09581669. doi:

10.1016/j.copbio.2010.01.010.

M. G. Poolman. ScrumPy: metabolic modelling with Python. *IEE Proceedings Systems biology*, 153(5), 2006.

M. G. Poolman, H. E. Assmus, and D. A. Fell. Applications of metabolic modelling to plant metabolism. *Journal of Experimental Botany*, 55(400):1177–1186, 2004a. ISSN 00220957. doi: 10.1093/jxb/erh090.

M. G. Poolman, K. V. Venkatesh, M. K. Pidcock, and D. A. Fell. A method for the determination of flux in elementary modes, and its application to *Lactobacillus rhamnosus*. *Biotechnology and Bioengineering*, 88(5):601–612, 2004b. ISSN 00063592. doi: 10.1002/bit.20273.

M. G. Poolman, B. K. Bonde, A. Gevorgyan, H. H. Patel, and D. A. Fell. Challenges to be faced in the reconstruction of metabolic networks from public databases. *IEE Proceedings: Systems Biology*, 153(5):379–384, 2006. ISSN 17412471. doi: 10.1049/ip-syb:20060012.

M. G. Poolman, L. Miguet, L. J. Sweetlove, and D. A. Fell. A genome-scale metabolic model of Arabidopsis and some of its properties. *Plant Physiol.*, 151(3):1570–1581, 2009. doi: 10.1104/pp.109.141267.

M. G. Poolman, D. A. Fell, and C. A. Raines. Elementary modes analysis of photosynthate metabolism in the chloroplast stroma. *European Journal of Biochemistry*, 270(3):430–439, 2003. ISSN 00142956. doi: 10.1046/j.1432-1033.2003.03390.

M. G. Poolman, C. Sebu, M. K. Pidcock, and D. A. Fell. Modular decomposition of metabolic systems via null-space analysis. *Journal of Theoretical Biology*, 249(4):691–705, 2007. ISSN 00225193. doi: 10.1016/j.jtbi.2007.08.005.

C. Reder. Metabolic control theory: A structural approach. *Journal of Theoretical Biology*, 135(2):175–201, 1988. ISSN 10958541. doi: 10.1016/S0022-5193(88)80073-0.

M. Reuter, A. Mallett, B. M. Pearson, and A. H. Van Vliet. Biofilm formation by *Campylobacter jejuni* is increased under aerobic conditions. *Applied and Environmental Microbiology*, 76(7):2122–2128, 2010. ISSN 00992240. doi: 10.1128/AEM.01878-09.

R. T. Rockafellar. The elementary vectors of a subspace of $\mathbf{R}^N$. In *Combinatorial Mathematics and its Applications*, pages 104–127. University of Washington, 1969.

R. C. Rodrigues, N. Haddad, D. Chevret, J. M. Cappelier, and O. Tresse. Comparison of proteomics profiles of *Campylobacter jejuni* strain Bf under microaerobic and aerobic conditions. *Frontiers in Microbiology*, 7(OCT):1–12, 2016. ISSN 1664302X. doi: 10.3389/fmicb.2016.01596.

A. Röhl and A. Bockmayr. Finding MEMo: minimum sets of elementary flux modes. *Journal of Mathematical Biology*, 79(5):1749–1777, 2019. ISSN 14321416. doi: 10.1007/s00285-019-01409-5.

A. Röhl, T. Riou, and A. Bockmayr. Computing irreversible minimal cut sets in genome-scale metabolic networks via flux cone projection. *Bioinformatics*, 35(15):2618–2625, 2019. ISSN 14602059. doi: 10.1093/bioinformatics/bty1027.

H. M. Sauro. *Enzyme Kinetics for Systems Biology.* Ambrosius Publishing, Seattle, 1st edition, 2012.

H. M. Sauro and B. Ingalls. Conservation analysis in biochemical networks : computational issues for software writers. *Biophys Chem.*, 109(1), 2003. doi: 10.1016/j.bpc.2003.08.009.

S. Schäuble, S. Schuster, and C. Kaleta. Hands-on metabolism: Analysis of complex biochemical networks using elementary flux modes. *Methods in Enzymology*, 500:437–456, 2011. ISSN 15577988. doi: 10.1016/B978-0-12-385118-5.00022-0.

S. Schuster, S. Klamt, W. Weckwerth, F. Moldenhauer, and T. Pfeiffer. Use of network analysis of metabolic systems in bioengineering. *Bioprocess and Biosystems Engineering*, 24(6):363–372, 2002. ISSN 16157591. doi: 10.1007/s004490100253.

S. Schuster and D. Fell. Modeling and Simulating Metabolic Networks. In T. Lengauer, editor, *Bioinformatics - From Genomes to Therapies*, volume 2. Wiley-VCH, Weinheim, 2007. ISBN 9783527312788.

S. Schuster and C. Hilgetag. On elementary flux modes in biochemical reaction systems at steady state. *Journal of Biological Systems*, 02(02):165–182, 1994. ISSN 0218-3390. doi: 10.1142/s0218339094000131.

S. Schuster and C. Hilgetag. What information about the conserved-moiety structure of chemical reaction systems can be derived from their stoichiometry? *Journal of Physical Chemistry*, 99(20):8017–8023, 1995. ISSN 00223654. doi: 10.1021/j100020a026.

S. Schuster and T. Hofer. Determining All Extreme Semi-positive Conservation Relations in Chemical Reaction Systems : A Test Criterion for Conservativity. *J. Chem Soc. Faraday Trans.*, 87(16):2561–2566, 1991.

S. Schuster, T. Dandekar, and D. A. Fell. Detection of elementary flux modes in biochemical networks: A promising tool for pathway analysis and metabolic engineering. *Trends in Biotechnology*, 17(2):53–60, 1999. ISSN 01677799. doi: 10.1016/S0167-7799(98)01290-6.

S. Schuster, D. A. Fell, and T. Dandekar. A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. *Nature Biotechnology*, 18(March), 2000. doi: 10.1038/73786.

J. M. Schwartz and M. Kanehisa. A quadratic programming approach for decomposing steady-state metabolic flux distributions onto elementary modes. *Bioinformatics*, 21 (SUPPL. 2):204–205, 2005. ISSN 13674803. doi: 10.1093/bioinformatics/bti1132.

M. J. Sellars, S. J. Hall, and D. J. Kelly. Growth of *Campylobacter jejuni* supported by respiration of fumarate, nitrate, nitrite, trimethylamine-N-oxide, or dimethyl sulfoxide requires oxygen. *Journal of Bacteriology*, 184(15):4187–4196, 2002. ISSN 00219193. doi: 10.1128/JB.184.15.4187-4196.2002.

G. Siebert. Citrate and isocitrate: Determination with aconitase and isocitric dehydrogenase. In H.-U. Bergmeyer, editor, *Methods of Enzymatic Analysis*, pages 318–323. Academic Press, 1965. ISBN 978-0-12-395630-9. doi: https://doi.org/10.1016/B978-0-12-395630-9.50068-2.

D. Singh. *Genome Scale Metabolic Modelling of Phaeodactylum tricornutum.* PhD thesis, Oxford Brookes University, 2017.

D. Singh and M. J. Lercher. Network reduction methods for genome-scale metabolic models. *Cellular and Molecular Life Sciences*, 77(3):481–488, 2020. ISSN 14209071. doi: 10.1007/s00018-019-03383-z.

W. R. Smith and R. R. Missen. What is Chemical Stoichiometry? *Chemical Engineering Education*, pages 26–32, 1979. ISSN 00092347.

H. S. Song, N. Goldberg, A. Mahajan, and D. Ramkrishna. Sequential computation of elementary modes and minimal cut sets in genome-scale metabolic networks using alternate integer linear programming. *Bioinformatics*, 33(15):2345–2353, 2017. ISSN 14602059. doi: 10.1093/bioinformatics/btx171.

J. Stelling and S. Klamt. Stoichiometric and Constraint-based Modeling. In Z. Szallasi, J. Stelling, and V. Periwal, editors, *System Modeling in Cellular Biology: From Concepts to Nuts and Bolts.* MIT Press, Cambridge, MA, 2006.

E. L. Stiefel, editor. *An Introduction to Numerical Mathematics.* Academic Press, 1963. ISBN 978-1-4832-0038-5. doi: https://doi.org/10.1016/B978-1-4832-0038-5.50002-8.

R. Sugimoto, N. Saito, T. Shimada, and K. Tanaka. Identification of ybha as the pyridoxal 5'-phosphate (plp) phosphatase in *Escherichia coli*: Importance of plp homeostasis on the bacterial growth. *The Journal of General and Applied Microbiology*, 63(6):362–368, 2017. doi: 10.2323/jgam.2017.02.008.

N. Tejera, L. Crossman, B. Pearson, E. Stoakes, F. Nasher, B. Djeghout, M. Poolman, J. Wain, and D. Singh. Genome-scale metabolic model driven design of a defined medium for *Campylobacter jejuni* M1cam. *Frontiers in Microbiology*, 11(June):1–13, 2020. ISSN 1664302X. doi: 10.3389/fmicb.2020.01072.

M. Terzer. *Large scale methods to enumerate extreme rays and elementary modes.* PhD thesis, ETH Zurich, 2009.

S. Thomas and D. A. Fell. The role of multiple enzyme activation in metabolic flux control. *Advances in Enzyme Regulation*, 38(1):65–85, 1998. ISSN 00652571. doi: 10.1016/S0065-2571(97)00012-5.

E. Torrents. Ribonucleotide reductases: Essential enzymes for bacterial life. *Frontiers in Cellular and Infection Microbiology*, 4(APR):1–9, 2014. ISSN 22352988. doi: 10.3389/fcimb.2014.00052.

C. T. Trinh, A. Wlaschin, and F. Srienc. Elementary mode analysis: A useful metabolic pathway analysis tool for characterizing cellular metabolism. *Applied Microbiol Biotechnol.*, 81(5):813–826, 2009. ISSN 15378276. doi: 10.1007/s00253-008-1770-1.Elementary.

E. Ullah, M. Yosafshahi, and S. Hassoun. Towards scaling elementary flux mode computation. *Briefings in Bioinformatics*, 00(May):1–11, 2019. ISSN 1467-5463. doi: 10.1093/bib/bbz094.

P. Unrean, K. L. Tee, and T. S. Wong. Metabolic pathway analysis for in silico design of efficient autotrophic production of advanced biofuels. *Bioresources and Bioprocessing*, 6(1), 2019. ISSN 21974365. doi: 10.1186/s40643-019-0282-4.

R. R. Vallabhajosyula, V. Chickarmane, and H. M. Sauro. Conservation analysis of large biochemical networks. *Bioinformatics*, 22(3):346–353, 2006. ISSN 13674811. doi: 10.1093/bioinformatics/bti800.

S. Van Dien. From the first drop to the first truckload: Commercialization of microbial processes for renewable chemicals. *Current Opinion in Biotechnology*, 24(6):1061–1068, 2013. ISSN 09581669. doi: 10.1016/j.copbio.2013.03.002.

J. Velayudhan and D. J. Kelly. Analysis of gluconeogenic and anaplerotic enzymes in *Campylobacter jejuni*: An essential role for phosphoenolpyruvate carboxykinase. *Microbiology*, 148(3):685–694, 2002. ISSN 13500872. doi: 10.1099/00221287-148-3-685.

S. Vijayakumar, P. Kaja-Mohideen Sheikh Mujibur Rahman, and C. Angione. A hybrid flux balance analysis and machine learning pipeline elucidates the metabolic response of cyanobacteria to different growth conditions. *iScience*, 23(12), 2020. ISSN 25890042. doi: 10.1016/j.isci.2020.101818.

S. Wagley, J. Newcombe, E. Laing, E. Yusuf, C. M. Sambles, D. J. Studholme, R. M. La Ragione, R. W. Titball, and O. L. Champion. Differences in carbon source utilisation distinguish *Campylobacter jejuni* from *Campylobacter coli*. *BMC Microbiology*, 14(1): 1–10, 2014. ISSN 14712180. doi: 10.1186/s12866-014-0262-y.

H. N. Westfall, D. M. Rollins, and E. Weiss. Substrate utilization by *Campylobacter jejuni* and *Campylobacter coli*. *Applied and Environmental Microbiology*, 52(4):700–705, 1986. ISSN 00992240. doi: 10.1128/aem.52.4.700-705.1986.

P. Willett. Similarity-based virtual screening using 2D fingerprints. *Drug Discovery Today*, 11(23-24):1046–1053, 2006. ISSN 13596446. doi: 10.1016/j.drudis.2006.10.005.

H. P. Williams. Fourier's method of linear programming and its Dual. *The American Mathematical Monthly*, 93(9):681–695, 1986. ISSN 0002-9890. doi: 10.1080/00029890.1986.11971923.

C. A. Woodall, M. A. Jones, P. A. Barrow, J. Hinds, G. L. Marsden, D. J. Kelly, N. Dorrell, B. W. Wren, and D. J. Maskell. *Campylobacter jejuni* gene expression in the chick cecum: Evidence for adaptation to a low-oxygen environment. *Infection and Immunity*, 73(8):5278–5285, 2005. ISSN 00199567. doi: 10.1128/IAI.73.8.5278-5285.2005.

M. M. Wösten, C. H. van de Lest, L. van Dijk, and J. P. van Putten. Function and regulation of the C4-dicarboxylate transporters in *Campylobacter jejuni*. *Frontiers in Microbiology*, 8(FEB):1–13, 2017. ISSN 1664302X. doi: 10.3389/fmicb.2017.00174.

Q. Zhao, Y. Su, Z. Wang, C. Chen, T. Wu, and Y. Huang. Identification of glutathione

(GSH)-independent glyoxalase III from Schizosaccharomyces pombe. *BMC Evolutionary Biology*, 14(1):1–18, 2014. ISSN 14712148. doi: 10.1186/1471-2148-14-86.

W. Zhou and L. Nakhleh. The strength of chemical linkage as a criterion for pruning metabolic graphs. *Bioinformatics*, 27(14):1957–1963, 2011. ISSN 13674803. doi: 10.1093/bioinformatics/btr271.

Appendix A

Metabolite/Reactions Keys

A.1 The Calvin Cycle

Table A.1: The abbreviations, common names, and chemical composition of the metabolites found in the Calvin cycle.

