universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„Exploring Microbial Community Landscapes by Modelling their Composition Spaces“

verfasst von / submitted by

Marianne Mießkes, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2023 / Vienna, 2023

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	UA 066862
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Chemie
Betreut von / Supervisor:	Univ.-Prof. Dipl.-Ing. Dr. Jürgen Zanghellini

Acknowledgements

I would like to sincerely thank Jürgen Zanghellini for giving me the opportunity to write my master's thesis in his group as well as continuously guiding me throughout this process. I am also grateful for the entire working group for their warmth, openness, and camaraderie, which made these past few months a truly enjoyable experience. As well as that, I would like to extend my heartfelt appreciation to my friends and family, for their continuous encouragement and emotional support when I needed it. A special thanks goes out to my friends Martin and Christian, who opened my eyes to the world of programming and inspired me to explore this fascinating field. Lastly, I want to thank my cats Pablo and Vallejo, for keeping my lap warm every step along the way.

Contents

1. List of Symbols

1.1. Metabolic models

$\boldsymbol{S}$	stoichiometric matrix	m	total number of metabolites
S_{ij}	cell in stoichiometric matrix	r	total number of reactions
$\boldsymbol{M}$	vector containing all metabolites	u_j	upper bound of reaction R_j
$\boldsymbol{R}$	vector containing all reactions	l_j	lower bound of reaction R_j
M_i	metabolite, with $i \in \{1...m\}$	$\boldsymbol{r}$	flux vector
R_j	reaction, with $j \in \{1...r\}$	r_j	flux of reaction R_j
R_μ	objective reaction	r_μ	flux of objective reaction

1.2. Community models

n	total number of community members	R_{μ_c}	community objective reaction
k	specific community member	r_{μ_c}	flux of the community objective reaction
M^k	metabolite of member k	$\boldsymbol{F}$	community composition
R^k	reaction of member k	F^k	abundance of member k
BM^k	biomass metabolite of member k	f^k	abundance reaction of member k
BM_c	biomass metabolite of community	f	slack variable
S^{ex}	stoichiometric matrix containing only the external reactions		

1.3. ECM-Project

P	convex polyhedron	v^*	specific starting node/vertex
G	graph	v	visited nodes/vertices
T	spanning tree	s	stack

2. Introduction

2.1. Microbial communities

Microbial communities consist of a group of individual microorganisms, which can interact in various ways. For instance, they can compete for limited resources, or cooperate through symbiosis. These interactions shape the structure as well as the function of a community.

Microbial communities find application in a variety of fields. They are commonly employed in biotechnological applications such as wastewater treatment [1–3], agriculture [4] and the production of various products, such as ethanol and methane which can be utilised as biofuels [5–7].

2.1.1. Wastewater treatment

In wastewater treatments, microbial communities are present in the activated sludge. Some biofilms can be utilised under a combination of anaerobic and aerobic conditions to remove nitrogen and phosphorus from the sewage [1]. Conventional methods of phosphate removal involve iron or aluminium salts, which serve as precipitation agents for phosphorus. Even though being very effective in the removal of phosphorus, these additional salts result in an increased amount of sludge, which is expensive to process further. As well as that, the treatment with such salts may result in heavy metal contamination in the sewage. Both of these effects can be mitigated by the employment of certain microbial communities. However, not all microorganisms are beneficial to wastewater treatment. Some bacteria cause filamentous bulking and foaming in the activated sludge [1]. This constitutes a major complication in wastewater treatment, as foaming prevents particles in the sludge from settling in the sedimentation tanks, which serve as a separation of biomass from the sewage. In that light, it can be said that the effectivity of the wastewater treatment plant is dependent on the microbial community composition and activity in the activated sludge [1].

2.1.2. Agriculture

Nitrogen fixation plays an essential role in agriculture and describes the reduction of molecular nitrogen into ammonia. As plants cannot process molecular nitrogen, but rather use ammonia as their primary nitrogen source, the conversion of these two molecules by microorganisms is an important topic and has been well explored. These nitrogen-fixating prokaryotes are present in the so-called rhizosphere, which describes the soil near the roots of the plants. The plants can interact and associate with these microbes and obtain their required nitrogen from the rhizosphere [4]. Another key substrate for plant growth is phosphorus, which primarily

appears as insoluble organic and inorganic materials in soil. These molecules can generally not be assimilated by plant cells, as they first have to be broken down into smaller parts before becoming soluble. Many different bacterial strains possess the ability of phosphate solubilisation, although different environmental conditions may affect the activity of those microorganisms. Additionally to those main functions, microbes can exhibit multiple other traits beneficial to plant growth [8].

2.1.3. Production of biofuels

For the production of biofuels, residues, wastes, and energy crops are digested by microbial communities under anaerobic conditions. Especially biogas produced by the conversion of straw, manure, and sewage sludge is a climate-friendly alternative to fossil fuels, as 98 percent fewer greenhouse gases are emitted during its usage in comparison to petrol [9]. Depending on the structure and members of the community, different products can be formed, which can later be utilised for different purposes. A very common biofuel produced in that way is methane, which can serve as a fuel in both heat and power generation. As well as that, it can also be employed as a fuel for vehicles. The fermentation process into methane is conducted by a variety of microorganisms that cross-feed metabolites, which means that they stand in syntrophic interrelation [5]. The members of this biogas-producing community have to carry out four distinct phases to produce methane:

1. Hydrolysis
2. Acidogenesis
3. Acetogenesis/Dehydrogenation
4. Methanation

In the first phase, polymers are hydrolysed into smaller molecules such as sugars, amino acids or long chain fatty acids. As well as that, hydrolysis may also result in the formation of carbon dioxide and hydrogen, which can immediately be metabolised by methanogens. The acidogenesis takes place in fermentative bacteria, which convert the long chain fatty acids into volatile fatty acids. Acetogenetic microorganisms metabolise these volatile fatty acids into acetate and hydrogen. Finally, the methanogenic bacteria consume acetate and convert it into methane and carbon dioxide, or metabolize hydrogen and carbon dioxide into methane. Interestingly, the addition of hydrogen producing bacteria raise the methane production, which suggests that hydrogen could be a limiting substrate for methanogens [5, 10]. However, large quantities of hydrogen may also inhibit the acetogenic bacteria. For this reason, the accumulation of hydrogen in the medium has to be prevented, which can be achieved by addition of hydrogen-consuming bacteria. Still, a high production of hydrogen is desired in order to augment methane production. As well as that, high quantities of acetate can lead to a drop in pH below 7.0, which in turn may also inhibit methanogenic bacteria. Already it becomes evident that the biogas production process is a very complex system that has to be well adapted in order to prevent process failure [5]. An overview of the stages in

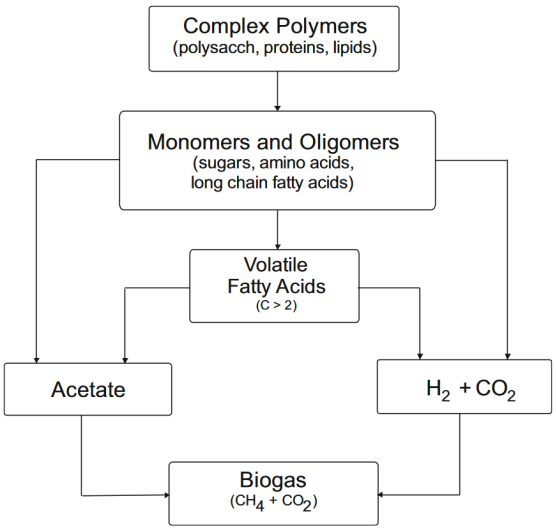


Figure 2.1.: Stages of the methane fermentation process. Taken from [5].

the methane fermentation process is shown in Figure 2.1. The anaerobic fermentation of materials into biogas constitutes an environmentally friendly and energy-efficient technology for energy production and has therefore received heightened attention. This is due to the fact that the burning of biofuels only emits as much carbon dioxide as the plants have absorbed from the atmosphere throughout their growth. Therefore, carbon dioxide produced by the employment of biofuels is considered to be climate-neutral. However, some greenhouse gas emissions still occur throughout the production of biofuels, for example during machinery use, the fertilisation process, and transport. Still, when compared to the conventional combustion of fossil fuels, biofuels pose an attractive alternative as an energy source. For that reason, two EU directives have been issued to promote the production and usage of biofuels. In 2012, 6.77 percent of fossil fuels were replaced by biofuels in Austria [9]. Nonetheless, issues in transport and distribution lead to the majority of the produced biofuels being utilised in heat and power generation.

2.1.4. Microbial communities in vertebrates and invertebrates

Microbial communities also play an important role in human health, as our body contains a multitude of microorganisms, forming the human microbiome [11].

The human body, like that of many other vertebrates, is home to a diverse array of microbial communities. These microorganisms, mainly bacteria, reside in various regions of the body, including the skin, mouth, and gut. While it might seem counterintuitive to host potentially harmful bacteria, this coexistence is made possible by the presence of the adaptive immune system. This immune system that vertebrates are equipped with is a remarkable defence mechanism that can “remember” previous encounters with microbes. This memory allows the immune system to adapt its response to future encounters with the same pathogens, creating a more tailored defence against reoccuring threats. In contrast, invertebrates possess only

an innate immune system, which lacks this memory and relies on pre-coded mechanisms to react to specific threats. This innate immune system cannot change over time. Nonetheless, invertebrates can exist without the same level of microbial control, largely due to differences in their interactions with resident bacteria [12].

Microbial communities residing in vertebrates, especially in regions like the mouth and gut, are incredibly dense and in direct contact with host tissues. This constant interaction poses a significant risk of infection, such as inflammation and sepsis [12, 13]. This is where the adaptive immune system comes into play. It enables vertebrates to manage the microbial communities in their bodies effectively, harnessing the microbes abilities for their own advantage, while keeping the risk of infection relatively low. This is because the memory-based immune system can recognise and respond to specific pathogens, ensuring the host's survival in this microbial-rich environment. Invertebrates, on the other hand, can lead long and healthy lives without the adaptive immune system [12]. The reason for this lies in the relatively lower density of resident bacteria within their bodies. Invertebrates do not require the same level of microbial control as vertebrates, making the adaptive immune system less critical for their survival. However, because of that, they do not gain the advantages of resident bacteria in their body.

The existence of multicellular eukaryotes alongside microbial communities predates the emergence of vertebrates. The interactions between these eukaryotes and microbes significantly influenced vertebrate evolution [12]. This influence is evident in the specialised adaptations of vertebrates' digestive tracts, such as the enlarged gut that precedes the stomach in foregut fermenters like cows and sheep, as well as an enlarged gut portion after the stomach in hindgut fermenters like horses and gorillas [14]. Herbivores, in particular, benefit from a longer retention of digested food in the gut. This extended contact time allows microbial communities to convert indigestible plant polysaccharides into digestible sugars, which are accessible to the host. These fermentative microbes possess enzymes, including glycoside hydrolases, to break down complex plant matter, and this process requires time.

The composition of the gut microbiome in vertebrates is profoundly influenced by their diet. Different host dietary tracts can also impact the microbial communities in the gut, with certain morphologies favouring specific microbial species and functions. In the gut, the adaptive immune system plays a crucial role by producing immunoglobulin A (IgA), which helps to maintain a balanced gut microbiota [14]. Studies have shown that deficiencies in IgA can significantly affect the microbial communities within the gut, as demonstrated in mice [15]. This highlights the intricate interplay between the adaptive immune system and the gut microbiota in vertebrates.

2.2. Analysis of microbial communities

Because of this multitude of applications, endeavours to develop new technologies based on the analysis of microbial communities have been taken.

Currently, germ-free rodents are primarily used for the investigation of the interaction of microbes and the immune system [13]. They are housed in sterile conditions to prevent any unwanted microorganisms getting into the body of the rodent. These germ-free rodents can either be used as a control group to wild-type animals that have an intact microbiome within

their body. As well as that, interactions of specific singular microorganisms or communities of microorganisms with the immune system can be observed by only introducing those desired microorganisms into the test tube. If one wants to test the memory of the adaptive immune system, one gets confronted by the problem that it is very unlikely to rid a rodent of its entire microbial population once it has been exposed to it. This issue can be solved by the employment of deletion strains of bacteria, which cannot synthesise certain important amino-acids. Whilst being able to be grown in a culture, once exposed to an in vivo experiment, these bacteria cannot survive. This makes an exposure of a germ-free rodent to certain strains of bacteria possible, whilst also enabling a return to its germ-free state afterwards. Later on, the memory of their immune system can be tested by subjecting the mice to the same bacteria species once more [13].

A method of analysing microbial communities is the computational model-based design approach. As the name suggests, this method works with a computational model of the metabolism of a species. This model can be employed to predict the uptake and secretion rates of metabolites, the growth rate of the species, as well as the flux through each reaction in the metabolism [11].

In this thesis, a novel approach has been developed for integrating various individual species models into a unified community model. This community model can be examined and analysed in a manner analogous to traditional single-species models. Additionally, it provides the option to calculate the composition spaces of the community. Whereas a single community composition characterises the relative abundance of individual species within the community, a composition space encompasses all possible community compositions.

2.3. Biochemical reaction networks

As a result of the huge amounts of experimental biological data produced by technological advances in measuring equipment, mathematical modelling techniques have been created in order to facilitate the analysis of biological systems. Mathematical models of such biological systems contain information of their cellular components, such as DNA, RNA, proteins, enzymes, metabolites, genes, etc., as well as their interactions with one another [16]. Depending on their function, one can differentiate between three major types of networks:

1. Metabolic networks: These contain metabolites which partake in metabolic reactions. These reactions are generally catalysed by an enzyme, which in turn is created from its corresponding gene. The network itself illustrates the uptake and processing of substrates by the cell.

2. Signal transduction networks: These networks are responsible for detecting signals from the environment as well as the current state within the cell. It generates suitable responses such as the modulation of gene expression, either increasing or decreasing it as needed.

3. Gene regulatory networks: These networks describe the dependencies and connections between genes.

These three classes of biological networks are implicitly linked to one another, especially signal transduction and gene regulatory networks both predominantly involve genes and proteins in their framework. Metabolic networks, on the other hand, can describe the flow of substrates and metabolites through the reaction pathways of a cell and thereby generate mass flow [16].

2.3.1. Metabolic models

Genome-scale metabolic models are instrumental in computational approaches to understand the behaviour of an organism in a certain condition. Concretely, a genome-scale metabolic model contains a metabolic network of the organism, and additional features connecting that metabolic network to the environment. The nodes of this metabolic network are represented by metabolites. These metabolites are converted by reactions, which define the edges of the network. Exchange or transport reactions are used to link the network to the external environment, allowing the flow of metabolites in and out of the cellular compartment. The role of these exchange reactions is to convert external metabolites, which represent the medium in which the organism resides, into internal metabolites. These internal metabolites taken up by the organism are then metabolised into different metabolites by the organism, until another exchange or transport reaction carries the internal metabolite out of the cellular compartment, thereby converting it into an external metabolite. As long as a metabolite is an internal metabolite, it cannot partake in reactions outside of the metabolic network, as the metabolite is considered to be physically inside the organism. Contrarily, an external metabolite can only start interacting with the reactions of the metabolic network once it has been transported into it by an exchange or transport reaction. The exchange reactions therefore play the essential role of connecting the metabolic network to its surroundings, and make the interaction of these two compartments possible [17].

There are multiple options for the creation of a genome-scale metabolic model:

1. Information of all the reactions and metabolites in the metabolic network can be taken from a variety of databases, such as BIGG [18], BRENDA [19], and KEGG [20]. These databases also provide stoichiometric details of the involved reactions, as well as the information on whether a reaction is reversible or not.

2. Alternatively, specific programs exist for the purpose of constructing a genome-scale model directly from the sequence of a genome. These programs include CarveMe [21], KBase [22] and ModelSEED [23].

However, models generated directly from genome sequences can be inaccurate, leading to gaps in the metabolic networks and incorrect assessments of reaction reversibility. Inconsistent naming of metabolites and reactions can also hinder the universal use of these models [17]. Some of the gaps in the metabolic network can be explained by the fact that not every transport reaction is encoded in the genome sequence. The single genes in the sequence solely translate into enzymes, which can function as a transporter of certain metabolites across the borders of two compartments. However, these enzymes only describe the options for active transport. In reality, many additional passive transports happen without the presence of an enzyme being necessary. Consequently, the genome sequence on its own is simply not enough to

sufficiently produce a feasible genome-scale metabolic model. In order to raise the quality of such a model, it has to be refined and curated first. Criteria commonly corrected in the curation process include the following [17]:

1. The mass balance of the reactions within the metabolic network needs to be checked in order to ensure that the mass conservation law is upheld by the model. This can be accomplished by comparing the atomic mass of the substrates and products of every reaction within the model, with the exception of exchange reactions. This exception arises due to the fact that the exchange reactions are inherently unbalanced, as they either introduce new metabolites from the environment into the system, or transport products out of the organism and into the environment. As the environment itself is not a part of the model, these exchange reactions are not balanced.

2. The reversibility of reactions has to be validated. Especially in the case of exchange or transport reactions, the reversibility of a reaction can have a major impact on the function of the organism. If, for instance, an organism had a transporter that can only excrete a certain metabolite, the reaction of the metabolic model should be irreversible.

3. Missing reactions have to be added in order to fill certain gaps that may be in the model. There are different approaches and tools that automatically fill the gaps of a metabolic model following specified criteria [24, 25].

4. Oftentimes, an artificial biomass reaction is added as well, in order to simulate the growth of an organism. This biomass reaction combines different organism-specific compounds including amino acids, DNA and RNA amongst others. Generally, this biomass reaction is best generated using experimental data from growth experiments in a defined medium.

A way of facilitating the curation process is to make reaction and metabolite names consistent with established naming rules in order to make the model compatible with them.

2.3.2. Metabolic models for microbial communities

In the field of systems biology, the conventional methods for investigating the behaviour of single cells are currently being expanded towards a broader perspective by encompassing multiple metabolic models which form a community. It is now understood that an organism's interactions with other organisms and the environment significantly influence its behaviour and function.

The generation of metagenome-assembled genomes (MAGs) is one approach for constructing community models [17]. The metagenome of a sample contains genetic information from all microorganisms within the sample, without distinguishing between individual species. This genetic data is obtained by sequencing methods, resulting in a vast array of genes which need to be ordered and assigned to the correct organism. This also constitutes one of the two major challenges of the metagenome approach:

1. Identify the contained members of the community in the sample and assign the sequenced genes to the respective member.

2. Infer function and behaviour of the individual members of the community.

Currently, the analysis of genome-scale metabolic models of single organisms is a more common approach than the utilisation of MAGs. However, only a fraction of the total number of microorganisms in the world have been sequenced and had their genome turned into a genome-scale metabolic model. Especially microorganisms that cannot be cultured alone because they rely on the co-members of their community to survive suffer an under-representation in online databases of metabolic networks. Due to that, the development of methods to capture metagenomes in high quality is an active field of research, as it currently represents a reliable approach of harnessing the genetic information from such organisms [17]. This approach involving metagenomics can be considered a top-down method.

However, this thesis revolves around constructing community models in a bottom-up manner. The method presented in this work assembles multiple singular genome-scale metabolic models which are taken from one of the numerous databases and assembling them into a combined community model. Generally, one can distinguish bottom-up approaches from top-down approaches by the feature that bottom-up methods require already existing work that has been conducted on individual models. The gathered information on these models is subsequently combined to create a larger community model. Generally, such bottom-up approaches are employed for smaller communities, containing models that can be easily cultured individually. In contrast to this, top-down approaches rely on data derived from metagenomics, with a follow-up differentiation between the individual models within the investigated community.

After deciding on a bottom-up or top-down approach, one must also decide on the modeling technique which should be employed. Importantly, one must differentiate between lumped and compartmentalised models (Figure 2.2). In a lumped model, all individual models are combined into a single model, which does not differentiate between the members. As there are no borders between the models of the organisms in the community, this model would act like a single enlarged organism, with all the functionalities of its members. However, this prevents the assignment of roles or functions of the single members of the community [17]. The compartmentalised model upholds the borders of the individual organisms. Therefore, each organism resides in its own compartment. An additional exchange compartment is added to the model, in which metabolites can be exchanged between the members of the community. This approach is able to provide predictions on whether the members cooperate with one another, or compete for limited substances in the exchange compartment. As I am interested in the interactions between the single members of the community, the method presented in this thesis exhibits a compartmentalised approach.

Furthermore, one can distinguish four different types of metabolic community models, each designed to fulfil a distinct purpose.

1. Network-based models
2. Graph-based models
3. Dynamic models
4. Steady-state models

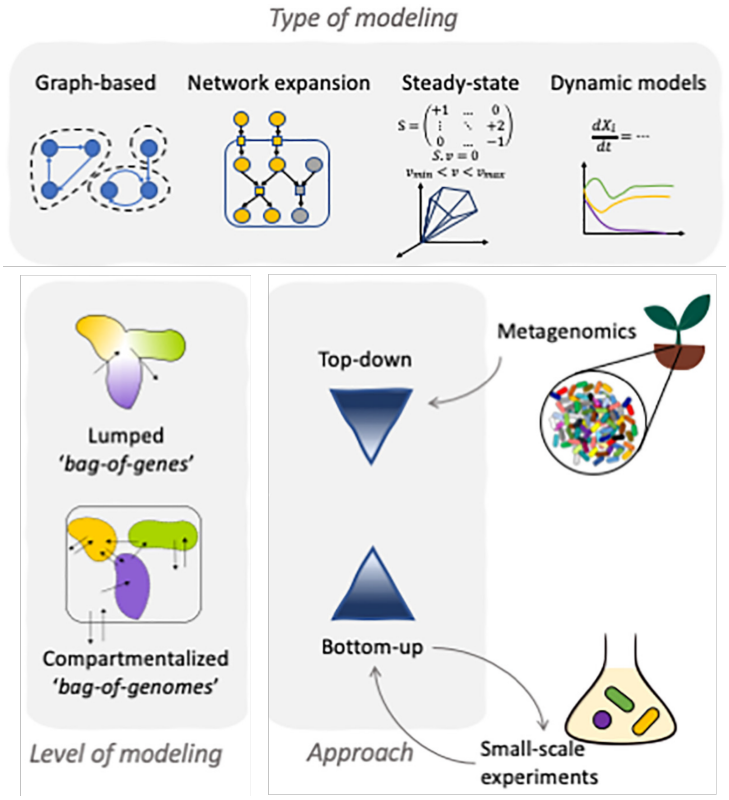


Figure 2.2.: Different types and approaches to metabolic modelling. Adapted from [17].

Network-based models are typically employed to predict whether members within a community compete or cooperate with one another. In order to do so, features and reactions of the individual members are compared. For instance, an overlap in substrate inputs might suggest that there is an option for competition in this community, as that metabolite is required by multiple members simultaneously [17]. The tool NetCooperate can predict the probability of cooperation by calculating a metabolic complementarity index and a biosynthetic support score [26].

The graph-based approach is also utilised for the prediction of competition in a community. Here, the overlap of metabolites in the models serves as a measure for the competitive potential. This stands to reason, as more similar models might have similar needs and therefore require the same substrates, which could become a limited source they must compete for.

Dynamic modelling makes the investigation of the composition of communities over a period of time possible. For this, longitudinal data is required, which can then be used to simulate the generation and stabilisation of the community composition [17].

Steady-state models assume that no accumulation of metabolites within the members in the community takes place. This steady-state principle therefore assumes that the concentration of internal metabolites remains constant. Due to that, no kinetic information of the reactions and the enzymes is required. This is a very beneficial concept, as the experimentational evaluation of every enzyme in an organism is not conceivable. Steady-state models are usually used in constraint-based analysis.

2.4. Constraint-based reconstruction and analysis

Such steady-state metabolic models can be simulated and analysed using constraint-based reconstruction and analysis (COBRA) methods, such as the python COBRApy toolbox [27]. A particular advantage of the constraint-based modelling approach is the fact that no extensive information on the mechanism and kinetics of the reactions is required [16]. Instead, certain biological constraints that should simulate a given state of the cell are defined. These constraints encompass a variety of facets such as mass conservation, thermodynamic directionality, and compartmentalization [27].

3. Aims

The overarching goal of this thesis is to create a method that is able to compute the composition space of a metabolic community model under specified constraints and biomass production. The composition space itself describes the area that encompasses every feasible composition of the members within the community.

First, a method for the assembly of a metabolic community model from individual metabolic models has to be created. Here, a compartmentalised approach will be followed, meaning that every member of the community should exist in its own compartment. This is relevant to distinguish the individual member models from one another. Additionally, various features have to be added to make the community model feasible. These include the introduction of a community biomass reaction to control the growth of the community, and a system to adapt the reaction constraints of each member to its relative abundance in the community.

For this, two different approaches will be developed. In the first method, additional reactions and metabolites will be introduced that should control the relative flux through the internal reactions of each member. This results in a metabolic model in which the internal reactions do not require upper or lower flux bounds. The flux through the reaction will instead be controlled by the added reactions. This adapted model will be referred to as the “equivalent bound-free community model”. The composition space of this community model can then be calculated via flux balance analysis.

The second method creates a community model that does not contain additional reactions controlling the abundance of the members. This community model will instead be used for the ECM-Project, which is a program that computes the elementary conversion modes of a metabolic system. Another aim within this thesis is to expand the ECM-Project to also work with community models and get it to output the vertices of the community composition space.

To summarise, two different approaches for the construction of a community model from single models, as well as two different methods for the subsequent computation of the community composition space will be created in this thesis. An overview of the project is depicted in Figure 3.1.

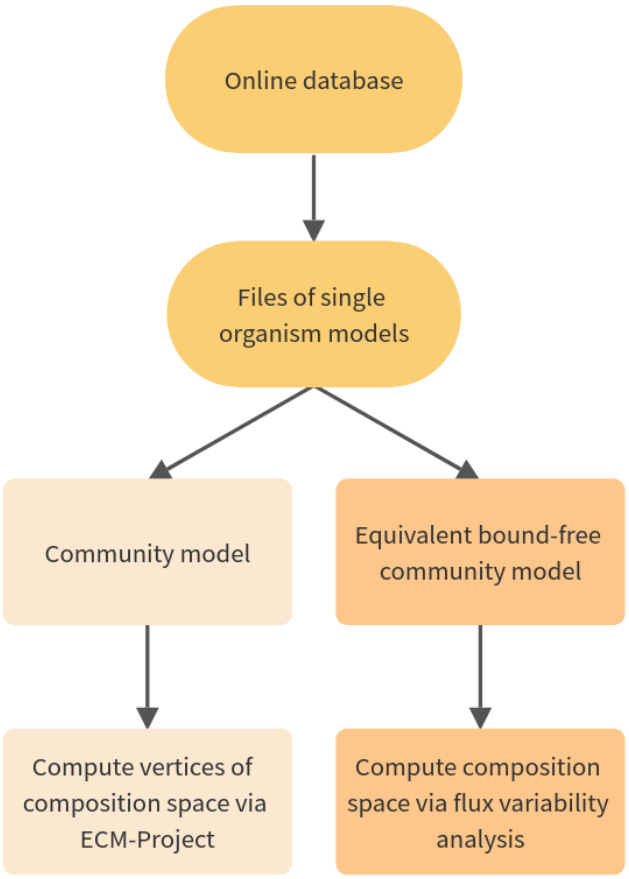


Figure 3.1.: Project overview.

4. Methods

4.1. Constructing a computational model

4.1.1. Building blocks of a metabolic model

A metabolic model is a system of metabolites that partake in metabolic reactions [16]. These reactions are generally catalysed by an enzyme, which in turn is produced by a gene. These four classes (metabolites, reactions, enzymes and genes) are the building blocks of a metabolic model. The enzyme-catalysed reactions convert metabolites into products at a set stoichiometry. The flow through one such reaction is called "flux" and describes how many mmol of metabolite are converted into products per gram dry weight each hour ($\mathrm{mmol\,gDW^{-1}\,h^{-1}}$). By computing the fluxes of each reaction, functional properties of the metabolic model can be discovered.

Some processes such as the transport of metabolites through the cell membrane or between different cell compartments can be included in the metabolic model as *pseudo reactions*. Especially the transport reactions across the cell membrane are important characteristics of a metabolic model [16]. These reactions transfer external metabolites into the cell, where they turn into internal species. The distinction between external and internal metabolites is of relevance as the internal species are included in the stoichiometric matrix $\boldsymbol{S}$, which will be described in greater detail in the following section. The internal species are balanced within the network, while the external species are thought of as limitless sources or sinks for specific metabolites.

4.1.2. Mathematical formulation

In order to analyse reconstructions of metabolic networks, they first have to be translated into a mathematical format [28]. The key element of this mathematical representation is a numerical matrix that contains information regarding the stoichiometry of metabolites in each reaction of the network [29]. Each column in this stoichiometric matrix $\boldsymbol{S}$ represents a reaction, while the rows of the matrix stand for the metabolites contained in the metabolic model. Considering a metabolic network with m metabolites participating in a total of r reactions, the stoichiometric matrix $\boldsymbol{S} \in \mathbb{R}^{m \times r}$ would therefore contain m rows and r columns. The content of a cell $S_{i,j}$ denotes the stoichiometric coefficient of the metabolite M_i with $i \in \{1, ..., m\}$ in the reaction R_j with $j \in \{1, ..., r\}$. The entry is positive if the metabolite is produced by the reaction, or negative if the metabolite is consumed. A stoichiometric coefficient of 0 signifies that the metabolite does not participate in the reaction [29]. Due to the fact that generally only a small fraction of all metabolites participate in any given reaction, the majority of entries in a column of a stoichiometric matrix are 0, which is why $\boldsymbol{S}$ can be denoted as a sparse matrix.

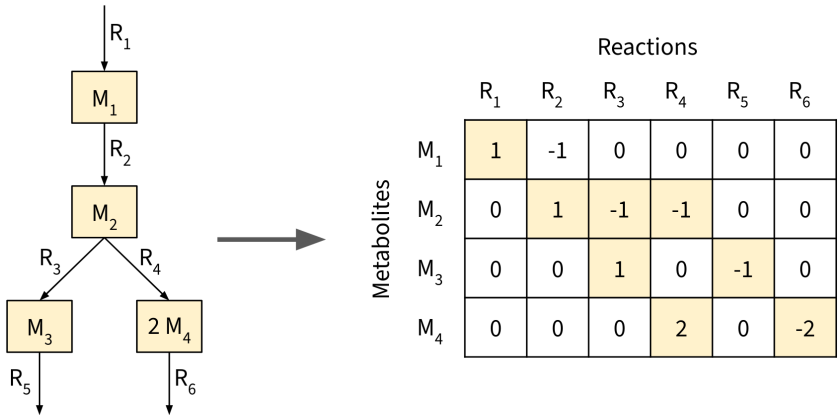


Figure 4.1.: Simple metabolic model with its corresponding stoichiometric matrix. M_{1-4} are metabolites, R_{1-6} are reactions. The entries in the stoichiometric matrix correspond to the stoichiometric coefficients of the reactions. Positive entries show that the metabolite in question is produced by the reaction. Negative entries show that the metabolite is consumed by the reaction. If the metabolite does not partake in the reaction, its entry is 0. R_4 produces two units of M_4, which corresponds to the entry of 2 in the cell M_4R_4. Adapted from [28].

An example metabolic model and its corresponding stoichiometric matrix are illustrated in Figure 4.1.

Assuming the steady-state condition, the stoichiometric matrix imposes constraints on the metabolic network, as the total amount of any metabolite produced must be equal to the amount consumed [29]. Apart from that, every reaction R_j can also be constrained by an upper bound u_j and a lower bound l_j. These bounds ensure that the flow of metabolites r_j through the reaction R_j lies between the lower and upper bound, therefore

$$l_j \leq r_j \leq u_j \,. \tag{4.1}$$

These can include substrate uptake and secretion rates, which can be controlled by setting the lower and upper bound of the reaction to the desired flux rate. Negative lower bounds indicate reversible reactions, while lower bounds set to 0 force the reaction to flow only in one direction, making it irreversible. One may also set the lower bound to a positive number, which introduces reactions that are always required to be active. One common example of such a reaction is the “ATP maintenance reaction”, which simulates the energy consumption of a cell which is not linked to the growth of the organism [29]. Other linear constraints can be added as well.

With all of these constraints in place, the flux through the reactions of a metabolic model is represented by the flux vector $\boldsymbol{r} \in \mathbb{R}^r$. Each entry r_j in $\boldsymbol{r}$ corresponds to the flux of a reaction R_j in the network. As the steady-state constraint demands the sum of all produced and consumed metabolites to be 0, the multiplication of the stoichiometric matrix $\boldsymbol{S} \in \mathbb{R}^{m \times r}$

with flux vector $\boldsymbol{r}$ must equal a vector $\mathbf{0}$ of length r, in which each entry is a 0,

$$\boldsymbol{S}\boldsymbol{r} = \mathbf{0} \tag{4.2}$$

This operation can also be formulated as a system of linear equations,

$$\begin{aligned} S_{11}r_1 + \ S_{12}r_2 + \cdots + \ S_{1r}r_r &= 0 \\ S_{21}r_1 + \ S_{22}r_2 + \cdots + \ S_{2r}r_r &= 0 \\ &\vdots \\ S_{m1}r_1 + S_{m2}r_2 + \cdots + S_{mr}r_r &= 0 \end{aligned} \tag{4.3}$$

A metabolic network generally contains more reactions than metabolites, meaning that $m \leq r$. Therefore the system in equation (4.2) contains more unknown variables than equations. This can also be referred to as an underdetermined system [30], which has the consequence that no unique solution exists for equation (4.2) and multiple flux vectors $\boldsymbol{r}$ can fulfil the constraints set by the system. The set of all feasible $\boldsymbol{r}$ forms an r-dimensional solution space, which is called the flux space.

It's very common to be interested in certain points of the flux space where a predetermined reaction μ (objective reaction) has the highest amount of flux. These points can be determined using flux balance analysis (FBA). FBA is a method that wants to maximise the objective function

$$Z = \max r_\mu \tag{4.4}$$

subject to equations (4.1) and (4.2), where r_μ is the flux through reaction μ [29]. The result of FBA is a vector $\boldsymbol{r}$ that lies within the flux space and therefore fulfils all of the constraints of the model, while also containing the highest possible flux through the objective reaction μ.