Abbreviation	**Common name**	**Chemical formula**
Ru5P	D-ribulose-5-phosphate	$C_5H_9O_8P$
SBP	D-sedoheptulose-1,7-biphosphate	$C_7H_{12}O_{13}P_2$
DHAP	dihydroxyacetone phosphate	$C_3H_5O_6P$
F6P	D-fructofuranose-6-phosphate	$C_6H_{11}O_9P$
E4P	D-erythrose-4-phosphate	$C_4H_7O_7P$
G1P	D-glucopyranose-1-phosphate	$C_6H_{11}O_9P$
R5P	D-ribose-5-phosphate	$C_5H_9O_8P$
G6P	D-glucopyranose-6-phosphate	$C_6H_{11}O_9P$
GAP	D-glyceraldehyde-3-phosphate	$C_3H_5O_6P$
X5P	D-xylulose-5-phosphate	$C_5H_9O_8P$
RuBP	ribulose-1,5-biphosphate	$C_5H_8O_{11}P_2$
BPGA	3-bisphospho-D-glycerate	$C_3H_4O_{10}P_2$
FBP	D-frunctose-1,6-biphosphate	$C_6H_{10}O_{12}P_2$
S7P	D-sedoheptulose-7-phosphate	$C_7H_{13}O_{10}P$
PGA	3-phospho-D-glycerate	$C_3H_4O_7P$

Table A.2: The enzyme abbreviations and corresponding reactions found in the Calvin cycle model. Metabolites labelled with a ‘x_’ prefix are external metabolites.

Step	**Label**	**Enzyme**	**Abbreviation**	**Stoichiometry**
Carbon fixation	r_1	RuBisCo	N/A	x_CO2 + RuBP + x_H2O ⟶ 2 PGA + 2 x_Proton
Reduction and output to biosynthesis	r_2	PGA kinase	PGK	PGA + ATP ⟷ BPGA + ADP
	r_3	GAP dehydrogenase	GAPdh	BPGA + x_NADPH + x_Proton ⟷ x_NADP + GAP + Pi
	r_4	triose-phosphate isomerase	TPI	GAP ⟷ DHAP
	r_5	FBP aldolase	Ald1	DHAP + GAP ⟷ FBP
	r_6	fructose 1,6-bisphosphatase	FBPase	FBP + x_H2O ⟶ F6P + Pi
Starch synthesis	r_7	phosphoglucose isomerase	PGI	F6P ⟷ G6P
	r_8	phosphoglucomutase	PGM	G6P ⟷ G1P
	r_9	starch synthase	StSynth	G1P + ATP + x_H2O ⟶ ADP + 2Pi + x_Starch + x_Proton
Starch degradation	r_{10}	starch phosphorylase	StPase	x_Starch + Pi ⟶ G1P
RuBP regeneration	r_{11}	F6P transketolase	TKL1	F6P + GAP ⟷ E4P + X5P
	r_{12}	FBP aldolase	Ald2	E4P + DHAP ⟷ SBP
	r_{13}	SBP biphosphatase	SBPase	SBP + x_H2O ⟶ S7P + Pi
	r_{14}	S7P transketolase	TKL2	GAP + S7P ⟷ X5P + R5P
	r_{15}	R5P isomerase	R5Piso	R5P ⟷ Ru5P
	r_{16}	Ru5P epimerase	X5Piso	X5P ⟷ Ru5P
	r_{17}	Ru5P kinase	Ru5Pk	Ru5P + ATP ⟶ RuBP + ADP + x_Proton
Sugar export	N/A	triose-phosphate translocator	TPT_PGA	PGA + x_Pi ⟶ Pi + x_PGA
	N/A	triose-phosphate translocator	TPT_GAP	GAP + x_Pi ⟶ Pi + x_GAP
	N/A	triose-phosphate translocator	TPT_DHAP	DHAP + x_Pi ⟶ Pi + x_DHAP
Light reactions	N/A	N/A	Lreac	ADP + Pi + x_Proton ⟶ ATP + x_H2O

A.2 Glycolysis

Table A.3: The abbreviations, common names, and chemical composition of the metabolites found in the model of glycolysis in Figure 5.3.

Label	Common Name	Empirical Formula
GLC	Glucose	$C_6H_{12}O_6$
G6P	glucose-6-phosphate	$C_6H_{11}O_9P$
F6P	fructose-6-phosphate	$C_6H_{11}O_9P$
FBP	fructose-1,6-biphosphate	$C_6H_{10}O_{12}P_2$
GAP	glyceraldehyde-3-phosphate	$C_3H_5O_6P$
DHAP	dihydroxyacetone phosphate	$C_3H_5O_6P$
BPGA	bisphosphoglycerate	$C_3H_4O_{10}P_2$
PGA	phosphoglycerate	$C_3H_4O_7P$
PEP	phospho-enol-pyruvate	$C_3H_2O_6P$
PYR	pyruvate	$C_3H_3O_3$

Table A.4: The enzyme abbreviations and corresponding reactions found in Figure 5.3.

Label	Common Name	EC number
r_1	glucokinase	2.7.1.1
r_2	phosphoglucose isomerase	5.3.1.9
r_3	phosphofructokinase	2.7.1.11
r_4	FBP aldolase	4.1.2.13
r_5	triose-phosphate isomerase	5.3.1.1
r_6	GAP dehydrogenase (phosphorylating)	1.2.1.12
r_7	PGA kinase	2.7.2.3
r_8	GAP dehydrogenase	1.2.1.9
r_9	enolase	4.2.1.11
r_{10}	pyruvate kinase	2.7.1.40

A.3 *C. jejuni*

Table A.5: The abbreviations, common names, and MetaCyc IDs of the metabolites found in the *C. jejuni* model.

Abbreviation	Common Name	MetaCyc ID
GLN	glutamine	GLN
GLT	glutamate	GLT
2KG	α-ketoglutarate	2-KETOGLUTARATE
SucCoA	succinyl-CoA	SUC-COA
SUC	succinate	SUC
FUM	fumarate	FUM
MAL	malate	MAL
OAA	oxaloacetate	OXALACETIC_ACID
PEP	phosphoenolpyruvate	PHOSPHO-ENOL-PYRUVATE
PGA	phospho-glycerate	2-PG
BPGA	bisphospho-glycerate	DPG
GAP	glyceraldehyde 3-phosphate	GAP
DHAP	dihydroxyacetone phosphate	DIHYDROXY-ACETONE-PHOSPHATE
FBP	fructofuranose 1,6-bisphosphate	FRUCTOSE-16-DIPHOSPHATE
F6P	fructofuranose 6-phosphate	FRUCTOSE-6P
E4P	erythrose 4-phosphate	ERYTHROSE-4P
S7P	seduloheptulose 7-phosphate	D-SEDOHEPTULOSE-7-P
R5P	ribose 5-phosphate	RIBOSE-5P
X5P	xylulose 5-phosphate	XYLULOSE-5-PHOSPHATE
Ru5P	ribulose 5-phosphate	RIBULOSE-5P
EN4P	erythronate 4-phosphate	ERYTHRONATE-4P
PAKB	hydroxy-2-oxo-4 phosphooxybutanoate	3OH-4P-OH-ALPHA-KETOBUTYRATE
POT	phosphooxy-threonine	4-PHOSPHONOOXY-THREONINE
AHAP	amino-1-hydroxyacetone 1-phosphate	1-AMINO-PROPAN-2-ONE-3-PHOSPHATE
DX5P	deoxy-xylulose 5-phosphate	DEOXYXYLULOSE-5P
PNP	pyridoxine 5′-phosphate	PYRIDOXINE-5P
PLP	pyridoxal 5′-phosphate	PYRIDOXAL_PHOSPHATE
PYR	pyruvate	PYRUVATE
Mq	menaquinol	MENAQUINOL
MqH	menaquinone	MENAQUINONE
Cy-Ox	cytochrome c oxidised	Cytochromes-C-Oxidized
Cy-Rd	cytochrome c reduced	Cytochromes-C-Reduced
Fd-Ox	oxidised ferredoxin	Oxidized-ferredoxins
Fd-Rd	reduced ferredoxin	Reduced-ferredoxins

Table A.6: The abbreviations, enzyme, and EC number of the reactions found in diagrams of the *C. jejuni* model.

Label	**Enzyme**	**EC number**
R1	glutaminase	3.5.1.38
R2a	glutamate synthase	1.4.7.1
R2b	glutamate dehydrogenase	1.4.1.4
R3	2-oxoglutarate synthase	1.2.7.3
R4	succinyl-CoA synthase	6.2.1.5
R5	succinate dehydrogenase	1.3.5.1
R6	fumerase	4.2.1.2
R7a	malate oxidoreductase (quinone)	1.1.5.4
R7b	malate dehydrogenase	1.1.1.37
R8	phosphoenolpyruvate carboxykinase	4.1.1.49
R9	enolase	4.2.1.11
R10	phosphoglycerate kinase	2.7.2.3
R11	glyceraldehyde 3-phosphate dehydrogenase	1.2.1.12
R12	triose-phosphate isomerase	5.3.1.1
R13	fructose-bisphosphate aldolase	4.1.2.13
R14	fructose 1,6-bisphosphatase	3.1.3.11
R15	fructofuranose 6-phosphate transketolase	2.2.1.1
R16	erythrose 4-phosphate dehydrogenase	1.2.1.72
R17	erythronate 4-phosphate dehydrogenase	1.1.1.290
R18	phosphohydroxythreonine aminotransferase	2.6.1.52
R19	hydroxythreonine 4-phosphate dehydrogenase	1.1.1.262
R20	deoxy-xylulose 5-phosphate synthase	2.2.1.7
R21	pyridoxine 5′-phosphate synthase	2.6.99.2
R22	transaldolase	2.2.1.2
R23	seduloheptulose 7-phosphate transketolase	2.2.1.1
R24	ribose 5-phosphate isomerase	5.3.1.6
R25	ribulose phosphate 3-epimerase	5.1.3.1
R26	nitrite reductase (NAD)	1.7.1.4
R27	oxygen reductase (cytochrome)	7.1.1.7
R28	nitrite reductase (cytochrome)	1.7.2.2
R29	nitrate reductase (cytochrome)	1.9.6.1
R30	pyridoxine 5′-phosphate oxidase	1.4.3.5
R31	catalase	1.11.1.6
R32	NADH peroxidase	1.11.1.1
R33	nitrate reductase (NAD)	1.7.99.4
R34	carbonic anhydrase	4.2.1.1
CV	proton translocating ATP synthase	7.1.2.2

Appendix B

Defining The Steady-State Solution Space

Consider the flux cone C, generated by an $m \times r$ stoichiometry matrix, $\mathbf{N}$,

$$C = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{N}\mathbf{v} = \mathbf{0}, \mathbf{v} \geq 0\}. \tag{B.1}$$

Definition B.0.0.1. *Pointed polyhedral cones are in general defined as a system of linear inequality constraints,* $\mathbf{A}\mathbf{v} \leq \mathbf{0}$ *where* $\mathbf{A}$ *has full column-rank.*

As discussed by Gagneur and Klamt [2004], C can be expressed as a pointed polyhedral cone:

$$C = \{\mathbf{v} \in \mathbb{R}^r | \mathbf{A}\mathbf{v} \leq \mathbf{0}\}, \tag{B.2}$$

where

$$\mathbf{A} = \begin{pmatrix} \mathbf{N} \\ -\mathbf{N} \\ -\mathbf{I}_{r\times r} \end{pmatrix}. \tag{B.3}$$

Including $-\mathbf{I}_{r\times r}$ in $\mathbf{A}$ ensures that $\mathbf{v} \geq \mathbf{0}$ and that $\mathbf{A}$ has full column-rank.

Reversible reactions can be included in this definition by splitting them into two irreversible forward and backward components as follows.

For simplicity, re-order its columns such that the reversible reactions consist of the first p columns. Then, these columns can be split into two, i.e. forwards and backwards components, $\mathbf{n}_i^+$ and $\mathbf{n}_i^-$, such that $\mathbf{n}_i^+ = \mathbf{n}_i$ and $\mathbf{n}_i^- = -\mathbf{n}_i$ with the corresponding fluxes being $\mathbf{v}_i^+, \mathbf{v}_i^- \geq \mathbf{0}$ (in the same manner as free variables are split in LP), creating a reconfigured $m \times q$ stoichiometry matrix $\bar{\mathbf{N}}$ where $q = r + p$, such that the columns of $\bar{\mathbf{N}}$ are $\mathbf{n}_1^+, \mathbf{n}_1^-, \ldots, \mathbf{n}_p^+, \mathbf{n}_p^-, \mathbf{n}_{p+1}, \ldots, \mathbf{n}_r$.

Therefore the matrix $\mathbf{A}$ in Equation (B.2) discussed above can be modified to create the matrix $\bar{\mathbf{A}}$ such that:

$$\bar{\mathbf{A}} = \begin{pmatrix} \bar{\mathbf{N}} \\ -\bar{\mathbf{N}} \\ -\mathbf{I}_{q\times q} \end{pmatrix}. \tag{B.4}$$

This produces the following flux cone

$$C = \{\bar{\mathbf{v}} \in \mathbb{R}^q | \quad \bar{\mathbf{A}}\bar{\mathbf{v}} \leq \mathbf{0}\}. \tag{B.5}$$

As discussed by Gagneur and Klamt [2004], any flux vector $\bar{\mathbf{v}}$ can be converted back to the corresponding vector $\mathbf{v}$ by subtracting the components of each reversible reaction ($v_i = \bar{v}_i^+ - \bar{v}_i^-$ for all indices i corresponding to reversible reactions). These authors proved that EMs resulting from such a reconfigured network are equivalent to the EMs of the original network (generated by $\mathbf{N}$), with the exception of two-cycle EMs generated by the forward and backwards component of the same reversible reaction.

Appendix C

Results From Chapter 3

Table C.1: The EMs of the Calvin cycle model.

	Relative Flux Measurements							
Reactions	**ElMo_0**	**ElMo_1**	**ElMo_2**	**ElMo_3**	**ElMo_4**	**ElMo_5**	**ElMo_6**	**ElMo_7**
RuBisCo	0	3	9	9	3	3	3	6
PGK	0	6	15	18	6	3	6	12
G3Pdh	0	6	15	18	6	3	6	12
TPI	0	3	6	6	4	1	1	5
Ald1	0	1	3	3	0	0	0	3
FBPase	0	1	3	3	0	0	0	3
PGI	0	0	0	0	-1	-1	-1	1
PGM	0	0	0	0	-1	-1	-1	1
StSynth	1	0	0	0	0	0	0	1
StPase	1	0	0	0	1	1	1	0
TKL1	0	1	3	3	1	1	1	2
Ald2	0	1	3	3	1	1	1	2
SBPase	0	1	3	3	1	1	1	2
TKL2	0	1	3	3	1	1	1	2
R5Piso	0	1	3	3	1	1	1	2
X5Piso	0	2	6	6	2	2	2	4
Ru5Pk	0	3	9	9	3	3	3	6
TPT_PGA	0	0	3	0	0	3	0	0
TPT_GAP	0	0	0	3	0	0	3	0
TPT_DHAP	0	1	0	0	3	0	0	0
LReac	1	9	24	27	9	6	9	19

Table C.2: A set of flux vectors corresponding to the reactions within the Calvin cycle (obtained via Flux Constraint Scanning). A list of reaction stoichiometries and abbreviations may be found in Appendix A.