A common example for the objective reaction is the biomass production, which simulates the growth of a cell. The biomass production can be simulated by adding a biomass reaction which produces biomass and consumes certain metabolites related to the growth of the organism. The stoichiometry of the biomass reaction is determined experimentally. Mathematically, the biomass reaction can be implemented by adding an additional column in the stoichiometric matrix, with negative entries in each row that corresponds to such a metabolite that should be converted into biomass. If properly normalised, the biomass reaction flux r_μ corresponds to the exponential growth rate of the organism [29].

4.2. Construction of a bound-free metabolic model

Consider a metabolic network represented by its stoichiometric matrix, $\boldsymbol{S} \in \mathbb{R}^{m\times r}$ containing the net stoichiometric coefficients of m internal metabolites in r reactions. The vector $\boldsymbol{r} \in \mathbb{R}^r$ denotes a flux distribution (flux vector) through the network, and its components r_j with $j \in \{1, ..., r\}$ are the respective reaction rates or fluxes.

In steady-state, any feasible flux distribution through the network is a solution of the flux polyhedron given by

$$\boldsymbol{S}\boldsymbol{r} = \boldsymbol{0}, \tag{4.5a}$$
$$\boldsymbol{l} \leq \boldsymbol{r} \leq \boldsymbol{u}, \tag{4.5b}$$

where the capacity constraints $\boldsymbol{l}$ and $\boldsymbol{u}$ denote the vectors of lower and upper bounds on the flux distribution. These limits also include irreversibility constraints, where $l_j = 0$ for irreversible reactions $R_j \in I_{\text{irr}}$.

By introducing the slack variable f and non-negative slack variable vectors $\boldsymbol{s}^l \in \mathbb{R}^r_{\geq 0}$ and $\boldsymbol{s}^u \in \mathbb{R}^r_{\geq 0}$ (associated with the lower and upper bounds of the reactions, respectively), the inhomogeneous set of equations in equation (4.5) can be transformed into a (higher-dimensional) set of homogeneous equations[31]

$$\begin{pmatrix} \boldsymbol{0} & \boldsymbol{S} & \boldsymbol{0} & \boldsymbol{0} \\ -\boldsymbol{l} & \boldsymbol{I}_r & -\boldsymbol{I}_r & \boldsymbol{0} \\ \boldsymbol{u} & -\boldsymbol{I}_r & \boldsymbol{0} & -\boldsymbol{I}_r \end{pmatrix} \begin{pmatrix} f \\ \boldsymbol{r} \\ \boldsymbol{s}^l \\ \boldsymbol{s}^u \end{pmatrix} = \boldsymbol{0}, \tag{4.6a}$$
$$\boldsymbol{s}^l \geq \boldsymbol{0}, \tag{4.6b}$$
$$\boldsymbol{s}^u \geq \boldsymbol{0}. \tag{4.6c}$$

Here $\boldsymbol{I}_r$ denotes the identity matrix of size r, where each cell on the diagonal either contains a 1 if the reaction in question is irreversible, or a 0 if the reaction is reversible.

By introducing the normalisation condition

$$f = 1, \tag{4.7}$$

the solution spaces for $\boldsymbol{r}$ in (4.5) and (4.6) become identical. Thus, (4.6) can be interpreted as an equivalent, yet "bound-free" metabolic network, represented by the (transformed) stoichiometric matrix

$$\tilde{\boldsymbol{S}} = \begin{pmatrix} 0 & \boldsymbol{S} & \boldsymbol{0} & \boldsymbol{0} \\ -\boldsymbol{l} & \boldsymbol{I}_r & -\boldsymbol{I}_r & \boldsymbol{0} \\ \boldsymbol{u} & -\boldsymbol{I}_r & \boldsymbol{0} & -\boldsymbol{I}_r \end{pmatrix},$$

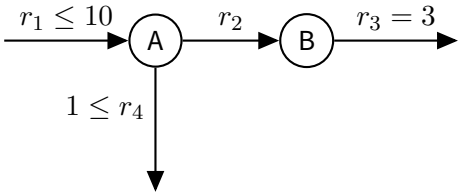
and (transformed) flux vector

$$\tilde{\boldsymbol{r}} = \begin{pmatrix} f & \boldsymbol{r} & \boldsymbol{s}^l & \boldsymbol{s}^u \end{pmatrix}^\mathsf{T},$$

including the irreversible reactions $\boldsymbol{s}^l$, and $\boldsymbol{s}^u$.

Consider the simple constrained metabolic network depicted in Figure 4.2a, composed of four irreversible reactions. Flux variability analysis[32] reveals the feasible ranges for reaction rates: r_1 spans $(4 \text{ to } 10)\,\mathrm{mmol\,g^{-1}\,h^{-1}}$, r_2 and r_3 are both $3\,\mathrm{mmol\,g^{-1}\,h^{-1}}$, and r_4 ranges from $(1 \text{ to } 7)\,\mathrm{mmol\,g^{-1}\,h^{-1}}$.

(a) Original metabolic network



(b) Equivalent, bound-free metabolic network

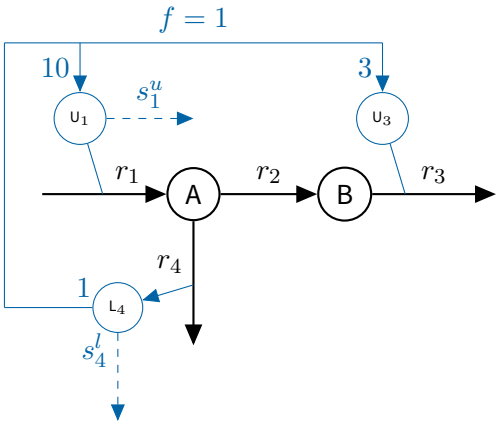


Figure 4.2.: Reconfiguring a constrained metabolic network (panel a) into an equivalent, bound-free metabolic network (panel b) via the set of equations (4.6). Each non-standard bound (in panel a) is substituted with a dummy metabolite (blue circles in panel b). These dummy metabolites are either consumed (for upper bounds) or produced (for lower bounds) by their corresponding reactions. An additional reaction, labelled f (blue line), is introduced to balance these dummy metabolites. It both produces (for upper bounds) and consumes (for lower bounds) these dummy metabolites, with stoichiometric coefficients matched to the corresponding flux bound values. Finally, sink reactions (blue dashed arrows) are added to all dummy metabolites that do not correspond to equality constraints.

The transformation of the toy network shown in Figure 4.2a into its equivalent bound-free network using equation (4.6) yields the network depicted in Figure Fig. 4.2b. Upon comparing Figure 4.2a and Figure 4.2b, it becomes apparent that the flux bounds in Figure 4.2a translate into stoichiometric coefficients for newly introduced dummy metabolites, which are either produced or consumed by the new reaction f. These dummy metabolites function as co-factors, and their availability constrains the respective original reactions r_1 to r_4. Any excess can be removed through additional sink reactions, corresponding to the variables $\boldsymbol{s}^l$ and $\boldsymbol{s}^u$. Note the simplification for the equality constraint $r_3 = 3\,\mathrm{mmol\,g^{-1}\,h^{-1}}$: U_3 is fully consumed by r_3, precisely determining the flux of r_3 and rendering both a sink for U_3 and a dummy metabolite for the lower bound unnecessary. For example, according to equation (4.7) reaction $f = 1\,\mathrm{mmol\,g^{-1}\,h^{-1}}$, which produces $3\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ U_3. U_3 can only be

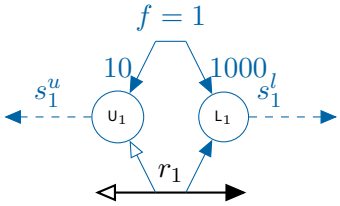


Figure 4.3.: Equivalent, bound-free metabolic network of a constrained, reversible reaction $-1000 \leq r_1 \leq 10$. Full and open arrows indicate forward and reverse directions of r_1, dashed reactions represent sink reactions for the dummy metabolites L_1, and U_1.

consumed by r_3 thus enforcing a flux of $3\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ through reaction r_3. Consequently also $r_2 = 3\,\mathrm{mmol\,g^{-1}\,h^{-1}}$. f also produces $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ U_1, which can be consumed by r_1 or s_1^u, effectively imposing an upper limit of $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ on r_1. Similarly, f drains $1\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ L_4, which enforces a minimum flux of $1\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ through r_4. Any surplus of L_4 is channelled through reaction s_4^l. A minimum flux of $r_1 = 4\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ and a maximum of $r_4 = 7\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ are found upon balancing these fluxes at metabolite A, ultimately fully recovering the limits obtained through flux variability analysis in the original network, as presented above.

In computational practice, the commonly adopted capacity constraints (l_j, u_j) are usually defined as (-1000,1000) for reversible reactions and (0,1000) for irreversible reactions. For instance, consider the reversible reaction $-1000 \leq r_1 \leq 10$, and its equivalent, bound-free network illustrated in Figure 4.3. Note that the negative lower flux bound results in the production of the corresponding dummy metabolite L_1 by the reaction f.

Let's assume r_1 operates at its maximum forward flux. This would lead to r_1 producing $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ of L_1, while reaction f contributes $1000\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ to the production of L_1. Thus, s_1^l needs to drain off a total flux of $1010\,\mathrm{mmol\,g^{-1}\,h^{-1}}$. Such a scenario would be problematic if openCOBRA [27, 33–35] default constraints were applied to s_1^l.

To ensure that sink reactions s_j^l and s_j^u never encounter capacity constraints, it is recommended to employ considerably larger upper bounds for them, such as $1\,000\,000\,\mathrm{mmol\,g^{-1}\,h^{-1}}$.

4.3. Constructing a community model

A community model is a metabolic model containing n single-species models that can interact and grow with one another. While different methods of constructing such a model exist, this work employed the compartmentalised approach, which places each member model in a separate compartment. An additional interspecies exchange compartment is added, which stands in contact with every member model as well as the medium (Figure 4.4). This interspecies exchange compartment makes the exchange of substrates and products between the community members possible. Metabolites from the external compartment, which are not balanced, can be transported into the interspecies exchange compartment, where they turn into internal (balanced) metabolites. These internal metabolites of the interspecies exchange

compartment have to be shared by the community members. This necessitates a merging of identical external metabolites of the single member models into one species in the interspecies exchange compartment.

Furthermore, a new pseudoreaction (4.8) has to be added to each member in the community, which converts the single-species biomass metabolite BM^k into the community biomass metabolite BM_c.

$$BM^k \rightarrow BM_c \tag{4.8}$$

with $k \in \{1...n\}$. Therefore, a community model containing three single species models would have three additional pseudoreactions added, as depicted in (4.8). Furthermore, a community biomass reaction R_{BM_c} is added, which transports the community biomass metabolite BM_c out of the interspecies exchange compartment. The corresponding rate for the community biomass reaction is the community growth rate, r_{BM_c} [36]. As the community growth rate will be the objective reaction of the community models presented in this thesis, the reaction R_{BM_c} and corresponding flux r_{BM_c} will be called R_{μ_c} and r_{μ_c}, respectively. An overview of the structure of a community model is depicted in Figure 4.4.

In contrast to single organism models, the growth of community models is a bit more complex to implement. This is due to the fact that a high r_{μ_c} can sometimes be achieved by some of the members behaving altruistically by neglecting its own growth and prioritising the production of byproducts that other members of the community require. This would result in organisms that behave egoistically to have a much higher biomass production r_μ^k, which in turn would lead to them outgrowing the altruistic members of the community. For this reason it has been argued that for continuous cultivation, each organism in the community needs to have the same specific growth rate. Following this logic, the fractional biomass abundance of each member in the community must be constant as well:

$$F^k = \frac{BM^k}{BM_c} \tag{4.9}$$

with F^k being the fractional abundance of member k in the community composition $\boldsymbol{F} = (F^1, ..., F^n)$ [36]. Taken together, all of abundances F^k must add up to 1.

$$\sum_{k=1}^{n} F^k = 1 \tag{4.10}$$

As mentioned previously, the balanced growth approach demands the specific growth rate of each member to be equal. This growth rate r_μ^k has to be adapted to the fractional abundance of the corresponding member in the community. Considering this, the specific growth rate can be defined as the product of the flux through the community biomass reaction r_{μ_c} and F^k.

$$r_\mu^k = F^k * r_{\mu_c} \tag{4.11}$$

In vitro, a smaller abundance of a member in the community would naturally lead to an overall smaller uptake rate and production of metabolites, as fewer cells of this organism are present.

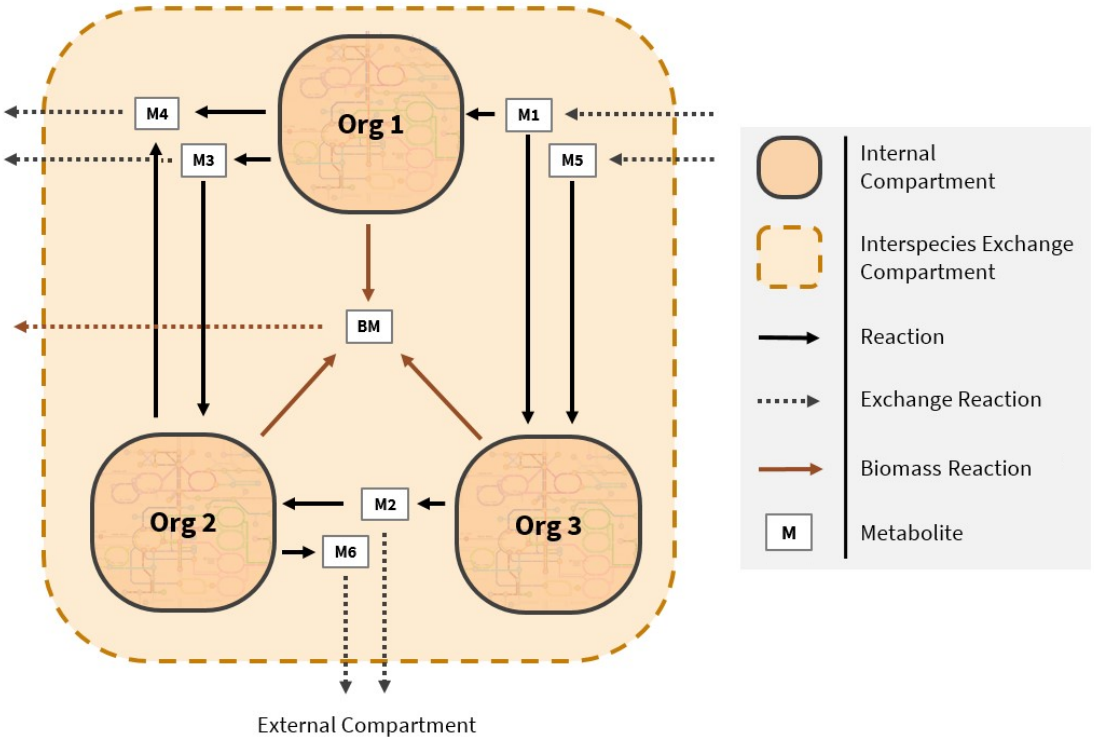


Figure 4.4.: Structure of an example community metabolic model. In a community model, an additional interspecies exchange compartment is introduced. In this compartment, metabolites produced by the members of the community can be exchanged with one another. The metabolites within the interspecies exchange compartment also adhere to the steady-state constraint. This makes the simulation of competitive or cooperative behaviour in the community possible, as only a limited amount of metabolites within the interspecies exchange compartment are present. An exception to this is of course the case in which a metabolite is imported into the interspecies exchange compartment from the external compartment, as the external compartment is not regulated by the steady-state constraint.

This can be simulated by adapting the upper and lower bounds u_j^k and l_j^k to the fractional abundance of the member:

$$F^k * l_j^k \leq F^k * r_j^k \leq F^k * u_j^k \tag{4.12}$$

where r_j^k is the flux and l_j^k and u_j^k are the lower and upper bounds for reaction j in organism k. In other words, the bounds of each reaction have to be multiplied by the fractional abundance of their corresponding organism to ensure a properly constrained community model. This also prevents a member with a fractional abundance of 0 to produce only byproducts while not generating any biomass (which would fulfil equation (4.11)), as according to equation (4.12), all bounds of the system would be set to 0, making any flux impossible.

4.3.1. Equivalent, bound-free community model

In order to compute the community composition space, the fractional abundance of the members and the bounds of the respective reactions would have to be adapted for each point in the composition space before the feasibility of the system could be checked. As this would require a large computation time, the approach of an equivalent, bound-free community model was created [37]. This approach results in a community model where the bounds of the internal reactions in the community members are unconstrained. In order to constrain the reactions without bounds being necessary, an additional reaction labelled f^k is introduced for each member k. The flux through f^k can range from 0 to 1 and corresponds to the relative abundance of the member in the community. Therefore, similar to equation (4.10) the sum of all f^k must equal 1.

$$\sum_{k=1}^{n} f^k = 1 \tag{4.13}$$

This results in a model that inherently adapts the reaction fluxes to the abundances of the members in the community, which saves computation time, as the model itself doesn't need to be changed for different community composition. As described in section 4.2, f^k carries dummy metabolites in a stoichiometry corresponding to the upper and lower bounds of the metabolic reactions in the model. These additional dummy metabolites as well as f^k can be integrated into a transformed stoichiometric matrix $\tilde{\boldsymbol{S}}^k$

$$\tilde{\boldsymbol{S}^k} = \begin{pmatrix} \mathbf{0} & \boldsymbol{S}^k & \mathbf{0} & \mathbf{0} \\ -\boldsymbol{l}^k & \boldsymbol{I}_r^k & -\boldsymbol{I}_r^k & \mathbf{0} \\ \boldsymbol{u}^k & -\boldsymbol{I}_r^k & \mathbf{0} & -\boldsymbol{I}_r^k \end{pmatrix}, \tag{4.14}$$

where $k \in \{1, ..., n\}$, with n being the total number of members contained in the community model. The transformed flux vector $\tilde{\boldsymbol{r}}^k$ includes the irreversible reactions $\boldsymbol{s}^{lk}$ and $\boldsymbol{s}^{uk}$.

$$\tilde{\boldsymbol{r}}^k = \begin{pmatrix} f^k & \boldsymbol{r}^k & \boldsymbol{s}^{lk} & \boldsymbol{s}^{uk} \end{pmatrix}^\mathsf{T}, \tag{4.15}$$

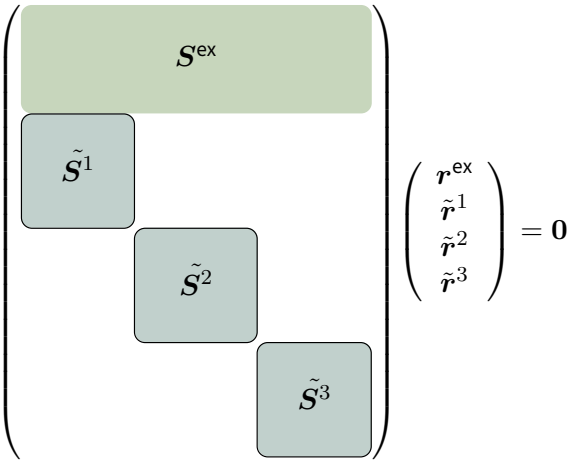
Each of the n members possesses a transformed stoichiometric matrix and flux vector. Merging these into a community would result into a community stoichiometric matrix $\tilde{\boldsymbol{S}}_c$

$$\tilde{\boldsymbol{S}}_c = \begin{pmatrix} \mathbf{0} & \boldsymbol{S}^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ -\boldsymbol{l}^1 & \boldsymbol{I}_r^1 & -\boldsymbol{I}_r^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \boldsymbol{u}^1 & -\boldsymbol{I}_r^1 & \mathbf{0} & -\boldsymbol{I}_r^1 & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{S}^2 & \mathbf{0} & \mathbf{0} & & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & -\boldsymbol{l}^2 & \boldsymbol{I}_r^2 & -\boldsymbol{I}_r^2 & \mathbf{0} & & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \boldsymbol{u}^2 & -\boldsymbol{I}_r^2 & \mathbf{0} & -\boldsymbol{I}_r^2 & & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \vdots & & & & & & & & \ddots & & & & \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & \mathbf{0} & \boldsymbol{S}^n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & -\boldsymbol{l}^n & \boldsymbol{I}_r^n & -\boldsymbol{I}_r^n & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & & \boldsymbol{u}^n & -\boldsymbol{I}_r^n & \mathbf{0} & -\boldsymbol{I}_r^n \end{pmatrix} \tag{4.16}$$

and a community flux vector $\tilde{\boldsymbol{r}}_c$

$$\tilde{\boldsymbol{r}}_c = \begin{pmatrix} f^1 & \boldsymbol{r}^1 & \boldsymbol{s}^{l1} & \boldsymbol{s}^{u1} & f^2 & \boldsymbol{r}^2 & \boldsymbol{s}^{l2} & \boldsymbol{s}^{u2} & \dots & f^n & \boldsymbol{r}^n & \boldsymbol{s}^{ln} & \boldsymbol{s}^{un} \end{pmatrix}^{\mathsf{T}}, \tag{4.17}$$

Following the steady-state constraint, a schematic depiction of a three-membered community with the external reactions of all members collected in $\boldsymbol{S}^{ex}$ would look as follows:



(4.18)

Here, $\boldsymbol{S}^{\text{ex}}$ contains the stoichiometric information of the external reactions of the community. These constitute the connections between the individual members, as any external metabolite present in the external compartment can be shared by the individual members. $\boldsymbol{r}^{\text{ex}}$ describes the flux vector of these external connections.

4.4. Used models

The research herein revolved around three distinct models, ranging from simplified toy models to intricate genome-scale metabolic frameworks.

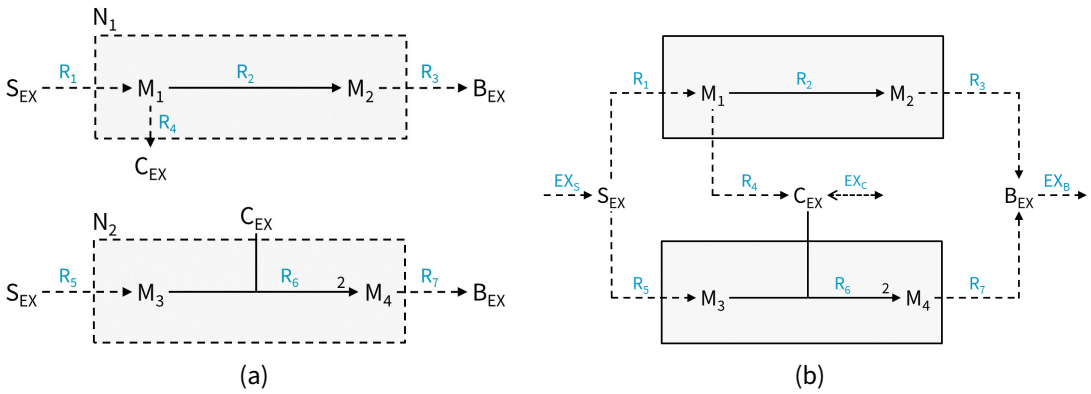


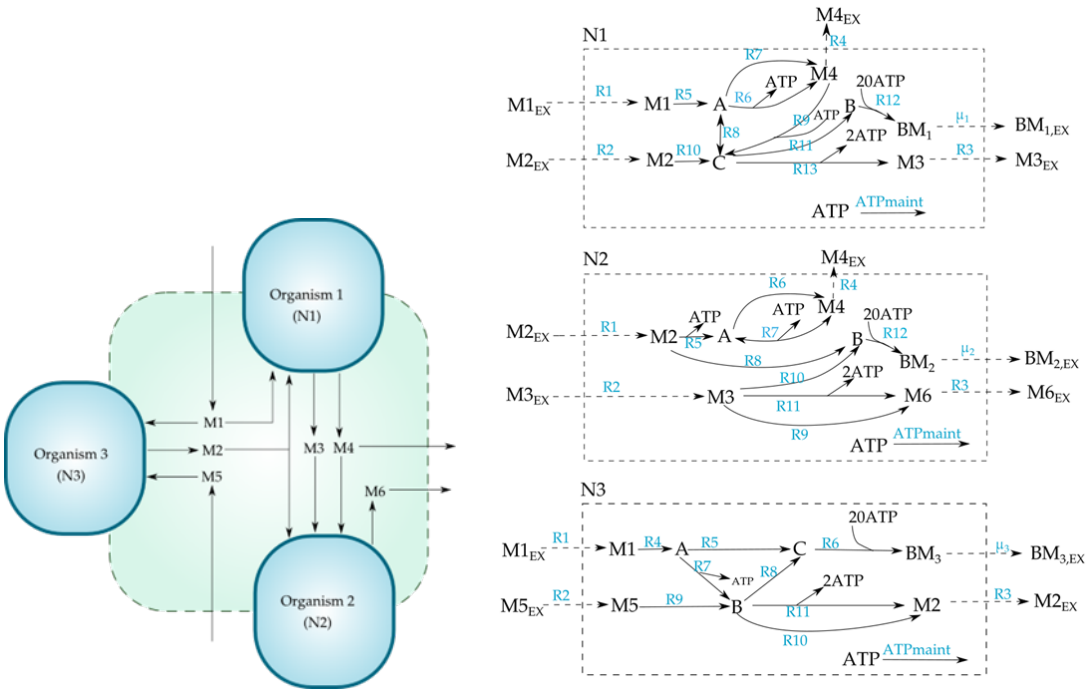
Figure 4.5.: Two-organism toy model. 4.5a: Detailed depiction of the individual members (N_1, N_2) in the community. Dashed arrows: exchange reactions; the index EX marks external metabolites.4.5b: Assembled community model of the two members. The additional exchanges EX_S, EX_C and EX_B were created, linking the medium to the interspecies exchange compartment.

4.4.1. Two-organism toy model

To initiate the study, a simple toy model depicting a two-organism community was introduced. This elementary model consists of only seven distinct metabolites and ten reactions, rendering it suitable for initial studies. The model consist of two cells. One cell is able to grow only on one nutrient source S, while the other cell also requires C to support growth. Using this model, the fundamental methods for calculating community compositions were developed and examined. The simplicity of the model facilitated a comprehensive exploration of interactions and the resultant composition space of the community. Moreover, this initial model served as the foundation for the development of the flux-bound free community model, as it could be validated manually. A schematic of this community model is depicted in Figure 4.5.

4.4.2. Three-organism toy model

Building upon this foundational exploration, a more intricate toy model was adopted from Koch et al [36]. This tripartite model, encompassing three distinct organisms, provided an intermediary complexity level, enabling a bridge between the simplicity of the initial toy model and the more advanced genome-scale metabolic model. The employment of this model allowed for the assessment of a three-organism community, while also retaining the valuable feature of manually verifying the results. An overview of the exchanges between the three organisms is depicted in Figure 4.6a, and a detailed description of the internal reactions of each model can be found in Figure 4.6b.



(a) (b)

Figure 4.6.: Three-organism toy model adopted from Koch et al [36]. 4.6a: Overview of the metabolite exchanges between the three members (N_1, N_2, N_3) of the community and the medium. 4.6b: Detailed depiction of the individual members (N_1, N_2, N_3) in the community. Dashed arrows: exchange reactions; the index EX marks external metabolites.

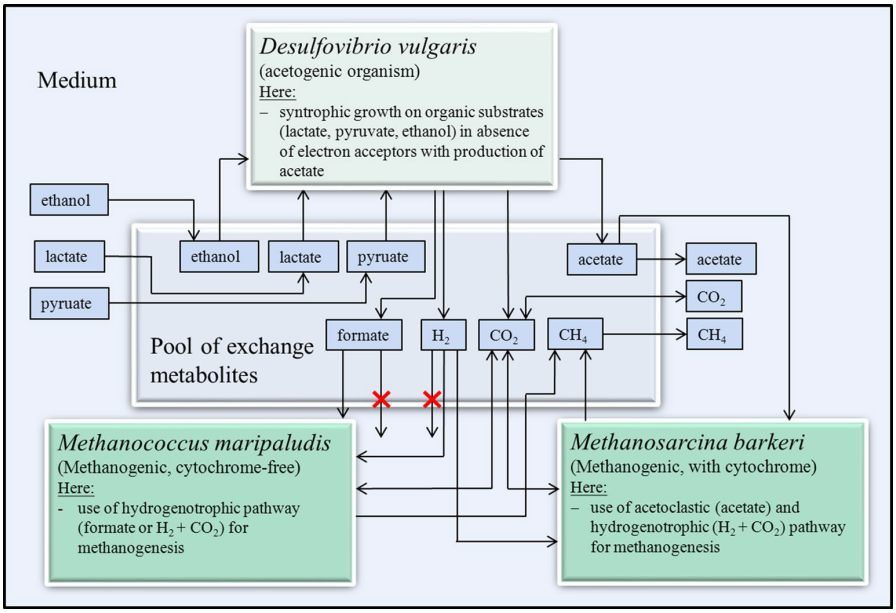


Figure 4.7.: Three-organism genome-scale community adopted from Koch et al [38]. Overview of the metabolite exchanges between the three members (*Desulfovibrio vulgaris*, *Methanococcus maripaludis*, *Methanosarcina barkeri*) of the community and the medium.

4.4.3. Three-organism genome-scale model

Lastly, the genome-scale model described by Koch et al. was investigated [38]. This model revolved around a community consisting of *Desulfovibrio vulgaris*, *Methanococcus maripaludis*, and *Methanosarcina barkeri*. These organisms demonstrated a synergistic relationship in the context of biogas production, particularly under anaerobic conditions.

D. vulgaris is an acetogenic organism, that can use ethanol, lactate, and pyruvate to grow under production of hydrogen, carbon dioxide, and acetate. *M. barkeri* and *M. maripaludis* are methanogenic organisms and therefore form a syntrophic community with the acetogenic *D. vulgaris*. This is due to the fact that methanogenic organisms can use the products of an acetogenic organism to produce biomass. As it is energetically unfavourable for *D. vulgaris* to produce formate and hydrogen if a high concentration of them is already present in the medium, *D. vulgaris* needs the other members of the community to take up hydrogen and formate to keep their concentration in the medium low. Simultaneously, the methanogens have an interest in keeping the hydrogen and formate concentrations low to raise the activity of *D. vulgaris*. *M. maripaludis* requires hydrogen and formate to grow. *M. barkeri* does not utilise formate as a substrate but solely relies on the presence of hydrogen and acetate, the latter being produced by *D. vulgaris*. Throughout this process, both *M. barkeri* and *M. maripaludis* produce methane and carbon dioxide as products. An outline of the community model composed of these three species is depicted in Figure 4.7.

4.5. Computation of feasible community composition spaces

With the models at hand, the computation of the community composition spaces can be initiated. This can be accomplished by two different methods, namely the ECM-Project method and the flux-variability method. The ECM-Project can utilise both the standard and the equivalent, bound-free model. As the computation time of the ECM-Project scales with the number of reactions in the metabolic model, the community model with constraining reaction bounds is the better suited model type.

4.6. ECM-Project Method

The ECM-Project method facilitates the calculation of all vertices within the composition space. The composition space of a community encompasses all potential configurations of community members coexisting while adhering to the community's set constraints.

Taken together, these constraints form a multidimensional polytope, wherein each linear inequality can be interpreted as a closed half-space. A convex polyhedron P can be formulated as

$$P = \{\boldsymbol{x} \in \mathbb{R}^n | \boldsymbol{A}\boldsymbol{x} \geq \boldsymbol{b}\} \tag{4.19}$$

where $\boldsymbol{A} \in \mathbb{R}^{m \times n}$ stands for a matrix and $\boldsymbol{x} \in \mathbb{R}^n$ signifies a vector. Similarly, the feasible flux space of a metabolic model can be described as

$$P = \{\boldsymbol{r} \in \mathbb{R}^r | \boldsymbol{S}\boldsymbol{r} = \boldsymbol{0}, \boldsymbol{I}\boldsymbol{r} \geq \boldsymbol{b}\} \tag{4.20}$$

where $\boldsymbol{S} \in \mathbb{R}^{m \times r}$ stands for the stoichiometric matrix and $\boldsymbol{r} \in \mathbb{R}^r$ signifies the flux vector. $\boldsymbol{I}$ represents the identity matrix, which represents a matrix in which each cell of the diagonal contains a 1 if the cell represents an irreversible reaction. For this, the value of $\boldsymbol{b}$ must be a 0. In case of a reversible reaction, the cell on the diagonal is filled with a 0.

Polytopes can be expressed by two different representations. The V-representation can be described as the convex hull of a finite set of points [39]. The vertices of a polytope define the minimal V-representation. The H-representation, on the other hand, is the bounded intersection of a finite number of closed half-spaces [39].

The computation of one representation from the other is called the vertex enumeration problem and is non-trivial. Commonly employed methods to address the convex hull problem include the Fourier-Motzkin elimination method (also known as the double description method) and the lexicographic reverse search (lrs) [39]. The ECM-project utilises mplrs, which is a parallel implementation of lrs by Avis [40].

The ECM-Project first projects the multidimensional flux cone onto a subspace and then enumerates the extreme rays of this resulting subcone. In the following, I will first discuss the projection of the polytope onto a subcone, which represents the community composition space. Following that, the enumeration of the vertices of that subcone using mplrs (multiprocessing lexicographic reverse search) will be explained.

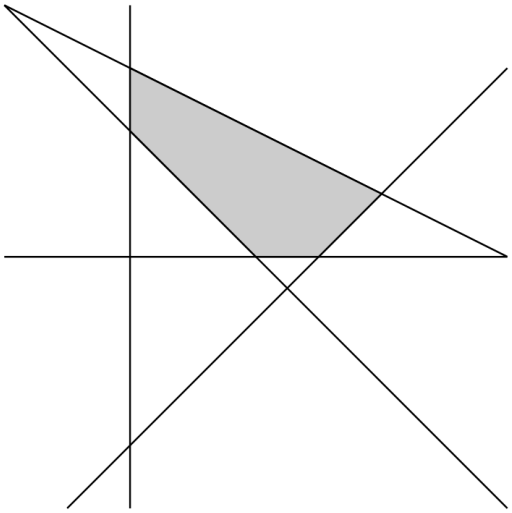


Figure 4.8.: Sketch of the solution space of (4.21)
. Taken from [41].