	Relative Flux Measurements as Constrained by the Corresponding PGA Export Flux										
Reactions	**PGA_0**	**PGA_5**	**PGA_10**	**PGA_15**	**PGA_20**	**PGA_25**	**PGA_30**	**PGA_35**	**PGA_40**	**PGA_45**	**PGA_50**
RuBisCo	0.00	5.00	10.00	15.00	20.00	25.00	30.00	35.00	40.00	45.00	50.00
PGK	0.00	5.00	10.00	15.00	20.00	25.00	30.00	35.00	40.00	45.00	50.00
G3Pdh	0.00	5.00	10.00	15.00	20.00	25.00	30.00	35.00	40.00	45.00	50.00
TPI	0.00	1.67	3.33	5.00	6.67	8.33	10.00	11.67	13.33	15.00	16.67
Ald1	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
FBPase	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
PGI	0.00	-1.67	-3.33	-5.00	-6.67	-8.33	-10.00	-11.67	-13.33	-15.00	-16.67
PGM	0.00	-1.67	-3.33	-5.00	-6.67	-8.33	-10.00	-11.67	-13.33	-15.00	-16.67
StSynth	100.00	90.00	80.00	70.00	60.00	50.00	40.00	30.00	20.00	10.00	0.00
StPase	100.00	91.67	83.33	75.00	66.67	58.33	50.00	41.67	33.33	25.00	16.67
TKL1	0.00	1.67	3.33	5.00	6.67	8.33	10.00	11.67	13.33	15.00	16.67
Ald2	0.00	1.67	3.33	5.00	6.67	8.33	10.00	11.67	13.33	15.00	16.67
SBPase	0.00	1.67	3.33	5.00	6.67	8.33	10.00	11.67	13.33	15.00	16.67
TKL2	0.00	1.67	3.33	5.00	6.67	8.33	10.00	11.67	13.33	15.00	16.67
R5Piso	0.00	1.67	3.33	5.00	6.67	8.33	10.00	11.67	13.33	15.00	16.67
X5Piso	0.00	3.33	6.67	10.00	13.33	16.67	20.00	23.33	26.67	30.00	33.33
Ru5Pk	0.00	5.00	10.00	15.00	20.00	25.00	30.00	35.00	40.00	45.00	50.00
TPT_PGA	0.00	5.00	10.00	15.00	20.00	25.00	30.00	35.00	40.00	45.00	50.00
TPT_GAP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
TPT_DHAP	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
LReac	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00	100.00

Table C.3: The decomposition of relative flux values obtained from Flux Constraint Scanning (found in Table C.2) into a set of EMs.

PGA (flux constraint)	ElMo_0	ElMo_5
0.00	100.00	0.00
5.00	90.00	1.67
10.00	80.00	3.33
15.00	70.00	5.00
20.00	60.00	6.67
25.00	50.00	8.33
30.00	40.00	10.00
35.00	30.00	11.67
40.00	20.00	13.33
45.00	10.00	15.00
50.00	0.00	16.67

Appendix D

The Left Null-Space

D.1 The Link Matrix

As first defined by Reder [1988], the link matrix, $\mathbf{L}$, transforms the stoichiometry matrix by removing its linearly dependent rows, such that the resulting reduced stoichiometry matrix, denoted by $\mathbf{N_r}$, has full row-rank.

Let the internal left null-space matrix, $\mathbf{G}$, be $m \times q$, where q is the rank of $\mathbf{G}$. Then, $\mathbf{N}$ has q linearly-dependent rows and $p = (m - q)$ linearly-independent rows (i.e. $p = \mathrm{Rank}(\mathbf{N})$). Therefore, every linearly-dependent row can be formed through a linear combination of some of the p linearly-independent rows.

Thus the stoichiometry matrix can be partitioned into independent and dependant components ($\mathbf{N}_\mathrm{r}$ and $\mathbf{N}_\mathrm{d}$, respectively):

$$\mathbf{N} = \begin{bmatrix} \mathbf{N}_\mathrm{r} \\ \mathbf{N}_\mathrm{d} \end{bmatrix}, \tag{D.1}$$

where $\mathbf{N_r}$ is $p \times r$ and $\mathbf{N_d}$ is $q \times r$.

These two matrices are related through the link matrix, $\mathbf{L}$,

$$\mathbf{L} = \begin{bmatrix} \mathbf{I}_{p\times p} \\ \mathbf{L}_0 \end{bmatrix}, \tag{D.2}$$

which is defined such that

$$\mathbf{L}_0\mathbf{N}_\mathrm{r} = \mathbf{N}_\mathrm{d}, \tag{D.3}$$

or, equivalently,

$$\begin{bmatrix} \mathbf{I}_{p\times p} \\ \mathbf{L}_0 \end{bmatrix} \mathbf{N}_\mathrm{r} = \begin{bmatrix} \mathbf{N}_\mathrm{r} \\ \mathbf{N}_\mathrm{d} \end{bmatrix} = \mathbf{N}, \tag{D.4}$$

where $\mathbf{L}_0$ is calculated by transforming $\mathbf{G}$ into the following form:

$$\mathbf{G} = \begin{bmatrix} \mathbf{L}_0^\top \\ -\mathbf{I}_{q\times q} \end{bmatrix}. \tag{D.5}$$

As a consequence of Equations D.4 and D.3, the vector of metabolite concentrations, $\mathbf{s}$, can also be partitioned into dependent and independent components, such that the stoichiometry of the reactions in a system can assume the form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix} \mathbf{s}_\mathrm{r} \\ \mathbf{s}_\mathrm{d} \end{bmatrix} = \begin{bmatrix} \mathbf{I}_{p\times p} \\ \mathbf{L}_0 \end{bmatrix} \boldsymbol{N}_\mathrm{r}\mathbf{v}, \tag{D.6}$$

and therefore,

$$\frac{\mathrm{d}\mathbf{s}_\mathrm{d}}{\mathrm{d}t} = \mathbf{L}_0 \frac{\mathrm{d}\mathbf{s}_\mathrm{r}}{\mathrm{d}t}. \tag{D.7}$$

Thus showing that the number of metabolites whose concentrations need to be known to calculate all of the concentration changes in the system is equivalent to the rank of $\mathbf{N}$, p.

D.2 The Atomic Matrix

Proposition D.2.0.1. *The left null-space of the external stoichiometry matrix can be constructed such that its basis includes the linearly independent vectors of the atomic matrix,* $\mathbf{A}$*, (along with some additional vectors if* $\mathrm{rank}(\mathbf{A}) < \mathrm{rank}(\ker(\mathcal{N}^\top))$*).*

Proof. Let the external stoichiometry matrix, $\mathcal{N}$, be an $l \times r$ matrix, where l is the number of metabolites in the system and r is the number of reactions.

The left null-space of $\mathcal{N}$ consists of all the vectors, $\mathbf{g}$, that satisfy

$$\mathcal{N}^\top \mathbf{g} = \mathbf{0}. \tag{D.8}$$

Let $\mathcal{G}$ be an lxt matrix whose columns are a set of basis vectors for the left null-space of $\mathcal{N}$:

$$\mathcal{G} = \{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_t\}. \tag{D.9}$$

Suppose the the atomic matrix, $\mathbf{A}$, is an $l \times p$ matrix, with rank k, and defined by the columns:

$$\mathbf{A} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k, \ldots, \mathbf{a}_p\}, \tag{D.10}$$

where, without loss of generality, the columns are ordered such that the first k columns are linearly independent. Therefore, $\mathbf{A}$ contains k linearly independent column vectors that form a basis, $\tilde{\mathbf{A}}$, for $\mathrm{Span}(\mathbf{A})$, i.e.

$$\tilde{\mathbf{A}} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k\}. \tag{D.11}$$

As discussed in Chapter 5.2.2, every column vector of $\mathbf{A}$ satisfies Equation (D.8). Therefore, the space generated by the columns of $\mathbf{A}$ must be a subset of the left null-space of $\mathcal{N}$, i.e.

$$\mathrm{Span}(\mathbf{A}) \subseteq \mathrm{Span}(\mathcal{G}). \tag{D.12}$$

There are three different scenarios to consider.

(1) **Suppose $t < k$.**

As the dimension of a space is equal to the number of basis vectors generating it, then

$$\dim(\text{Span}(\mathcal{G})) < \dim(\text{Span}(\mathbf{A})). \tag{D.13}$$

But since all of the vectors of $\mathbf{A}$ are elements of the left null-space, then

$$\text{Span}(\mathbf{A}) \subseteq \text{Span}(\mathcal{G}). \tag{D.14}$$

Therefore,

$$\dim(\text{Span}(\mathbf{A})) \leq \dim(\text{Span}(\mathcal{G})), \tag{D.15}$$

which is a contradiction.

Hence, if the dimension of the left null-space is less than the number of assumed species in the model, some of the column vectors of $\mathbf{A}$, (at least p - t), must be linearly dependent.

(2) **Suppose $t > k$.**

The Steinitz exchange lemma states that any finite linearly independent set in a vector space can be extended to form a basis.

Therefore, if $\tilde{\mathbf{A}} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k\}$ is a set of linearly independent vectors of the left null-space, and $\mathcal{G} = \{\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_t\}$ is a basis for the left null-space, than there exist an alternative basis for the left null-space such that

$$\text{Span}(\mathcal{G}) = \text{Span}(\{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_p, \tilde{\mathbf{g}}_{k+1}, \ldots, \tilde{\mathbf{g}}_t\}), \tag{D.16}$$

where $\tilde{\mathcal{G}} = \{\tilde{\mathbf{g}}_1, \tilde{\mathbf{g}}_2, \ldots, \tilde{\mathbf{g}}_t\}$ is an appropriate re-ordering of $\mathcal{G}$.

(3) **Suppose $t = p$.**

Since $\text{Span}(\mathbf{A}) \subseteq \text{Span}(\mathcal{G})$, and any two bases of a vector space must have the same number of elements, then $\tilde{\mathbf{A}} = \{\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_k\}$ must be an alternative basis for the left null-space. □

D.3 The Similarity Measure

D.3.1 The Similarity Measure, ϕ, is Invariant to the Choice of Orthonormal Basis

This subsection shows that the similarity measure, ϕ^G, defined by Equation (5.16), is invariant to the choice of left null-space basis, $\mathbf{G}$, given that $\mathbf{G}$ is orthonormal. The following is derived from Lemma 1 and Theorem 1 in Poolman et al. [2007].

Definition D.3.1.1. *If a matrix $\mathcal{A}$ is defined here as an orthonormal basis, then the columns of $\mathcal{A}$ are normalized basis vectors where each column vector is orthogonal to every other column vector.*

Let $\mathcal{A}$ and $\mathcal{B}$ be two different orthonormal bases for the same space, of size $m \times q$.

Let the xth row of $\mathcal{A}$ be denoted by $\mathcal{A}_x$ and the ith column of $\mathcal{A}$ be $\mathbf{a}_i$, such that the matrices $\mathcal{A}$ and $\mathcal{B}$ can be represented by the set of their columns $\mathcal{A} = [\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_q]$ and $\mathcal{B} = [\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_q]$.

Then, the dot product between rows x and y can be denoted by $\mathcal{A}_x\mathcal{A}_y^\top$ such that the similarity measure applied to the rows x and y, $\phi_{xy}^{\mathcal{A}}$, is

$$\phi_{xy}^{\mathcal{A}} = \cos(\theta_{xy}^{\mathcal{A}}) = \frac{\mathcal{A}_x\mathcal{A}_y^\top}{\sqrt{\mathcal{A}_x\mathcal{A}_x^\top}\sqrt{\mathcal{A}_y\mathcal{A}_y^\top}}. \tag{D.17}$$

Therefore, to demonstrate that this measure is independent of the choice of basis, it is sufficient to show that the dot product between any two rows of an orthonormal basis is invariant to the choice of basis i.e. $\mathcal{A}_x\mathcal{A}_y^\top = \mathcal{B}_x\mathcal{B}_y^\top$

Lemma D.3.1.1. *The product $\mathcal{A}\mathcal{A}^\top$ where $\mathcal{A}$ is an orthonormal basis is invariant of the choice of basis.*

Proof. As defined previously in this section, let $\mathcal{A}$ and $\mathcal{B}$ be two orthonormal basis for the same space.

As the columns of $\mathcal{A}$ and $\mathcal{B}$ are orthonormal, then the matrices are orthogonal and satisfy

$$\mathcal{A}^\top\mathcal{A} = \mathbf{I}_{q\times q} = \mathcal{B}^\top\mathcal{B}. \tag{D.18}$$

Moreover, since the columns of the matrices $\mathcal{A}$ and $\mathcal{B}$ are two orthonormal bases that span the same space, these matrices are related by an orthogonal matrix $\mathbf{P}$, such that

$$\mathcal{A} = \mathcal{B}\mathbf{P}. \tag{D.19}$$

Therefore,

$$\begin{aligned}\mathcal{A}\mathcal{A}^\top &= \mathcal{B}\mathbf{P}(\mathcal{B}\mathbf{P})^\top = \mathcal{B}\mathbf{P}\mathbf{P}^\top\mathcal{B}^\top \\ &= \mathcal{B}\mathbf{I}\mathcal{B}^\top = \mathcal{B}\mathcal{B}^\top,\end{aligned} \tag{D.20}$$

and hence,

$$\mathcal{A}\mathcal{A}^\top = \mathcal{B}\mathcal{B}^\top. \tag{D.21}$$

□

Theorem D.3.1.1. *ϕ_{xy} is invariant to the choice of orthonormal basis.*

Proof. Consider the two orthonormal bases $\mathcal{A}$ and $\mathcal{B}$, as discussed above.