4.6.1. Projection of polytopes

In the following, a brief overview on the projection of polytopes onto selected dimensions will be given.

Suppose you have a d-dimensional H-representation made up from d halfspaces. The intersection of these halfspaces form a d-dimensional polytope. If you are interested in only a subset of the d dimensions, which will be denoted as subset a, then you have to project the dimensions in subset $c = d - a$ onto the dimensions in a. This is done in a stepwise fashion, meaning that only one dimension is projected at a time. For example, if c contains five dimensions, five projection steps must be performed to obtain a.

Mplrs employs the Fourier-Motzkin elimination [40] for the projection of the polytope. A d-dimensional polytope can be expressed as a set of linear inequalities with multiple variables. The aim of the Fourier-Motzkin elimination is to eliminate one variable after another in a stepwise fashion. Let's consider the following exemplary system, taken from [41]:

$$\begin{aligned} x &\geq 0 \\ x + 2y &\leq 6 \\ x + y &\geq 2 \\ x - y &\leq 3 \\ y &\geq 0 \end{aligned} \tag{4.21}$$

The space described by this system of inequalities is illustrated in Figure 4.8. Under the as-

sumption that

$$\begin{aligned} a \leq & x \\ x \leq & \, b, \end{aligned} \tag{4.22}$$

then the following must also be true:

$$a \leq b \tag{4.23}$$

We can now rewrite (4.21) to show only x on one side of the equation:

$$\begin{aligned} 0 \leq & x \\ x \leq & \, 6 - 2y \\ 2 - y \leq & x \\ x \leq & \, 3 + y \\ y \geq & \, 0. \end{aligned} \tag{4.24}$$

There are now multiple inequalities that define x. But we can further simplify (4.24) by stating that only the terms closest to the minimum and maximum of all possible x define the interval boundaries of x.

$$\begin{aligned} x \leq & \, \min\{6 - 2y, 3 + y\} \\ \max\{0, 2 - y\} \leq & x \\ y \geq & \, 0. \end{aligned} \tag{4.25}$$

By adhering to (4.23), we can express this system without the variable x as

$$\begin{aligned} \max\{0, 2 - y\} \leq & \, \min\{6 - 2y, 3 + y\} \\ y \geq & \, 0. \end{aligned} \tag{4.26}$$

This leaves us with the one-dimensional system

$$\begin{aligned} 0 \leq & \, 6 - 2y \\ 0 \leq & \, 3 + y \\ 2 - y \leq & \, 6 - 2y \\ 2 - y \leq & \, 3 + y \\ 0 \leq & \, y. \end{aligned} \tag{4.27}$$

This is a simple inequality system containing only one variable and can now be solved.

$$0 \leq y \leq 3 \tag{4.28}$$

Therefore, we can conclude that the solution of this inequality system is an area that can be described by the following two expressions:

$$\begin{aligned} 0 \leq & y \leq 3 \\ \max\{0, 2 - y\} \leq & x \leq \min\{6 - 2y, 3 + y\} \end{aligned} \tag{4.29}$$

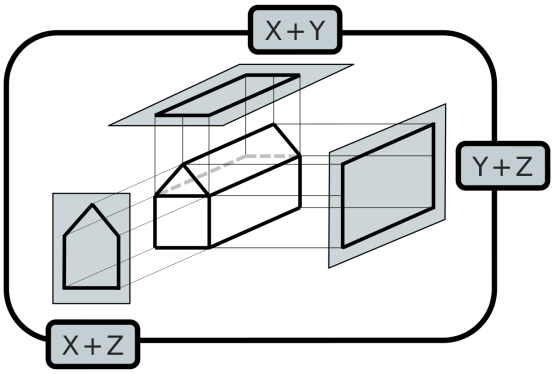


Figure 4.9.: Projection of a three-dimensional shape onto a two-dimensional plane. The shaded areas represent the two-dimensional plane in which the projection lies.

In a system with d variables, this projection procedure must be repeated iteratively until the system can be expressed solely by the desired variables. Naturally, during each projection, information loss occurs. This becomes more apparent when you think about projection as a way to determine the shadow of a multidimensional object. The shadow of an object still inherently provides some context of the shape of an object, but does not contain information on the removed dimensions. In Figure 4.9, a three-dimensional object in the form of a house is projected onto a two-dimensional space. Depending on the dimensions that are determined to be of interest, the projection can be conducted in three different ways, resulting in three different “shadows”.

4.6.2. Lexicographic reverse search

Reverse search is a method of identifying all vertices of a polytope P, which form the V-representation of P. This involves the initial selection of a specific vertex $\boldsymbol{v}^*$ of P. From there, paths are traced along the edges of the polytope until no further vertices remain. This procedure can be compared to the traversal of a spanning tree. A spanning tree T is a subgraph within a graph $G = (V, E)$ which contains all vertices V, but must not include all edges E of G. In fact, a spanning tree describes a subgraph with the fewest possible edges while still encompassing all of G's vertices. This design prevents cycles within the tree structure, as the removal of an edge belonging to a cycle inevitably decreases the tree's edge count while maintaining the same number of enclosed vertices. An illustration of this concept is presented in Figure 4.10 for a simple graph G and its corresponding trees. Within the mplrs application in this project, graph G encompasses the composition space of a metabolic model, as well as the flux space of the reactions.

One method of determining a spanning tree of a graph is the repeated application of the simplex algorithm for linear programming. The simplex method requires a system of inequalities that construct a polyhedron $P \in \mathbb{R}^d$ along with a contained vertex [43]. The algorithm then progresses by tracing along the edges of P while seeking a vertex that maximises a given objective function. This procedure is repeated for each vertex within the polyhedron, eventu-

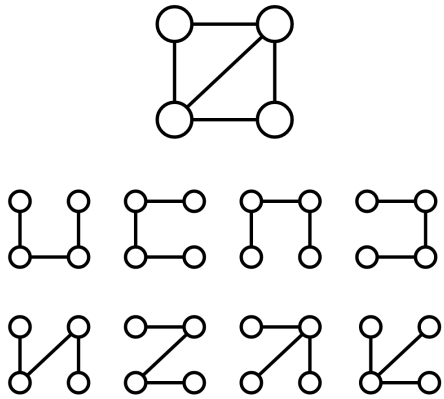


Figure 4.10.: A connected graph G (top) and its corresponding spanning trees T_i (bottom). Each node contained in G must also be contained in T_i. Adapted from [42].

ally yielding a spanning tree for G, where the optimal vertex functions as the tree's root. To ensure a unique path for each starting vertex, the Bland's rule, also referred to as the least subscript rule is employed [44].

The lrs algorithm, in contrast, reverses the simplex method by taking the optimal vertex (and thereby the root of the tree) as the initial vertex $\boldsymbol{v}^*$. This is done by initially selecting a random vertex and creating an objective function which has that vertex as its optimum. Subsequently, the algorithm starts tracing the tree in depth-first order [43]. Figure 4.11 depicts the concept of a depth-first order search (DFS) with a simple model. One of the adjacent nodes to $\boldsymbol{v}^*$ (node 1) is selected based on a predetermined criterion. In this example, the leftmost unexplored node (node 2) is picked first. Next, the adjacent nodes of node 2 which are further down the tree are computed, and so on. When a leaf node is encountered, indicating the end of one particular path, the traversal transitions to a "backtracking" phase. In the example, this backtracking initiates upon encountering node 4. Reversing back up the tree can simply be accomplished by adhering to Bland's rule, as it unfailingly guides towards the node closer to the optimal vertex $\boldsymbol{v}^*$. This backtracking continues stepwise until a node surfaces that still possesses some unexplored nodes further down the tree. At this juncture, the backtracking halts until this selected branch is fully explored. The lrs algorithm employs the same principle, selecting nodes in lexicographic order.

A deciding advantage of lrs compared to conventional depth-first searches is the absence of a stack [45]. Usually, a successful backtracking procedure requires the visited nodes to be stored in the list $\boldsymbol{v}$, as well as the unexplored nodes in the stack $\boldsymbol{s}$. When the path encounters a new node, this node is added to $\boldsymbol{v}$, while its adjacent nodes $n_i \notin \boldsymbol{v}$ are put on top of the stack $\boldsymbol{s}$. Then, the topmost object of the stack is viewed and selected as the next node to be visited. If a node does not possess any $n_i \notin \boldsymbol{v}$, no new objects are added to the stack, which signals the start of the backtracking phase. As always, the topmost item in the stack is visited, which is now a node at the same level as the current node or further up the tree. This concept is illustrated in Figure 4.12. In contrast to this, lrs works without storing the visited nodes or unexplored nodes in any data structure. The algorithm always knows which node

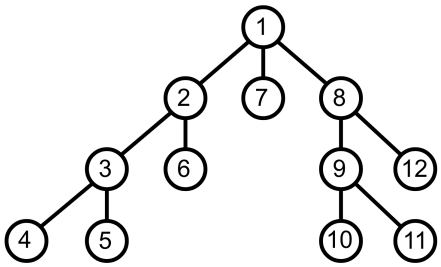


Figure 4.11.: Example of a depth-first order search (DFS). The numbers within the nodes represent the order in which the path of the spanning tree is traced.

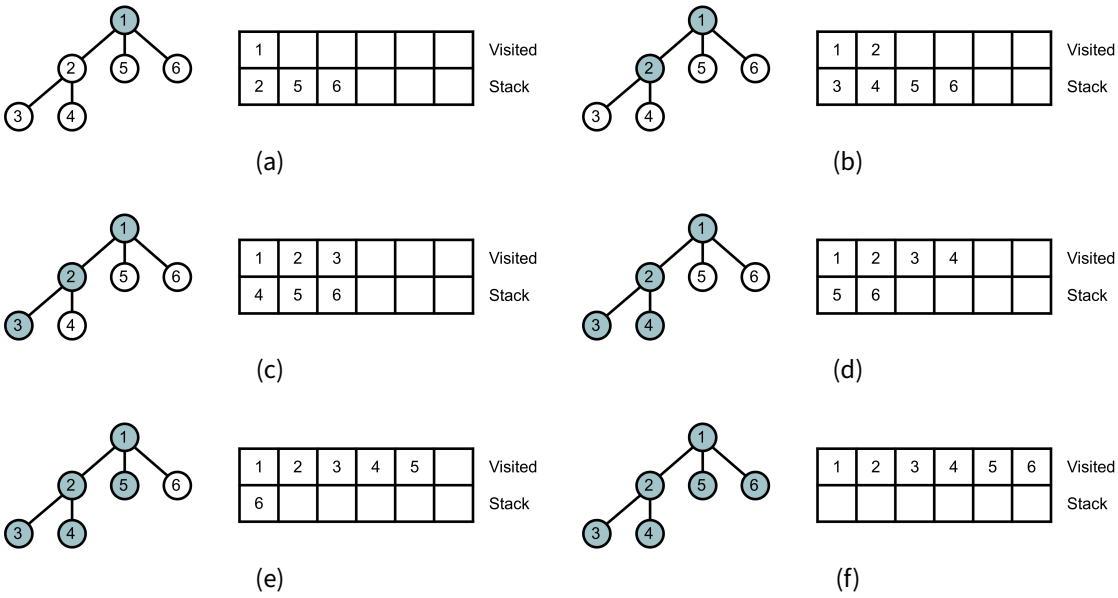


Figure 4.12.: Example of a depth-first search algorithm using the stack. 4.12a: n_1 is put into $\boldsymbol{v}$. The adjacent nodes $n_{2,5,6}$ are put into the stack $\boldsymbol{s}$. The node on the top of the stack is n_2, which will be visited next. 4.12b: n_2 is added to $\boldsymbol{v}$. As n_2 has two adjacent nodes $n_{3,4} \notin \boldsymbol{v}$, they are pushed to the top of the stack. 4.12c: n_3 is a leaf node and is not adjacent to any other unexplored nodes. Therefore, no new nodes are added to the stack in this step and the backtracking phase starts. In 4.12d, 4.12e and 4.12f, the stack is disassembled in a stepwise fashion, until every node has been visited.

to visit next as it traces a path in lexicographic order. In the backtracking phase, Bland's rule is used to return to the parent node, which is followed by the selection of the next node in the lexicographic order [43]. Therefore, only one node must be stored at any given time, to prevent backtracking to the parent node and selection of the just visited node. The absence of a stack is a key feature that enables the parallelisation of the lrs process over a network of computers. Moreover, the mplrs algorithm can be paused at any point and resumed at a later time with only the information of the current node [45]. This is especially of use for very large models, where computation may be interrupted at some point.

4.6.3. Parallel lexicographic reverse search

Parallel execution of the lrs algorithm has been achieved by the mplrs module [40], which utilises a standard message passing interface (MPI) [46] and improved load balancing in order to distribute the computation of the V-representation over multiple processes. This, in turn, leads to a significant improvement in performance, especially for medium to large-scale networks. Three different roles are assigned to the processes running in parallel: a *master*, multiple *workers*, and a *consumer*. One process becomes the master and is responsible for distributing the input file and later on the subproblems to the workers. It is also responsible for checkpointing and stopping other processes. The workers receive the subproblems from the master and start running the lrs algorithm on their subproblem. Each worker has a certain *budget* that it can expend. Concretely, the budget describes the number of nodes a worker can visit. As soon as the budget is depleted, the worker starts the backtracking process, returning unexplored children as new subproblems to the master. The master then goes on to redistribute all of these new subproblems to idle workers. If the subproblem is larger than a certain threshold the budget of the worker depletes before the subproblem is solved. The worker then stops at that point and sends the unfinished subproblems back to the master. If the subproblem is within the set budget, the worker can completely evaluate the remaining subtree and pass the output to the consumer. The consumer collects all outputs received from workers and delivers a continuous output stream to the user.

Initially, the master sends the input file, which is the polyhedron for which the vertices should be calculated, to all workers. Following that, it provides a single worker with a subproblem. This worker runs the lrs code and visits the nodes of his subproblem in lexicographic order until its budget is depleted. Only after the first worker returns multiple subproblems can the parallel computation begin. Depending on the input polyhedron, certain parameters are set for this initial worker that limit the depth and budget of this first calculation. This ensures that a large amount of subproblems are returned in a short amount of time. The master is also able to set a checkpoint during the calculation [45]. For this, the master stops distributing subproblems to idle workers and tells the remaining workers to quit. While the workers backtrack to the starting vertex, the master collects all subproblems and saves them in a checkpoint file. This file can later on be reloaded by the master, who will then assign the subproblems to the workers again.

4.6.4. Constructing the input matrix for mplrs

As discussed in the previous section, mplrs calculates the vertices of an input polytope. We defined the polytope of a metabolic models flux space as

$$P = \{\boldsymbol{r} \in \mathbb{R}^r | \boldsymbol{S}\boldsymbol{r} = \boldsymbol{0}, \boldsymbol{I}\boldsymbol{r} \geq \boldsymbol{b}\} \tag{4.30}$$

with $\boldsymbol{S} \in \mathbb{R}^{m \times r}$ being the stoichiometric matrix and $\boldsymbol{r} \in \mathbb{R}^r$ being the flux vector. $\boldsymbol{b}$ is a vector of variables. P can also be depicted as a set of equations.

$$\begin{aligned} S_{11}r_1 + S_{12}r_2 + \cdots + S_{1r}r_r + b_1 \geq 0 \\ S_{21}r_1 + S_{22}r_2 + \cdots + S_{2r}r_r + b_2 \geq 0 \\ \vdots \\ S_{m1}r_1 + S_{m2}r_2 + \cdots + S_{mr}r_r + b_m \geq 0 \end{aligned} \tag{4.31}$$

If we compare this set of equations to the steady-state constraint

$$\boldsymbol{S} * \boldsymbol{r} = \boldsymbol{0} \tag{4.32}$$

we can conclude that b_i in each of the equations in (4.31) must have the value 0. Furthermore, the sign of the equations should be replaced with $=$ instead of $\geq$. This can be done by defining a "linearity set", which contains the indexes of the rows that should be equalities.

The mplrs input is in the form of a matrix. The matrix corresponding to P would look as follows:

$$\begin{array}{c} \\ M_1 \\ M_2 \\ \\ M_m \end{array} \begin{array}{c} \begin{array}{cccccc} b & R_1 & R_2 & & R_r \end{array} \\ \begin{pmatrix} 0 & S_{11} & S_{12} & \cdots & S_{1r} \\ 0 & S_{21} & S_{22} & & S_{2r} \\ \vdots & & & \ddots & \\ 0 & S_{m1} & S_{m2} & & S_{mr} \end{pmatrix} \end{array} \tag{4.33}$$

with the linearity list

$$linearity = \{1, 2, ..., m\}. \tag{4.34}$$

In (4.33), M_i and R_j describe the metabolites and reactions of the model. S_{ij} therefore describes the stoichiometric coefficient of metabolite M_i in reaction R_j.