As defined by Equation (D.17), the similarity measure applied to the rows x and y of the matrix $\mathcal{A}$ is:

$$\phi_{xy}^{\mathcal{A}} = \cos(\theta_{xy}^{\mathcal{A}}) = \frac{\mathcal{A}_x \mathcal{A}_y^\top}{\sqrt{\mathcal{A}_x \mathcal{A}_x^\top}\sqrt{\mathcal{A}_y \mathcal{A}_y^\top}}. \tag{D.22}$$

As shown by Lemma D.3.1.1 $\mathcal{A}\mathcal{A}^\top = \mathcal{B}\mathcal{B}^\top$, and consequentially $\mathcal{A}_x \mathcal{A}_y^\top = \mathcal{B}_x {\mathcal{B}_y}^\top$. Therefore

$$\phi_{xy}^{\mathcal{A}} = \frac{\mathcal{A}_x \mathcal{A}_y^\top}{\sqrt{\mathcal{A}_x \mathcal{A}_x^\top}\sqrt{\mathcal{A}_y \mathcal{A}_y^\top}} = \frac{\mathcal{B}_x \mathcal{B}_y^\top}{\sqrt{\mathcal{B}_x \mathcal{B}_x^\top}\sqrt{\mathcal{B}_y \mathcal{B}_y^\top}} = \phi_{xy}^{\mathcal{B}}. \tag{D.23}$$

□

D.3.2 The Pearson Correlation Coefficient

Following the approach of Poolman et al. [2007], this subsection shows that applying the similarity measure to the rows of an orthonormal left null-space basis, is equivalent to the Pearson's Population Correlation Coefficient, ρ, applied to the same rows when given all possible vectors that are spanned by the basis.

Therefore, ϕ_{xy}^G applied to the left null-space of the internal stoichiometry matrix is equivalent to ρ_{xy} applied to all potential conservation relations of the system. While, ϕ_{xy}^G applied to the left null-space of the external stoichiometry matrix is equivalent to ρ_{xy} applied to all potential vectors spanned by the space (including the maximal moiety vectors, $\mathbf{z}$).

The Pearson's Correlation Coefficient, ρ, is defined as follows.

Two variables are said to be correlated when changes in one variable are met with similar changes in the other. Consider a pair of variables $\mathbf{x}$ and $\mathbf{y}$. Then, the Pearson's population correlation coefficient, ρ_{xy}, is given by

$$\rho_{xy} = \frac{\mathrm{Cov}(\mathbf{x}, \mathbf{y})}{\sigma_x \sigma_y}, \tag{D.24}$$

where σ_x and σ_y are the standard deviation of $\mathbf{x}$ and $\mathbf{y}$ respectively, and $\mathrm{Cov}(\mathbf{x}, \mathbf{y})$ is the covariance [Field et al., 2012, pages 456,461].

The standard deviation measures the dispersion of the data within a variable, $\mathbf{x}$, in relation to its mean, $\overline{x}$, where a low value indicates that data-points are clustered close to the mean, and is defined as

$$\sigma_x^2 = \frac{1}{N}\sum_{i=1}^{n}(x_i - \overline{x})^2, \tag{D.25}$$

where N is the size of the population.

The covariance is a measure of the linear association between the two variables, namely

$$\mathrm{Cov}(\mathbf{x}, \mathbf{y}) = \frac{1}{N}\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y}). \tag{D.26}$$

A positive covariance indicates that as the values of one variable deviate from its mean in one direction so do the values of the other variable (with respect to their corresponding mean), e.g both variables increase. Deviation in the opposite directions results in a negative covariance. The covariance is unit-dependent and thus does not reveal the strength of the linear association between $\mathbf{x}$ and $\mathbf{y}$. This problem is counteracted in ρ_{xy} by accounting for the standard deviation of the individual variables.

Substituting Equations (D.25) and (D.26) into (D.24) yields an approximate formula for Pearson's Correlation Coefficient, referred to as Pearson's Sample Correlation Coefficient and denoted by r_{XY}:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(x_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}}, \tag{D.27}$$

where a value of $r_{xy} = \pm 1$ indicates a perfect linear association between the variables, where one increases proportionally to the other if $r_{xy} = 1$ and inversely if $r_{xy} = -1$. The coefficient is zero if there is no linear relationship between the variables.

Proposition D.3.2.1. *Measuring the cosine of the angle between two rows, x and y, of an orthonormal left null-space basis, ϕ_{xy}^{G}, is equivalent to measuring Pearson's Correlation Coefficient, ρ_{xy}, between the corresponding rows of a matrix that spans the set of all possible conservation relations of the network.*

Proof. Let $\mathbf{G}$ be an $m \times q$ orthonormal left null-space matrix for an internal stoichiometry matrix $\mathbf{N}$, with m metabolites and r reactions. Consider a $q \times j$ matrix of random numbers, $\mathbf{Q}$, whose elements are independently and identically distributed from a distribution with mean μ and variance σ^2, where q is the dimension of the left null-space and j is arbitrarily large.

Let $\mathbf{q}_i$ denote the ith column of $\mathbf{Q}$ and let $\mathbf{Q}_x$ denote the xth row of $\mathbf{Q}$, such that Q_{xi} is the element at row x and column i.

Since $\mathbf{G}$ is a basis for the left null-space of $\mathbf{N}$,

$$\mathbf{N}^{\top}\mathbf{G} = \mathbf{O}_{r \times q}. \tag{D.28}$$

Define an $m \times j$ matrix $\mathbf{P}$ such that $\mathbf{P} = \mathbf{GQ}$. Then,

$$\mathbf{N}^{\top}\mathbf{P} = \mathbf{N}^{\top}\mathbf{GQ} = \mathbf{O}_{r \times j}. \tag{D.29}$$

Therefore any column of $\mathbf{P}$, $\mathbf{p}_i$, is within the left null-space of $\mathcal{N}$.

As previously, let the xth row of $\mathbf{P}$ be denoted by $\mathbf{P}_x$, such that P_{xi} refers to the element of $\mathbf{P}$ at row x and column i.

As defined by Equation (D.27), Pearson's sample correlation coefficient, r_{xy}, between the rows x and y (metabolites) of $\mathbf{P}$ is:

$$r_{xy} = \frac{\sum_{i=1}^{j}(P_{xi} - \bar{\mathbf{P}}_x)(P_{yi} - \bar{\mathbf{P}}_y)}{\sqrt{\sum_{i=1}^{j}(P_{xi} - \bar{\mathbf{P}}_x)^2}\sqrt{\sum_{i=1}^{j}(P_{yi} - \bar{\mathbf{P}}_y)^2}}, \tag{D.30}$$

where $P_{xi} =< \mathbf{G}_x, \mathbf{q}_i^\top >= \mathbf{G}_x\mathbf{q}_i$. and the sample mean of row x in $\mathbf{P}$, $\bar{\mathbf{P}}_x$, is

$$\bar{\mathbf{P}}_x = \frac{\sum_{i=1}^{j} \mathbf{G}_x\mathbf{q}_i}{j} = \mathbf{G}_x\frac{\sum_{i=1}^{j} \mathbf{q}_i}{j} = \mathbf{G}_x\widehat{\boldsymbol{\mu}}. \tag{D.31}$$

Where $\widehat{\boldsymbol{\mu}}$ is a vector of length q such that each element consists of the mean value of the corresponding row of $\mathbf{Q}$.

Therefore,

$$r_{xy} = \frac{\sum_{i=1}^{j}(\mathbf{G}_x\mathbf{q}_i - \mathbf{G}_x\widehat{\boldsymbol{\mu}})(\mathbf{G}_y\mathbf{q}_i - \mathbf{G}_y\widehat{\boldsymbol{\mu}})^\top}{\sqrt{\sum_{i=1}^{j}(\mathbf{G}_x\mathbf{q}_i - \mathbf{G}_x\widehat{\boldsymbol{\mu}})(\mathbf{G}_x\mathbf{q}_i - \mathbf{G}_x\widehat{\boldsymbol{\mu}})^\top}\sqrt{\sum_{i=1}^{j}(\mathbf{G}_y\mathbf{q}_i - \mathbf{G}_y\widehat{\boldsymbol{\mu}})(\mathbf{G}_y\mathbf{q}_i - \mathbf{G}_y\widehat{\boldsymbol{\mu}})^\top}}. \tag{D.32}$$

However,

$$(\mathbf{G}_x\mathbf{q}_i - \mathbf{G}_x\widehat{\boldsymbol{\mu}})(\mathbf{G}_y\mathbf{q}_i - \mathbf{G}_y\widehat{\mu})^\top = \mathbf{G}_x(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})^\top\mathbf{G}_y^\top, \tag{D.33}$$

and thus

$$r_{xy} = \frac{\mathbf{G}_x \sum_{i=1}^{j}(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})^\top\mathbf{G}_y^\top}{\sqrt{\mathbf{G}_x(\sum_{i=1}^{j}(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})^\top)\mathbf{G}_x^\top}\sqrt{\mathbf{G}_y(\sum_{i=1}^{j}(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})^\top)\mathbf{G}_y^\top}}. \tag{D.34}$$

Let $\mathbf{V} = \frac{1}{j}\sum_{i=1}^{j}(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})(\mathbf{q}_i - \widehat{\boldsymbol{\mu}})^\top$.

Note that $\mathbf{V}$ is a matrix where as j becomes sufficiently large the leading diagonal of $\mathbf{V}$ approaches σ^2 and all other elements approach 0:

$$\lim_{j\to\infty} \mathbf{V} = \sigma^2\mathbf{I}_{q\times q}. \tag{D.35}$$

This is because the co-variance of any two rows, with index x and y, of $\mathbf{Q}$ is $\frac{1}{j}\sum_{i=1}^{j}(\mathbf{q}_{xi} - \widehat{\mu}_x)(\mathbf{q}_{yi} - \widehat{\mu}_y)$ which converges to σ^2 if $x = y$ (using the weak law of large numbers), and to 0 otherwise, as $j \to \infty$ (since the covariance of two independent random variables is 0).

Hence,

$$\begin{aligned}\lim_{j\to\infty} r_{xy} = \rho_{xy} &= \frac{\mathbf{G}_x\sigma^2\mathbf{I}_{q\times q}\mathbf{G}_y^\top}{\sqrt{\mathbf{G}_x\sigma^2\mathbf{I}_{q\times q}\mathbf{G}_x^\top}\sqrt{\mathbf{G}_y\sigma^2\mathbf{I}_{q\times q}\mathbf{G}_y^\top}} = \frac{\sigma^2\mathbf{G}_x\mathbf{G}_y^\top}{\sigma^2\sqrt{\mathbf{G}_x\mathbf{G}_x^\top}\sqrt{\mathbf{G}_y^\top\mathbf{G}_y^\top}} \\ &= \frac{\mathbf{G}_x\mathbf{G}_y^\top}{\sqrt{\mathbf{G}_x\mathbf{G}_x^\top}\sqrt{\mathbf{G}_y^\top\mathbf{G}_y^\top}} = \cos(\theta_{xy}^{\mathbf{G}}) = \phi^G.\end{aligned} \tag{D.36}$$

□

D.3.3 A ϕ of Zero.

Proposition D.3.3.1. *If there exists no conservation relation, $\mathbf{g}_i$, such that $g_{xi} \neq 0$ and $g_{yi} \neq 0$, then $\phi_{xy}^G = 0$.*

Proof. Recall the $m \times j$ matrix $\mathbf{P}$ and the $m \times q$ orthogonal left null-space matrix $\mathbf{G}$ defined in Proposition D.3.2.1 (Equation D.29). As $j \to \infty$ the columns of $\mathbf{P}$ consist of the collection of vectors that can be spanned by the columns of $\mathbf{G}$ i.e. all of the potential conservation relations of the system. Suppose that there exists no conservation relation $\mathbf{g}_i$, such that $g_{xi} \neq 0$ and $g_{yi} \neq 0$, then there must be no column in $\mathbf{P}$ such that both $P_{xi} \neq 0$ and $P_{yi} \neq 0$. Therefore, $\rho_{xy} = 0$ and $\phi^G_{xy} = 0$.

However, note that the reverse is not necessarily true.

□

D.3.4 A ϕ of One

As shown by Proposition D.3.2.1, applying the similarity measure to the rows of an orthonormal left null-space basis, $\mathbf{G}$, is equivalent to Pearson's Correlation Coefficient, ρ, applied to the rows of a matrix $\mathbf{P}$ having infinitely many columns that correspond to the possible conservation relations of the network (i.e. the vectors that can be spanned by the columns $\mathbf{G}$). Therefore, for any two rows (metabolites) x and y where $\phi^G_{xy} = 1$, the rows x and y in $\mathbf{P}$ are proportional to each other such that there exists a constant, λ, where for any any left null-space vector, $\mathbf{g}$, spanned by the columns of $\mathbf{G}$, $g_x = \lambda g_y$.

Hence, proportional rows in the internal left null-space matrix $\mathbf{G}$, must be present in the same proportion in all conservation relations. Whilst, proportional rows in the external left null-space matrix $\mathcal{G}$, must be present in the same proportion in all maximal conserved moiety vectors, $\mathbf{z}$, including the column vectors of the atomic matrix $\mathbf{A}$.

Moreover, if two metabolites, x and y are parallel in the internal left null-space, then it is not possible to construct a link matrix, $\mathbf{L}$, where the metabolites are considered to be independent.

For example, consider a four-metabolite system with the following two conservation relations, in which the metabolites A and B are parallel in the left null-space matrix, such that their concentration changes are:

$$\mathbf{g}_1 : \frac{\mathrm{dA}}{\mathrm{d}t} + \frac{\mathrm{dB}}{\mathrm{d}t} + \frac{\mathrm{dC}}{\mathrm{d}t} = 0 \text{ and } \mathbf{g}_2 : \frac{\mathrm{dA}}{\mathrm{d}t} + \frac{\mathrm{dB}}{\mathrm{d}t} + \frac{\mathrm{dD}}{\mathrm{d}t} = 0, \tag{D.37}$$

where A, B, C, and D are metabolite concentrations. A left null-space matrix of the form as in Equation D.5 for this system is:

$$\mathbf{G} = \begin{pmatrix} \mathbf{L}_0^\top \\ -\mathbf{I} \end{pmatrix} = \begin{array}{c} \mathrm{A} \\ \mathrm{B} \\ \mathrm{C} \\ \mathrm{D} \end{array} \begin{pmatrix} -1 & -1 \\ -1 & -1 \\ -1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{D.38}$$

Since the rows pertaining to metabolites A and B are parallel in $\mathbf{G}$, there is no possible permutation of the rows of $\mathbf{G}$ through which metabolites A or B can form part of an identity sub-matrix in $\mathbf{G}$, which is required for their concentration to be considered independent in the link matrix $\mathbf{L}$.

Appendix E

Source-Code

These algorithms described in this thesis are implemented as add-ons to ScrumPy which can be installed using the following link: `mudshark.brookes.ac.uk/ScrumPy`.

The code for Algorithm 1 in Chapter 3 is released as part of ScrumPy (as the *LPEMs* module in the *Structural* directory). The source code for the algorithms in Chapters 4 and 5 is available in the appended CD (hard-bound thesis only) or alternatively, from the author upon request. It is organised in the following directories: *Util*, *Compress*, and *LeftNS*.

***Util*:** modules that provide helper functions that are used by the algorithms in the below modules (such as a general *tree* class and functions that manipulate datasets)

***Compress*:** modules that implement the algorithms presented in Chapter 4:

- *Compress*: a class called *CompressModel* that implements the algorithms described in Chapter 4.3 as its attributes, using helper functions and classes as defined below. This class requires a model instance as an input. As discussed in Chapter 4.3, Algorithm 2 is implemented as the attribute function *Compress* which modifies the class's associated model, while Algorithms 3 and 4 are called *OriginalSMX* and *DecompressDataSet* and each return the respective algorithms' output. Algorithm 6 is achieved by applying an attribute called *CompressVector* that compresses the given vector and subsequently applying *DecompressDataSet* to its output.
- *DeadReacs*: helper functions that find and eliminate dead reactions.
- *IsoForms*: helper class that calculates, defines, and eliminates iso-stoichiometric groups,
- *Tree*: the tree class as described in Chapter 4.3.5.5.

***LeftNS*:** modules that implements the algorithms presented in Chapter 5:

- *LeftNS*: defines a left null-space matrix that has an associated correlation matrix (as an attribute called *diffMtx*) whose associated functions relate metabolites (such as by identifying groups of parallel and orthogonal metabolites) as described in

Chapters 5.3.1 and 5.3.2.1. This is achieved by a function called *GetLeftNS* that takes a stoichiometry matrix as an input and returns the left null-space matrix defined above as an output.

- *AtomsMatrix*: calculates and stores an atomic matrix using database information as described in Chapter 5.3.2.2. This is achieved through a function called *AtomsMatrix* that requires a stoichiometry matrix and ScrumPy *BioCyc database* object as an input. Additionally, a method that creates an atomic matrix by reading information from a *tsv* file, as well as a method that extends the atomic matrix using the conventional technique described in Chapter 4.6 are also present.

- *ExtendAtomic*: implements the *ExtendA* algorithm as described in Chapter 5.3.2.3. This is done by a class called *ExtendAtomic* that takes a stoichiometry matrix and an incomplete atomic matrix as an input (which is then modified by the *ExtendA* algorithm).

- *Fingerprints*: calculates metabolite fingerprint correlation as described in Chapter 5.3.2.4. This information is calculated by a class called *Fingerprints* that requires metabolite names as an output and calculates a correlation matrix (as an attribute called *diffMtx*).

- *ClassifyMets*: implements the network pruning and pathway finding algorithms as part of a class called *LeftNSCluster*, as described in Chapter 5.3.3. This class requires a stoichiometry matrix as an input, the network pruning algorithm is implemented as an attribute function called *classifyMets* and similarly the pathway finding algorithm is implemented as *GetPathway.*

Appendix F

Research Paper: A Novel Algorithm to Calculate Elementary Modes: Analysis of *Campylobacter jejuni* Metabolism.

The research paper titled *A Novel Algorithm to Calculate Elementary Modes: Analysis of* Campylobacter jejuni *Metabolism* is published as a pre-print on bioRxiv a copy of which can be found at www.biorxiv.org/content/10.1101/2023.01.11.521685v1 and below. This paper is also currently under review at BioSystems[1] (Elsevier).

[1] sciencedirect.com/journal/biosystems

A Novel Algorithm to Calculate Elementary Modes: Analysis of *Campylobacter jejuni* Metabolism

Yanica Said*[1,2] Dipali Singh[3] Cristiana Sebu[2] Mark Poolman[1]

August 24, 2023

Abstract

We describe a novel algorithm, ‘LPEM’, that given a steady-state flux vector from a (possibly genome-scale) metabolic model, decomposes that vector into a set of weighted elementary modes such that the sum of these elementary modes is equal to the original flux vector.

We apply the algorithm to a genome scale metabolic model of the human pathogen *Campylobacter jejuni*. This organism is unusual in that it has an absolute growth requirement for oxygen, despite being able to operate the electron transport chain anaerobically.

We conclude that 1) Microaerophilly in *C. jejuni* can be explained by the dependence of pyridoxine 5′-phosphate oxidase for the synthesis of pyridoxal 5′- phosphate (the biologically active form of vitamin B6), 2) The LPEM algorithm is capable of determining the elementary modes of a linear programming solution describing the simultaneous production of 51 biomass precursors, 3) Elementary modes for the production of individual biomass precursors are significantly more complex when all others are produced simultaneously than those for the same product in isolation and 4) The sum of elementary modes for the production of all precursors in isolation requires a greater number of reactions and overall total flux than the simultaneous production of all precursors.

Keywords: Genome Scale Metabolic Model, Elementary Modes, Linear Programming, *Campylobacter jejuni*, Microaerophilly

[1] Cell Systems Modelling Group, Oxford Brookes University, Oxford, OX3 0BP, UK
[2] Department of Mathematics, University of Malta, Msida, MSD 2080, Malta, UK [3] Quadram Institute Bioscience, Norwich Research Park, Norwich, NR4 7UQ,
* *Address for correspondence*: Yanica Said, Cell Systems Modelling Group, Oxford Brookes University, Oxford, OX3 0BP, UK.
Corresponding author e-mail: `18079176@brookes.ac.uk`

1 Introduction

1.1 Camplylobacter

Campylobacter, a genus of Gram-negative, curved or spiral, highly motile bacilli, are foodborne pathogens and the leading cause of acute bacterial gastroenteritis worldwide. The most common routes of human infection is fecal-oral or via contaminated meat, with poultry, in which it is a common gut commensal, being of particular concern. However, it is also found in the gut of wild birds, other farmed animals such as pigs and cattle, pets, as well as soil and contaminated water sources. The ability of this pathogen to persist in the food chain and to contaminate food products poses a serious challenge for food safety and global health.

Campylobacteriosis presents with typical symptoms of gastrointestinal infection, including diarrhoea (often bloody), fever, nausea, and abdominal pain. It is a notifiable disease in the UK with approximately 52,000 cases reported in 2016[1], with an annual cost estimated as £700 million Daniel et al. (2020). Although the disease is commonly self-limiting, it can be associated with a number of serious or life-threatening sequelae including colitis, reactive arthritis, and Guillain-Barré syndrome (GBS), a neurological autoimmune disorder causing muscle weakness and even paralysis requiring ventilation in severe cases. The association of *Campylobacter spp.* with GBS is assumed to be due to the similarity of a lipopolysaccharide cell wall component with a human ganglioside Hadden et al. (2002); Poropatich et al. (2010).

1.1.1 Microbiology and biochemistry

Campylobacter jejuni is well known for being microaerophilic, having an absolute requirement for oxygen and growing optimally at a pO_2 of ≈ 50 mbar but is non-viable (or at least unculturable) at atmospheric pO_2. The reason for the requirement for O_2 is not well understood, although as *C. jejuni* is able to respire anaerobically, it has been proposed that this is a biosynthetic requirement. It is of note that although the intestinal lumen is generally thought of as an anaerobic environment, in regions close to the mucosa, pO_2 has been reported as being as high as 60 mbar Albenberg et al. (2014); Zheng et al. (2015).

Another unusual feature of *C. jejuni* metabolism is that it lacks the enzymes forming the initial steps of glycolysis (glucokinase and phosphofructokinase) as well as those of oxidative limb of the oxidative pentose phosphate pathway (glucose-6-phosphate dehydrogenase, phosphogluconolactonase and phosphogluconate dehydrogenase) Parkhill

[1] https://www.gov.uk/government/publications/campylobacter-infection-annual-data, accessed December 2022.

et al. (2000); Stahl et al. (2012). It also lacks the monosaccharide transporters found in most other bacteria, although it has been suggested that some *Campylobacter spp.* strains are capable of catabolising fucose (a common saccharide residue in intestinal mucins), otherwise the preferred carbon sources are pyruvate, TCA cycle intermediates, and amino acids Hofreuter (2014); Tejera et al. (2020); Wagley et al. (2014).

C. jejuni possess a branched electron transport chain (ETC) with the flexibility to utilise a number of substrates as electron donors including H_2 and formate (both present in the intestinal lumen as metabolic by-products of other gut microbes) Bernalier-Donadille (2010); Kelly (2008); Weerakoon et al. (2009). O_2 can act as the terminal electron acceptor, although NO_3, NO_2, SO_4 and fumarate can also fulfil this function Stoakes et al. (2022); van der Stel and Wösten (2019); van der Stel et al. (2017), allowing for the anaerobic operation of the ETC. However, despite this, *C. jejuni* is unable to grow anaerobically Kaakoush et al. (2007); Sellars et al. (2002) and this has been proposed to be due the existence of oxygen dependent reactions involved in the synthesis of one or more biomass components, specifically: 1) coproporphyrinogen oxidase (EC 1.3.3.3) - a component of heme biosynthesis, and ii) that O_2 is required for the post-translational modification of the enzyme ribonucleotide reductase (RNR) Sellars et al. (2002). However, as it has also been reported that the *C. jejuni* genome encodes for the O_2 independent coproporphyrinogen dehydrogenase (EC 1.3.98.3) capable of fulfilling the same physiological role as its O_2 dependent counterpart de Vries et al. (2015); Parkhill et al. (2000), in which case the latter proposal would appear unlikely to be correct.

In order to further investigate the O_2 dependence of *Campylobacter spp.* we here describe the analysis of a genome-scale metabolic model (GSM) of *C. jejuni* in order to investigate oxygen utilisation for the production of individual biomass precursors whilst growing in a minimal medium. We also present a novel algorithm to identify the pathways (Elementary Modes - see below) utilised when all biomass precursors are synthesised simultaneously.

1.2 Modelling Background

GSMs are computational representations of all reactions (assumed to be) present in an organism's repertoire, typically derived in the first instance from an annotated genome, with reactions defined in terms of stoichiometry, directionality and reversibility, but without any information about reaction kinetics. Such models can be referred to as stoichiometric, or *structural* models.

The theoretical basis for the analysis of structural models can be divided into two broad categories, those derived from *null-space* analysis, and in particular Elementary

Modes Analysis (EMA) Schuster et al. (1999, 2000), and those derived from *linear programming* (LP) Fell and Small (1986); Orth et al. (2010) ; both assume steady-state conditions and both can be regarded as pathway identification tools. The essential difference between the two is that whilst EMA identifies all possible independent routes through the network, LP results typically identify a single (possibly non-unique) pathway through the network that satisfies given optimisation criteria, and the two can be regarded as complementary approaches.

An Elementary Mode (EM) is a minimal metabolic sub-network capable of sustaining a steady-state flux whilst respecting reversibility criteria, associated with a net conversion of substrates into products, and any steady-state flux distribution of the network can be obtained from a non-negative summation of EMs Schuster and Hilgetag (1994); Schuster et al. (2000). Given the complete set of EMs for a network and an associated flux vector, it is possible to then calculate the flux carried by each EM Poolman et al. (2004); Schwartz and Kanehisa (2005).

A well-known draw-back of EMA is the fact that it suffers from combinatorial explosion (is NP-hard - Acuña et al. (2009); Klamt and Gilles (2004)), and therefore obtaining the complete set of EMs of a GSM is impractical for most purposes, not only in terms of the computational resources required but also the unwieldy size of any resultant data-set should such a calculation be possible.

In contrast, flux vectors (modes which may or may not be elementary) with defined properties can be obtained very rapidly using LP; methods based on this (such those related to flux balance analysis (FBA) Orth et al. (2010); Schilling et al. (2000)) are currently the predominant means of analysing GSMs. One draw-back of using LP in the context of a GSM, and in particular where the aim of the study is to account for growth (and therefore the production of all biomass precursors), is that the resulting flux vector still defines a large network that requires further analysis to be easily understood.

However, the solution to a given linear program applied to a GSM will generally contain a much smaller number of reactions than the original, and under some circumstances may be small enough to be treated as a sub-network amenable for EMA (Hartman et al., 2014; Mesfin, 2020).

A number of methods for decomposing flux-vectors into EMs without having previously determined the complete set of EMs have also been described. These utilise LP and/or Mixed Integer Linear Programming (MILP) as strategies for finding individual EMs that fulfil a set of desired attributes. Oddsdóttir et al. (2015), obtained EMs that account for a set of observed transporter fluxes. Their algorithm starts from a matrix, E, whose columns, E_i, comprise of a small sub-set of EMs, and a weighting vector, w.

Two optimization problems are simultaneously solved; a least-squares data fitting master problem that iteratively improves upon the weighting vector w, aiming to obtain a product, **E**w, that is consistent with the external flux measurements, and an additional sub-problem that after each iteration of the master, obtains a new EM to be added to the matrix **E** such that the least-square fit is improved. The algorithm concludes when the fit can no longer be improved by the addition of more EMs. This strategy requires solving a quadratic programme at each iteration, which due to being inherently more computationally costly than LP limits this algorithm's suitability for large models.

Hung et al. (2011), proposed an algorithm that decomposes a steady-state flux vector, v, in series of stages. At each stage, a succession of MILPs reduce the number of non-zero fluxes in v (whilst considering the values of v as an upper-bound) until a solution which cannot be reduced further (and therefore is an EM) is obtained. The EM is subtracted from v and the process is repeated until all constituent EMs have been discovered. A drawback of this algorithm is the heavy use of MILPs, which are more computationally costly then LP.

Jungers et al. (2011), used LP to decompose flux vectors into a minimal number of EMs. This algorithm extracts EMs from the solution-space generated by combining the network's stoichiometry with the steady-state flux-vector, v. The first EM is chosen at random from this space, whilst subsequent EMs are chosen such that the difference from them and the previous modes is maximized. This technique reduces the dimension of the solution-space one step at a time, and, as a consequence has the advantage that the maximum number of EMs obtained must be equivalent to the dimension of the null-space. However, it is likely to produce different results each time that it is used, making the replication of findings difficult.

In this work we describe an algorithm (LPEM) which uses LP to extract a set of EMs that account for a steady-state flux vector (which itself may be an LP solution) from a (possibly genome-scale) metabolic model and demonstrate its application and utility by investigating the phenomenon of micro-aerophilly in *C. jejuni*. The approach is simialr to that described by Hung et al. (2011) but has the advantage that it requires only LP, and not MILP. This algorithm is implemented as part of the open-source metabolic modelling software package, ScrumPy Poolman (2006, 2020).