It is evident that the upper and lower bound constraints are not accounted for in P. This can be remedied by adding additional linear equations to system (4.31):

$$\begin{aligned} -l_1 + r_1 \geq 0 \\ -l_2 + r_2 \geq 0 \\ \vdots \\ -l_r + r_r \geq 0 \end{aligned} \tag{4.35}$$

as well as

$$\begin{aligned} u_1 + r_1 &\geq 0 \\ u_2 + r_2 &\geq 0 \\ &\vdots \\ u_r + r_r &\geq 0 \end{aligned} \tag{4.36}$$

with l_j being the lower bound of reaction R_j with $j \in \{1, 2, ..., r\}$ and u_j being the upper bound. To integrate this into the mplrs input matrix (4.33), b_i takes on the form of $-l_j$ or u_j, and a 1 is listed in the column of the corresponding reaction R_j. The remaining cells of the row (which correspond to other reactions) contain a 0. Including the lower bound constraints $\boldsymbol{l}$ and upper bound constraints $\boldsymbol{u}$, the mplrs input matrix would look like this:

$$\begin{array}{c} \\ M_1 \\ M_2 \\ \vdots \\ M_m \\ \\ l_1 \\ l_2 \\ \vdots \\ l_r \\ \\ u_1 \\ u_2 \\ \vdots \\ u_r \end{array} \begin{array}{c} b \\ \left(\begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \\ \\ -l_1 \\ -l_2 \\ \vdots \\ -l_r \\ \\ u_1 \\ u_2 \\ \vdots \\ u_r \end{array} \right. \end{array} \begin{array}{c} R_1 \\ \begin{array}{c} S_{11} \\ S_{21} \\ \vdots \\ S_{m1} \\ \\ 1 \\ 0 \\ \vdots \\ 0 \\ \\ 1 \\ 0 \\ \vdots \\ 0 \end{array} \end{array} \begin{array}{c} R_2 \\ \begin{array}{c} S_{12} \\ S_{22} \\ \vdots \\ S_{m2} \\ \\ 0 \\ 1 \\ \vdots \\ 0 \\ \\ 0 \\ 1 \\ \vdots \\ 0 \end{array} \end{array} \begin{array}{c} \\ \begin{array}{c} \dots \\ \dots \\ \\ \dots \\ \\ \dots \\ \dots \\ \\ \dots \\ \\ \dots \\ \dots \\ \\ \dots \end{array} \end{array} \begin{array}{c} R_r \\ \begin{array}{c} S_{1r} \\ S_{2r} \\ \vdots \\ S_{mr} \\ \\ 0 \\ 0 \\ \vdots \\ 1 \\ \\ 0 \\ 0 \\ \vdots \\ 1 \end{array} \left. \begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{array} \right) \end{array} \tag{4.37}$$

with the linearity list

$$linearity = \{1, 2, ..., m\}. \tag{4.38}$$

The linearity list stays the same, as the added rows correspond to inequalities. Such an input matrix would enable mplrs to calculate the feasible flux polyhedron of a metabolic model, which is an r-dimensional space encompassing all feasible flux vectors $\boldsymbol{r}$. Therefore, the flux space of a metabolic model could be calculated in that manner. However, we are not necessarily interested in the flux space, but rather the composition space of the community. For that reason, an additional column f^k for each member k of the community must be added to the matrix. f^k is a variable corresponding to the relative abundance of the member k in the community. The sum of all f^k must equal 1.

$$\sum_{k=1}^{n} f^k = 1 \tag{4.39}$$

where n is the total number of members in the community. These f^k columns serve the purpose of weighting the reaction fluxes as well as the upper and lower bounds of the community. Suppose $\boldsymbol{S^k}$ is the stoichiometric matrix of member k with $k \in \{1, 2, ...n\}$, then the set of equations (4.31) can be described as

$$
\begin{aligned}
&S^1_{11}r^1_1 \cdot f^1 + S^1_{12}r^1_2 \cdot f^1 + \cdots + S^1_{1r}r^1_r \cdot f^1 + b^1_1 \cdot f^1 \geq 0 \\
&\qquad \vdots \\
&S^1_{m1}r^1_1 \cdot f^1 + S^1_{m2}r^1_2 \cdot f^1 + \cdots + S^1_{mr}r^1_r \cdot f^1 + b^1_m \cdot f^1 \geq 0 \\
&S^2_{11}r^2_1 \cdot f^2 + S^2_{12}r^2_2 \cdot f^2 + \cdots + S^2_{1r}r^2_r \cdot f^2 + b^2_1 \cdot f^2 \geq 0 \\
&\qquad \vdots \\
&S^2_{m1}r^2_1 \cdot f^2 + S^2_{m2}r^2_2 \cdot f^2 + \cdots + S^2_{mr}r^2_r \cdot f^2 + b^2_m \cdot f^2 \geq 0 \\
&\qquad \vdots \\
&S^n_{11}r^n_1 \cdot f^n + S^n_{12}r^n_2 \cdot f^n + \cdots + S^n_{1r}r^n_r \cdot f^n + b^n_1 \cdot f^n \geq 0 \\
&\qquad \vdots \\
&S^n_{m1}r^n_1 \cdot f^n + S^n_{n2}r^n_2 \cdot f^n + \cdots + S^n_{mr}r^n_r \cdot f^n + b^n_m \cdot f^n \geq 0
\end{aligned}
\tag{4.40a}
\tag{4.40b}
\tag{4.40c}
$$

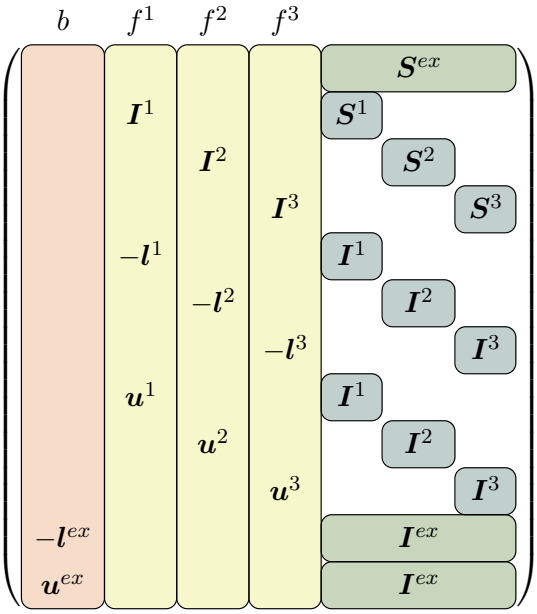
In other words, the stoichiometric matrix of each member is multiplied by its corresponding abundance in the community. This adapts the stoichiometry of the internal reactions of each member. The higher f^k and therefore the abundance of member k with $k \in \{1, 2, ...n\}$ is, the heavier the fluxes through their reactions are weighted. As well as that, the lower bounds and upper bounds have to be adapted to the relative abundances.

$$
\begin{aligned}
r^k_1 \cdot f^k - l^k_1 \cdot f^k &\geq 0 \\
&\vdots \\
r^k_r \cdot f^k - l^i_r \cdot f^k &\geq 0
\end{aligned}
\tag{4.41a}
$$

$$
\begin{aligned}
r^k_1 \cdot f^k - u^k_1 \cdot f^k &\leq 0 \\
&\vdots \\
r^k_r \cdot f^k - u^k_r \cdot f^k &\leq 0
\end{aligned}
\tag{4.41b}
$$

Therefore, the lower bound and upper bound values have to be entered into the f^k column of the mplrs input matrix. A three-membered community model would therefore have an input

matrix like this:



(4.42)

with $\boldsymbol{S}^{ex}$ being the stoichiometric matrix of the external reactions and metabolites that are shared between the community members, while $\boldsymbol{S}^k$ make up the stoichiometric matrices of the members. $\boldsymbol{I}^k$ is the identity matrix of the internal reactions in member k, while $\boldsymbol{I}^{ex}$ is the identity matrix of the external reactions. Therefore, each reaction becomes linked to its bounds, with the internal reaction bounds being multiplied by the relative abundance f^k. Consequently, low f^k lead to smaller reaction bounds in the community member. External reactions are restrained by their original bounds which are not multiplied by any f^k.

Now, only two constraints remain to be added to the mplrs input matrix. The first one is described by equation (4.39), meaning that the sum of all f^k must equal 1. This can be enforced by adding the following row to the mplrs input matrix:

$$\begin{array}{cccc} b & f^1 & f^2 & f^3 \\ \left(\begin{array}{|c|c|c|c|} \hline -1 & 1 & 1 & 1 \\ \hline \end{array}\right. & & & \left.\right) \end{array} \tag{4.43}$$

The second remaining constraint that has to be added concerns itself with balanced growth. As described in section 4.3, we assume balanced growth for the community models, meaning that each member of the community has the same growth rate as the overall community growth rate r_{μ_c}. The growth rates r_μ^k of the members get adapted by the relative abundance of the corresponding member, therefore

$$r_\mu^k = r_{\mu_c} \tag{4.44}$$

$$\sum_{k=1}^{n} r_\mu^k \cdot f^k = r_{\mu_c} \tag{4.45}$$

To ensure that this constraint is met, the following row has to be added to the input matrix for each member k in the community.

$$\begin{array}{c} \begin{array}{cccccccccc} b & f^1 & f^2 & f^3 & \dots & R^1_\mu & \dots & R^2_\mu & \dots & R^3_\mu & \dots \end{array} \\ \left(\begin{array}{cccc} & -r_{\mu_c} & & & & 1 & & & & & \\ & & -r_{\mu_c} & & & & & 1 & & & \\ & & & -r_{\mu_c} & & & & & & 1 & \end{array} \right) \end{array} \tag{4.46}$$

The finished mplrs input matrix would therefore be constructed like this:

$$\begin{array}{c} \begin{array}{cccc} b & f^1 & f^2 & f^3 \end{array} \\ \left(\begin{array}{cccc|c} & \boldsymbol{I}^1 & & & \boldsymbol{S}^{ex} \\ \rightarrow & & & & \boldsymbol{S}^1 \\ \rightarrow & & \boldsymbol{I}^2 & & \boldsymbol{S}^2 \\ \rightarrow & & & \boldsymbol{I}^3 & \boldsymbol{S}^3 \\ & -\boldsymbol{l}^1 & & & \boldsymbol{I}^1 \\ & & -\boldsymbol{l}^2 & & \boldsymbol{I}^2 \\ & & & -\boldsymbol{l}^3 & \boldsymbol{I}^3 \\ & \boldsymbol{l}^1 & & & \boldsymbol{I}^1 \\ & & \boldsymbol{l}^2 & & \boldsymbol{I}^2 \\ & & & \boldsymbol{l}^3 & \boldsymbol{I}^3 \\ -\boldsymbol{u}^{ex} & & & & \boldsymbol{I}^{ex} \\ \boldsymbol{l}^{ex} & & & & \boldsymbol{I}^{ex} \\ \rightarrow -1 & 1 & 1 & 1 & \\ \rightarrow & -r_{\mu_c} & & & 1 \\ \rightarrow & & -r_{\mu_c} & & \quad 1 \\ \rightarrow & & & -r_{\mu_c} & \quad\quad 1 \end{array} \right) \end{array} \tag{4.47}$$

Note that the 1s in the last three rows are placed in the columns corresponding to reactions R^1_μ, R^2_μ, R^3_μ as shown in equation 4.46. The linearity set would include the indices of the rows contained in the sections that are marked with an arrow ($\rightarrow$).

With this input matrix, mplrs can generate the halfspace-representation (H-representation) of the $\sum_1^n r^k + n + 1$-dimensional polyeder P. Each column (apart from the very first b-column) represents one dimension in P. The composition space of the community model is represented by the n dimensions of the f^k columns, as these represent the fractions of the members in the community at any given feasible point in the flux space. To obtain the vertices of this composition space, the polyeder P has to be projected onto the f^k dimensions. Following that, the vertices of the n-dimensional polytope can be calculated via mplrs.

4.6.5. Constructing a composition space from computed vertices

The composition space can be easily created by calculating the convex hull of the gained vertices. This results in a multidimensional space, wherein each dimension signifies the abundance of a member in the community. Any one point within that space constitutes a feasible community composition and can be described as a set of the individual vertices. This provides the possibility of gaining a comprehensive overview of the communities dependencies. Varying the biomass yield can lead to different community compositions, as the members may have to work more closely together to output high biomass yields.

4.7. Flux variability analysis method

In the flux variability analysis method, the equivalent, bound-free community model described in section 4.2 is required. Here, the model contains an added fraction reaction for each member in the community. The flux through these fraction reactions f^k with $k \in \{1, 2, ...n\}$, which can range from 0 to 1 stands for the relative abundance of the corresponding member in the community. By finding out the minimal and maximal flux through these reactions, the community composition space can be approximated.

This can easily be achieved by employing flux variability analysis (FVA). FVA is a method that can compute the flux range of all reactions in a model that still satisfy the set constraints [47]. The computation of the composition space with n members using FVA follows the following steps:

1. Calculate maximal and minimal flux of f^1
2. Calculate maximal and minimal flux of f^2 x times. At every iteration, the flux of f^1 is fixed to a different point in the range of the minimal and maximal flux of f^1.
3. Repeat the second step for each f^k reaction in the community model whilst fixating the flux of the already computed f^k reactions to different points between their maximal and minimal fluxes.

The result of this approach is a collection of n dimensional points, which trace the edges of the community composition space. The higher the amount of iterations x is, the better the resolution of the resulting composition space.

5. Results and Discussion

5.1. Two-organism toy model

The simple two-organism toy model is primarily used as a proof of concept. The comparatively small size of the network makes manual validation of the computed results possible. The structure of the model is depicted in Figure 5.1a. The upper bounds of R_1 and R_5 were set to $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$. This limits the uptake rate of substrate S_{EX} to a maximum of $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$. An uptake of $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$ is considered the conventional amount, as the bacterium E. coli shows an glucose uptake of $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$. Setting the uptake constraints in that manner makes the model comparable to other models. The community composition of this model was calculated under three distinct conditions:

1. EX_C is unconstrained. Therefore, the uptake and excretion of C_{EX} is allowed
2. The lower bound of EX_C is set to 0. Therefore, only the excretion of C_{EX} is allowed.
3. The lower and upper bound of EX_C are set to 0. Therefore, neither uptake nor excretion of C_{EX} are possible.

5.1.1. Flux variability analysis method

A bound-free model was created using PyCoMo[37]. Here, multiple reactions are added to the model, its new structure is depicted in Figure 5.1. Metabolite F is created by the two reactions f^1 and f^2. It is transported out of the system by f^{fin}, which has its flux set to 1. This imposes the constraint of $f^1 + f^2 = 1$. As mentioned above, R_1 and R_5, which represent the uptake reactions of S_{EX} of the two members N_1 and N_2 were constrained by an upper bound of $10\,\mathrm{mmol\,g^{-1}\,h^{-1}}$. Therefore, the fraction reactions f^1 and f^2 introduced in the bound free model generate the dummy metabolites U_1 and U_5, respectively. U_1 is produced in a stoichiometry of 10 by f^1 and consumed by R_1 at a stoichiometry of 1. Similarly, U_5 is produced by f^2 in a stoichiometry of 10 and consumed by R^5 in a stoichiometry of 1. Therefore, if the flux $r_{R_1} = 10$, then $r_{f^1} = 1$ because R_1 requires ten units of the dummy metabolite U_1. As $f^1 + f^2 = 1$, $r_{f^2} = 0$, which results in $r_{R_5} = 0$, as no dummy metabolite U_5 is produced by f^2, which would be a required metabolite of R_5.