2 Methods

2.1 The Model

The model used here is that described by Stoakes et al. (2022) (a revision of that previously described by Tejera et al. (2020)). As the genome lacks annotation for enzymes involved in glyoxylate shunt (malate synthase and isocitrate lyase) there is no sink for glycolaldehyde, (a by-product of folate synthesis) and therefore a corresponding transporter added. The model consists of 1029 reactions, 92 transporters and 988 metabolites. It is capable of producing biomass components in a defined media as described in Tejera et al. (2020).

2.2 Model analysis

Initial model analysis was performed using the linear program defined by Eq. (1).

$$\begin{array}{ll} \min & \sum_{i=1}^{n} |w_i v_i| \\ \text{subject to} & \left\{ \begin{array}{lcl} \mathbf{Nv} & = & \mathbf{0}\,, \\ v_b & = & t_b, \text{ for one or all } b \in \{1,2,\dots,B\}, \\ v_{\mathrm{O}_2} & = & 0 \text{ or unconstrained.} \end{array} \right. \end{array} \tag{1}$$

The objective is to determine a flux vector $\mathbf{v}$ whose weighted sum of fluxes is minimised. Each component, v_i , $i = (1...n)$, where n is the number of reactions, v_i corresponds to the flux being carried by the i^{th} reaction and w_i is the associated weighting coefficient. Except where specified, weighting coefficients, w_i, were set to 1 giving equal weight to the minimisation of flux carried by each reaction.

This is subject to the steady-state condition $\mathbf{Nv} = \mathbf{0}$ and the production of one or more biomass components may be taken into account by setting a fixed constraint, $v_b = t_b$, where the suffix b denotes the b^{th} out of B exported biomass components whose transport flux, v_b, is defined by its relative abundance, t_b (m.gdw^{-1}) multiplied by the growth-rate which, for the purposes of this study, was arbitrarily set to unity. In order to investigate the O_2 requirement the associated transport flux, v_{O_2}, was either set to zero or unconstrained and the corresponding penalty weighting, w_{O_2}, was set either to 1 or 10^6 as described below.

With v_{O_2} set to zero, in combination with a demand for all biomass components, Eq. (1) has no feasible solution, demonstrating that at least one biomass component has an absolute requirement for O_2. In order to identify the O_2 dependent biomass component(s), Eq. (1) was solved repeatedly for each biomass component at a time, with the oxygen

Panel 1 The five solution sets used to analyse the oxygen requirement of *C. jejuni*

1. Individual solutions for $v_b = t_b$ for a given $b \in \{1 \dots B\}, v_{O_2} = 0$
2. Individual solutions for $v_b = t_b$ for a given $b \in \{1 \dots B\}, w_{O_2} = 1$
3. Individual solutions for $v_b = t_b$ for a given $b \in \{1 \dots B\}, w_{O_2} = 10^6$
4. A single solution for $v_b = t_b$, for all $b \in \{1 \dots B\}, w_{O_2} = 1$
5. A single solution for $v_b = t_b$, for all $b \in \{1 \dots B\}, w_{O_2} = 10^6$

transport flux constrained to zero. Obtaining a feasible solution with such a constraint demonstrates that this component can be produced without oxygen, conversely, failure to obtain a solution demonstrates that the synthesis of that component has an absolute dependence on oxygen.

In order to identify a solution accounting for all biomass components while minimising v_{O_2}, w_{O_2}, was set to an arbitrarily high value of 10^6 and Eq. (1) re-solved. Thus five sets of flux distributions were obtained, as shown in Table 1.

2.3 Identifying EMs in a Steady-State Flux Vector

In order to identify individual EMs in LP solutions when all biomass components are produced simultaneously we developed an algorithm combining LP with null-space as follows.

The algorithm proceeds by iteratively removing one or more reactions at a time from an initial solution, $\mathbf{v}$, and identifying an associated subsystem, $\mathbf{v}'$, which either contains a single EM, or can be decomposed into a set of EMs. Then, $\mathbf{v}'$ is subtracted from $\mathbf{v}$ and the process continues until $\mathbf{v} = \mathbf{0}$. We thereby obtain a matrix , $\mathbf{E}$, whose column vectors consist of EMs, such that:

$$\sum_{i=1}^{m} \mathbf{E}_i = \mathbf{v},$$

where m is the number of EMs. Note that, in this case, the EMs are not normalised in order for the magnitude of each $\mathbf{E}_i$ to reflect the contribution of that EM to $\mathbf{v}$.

At each iteration, the reaction to be eliminated, $\mathbf{v}_{\text{targ}}$, is selected as the reaction with the smallest (absolute) flux in $\mathbf{v}$. The elimination is achieved by obtaining a solution, $\mathbf{v}'$, to the linear program:

$$\begin{array}{ll} \min & \sum_{i=1}^{n} |\mathbf{v}'_i| \\ \text{subject to} & \begin{cases} \mathbf{N}\mathbf{v}' = \mathbf{0}\,, \\ \mathbf{v}'_{\text{targ}} = \mathbf{v}_{\text{targ}}\,, \\ |\mathbf{v}'_i| \le |\mathbf{v}_i|,\ \text{sign}(\mathbf{v}'_i) = \text{sign}(\mathbf{v}_i),\ \text{for all } i \in \{1, 2, \dots, n\}\,. \end{cases} \end{array} \tag{2}$$

Given these definitions the algorithm can be described by the pseudo-code in presented in Algorithm 1.

Algorithm 1 : LPEM - Using LP to decompose a steady-state flux vector, v, into a matrix of EMs, $\mathbf{E}$

```
1:  while $v > 0$ do
2:     $\mathbf{v}_{\text{targ}} = \min(\mathbf{v})$
3:     solve Eq. (2)
4:     if $v'$ is elementary then
5:        add $v'$ to $\mathbf{E}$
6:     else
7:        decompose $v'$ into a set of EMs
8:        add these EMs to $\mathbf{E}$
9:     end if
10:    $v \leftarrow v - v'$
11: end while.
```

Verifying that v' represents a single EM (step 4) is readily achieved by determining the dimension of null-space of the subsystem it represents: if it is 1, then v' is an EM. If not, step 7 decomposes v into EMs using the algorithm by Schuster et al. (1999), and flux assigned to each EM as described in Poolman et al. (2004). The constraints defined in Eq. (2) along with the subtraction in step 10 ensure that the number of non-zero elements in v decrease by at least one in each iteration of the algorithm, thereby guaranteeing completion.

3 Results

3.1 Model responses penalties and constraints described in Panel 1

No solution exists when attempting to solve Eq. (1) with the oxygen uptake constrained to zero, demonstrating an absolute requirement for oxygen to account for biomass production in this model. The results obtained under the different conditions described in Panel 1 were as follows:

1. **Individual biomass components, oxygen uptake set to zero**

   Solutions could be found for all individual biomass components, with the exception of pyridoxal 5′-phosphate (PLP), the active form of vitamin B6 and a co-substrate for many enzyme catalysed reactions (see discussion).

2. **Individual biomass components, with no constraint or penalty on oxygen uptake**

   With no constraints on oxygen uptake, the synthesis of 45 of the 51 biomass precursors utilised oxygen (including PLP synthesis), and all of these used proton translocating ATP synthase to satisfy some, or all, of their ATP demand; 43 (including PLP) utilised cytochrome oxidase and/or NADH oxidase to generate some or all of the necessary proton gradient to drive ATP synthesis.

3. **Individual biomass components, with imposed oxygen uptake penalty**

   With the oxygen uptake penalty (weighting factor w_{O_2} in Eq. (1)) set to 10^6 none of the biomass precursors, with the exception of PLP, utilised oxygen. 43 of these (including PLP) utilised the electron transport chain for ATP generation with NO_3 or NO_2 acting as the terminal electron acceptor.

4. **Simultaneous production of all biomass components, with no oxygen uptake penalty**

   When Eq. (1) was solved to account for the simultaneous production of all biomass components, the resulting solution contained 324 reactions (excluding transport processes). The major carbon and nitrogen source was glutamine, accounting for 51% and 87% of total carbon and nitrogen uptake respectively. Most of the remaining carbon uptake was satisfied by the uptake of pyruvate (39%) with the rest of the carbon and nitrogen demand satisfied by serine, methionine and cysteine. The latter two amino acids satisfied the demand for sulphur. Excretion of carbon by-product was mainly in the form of carbon dioxide and carbonic acid (73%) with the remainder exported as acetate. Excess nitrogen was excreted in the form of NH_4.

5. **Simultaneous production of all biomass components, with imposed oxygen uptake penalty**

   When the oxygen uptake penalty of 10^6 was imposed on Eq. (1) with the demand for the synthesis of all biomass precursors, the resulting solution also contained 324 reactions of which 320 were common to the solution obtained with no imposed penalty. The fact that the two have the same number of reactions is assumed to be coincidence. All changes to the flux distribution (qualitative and quantitative) were associated with the redirection of the ETC from aerobic to anaerobic operation.

3.2 EMs of the production of biomass components

When solutions for individual biomass components were generated, no constraint was placed on the production of other biomass components, thus allowing the potential for the production of other components as by-products. Nonetheless, each solution resulted in the generation of the target product only, and represented a single EM (see discussion below). However, the total number of reactions utilised by all individual solutions was substantially greater than the number of reactions required to solve Eq. (1) for all components simultaneously (423 vs 397 and 434 vs 395 for penalised and unpenalised O_2 uptake respectively). Therefore the solution obtained for the simultaneous production of all components is not simply the sum of the individual solutions for each component.

To further investigate this, the LPEM algorithm was used to identify EMs utilised for each biomass component in whole solutions (Panel 1.4 and 1.5). The sets of EMs thus calculated comprised a total of 62 EMs for both the penalised and unpenalised solutions. All of the unpenalised EMs utilised oxygen but only one (responsible for PLP synthesis) EM in the penalised set did so. Although the EM responsible for PLP synthesis with penalised O_2 uptake was not the same as the equivalent LP solution, the rate of O_2 uptake was the same for both.

The major difference between the EMs extracted from the whole solution and the individual LP solutions was that most EMs (all but one) generated more than one product and, conversely, most end products were generated by more than one EM.

3.3 EMs of PLP production

The solution for the synthesis of PLP in the absence of demand for other products is presented in Fig. 1 and summarised in Table 1. This is a superset of the the PLP synthesis pathway reported in metacyc (PYRIDOXSYN-PWY), which in turn is derived from the pathways proposed by Fitzpatrick et al. (2007) and described as the DXP-dependent and DXP-independent pathways. The pathway presented here is more complete as it balances all reactions starting from the external substrates glutamine and pyruvate, as well as generating necessary ATP and reductant. It is interesting to note that although both utilise the ETC for ATP generation, neither glycolysis (some reactions of glycolysis are present but run in the reverse, gluconeogenic, direction) nor the TCA cycle is used; sufficient reductant is generated by the oxidation of other substrates to satisfy the demand for reductant by the ETC. In fact, there is a slight excess of reductant generated which is then balanced by the reduction of external NO_3 to NH_3, which is in turn simply excreted (reactions R26 and R33 in Fig. 1).

The EM obtained for the synthesis of PLP in the context of simultaneous synthesis of other biomass components, without imposing a penalty on O_2 uptake is considerably more complex than the solution obtained for PLP as a single product (140 vs 46 reactions). However it does utilise the majority of the reactions depicted in Fig. 1, with the exception of those associated with oxidising excess reductant (R7a, R2a, R32, R33, R26 (NADP), R27). The other difference between this EM and the simple solution is that the EM also produces small amounts of other biomass components: DTTP, FAD, and valine.

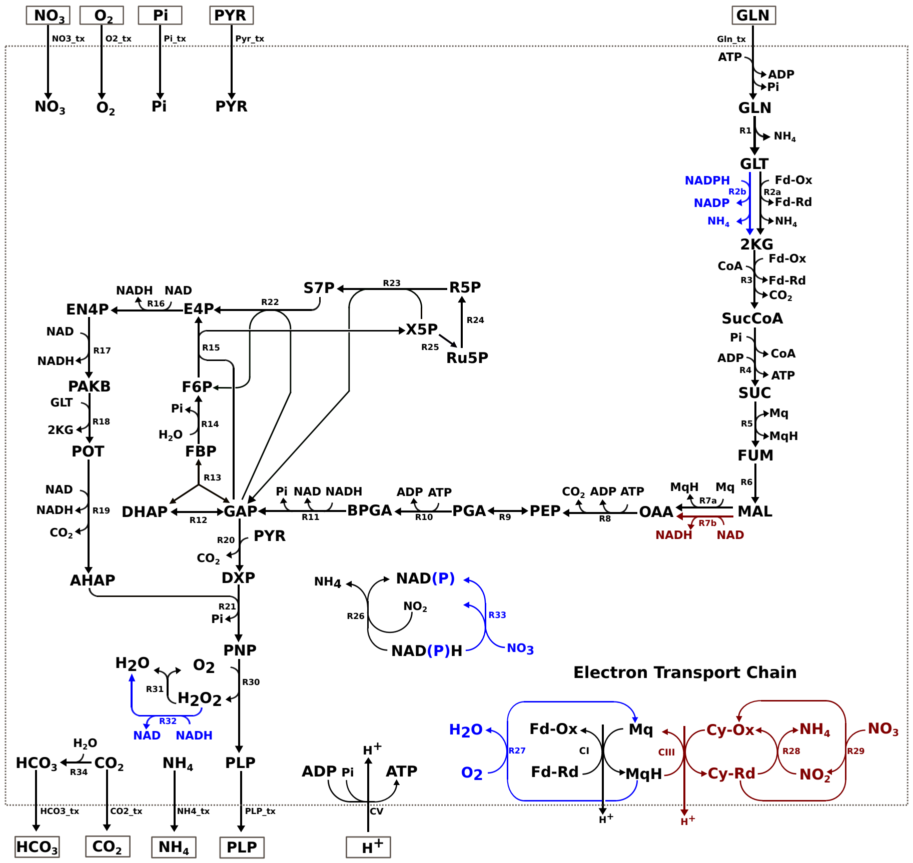


Figure 1: PLP synthesis in the *Campylobacter spp.* model with and without out a penalty on O_2 uptake. Reactions in black are active under both the conditions. Reactions in blue are active when there is no penalty on O_2 uptake and reactions in red are active only when the penalty is imposed. See key tables 1 and 2 for complete descriptions of reaction and metabolite abbreviations.

A similar situation was found when comparing the EM producing PLP in the context of simultaneous synthesis of other biomass components, with the penalty on O_2 imposed (136 vs 46 reactions), and contained 41 reactions in common with the simple solutions. Of the those that were absent, one was associated with the oxidation of excess reductant (nitrate reductase, R26 in Fig. 1), and two were associated with a move away from the use of membrane bound electron carriers to nicatinamides (malate oxidoreductase (R7a) and glutamate synthase (R2a) in Fig. 1). Somewhat unexpectedly, this EM did not utilise transaldolase (R22) or sedoheptulose 7-phosphate transketolase (R23) indicating that this EM has an alternative source of the pentose phosphate pathway intermediate, X5P. Again this EM also produced a number of other biomass precursors, in this case histidine, phosphatidyl-serine, FAD and valine.

3.4 Algorithm Performance

The LP solutions for production of all biomass components with free and penalised O_2 uptake were comprised of 390 and 392 reactions respectively, the dimension of null-

Table 1: Modes of PLP production with or without a background demand for other biomass components and with or without O_2 uptake penalty. The solutions for PLP production are single EMs.

Penalty	+Biomass	Reactions	Total flux	O_2 uptake	Transporters
No	No	46	4.6×10^{-4}	2.9×10^{-5}	10
Yes	No	47	5.0×10^{-4}	3.0×10^{-6}	10
No	Yes	140	1.1×10^{-3}	3.8×10^{-5}	15
Yes	Yes	136	1.3×10^{-3}	3.0×10^{-6}	16

spaces of the subsystems defined by these solutions was 51 in each case. The LPEMs algorithm decomposed these two solutions into 52 and 55 EMs, remarkably close to the dimension of their respective null-spaces. Using a commodity desk-top PC with a 2.6 GHz AMD Ryzen processor the determination of the EMs for both solutions took about 70 seconds and required less than 1 GB of available memory.

4 Discussion and Conclusion

4.1 Micro-aerophilly in *C. jejuni*

The results presented above demonstrate that, in this model, O_2 is essential for the synthesis of PLP, the biologically active form of vitamin B6. This is an enzyme bound co-factor for more than 140 reactions, mainly those involved in amino-acid metabolism and predominantly transferase and lyase reactions Eliot and Kirsch (2004); Percudani and Peracchi (2003, 2009). Animals are unable to synthesise PLP, and it is therefore an essential vitamin. However, plants, fungi and bacteria are able to synthesize PLP via one of two reported routes: the DXP (deoxy-xylulose 5-phosphate) dependent and DXP independent pathway. In the DXP independent pathway, PLP is synthesised by single heterodimeric complex from glutamine, ribose 5-phosphate, and glyceraldehyde 3-phosphate. Organisms lacking this pathway, mainly the proteobacteria, utilise the DXP dependent pathway, commonly depicted as using erythrose 4-phosphate and glyceraldehyde 3-phosphate as the starting point. The pathway depicted in Fig. 1 is in fact a superset of the DXP dependent pathway.

Many other organisms which use the DXP dependent pathway have additional associated reactions and transporters, allowing the uptake of additional precursors as well as greater metabolic flexibility (Fig. 2), and in particular the potential to bypass the O_2 dependent pyridoxine 5′-phosphate oxidase (R30 in Fig. 1 and 2) step Ito and Downs (2020); Sugimoto et al. (2017). However, these have not been reported *C. jejuni* M1cam and therefore O_2 is an absolute requirement for PLP synthesis.

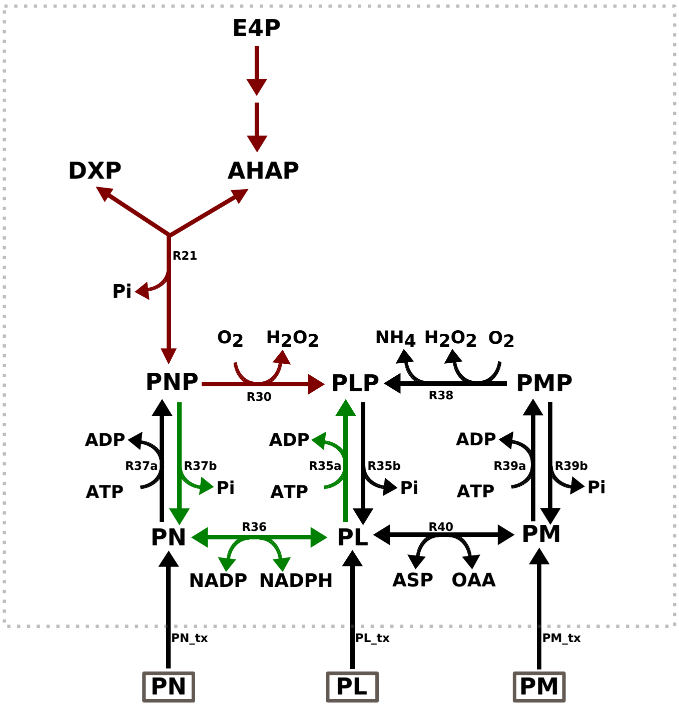


Figure 2: Reactions involved in PLP metabolism in *Escherichia coli*. Reactions in red are common to *C. jejuni*, those in green allow for the bypass of of the O_2 dependent pyridoxine 5′-phosphate oxidase step (R30). *** Reactions labelled R35a/R37a/R39a are all catalysed by the same enzyme (similarly for reactions labelled R35b/R37b/R39b and R30/R38). See key tables 3 for a complete key.

4.2 Oxygen dependence of PLP synthesis

That the LP solutions of all biomass precursors individually, with free and restricted O_2, were single EMs is unsurprising as the solutions of linear programs that contain only one non-zero flux constraint have been shown to always consist of a single EMs Maarleveld (2015). What is more relevant is that this allows the unambiguous identification of PLP production as the reason for the absolute dependence on O_2 for this model to account for growth, although this does not unambiguously identify which O_2 consuming reactions are responsible for the dependence. The model contains a total of 27 reactions utilising O_2 as a substrate, making it impractical to identify the essential reactions by a combinatorial search strategy. However the problem may be readily solved by using the technique of Enzyme Subsets analysis Pfeiffer et al. (1999) which identifies sets of reactions in a network which must carry flux in a fixed ratio in *any* steady-state. A corollary of this is that if any one reaction in a subset carries zero flux at a given steady state, then every other reaction must also carry zero flux. Determining the enzyme subsets of this model reveals that pyridoxine (pyridoxamine) 5′-phosphate oxidase (R30 in Fig. 1) is in the same subset as the PLP transporter and therefore it is the reaction responsible for the absolute dependency on O_2 for the synthesis of PLP. It is interesting to note that, although the model contains the catalase reaction (R31 in Fig. 1), this cannot be used to generate internal O_2 as a substrate for these reactions, as the generation of hydrogen peroxide itself must ultimately depend on an exogenous O_2 source.

4.3 Sum of individual solutions compared to the decomposition of the whole solution

By using LP to identify pathways for precursor synthesis in isolation and applying the LPEM algorithm to an LP solution accounting for the simultaneous production of all precursors, two sets of results were obtained. Although it might be intuitively expected that the solution that accounts for all biomass precursors would be equivalent to the sum of the 51 pathways that produce each precursor individually, this was not the case: the summation of individual solutions required more reactions and greater total flux. This suggests that the optimal solution for the synthesis of a single product in isolation is not necessarily optimal in the presence of demand for additional products, and that therefore individual solutions must be interpreted with care in the context of a growing organism. The explanation for this appears to be that optimal individual solutions also generate by-products, that must then be further metabolised before they can be exported. However, when multiple products must be synthesised such by-products may be utilised for the synthesis of other products. For example, the solution for PLP synthesis in isolation (Fig. 1) generates excess reductant, which is then oxidised by the reduction of NO_3 to NH_3 which is subsequently exported. However, when there is a requirement to synthesise other products, these by-products become useful intermediates. A similar observation was originally made by Fell and Small (1986) in one of earliest papers describing the application of LP to metabolic networks.

4.4 Conclusion

The LPEM algorithm provides a relatively simple and computationally efficeint way to leverage the advantages of FBA and Elementary Modes Analysis. This has shown that the actual modes utilised by an organism *in vivo* may be rather more complicated than consideration of individual FBA solutions would suggest, but that these may nonetheless be more efficeint both in terms of the total number of reactions required and of the overall flux they carry. Applying the approaches described here suggests that the reason for micro-aerophilly in the pathogen *C. jejuni* is the dependence on oxygen for the production of PLP, although this may not be exclusive.

Keys to figures

Key 1 Fig. 1 (Reactions)

Label	Enzyme	Metacyc ID	EC number
R1	glutaminase	GLUTAMIN-RXN	3.5.1.38
R2a	glutamate synthase	RXN-12878	1.4.7.1
R2b	glutamate dehydrogenase	GLUTDEHYD-RXN	1.4.1.4
R3	2-oxoglutarate synthase	2-OXOGLUTARATE-SYNTHASE-RXN	1.2.7.3
R4	succinyl-CoA synthase	SUCCCOASYN-RXN	6.2.1.5
R5	succinate dehydrogenase	SUCCINATE-DEHYDROGENASE-MENAQUINONE-RXN	1.3.5.1
R6	fumerase	FUMHYDR-RXN	4.2.1.2
R7a	malate oxidoreductase (quinone)	RXNI-3	1.1.5.4
R7b	malate dehydrogenase	MALATE-DEH-RXN	1.1.1.37
R8	phosphoenolpyruvate carboxykinase	PEPCARBOXYKIN-RXN	4.1.1.49
R9	enolase	2PGADEHYDRAT-RXN	4.2.1.11
R10	phosphoglycerate kinase	PHOSGLYPHOS-RXN	2.7.2.3
R11	glyceraldehyde 3-phosphate dehydrogenase	GAPOXNPHOSPHN-RXN	1.2.1.12
R12	triose-phosphate isomerase	TRIOSEPISOMERIZATION-RXN	5.3.1.1
R13	fructose-bisphosphate aldolase	F16ALDOLASE-RXN	4.1.2.13
R14	fructose 1,6-bisphosphatase	F16BDEPHOS-RXN	3.1.3.11
R15	fructofuranose 6-phosphate transketolase	2TRANSKETO-RXN	2.2.1.1
R16	erythrose 4-phosphate dehydrogenase	ERYTH4PDEHYDROG-RXN	1.2.1.72
R17	erythronate 4-phosphate dehydrogenase	ERYTHRON4PDEHYDROG-RXN	1.1.1.290
R18	phosphohydroxythreonine aminotransferase	PSERTRANSAMPYR-RXN	2.6.1.52
R19	hydroxythreonine 4-phosphate dehydrogenase	RXN-13179	1.1.1.262
R20	deoxy-xylulose 5-phosphate synthase	DXS-RXN	2.2.1.7
R21	pyridoxine 5′-phosphate synthase	PDXJ-RXN	2.6.99.2
R22	transaldolase	TRANSALDOL-RXN	2.2.1.2
R23	seduloheptulose 7-phosphate transketolase	1TRANSKETO-RXN	2.2.1.1
R24	ribose 5-phosphate isomerase	RIB5PISOM-RXN	5.3.1.6
R25	ribulose phosphate 3-epimerase	RIBULP3EPIM-RXN	5.1.3.1
R26	nitrite reductase (NAD)	RXN0-6377	1.7.1.4
R27	oxygen reductase (cytochrome)	RXN0-5266	7.1.1.7
R28	nitrite reductase (cytochrome)	1.7.2.2-RXN	1.7.2.2
R29	nitrate reductase (cytochrome)	NITRATE-REDUCTASE-CYTOCHROME-RXN	1.9.6.1
R30	pyridoxine 5′-phosphate oxidase	PNPOXI-RXN	1.4.3.5
R31	catalase	CATAL-RXN	1.11.1.6
R32	NADH peroxidase	NADH-PEROXIDASE-RXN	1.11.1.1
R33	nitrate reductase (NAD)	NITRATE-REDUCTASE-NADPORNOPH-RXN	1.7.99.4
R34	carbonic anhydrase	RXN0-5224	4.2.1.1
CV	proton translocating ATP synthase	ATPSYN-RXN	7.1.2.2

Key 2 Fig. 1 (Metabolites)

Abbreviation	Common Name	BioCyc ID
GLN	glutamine	GLN
GLT	glutamate	GLT
2KG	α-ketoglutarate	2-KETOGLUTARATE
SucCoA	succinyl-CoA	SUC-COA
SUC	succinate	SUC
FUM	fumarate	FUM
MAL	malate	MAL
OAA	oxaloacetate	OXALACETIC_ACID
PEP	phosphoenolpyruvate	PHOSPHO-ENOL-PYRUVATE
PGA	phospho-glycerate	2-PG
BPGA	bisphospho-glycerate	DPG
GAP	glyceraldehyde 3-phosphate	GAP
DHAP	glycerone phosphate	DIHYDROXY-ACETONE-PHOSPHATE
FBP	fructofuranose 1,6-bisphosphate	FRUCTOSE-16-DIPHOSPHATE
F6P	fructofuranose 6-phosphate	FRUCTOSE-6P
E4P	erythrose 4-phosphate	ERYTHROSE-4P
S7P	seduloheptulose 7-phosphate	D-SEDOHEPTULOSE-7-P
R5P	ribose 5-phosphate	RIBOSE-5P
X5P	xylulose 5-phosphate	XYLULOSE-5-PHOSPHATE
Ru5P	ribulose 5-phosphate	RIBULOSE-5P
EN4P	erythronate 4-phosphate	ERYTHRONATE-4P
PAKB	hydroxy-2-oxo-4 phosphooxybutanoate	3OH-4P-OH-ALPHA-KETOBUTYRATE
POT	phosphooxy-threonine	4-PHOSPHONOOXY-THREONINE
AHAP	amino-1-hydroxyacetone 1-phosphate	1-AMINO-PROPAN-2-ONE-3-PHOSPHATE
DX5P	deoxy-xylulose 5-phosphate	DEOXYXYLULOSE-5P
PNP	pyridoxine 5′-phosphate	PYRIDOXINE-5P
PLP	pyridoxal 5′-phosphate	PYRIDOXAL_PHOSPHATE
PYR	pyruvate	PYRUVATE
Mq	menaquinol	MENAQUINOL
MqH	menaquinone	MENAQUINONE
Cy-Ox	cytochrome c oxidised	Cytochromes-C-Oxidized
Cy-Rd	cytochrome c reduced	Cytochromes-C-Reduced
Fd-Ox	oxidised ferredoxin	Oxidized-ferredoxins
Fd-Rd	reduced ferredoxin	Reduced-ferredoxins

Key 3 Fig. 2

Reactions

Label	Enzyme	Metacyc ID	EC number
R35a	pyridoxal kinase	PYRIDOXKIN-RXN	2.7.1.35
R35b	PLP phosphatase	3.1.3.74-RXN	3.1.3.74
R36	pyridoxal reductase	PYRIDOXINE-4-DEHYDROGENASE-RXN	1.1.1.65
R37a	pyridoxine kinase	PNKIN-RXN	2.7.1.35
R37b	PNP phosphatase	RXN-14181	3.1.3.74
R38	PMP oxidase	PMPOXI-RXN	1.4.3.5
R39a	pyridoxamine kinase	PYRAMKIN-RXN	2.7.1.35
R39b	PMP phosphatase	RXN-14046	3.1.3.74
R40	PM–OAA transaminase	PYROXALTRANSAM-RXN	2.6.1.31

Metabolites

Abbreviation	Common Name	BioCyc ID
PN	pyridoxine	PYRIDOXINE
PL	pyridoxal	PYRIDOXAL
PM	pyridoxamine	PYRIDOXAMINE
PMP	pyridoxamine 5′-phosphate	PYRIDOXAMINE-5P
ASP	aspartate	L-ASPARTATE

Conflicts of Interest

The authors declare they have no competing interest.