Similarly, the biomass reactions R_3 and R_7 are constrained by the fraction reactions. The upper and lower bounds of these reactions are set to r_{μ_c}. For this community, the maximum biomass yield $r_{\mu_{max}} = 10$, therefore EX_B was set to numbers of increasing size between 0 and μ_{max}. This is done in order to observe whether the community behaves differently with different growth rates. The community compositions that were computed using the flux

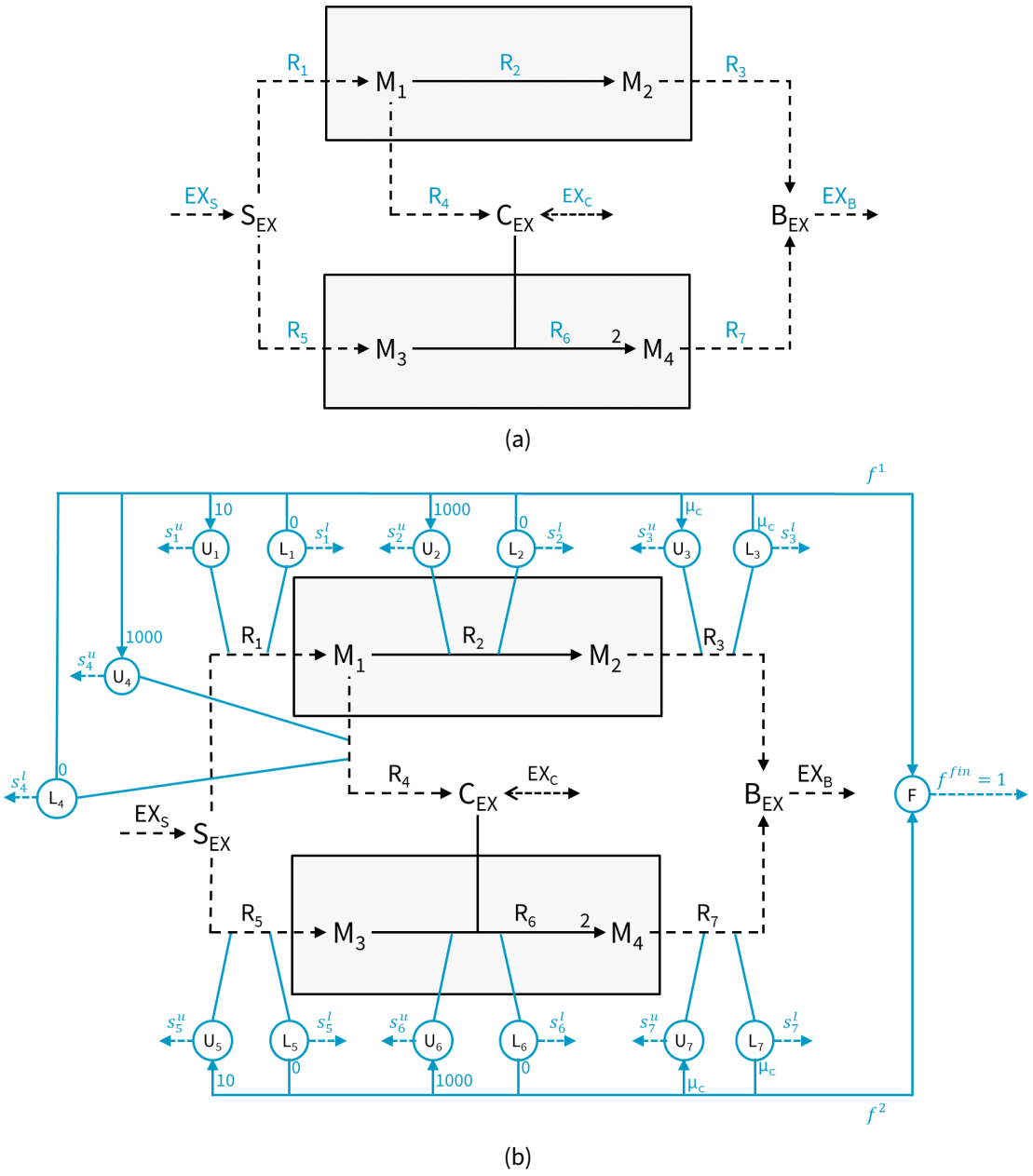
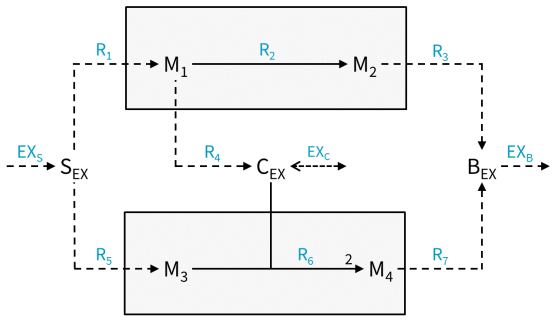
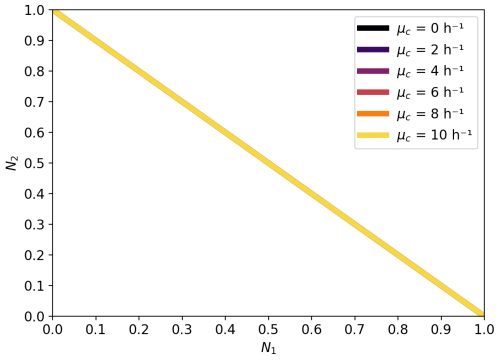


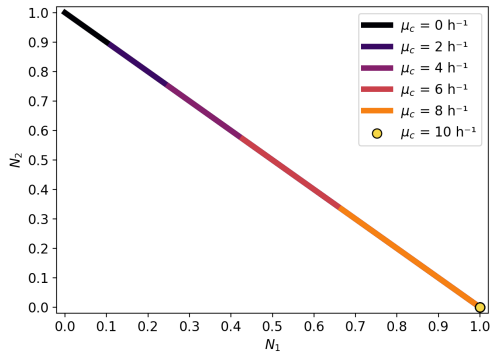
Figure 5.1.: Two-organism toy model. Dashed arrows: exchange reactions; the index EX marks external metabolites. 5.1a: Assembled community model of the two members. 5.1b: Equivalent, bound-free community model. The blue arrows represent the added reactions which impose the constraints on the network.



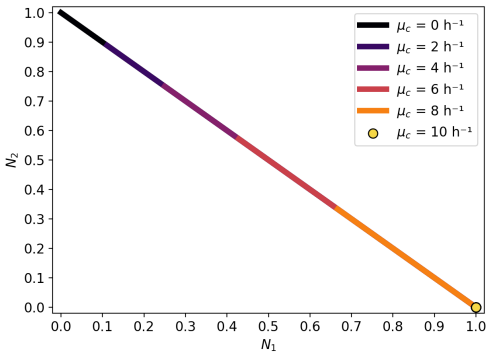
(a) Structure of model.



(b) EX_C is reversible.



(c) EX_C is irreversible.



(d) EX_C is deactivated.

Figure 5.2.: Results of the 2-member toy model. 5.2a: Structure of the toy model. For simplicity, the fraction reactions f^k are not included in the depiction.5.2b: Composition space of the model with the reaction EX_C being unconstrained. 5.2c: Composition space of the model with the lower bound of reaction C_{EX} being set to zero. 5.2d: Composition space of the model with the lower and upper bounds of reaction C_{EX} being set to zero.

variability analysis method 4.7 are shown in Figure 5.2. 5.2b shows the composition space of the model with the reaction EX_C being unconstrained. Therefore, the uptake and excretion of C_EX are possible. The point where $N_1 = 1$ and $N_2 = 2$ would stand for a community composition where 100% of the community would consist of organism N_1. Similarly, the point with $N_1 = 0$ and $N_2 = 1$ describes a community with 100% N_2. As every point between these two extremes is feasible as well, the maximum biomass production $r_{\mu_{max}} = 10$ can be achieved by N_1 and N_2 acting individually or cooperatively. 5.2c depicts the composition space of the model with the lower bound of reaction C_{EX} being set to zero. Therefore, only the excretion of C_{EX} is possible, the uptake of C_{EX} is not allowed. Here, the community space gets smaller, the higher the biomass demand is set. At the maximum biomass production $r_{\mu_{max}} = 10$, only a single point at $(1, 0)$ is feasible. Therefore, it can be said that N_1 can function individually at any biomass production rate. In contrast to that, N_2 requires C_EX to be produced by N_1, which necessitates a cooperation between the two members. 5.2d shows the composition space of the model with the lower and upper bounds of reaction C_{EX} being set to zero. Therefore, neither uptake nor excretion of C_{EX} are possible. The results of this calculation are the same as in 5.2c. It can therefore be concluded, that the export of C_{EX} does not have an impact on the community composition.

5.1.2. ECM-Project method

In order to calculate the vertices of the composition space, the mplrs input matrix was created.

	b	f^1	f^2	R_1	R_2	R_4	R_3	EX_S	EX_C	R_5	R_6	R_7	EX_B
S_{EX}	0	0	0	−1	0	0	0	1	0	−1	0	0	0
B_{EX}	0	0	0	0	0	0	1	0	0	0	0	1	−1
C_{EX}	0	0	0	0	0	1	0	0	−1	0	−1	0	0
M_1	0	0	0	1	−1	−1	0	0	0	0	0	0	0
M_2	0	0	0	0	1	0	−1	0	0	0	0	0	0
M_3	0	0	0	0	0	0	0	0	0	1	−1	0	0
M_4	0	0	0	0	0	0	0	0	0	0	2	−1	0
l_{R_1}	0	0	0	1	0	0	0	0	0	0	0	0	0
u_{R_1}	0	−10	0	1	0	0	0	0	0	0	0	0	0
l_{R_2}	0	0	0	0	1	0	0	0	0	0	0	0	0
u_{R_2}	0	−1000	0	0	1	0	0	0	0	0	0	0	0
l_{R_4}	0	0	0	0	0	1	0	0	0	0	0	0	0
u_{R_4}	0	−1000	0	0	0	1	0	0	0	0	0	0	0
l_{R_3}	0	0	0	0	0	0	1	0	0	0	0	0	0
u_{R_3}	0	−1000	0	0	0	0	1	0	0	0	0	0	0
l_{R_5}	0	0	0	0	0	0	0	0	0	1	0	0	0
u_{R_5}	0	0	−10	0	0	0	0	0	0	1	0	0	0
l_{R_6}	0	0	0	0	0	0	0	0	0	0	1	0	0
u_{R_6}	0	0	−1000	0	0	0	0	0	0	0	1	0	0
l_{R_7}	0	0	0	0	0	0	0	0	0	0	0	1	0
u_{R_7}	0	0	−1000	0	0	0	0	0	0	0	0	1	0
l_{EX_S}	1000	0	0	0	0	0	0	1	0	0	0	0	0
u_{EX_S}	−1000	0	0	0	0	0	0	1	0	0	0	0	0
l_{EX_C}	1000	0	0	0	0	0	0	0	1	0	0	0	0
u_{EX_C}	0	0	0	0	0	0	0	0	1	0	0	0	0
l_{EX_B}	r_{μ_c}	0	0	0	0	0	0	0	0	0	0	0	1
u_{EX_B}	$-r_{\mu_c}$	0	0	0	0	0	0	0	0	0	0	0	1
$\sum_i^2 f^k = 1$	−1	1	1	0	0	0	0	0	0	0	0	0	0
$r_\mu^1 = f^1 r_{\mu_c}$	0	$-r_{\mu_c}$	0	0	0	0	1	0	0	0	0	0	0
$r_\mu^2 = f^2 r_{\mu_c}$	0	0	$-r_{\mu_c}$	0	0	0	0	0	0	0	0	1	0
$f^1 \geq 0$	0	1	0	0	0	0	0	0	0	0	0	0	0
$f^2 \geq 0$	0	0	1	0	0	0	0	0	0	0	0	0	0

(5.1)

As this community consists of two individual members, two f-columns (yellow) are present in the matrix. Beginning from the top, the first green block represents the stoichiometric matrix of the exchange reactions S^{ex}. The following two blocks blue blocks are the stoichiometric matrices of the two members S^1 and S^2. After that, the upper and lower bound constraints are set. This is done by putting the upper and lower bounds into the f-column of the respective member. The third and fourth blue blocks represent the identity matrix of the two members I^1 and I^2. The final green block represents the identity matrix of the external reactions I^{ex}, the upper and lower bounds of these reactions are put into the b-column (red). The final rows represent additional constraints that the community model must adhere to.

Table 5.1.: Vertices of community composition spaces of two-member toy model with the reaction EX_C being unconstrained. Depicted in 5.2b.

$r_{\mu_c}\,[h^{-1}]$	f^1	f^2
0	0	1
	1	0
2	0	1
	1	0
4	0	1
	1	0
6	0	1
	1	0
8	0	1
	1	0
10	0	1
	1	0

Table 5.2.: Vertices of community composition spaces of two-member toy model with the reaction EX_C being constrained. Depicted in 5.2d and 5.2c.

$r_{\mu_c}\,[h^{-1}]$	f^1	f^2
0	0	1
	1	0
2	0.11	0.89
	1	0
4	0.25	0.75
	1	0
6	0.43	0.57
	1	0
8	0.67	0.33
	1	0
10	1	0

Following the construction of the input matrix, the vertices of the composition space could be calculated for each r_{μ_c}, shown in Tables 5.1 and 5.2. With the vertices at hand, the composition space can be obtained by creating the convex hull. The results were identical with the results computed via the FVA method and are depicted in Figure 5.2.

5.2. Three-organism toy model

To test the devised methods for larger communities, a community model with three members was tested. This model, which was created by Koch et al.[36] contains the constraints depicted in Table 5.3. Three distinct cases were studied with this toy model to illustrate the effect of a changing environment on the community composition:

1. The exchange of metabolite M2 is allowed, therefore M2 can be transported into and out of the interspecies exchange compartment.

N_1	N_2	N_3
$r_{R_1} \leq 40\,\mathrm{mmol\,g^{-1}\,h^{-1}}$	$r_{R_1} \leq 40\,\mathrm{mmol\,g^{-1}\,h^{-1}}$	-
$r_{R_2} \leq 30\,\mathrm{mmol\,g^{-1}\,h^{-1}}$	$r_{R_2} \leq 30\,\mathrm{mmol\,g^{-1}\,h^{-1}}$	-
$r_{R_3} \leq 50\,\mathrm{mmol\,g^{-1}\,h^{-1}}$	-	$r_{R_3} \leq 50\,\mathrm{mmol\,g^{-1}\,h^{-1}}$
$r_{ATP_{maint}} \geq 5\,\mathrm{mmol\,g^{-1}\,h^{-1}}$	$r_{ATP_{maint}} \geq 2\,\mathrm{mmol\,g^{-1}\,h^{-1}}$	$r_{ATP_{maint}} \geq 3\,\mathrm{mmol\,g^{-1}\,h^{-1}}$

Table 5.3.: Constraints of three-membered toy model. Adapted from Koch et al.[36].

2. No exchange of metabolite M2 is allowed.

3. No exchange of metabolites M2 and M5 is allowed.

The exchange of metabolite M3 is not allowed in all of the three cases above.

5.2.1. Flux variability analysis Method

Inspecting the community structure of the three-membered model 5.3a, one might notice that the introduction of an exchange reaction for M2 would make the member N3 independent from the other organisms. This is due to the fact that N3 can take either M1 or M5 as substrate, which are both taken from the external compartment. Therefore, it is independent of the production rate of N1 and N2. However, it produces M2 as a byproduct, which as stated in the steady-state constraint, must be consumed in the same quantity as it is produced. Without an exchange reaction for M2, N3 would have to rely on N1 or N2 to take up M2 and convert it into a different metabolite. Introducing an exchange reaction circumvents that dependency of N3 on the other members of the community, as with the exchange reaction, M2 can be transported out of the system, thereby adhering to the steady-state constraint. For this scenario, the composition spaces for community biomass production rates (r_{μ_c}) ranging from $0\,\mathrm{h^{-1}}$ to $7\,\mathrm{h^{-1}}$ were computed and are depicted in Figure 5.3b. Up to a biomass production of $r_{\mu_c} = 3\,\mathrm{h^{-1}}$, community compositions containing only one member are possible. For example, the point (1,0,0) represents a community composition of 100% N1, 0% N2 and 0% N3. As the points (0,1,0) and (0,0,1) are also included in the composition space, it can be said that all members of the community can work independently from one another up to $r_{\mu_c} = 3\,\mathrm{h^{-1}}$. N1 only requires M1 and M2 as substrates, which are both transported into the shared compartment from the medium, as they both possess exchange reactions. However, N1 produces M3 and M4 as byproducts. As there is no exchange reaction for M3 in this scenario, M3 has to be consumed by organism N2. Therefore it can be concluded that up to $r_{\mu_c} = 3\,\mathrm{h^{-1}}$, N1 can work independently without producing any M3 as a byproduct. At higher biomass production rates, the point (1,0,0) is not present in the community composition space anymore, meaning that N1 cannot work independently anymore. In fact, N1 and N2 have to start working together in order to produce the demanded biomass. They can grow without involving N3 up to $r_{\mu_c} = 5\,\mathrm{h^{-1}}$. N3 on the other hand, does not need to cooperate with the other members at all, it can feasibly produce

biomass independently up to a rate of $r_{\mu_c} = 7\,\mathrm{h}^{-1}$, where the point (0,0,1) remains the only feasible community composition.