Acknowledgements

YS The research work disclosed in this publication is partially funded by the *Tertiary Education Scholarships Scheme* (Malta) and Oxford Brookes University (*Nigel Groome Studentship*)

DS Is funded by the UK Biotechnology and Biological Sciences Research Council (BBSRC), through the Institute Strategic Research Programme (ISP) BB/R012504/1 Microbes in the Food Chain and its constituent project Microbial Survival in the Food Chain BBS/E/F/000PR10349.

MP Is funded by Oxford Brookes University and the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement number 956154 (INNOTARGETS).

References

V. Acuña, F. Chierichetti, V. Lacroix, A. Marchetti-Spaccamela, M. F. Sagot, and L. Stougie. Modes and cuts in metabolic networks: Complexity and algorithms. *BioSystems*, 95(1):51–60, 2009. ISSN 03032647. doi: 10.1016/j.biosystems.2008.06.015.

L. Albenberg, T. V. Esipova, C. P. Judge, K. Bittinger, J. Chen, A. Laughlin, S. Grunberg, R. N. Baldassano, J. D. Lewis, H. Li, S. R. Thom, F. D. Bushman, S. A. Vinogradov, and G. D. Wu. Correlation between intraluminal oxygen gradient and radial partitioning of intestinal microbiota. *Gastroenterology*, 147(5):1055–63.e8, November 2014. doi: 10.1053/j.gastro.2014.07.020.

A. Bernalier-Donadille. Fermentative metabolism by the human gut microbiota. *Gastroentérologie Clinique et Biologique*, 34:S16–S22, 2010. ISSN 0399-8320. doi: 10.1016/S0399-8320(10) 70016-6.

N. Daniel, N. Casadevall, P. Sun, D. Sugden, and V. Aldin. The burden of foodborne disease in the UK 2018. https://www.food.gov.uk/sites/default/files/media/document/the-burden-of-foodborne-disease-in-the-uk_0.pdf, 2020.

S. P. W. de Vries, S. Gupta, A. Baig, J. Lapos Heureux, E. Pont, D. P. Wolanska, D. J. Maskell, and A. J. Grant. Motility defects in *Campylobacter jejuni* defined gene deletion mutants caused by second-site mutations. *Microbiology*, 161(12):2316–2327, 2015. ISSN 1465-2080. doi: 10.1099/mic.0.000184.

A. C. Eliot and J. F. Kirsch. Pyridoxal phosphate enzymes: Mechanistic, structural, and evolutionary considerations. *Annual Review of Biochemistry*, 73(1):383–415, 2004. doi: 10.1146/annurev.biochem.73.011303.074021.

D. a. Fell and J. R. Small. Fat synthesis in adipose tissue. *The Biochemical journal*, 238(3):781–786, 1986. ISSN 02646021.

T. B. Fitzpatrick, N. Amrhein, B. Kappes, P. Macheroux, I. Tews, and T. Raschle. Two independent routes of de novo vitamin b6 biosynthesis: Not that different after all. *Biochemical Journal*, 407 (1):1–13, 2007. doi: 10.1042/bj20070765.

R. Hadden, N. Gregson, R. Gold, K. Smith, and R. Hughes. Accumulation of immunoglobulin across the 'blood-nerve barrier' in spinal roots in adoptive transfer experimental autoimmune neuritis. *Neuropathology and Applied Neurobiology*, 28(6):489–497, 2002. ISSN 0305-1846. doi: 10.1046/j.1365-2990.2002.00421.x.

H. B. Hartman, D. A. Fell, S. Rossell, P. R. Jensen, M. J. Woodward, L. Thorndahl, L. Jelsbak, J. E. Olsen, A. Raghunathan, S. Daefler, and M. G. Poolman. Identification of potential drug targets in *salmonella enterica* sv. typhimurium using metabolic modelling and experimental validation. *Microbiology*, 160(6):1252–1266, 2014. doi: 10.1099/mic.0.076091-0.

D. Hofreuter. Defining the metabolic requirements for the growth and colonization capacity of *Campylobacter jejuni*. *Frontiers in Cellular and Infection Microbiology*, 4, 2014. ISSN 2235-2988. doi: 10.3389/fcimb.2014.00137.

S. Hung, J. Chan, and P. Ji. Decomposing flux distributions into elementary flux modes in genome-scale metabolic networks. *Bioinformatics*, 27(16):2256–2262, 2011. ISSN 14602059. doi: 10.1093/bioinformatics/btr367.

T. Ito and D. M. Downs. Pyridoxal reductase, pdxi, is critical for salvage of pyridoxal in *Escherichia coli*. *Journal of Bacteriology*, 202(12):e00056–20, 2020. doi: 10.1128/JB.00056-20.

R. M. Jungers, F. Zamorano, V. D. Blondel, A. V. Wouwer, and G. Bastin. Fast computation of minimal elementary decompositions of metabolic flux vectors. *Automatica*, 47(6):1255–1259, 2011. ISSN 00051098. doi: 10.1016/j.automatica.2011.01.011.

N. O. Kaakoush, W. G. Miller, H. De Reuse, and G. L. Mendz. Oxygen requirement and tolerance of *Campylobacter jejuni*. *Research in Microbiology*, 158(8-9):644–650, OCT-NOV 2007. ISSN 0923-2508. doi: 10.1016/j.resmic.2007.07.009.

D. J. Kelly. Complexity and versatility in the physiology and metabolism of *Campylobacter jejuni*. In *Campylobacter , Third Edition*, pages 41–61. American Society of Microbiology, 2008. doi: 10.1128/9781555815554.ch3.

S. Klamt and E. Gilles. Minimal cut sets in biochemical reaction networks. *Bioinformatics*, 20(2): 226–234, 2004. doi: 10.1093/bioinformatics/btg395.

T. Maarleveld. *Fluxes and Fluctuations in Biochemical Models*. PhD thesis, Vrije Universiteit Amsterdam, 2015.

N. A. Mesfin. *Structural metabolic modelling of the acetogen* Acetobacterium woodii. PhD thesis, Oxford Brookes University, 2020.

H. Æ. Oddsdóttir, E. Hagrot, V. Chotteau, and A. Forsgren. On dynamically generating relevant elementary flux modes in a metabolic network using optimization. *Journal of Mathematical Biology*, 71(4):903–920, 2015. ISSN 14321416. doi: 10.1007/s00285-014-0844-1.

J. D. Orth, I. Thiele, and B. O. Palsson. What is flux balance analysis? *Nature Biotechnology*, 28 (3):245–248, 2010. ISSN 10870156. doi: 10.1038/nbt.1614.

J. Parkhill, B. Wren, K. Mungall, J. Ketley, C. Churcher, D. Basham, T. Chillingworth, R. Davies, T. Feltwell, S. Holroyd, K. Jagels, A. Karlyshev, S. Moule, M. Pallen, C. Penn, M. Quail, M. Rajandream, K. Rutherford, A. van Vliet, and B. Barrell. The genome sequence of the food-borne pathogen *Campylobacter jejuni* reveals hypervariable sequences. *Nature*, 403:665–8, 03 2000. doi: 10.1038/35001088.

R. Percudani and A. Peracchi. A genomic overview of pyridoxal-phosphate-dependent enzymes. *EMBO reports*, 4(9):850–854, 2003. doi: 10.1038/sj.embor.embor914.

R. Percudani and A. Peracchi. The b6 database: A tool for the description and classification of vitamin b6-dependent enzymatic activities and of the corresponding protein families. *BMC bioinformatics*, 10:273, 10 2009. doi: 10.1186/1471-2105-10-273.

T. Pfeiffer, I. Sanchez-Valdenebro, J. Nuno, F. Montero, and S. Schuster. Metatool: for studying metabolic networks. *Bioinformatics*, 15(3):251–257, 1999. doi: 10.1093/bioinformatics/15.3.251.

M. G. Poolman. Scrumpy: metabolic modelling with python. *Systems biology*, 153(5):375–378, 2006. doi: 10.1049/ip-syb:20060010.

M. G. Poolman, K. V. Venkatesh, M. K. Pidcock, and D. A. Fell. A method for the determination of flux in elementary modes, and its application to *Lactobacillus rhamnosus*. *Biotechnol. Bioeng.*, 88(5):601–612, 2004. doi: 10.1002/bit.20273.

M. G. Poolman. Scrumpy. `gitlab.com/MarkPoolman/scrumpy`, 2020.

K. O. Poropatich, C. L. F. Walker, and R. E. Black. Quantifying the association between Campylobacter infection and Guillain-Barre syndrome: A systematic review. *Journal of Health Population and Nutrition*, 28(6):545–552, DEC 2010. ISSN 1606-0997. doi: 10.3329/jhpn.v28i6.6602.

C. H. Schilling, J. S. Edwards, D. Letscher, and B. Ø. Palsson. Combining pathway analysis with flux balance analysis for the comprehensive study of metabolic systems. *Biotechnol Bioeng.*, 71 (4):286–306, 2000.

S. Schuster and C. Hilgetag. On elementary flux modes in biochemical systems at steady state. *J.Biol.Syst.*, 2:165–182, 1994. doi: 10.1142/S0218339094000131.

S. Schuster, T. Dandekar, and D. Fell. Detection of elementary flux modes in biochemical networks: a promising tool for pathway analysis and metabolic engineering. *Trends. Biotech.*, 17(2):53–60, 1999. doi: 10.1016/s0167-7799(98)01290-6.

S. Schuster, D. Fell, and T. Dandekar. A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks. *Nature Biotech.*, 18:326–332, 2000. doi: 10.1038/73786.

J.-M. Schwartz and M. Kanehisa. A quadratic programming approach for decomposing steady-state metabolic flux distributions onto elementary modes. *Bioinformatics*, 21(Suppl 2):ii204–ii205, 2005. ISSN 1460-2059. doi: 10.1093/bioinformatics/bti1132.

M. J. Sellars, S. J. Hall, and D. J. Kelly. Growth of *Campylobacter jejuni* supported by respiration of fumarate, nitrate, nitrite, trimethylamine-n-oxide, or dimethyl sulfoxide requires oxygen. *Journal of Bacteriology*, 184(15):4187–4196, 2002. ISSN 0021-9193. doi: 10.1128/JB.184.15.4187-4196.2002.

M. Stahl, J. Butcher, and A. Stintzi. Nutrient acquisition and metabolism by *Campylobacter jejuni*. *Frontiers in Cellular and Infection Microbiology*, 2, 2012. ISSN 2235-2988. doi: 10.3389/fcimb.2012.00005.

E. Stoakes, G. M. Savva, R. Coates, N. Tejera, M. G. Poolman, A. J. Grant, J. Wain, and D. Singh. Substrate utilisation and energy metabolism in non-growing *Campylobacter jejuni* M1cam. *Microorganisms*, 10(7), 2022. ISSN 2076-2607. doi: 10.3390/microorganisms10071355.

R. Sugimoto, N. Saito, T. Shimada, and K. Tanaka. Identification of ybha as the pyridoxal 5'-phosphate (plp) phosphatase in *Escherichia coli*: Importance of plp homeostasis on the bacterial growth. *The Journal of General and Applied Microbiology*, 63(6):362–368, 2017. doi: 10.2323/jgam.2017.02.008.

N. Tejera, L. Crossman, B. Pearson, E. Stoakes, F. Nasher, B. Djeghout, M. Poolman, J. Wain, and D. Singh. Genome-scale metabolic model driven design of a defined medium for *Campylobacter jejuni* M1cam. *Frontiers in Microbiology*, 11, JUN 19 2020. ISSN 1664-302X. doi: 10.3389/fmicb.2020.01072.

A.-X. van der Stel and M. M. S. M. Wösten. Regulation of respiratory pathways in campylobacterota: A review. *Frontiers in Microbiology*, 10, 2019. ISSN 1664-302X. doi: 10.3389/fmicb.2019.01719.

A.-X. van der Stel, F. C. Boogerd, S. Huynh, C. T. Parker, L. van Dijk, J. P. M. van Putten, and M. M. S. M. Wösten. Generation of the membrane potential and its impact on the motility, atp production and growth in *Campylobacter jejuni*. *Molecular Microbiology*, 105(4):637–651, 2017. doi: 10.1111/mmi.13723.

S. Wagley, J. Newcombe, E. Laing, E. Yusuf, C. M. Sambles, D. J. Studholme, R. M. La Ragione, R. W. Titball, and O. L. Champion. Differences in carbon source utilisation distinguish *Campylobacter jejuni* from *Campylobacter coli*. *BMC Microbiology*, 14, 2014. ISSN 1471-2180. doi: 10.1186/s12866-014-0262-y.

D. R. Weerakoon, N. J. Borden, C. M. Goodson, J. Grimes, and J. W. Olson. The role of respiratory donor enzymes in *Campylobacter jejuni* host colonization and physiology. *Microbial Pathogenesis*, 47(1):8–15, 2009. ISSN 0882-4010. doi: 10.1016/j.micpath.2009.04.009.

L. Zheng, C. J. Kelly, and S. P. Colgan. Physiologic hypoxia and oxygen homeostasis in the healthy intestine. A Review in the Theme: Cellular Responses to Hypoxia. *Am J Physiol Cell Physiol*, 309 (6):C350–60, 2015. doi: 10.1152/ajpcell.00191.2015.