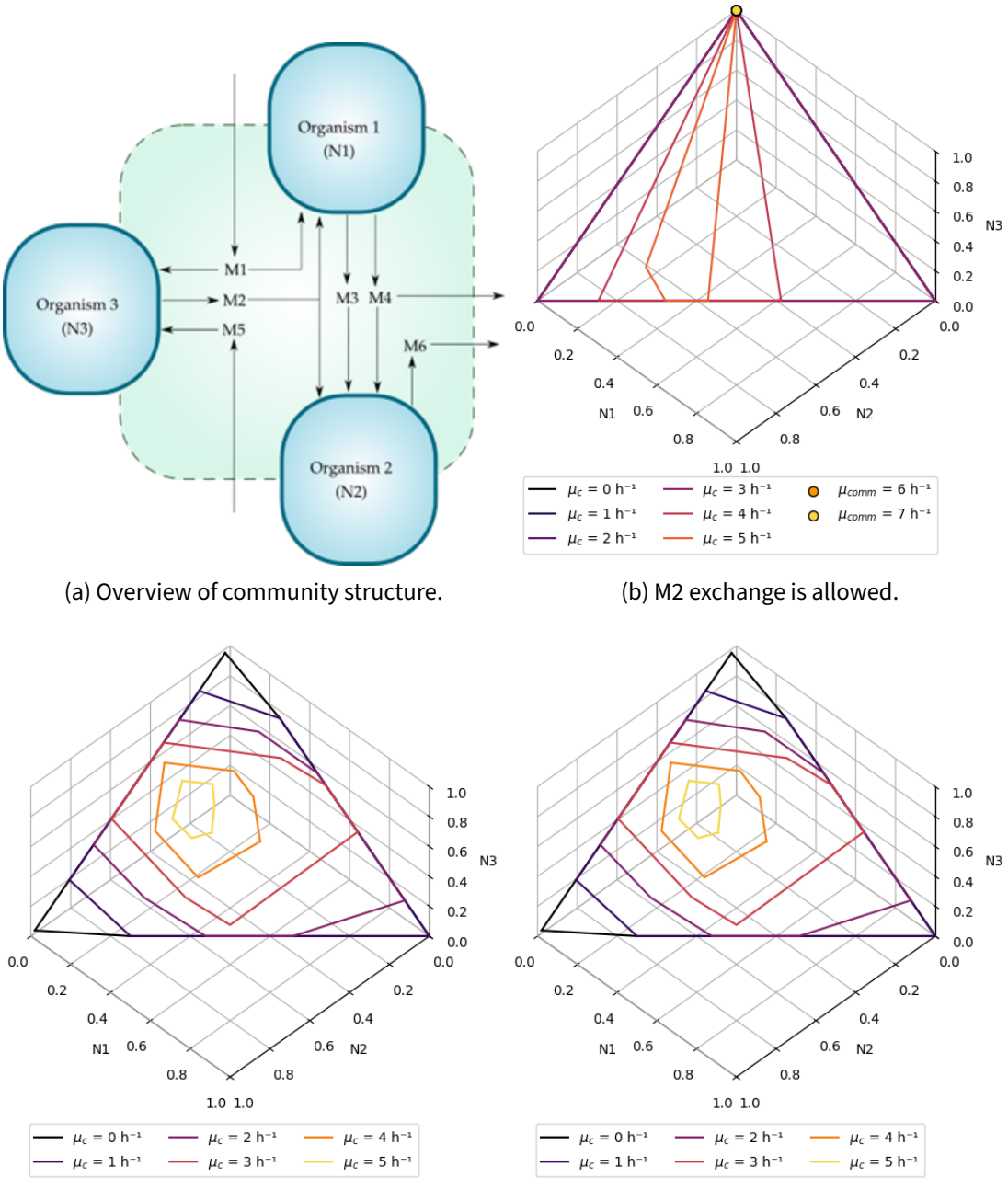
When the exchange of M2 becomes prohibited, the composition spaces of the community change drastically. This scenario is depicted in Figure 5.3c. It represents the structure of the community shown in 5.3a and therefore corresponds to the structure proposed by Koch et al.[36]. Here the only instance in which independent function is possible is at $r_{\mu_c} = 0\,\mathrm{h}^{-1}$, represented by the point (1,0,0). Here, N1 can feasibly work at a relative abundance of 1.0, as it takes M1 as a substrate and only produces M4 as a byproduct. As the production of biomass necessitates a higher internal production of ATP, N1 has to follow a pathway that produces M3 at higher r_{μ_c}. Consequently, N2 needs to take up the produced M3 and metabolise it, producing M6. N1 and N2 can feasible work without interacting with N3 up to $r_{\mu_c} = 2\,\mathrm{h}^{-1}$. These compositions contain every point in the composition space in which the value of the z-axis (N3) remains 0. At higher rates, the uptake constraints in table 5.3 prevent the organisms of metabolising enough substrates to meet the biomass demand. Therefore, they require N3 to produce biomass as well as M2, which can be employed as an additional substrate by N1 and N2. The feasible community composition space shrinks with the rising demand for biomass production.

Finally, 5.3d shows the community composition with the added constraint that no M5 exchange is allowed. This measure does not impact the composition space at all. Therefore it can be concluded that N3 does not require M5 to yield optimal biomass rates.

5.2.2. ECM-Project method

Constructing the mplrs input matrix using ECM-Project, the vertices of the composition spaces could also be computed. The resulting composition spaces are shown in Figure 5.4, the vertices of the composition spaces are shown in Tables 5.4, 5.5 and 5.6. Slight differences in the composition spaces can be detected if we compare the plots 5.4b, 5.4c, and 5.4d to 5.3b, 5.3c and 5.3d, respectively. This is due to the fact that the composition spaces computed by ECM-Project are exact, while the FVA-method only yields approximations. As described in section 4.7, the more iterations of the flux variability analysis are performed, the more accurate the resulting flux space becomes. Of course, a higher number of iterations also results in a larger computation time. In Figure 5.3, 5 iterations were conducted.

Figure 5.5 shows a comparison of the computed flux spaces using different amounts of steps in the FVA-method. It is evident that 2 steps is not enough to gain an accurate impression of the community composition spaces. In contrast to that, the composition space computed with 20 steps already shows a fairly similar shape to the exact one from the ECM-Project. Table 5.7 summarises the computation times for the community composition spaces shown in Figure 5.5.



(a) Overview of community structure.

(b) M2 exchange is allowed.

(c) M2 exchange is not allowed.

(d) M2 and M5 exchange is not allowed.

Figure 5.3.: Community composition space computed by the FVA method. The axes in the plots represent the relative abundance of the corresponding member in the community. 5.3a: Overview of the three-member community. 5.3b: Composition space of community in which an additional exchange reaction for M2 is introduced. The exchange of M3 is not allowed. 5.3c: Composition space of community in which the exchange of M2 and M3 is not allowed. 5.3d: Composition space of community in which the exchange of M2, M3 and M5 is not allowed.

Table 5.4.: Vertices of community composition spaces of three-member toy model in which an additional exchange reaction for M2 is introduced. The exchange of M3 is not allowed. Depicted in 5.4b.

$r_{\mu_c}\,[h^{-1}]$	f^1	f^2	f^3
0	0	0	1
	0	1	0
	1	0	0
1	0	0	1
	0	1	0
	1	0	0
2	0	0	1
	0	1	0
	1	0	0
3	0	0	1
	0	1	0
	1	0	0
4	0	0	1
	0.61	0.39	0
	0.09	0.91	0
5	0	0	1
	0.43	0.57	0
	0.24	0.76	0
6	0	0	1
7	0	0	1

Table 5.5.: Vertices of community composition spaces of three-member toy model in which the exchange of M2 and M3 is not allowed. Depicted in 5.4c.

$r_{\mu_c}\,[h^{-1}]$	f^1	f^2	f^3
0	0.03	0	0.97
	1	0	0
	0	0.02	0.98
	0.02	0.98	0
	0	0.98	0.02
1	0	0.15	0.85
	0	0.81	0.19
	0.20	0	0.80
	0.24	0.76	0
	1	0	0
2	0.38	0.62	0
	0.81	0.19	0
	0.88	0	0.12
	0	0.68	0.32
	0.31	0	0.69
	0	0.25	0.75

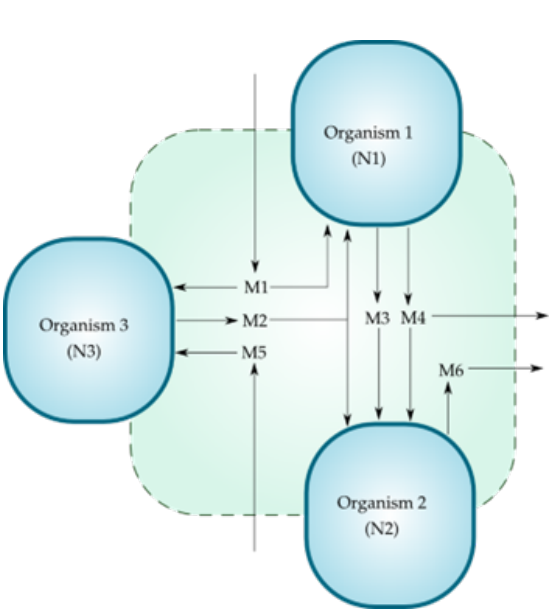
$r_{\mu_c}\,[h^{-1}]$	f^1	f^2	f^3
3	0.50	0.46	0.04
	0.43	0.53	0.04
	0.64	0	0.36
	0	0.60	0.40
	0.40	0	0.60
	0	0.33	0.67
4	0.04	0.37	0.60
	0.05	0.52	0.43
	0.27	0.17	0.56
	0.34	0.48	0.19
	0.41	0.26	0.33
5	0.11	0.35	0.53
	0.14	0.44	0.41
	0.20	0.27	0.52
	0.25	0.42	0.32
	0.28	0.37	0.36

Table 5.6.: Vertices of community composition spaces of three-member toy model in which the exchange of M2, M3 and M5 is not allowed. Depicted in 5.4d.

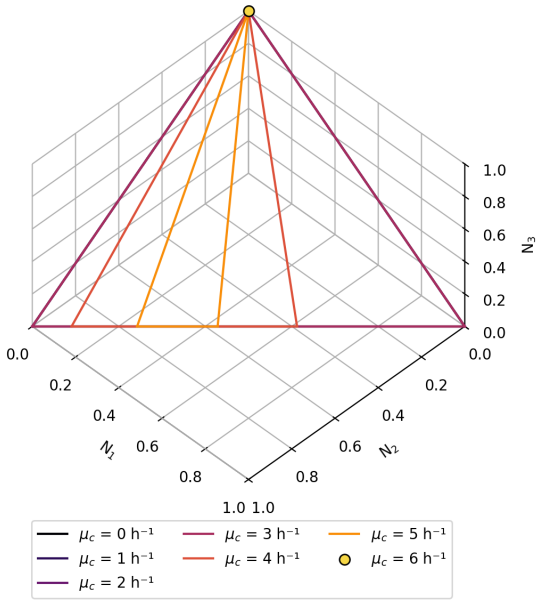
$r_{\mu_c}\,[h^{-1}]$	f^1	f^2	f^3
0	0.03	0	0.97
	1	0	0
	0	0.02	0.98
	0.02	0.98	0
	0	0.98	0.02
1	0	0.15	0.85
	0	0.81	0.19
	0.20	0	0.80
	0.24	0.76	0
	1	0	0
2	0.38	0.62	0
	0.81	0.19	0
	0.88	0	0.12
	0	0.68	0.32
	0.31	0	0.69
	0	0.25	0.75
3	0.50	0.46	0.04
	0.43	0.53	0.04
	0.64	0	0.36
	0	0.60	0.40
	0.40	0	0.60
	0	0.33	0.67
4	0.04	0.37	0.60
	0.05	0.52	0.43
	0.27	0.17	0.56
	0.34	0.48	0.19
	0.41	0.26	0.33
5	0.11	0.35	0.53
	0.14	0.44	0.41
	0.20	0.27	0.52
	0.25	0.42	0.32
	0.28	0.37	0.36

Table 5.7.: Comparison of computation times for composition spaces shown in Figure 5.5. The computation time rises with an increased number of iteration steps.

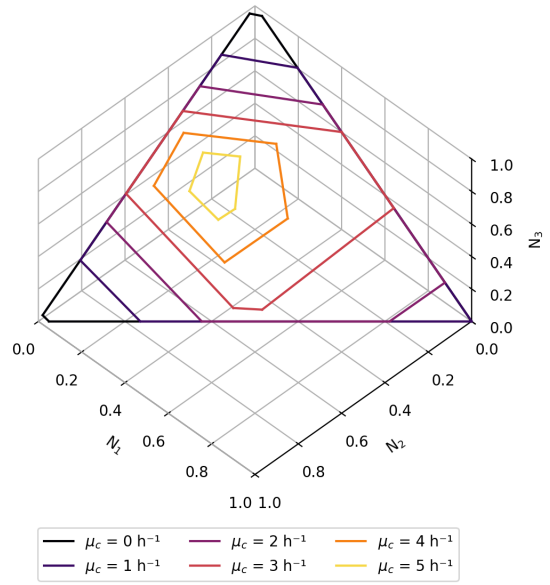
Method	Computation time $[s]$
Flux variability analysis (2 steps)	11.0
Flux variability analysis (5 steps)	23.3
Flux variability analysis (10 steps)	43.6
Flux variability analysis (20 steps)	83.2
ECM-Project	65.7



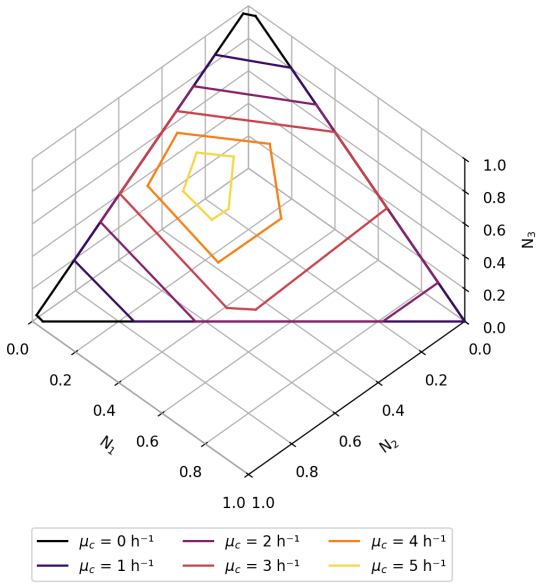
(a) Overview of community structure.



(b) M2 exchange is allowed.



(c) M2 exchange is not allowed.



(d) M2 and M5 exchange is not allowed.

Figure 5.4.: Community composition space computed by the ECM-Project method. The axes in the plots represent the relative abundance of the corresponding member in the community. 5.4a: Overview of the three-member community. 5.4b: Composition space of community in which an additional exchange reaction for M2 is introduced. The exchange of M3 is not allowed. 5.4c: Composition space of community in which the exchange of M2 and M3 is not allowed. 5.4d: Composition space of community in which the exchange of M2, M3 and M5 is not allowed.

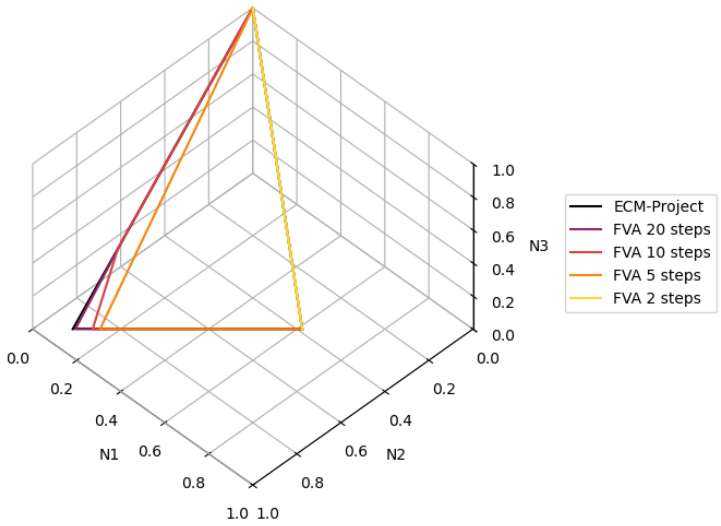


Figure 5.5.: Comparison of plots using ECM-Project or FVA with different stepsizes. The composition spaces describe the three-membered toy model in which the exchange of M2 is allowed. The growth rate was set to $r_{\mu_c} = 4\,\mathrm{h}^{-1}$. The step size is correlated to the accuracy of the resulting composition space.

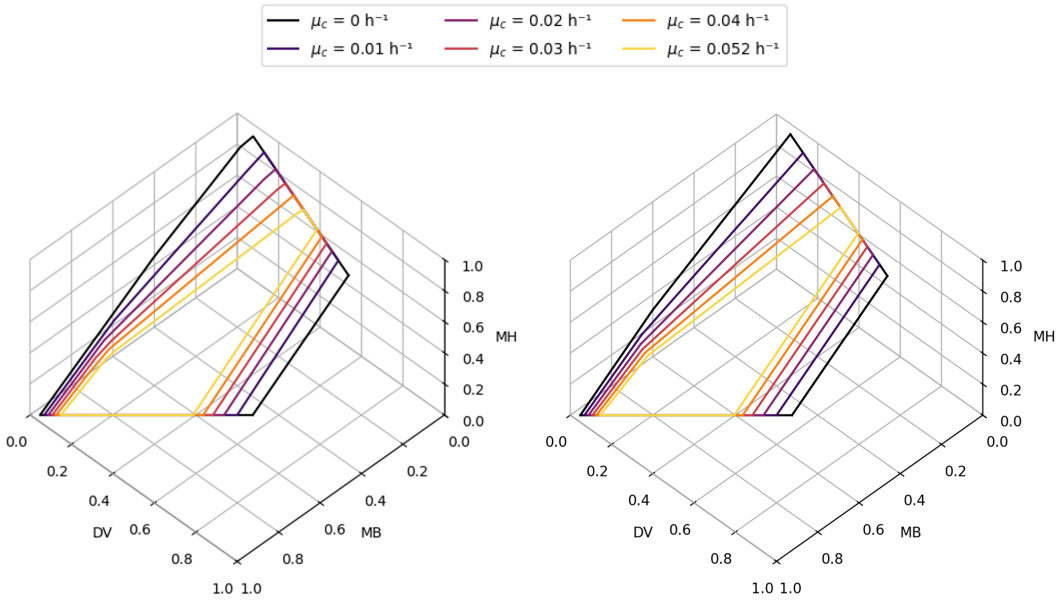
5.3. Three-organism genome-scale model

The two-organism genome-scale model encompassing the metabolic models of *Desulfovibrio vulgaris* (DV), *Methanococcus maripaludis* (MM), and *Methanosarcina barkeri* (MB) contains 355 reactions. This higher amount of reactions results in a larger computational effort in the ECM-Project method, as the computation time required by the mplrs algorithm scales with the number of columns in the matrix. The total time consumed for calculating a composition space of this community model using the ECM-Project method were 2206.8 seconds. In contrast to that, using the FVA-method with 20 calculation steps only took 84.1 seconds. It can therefore be concluded that the FVA-method is better suited for larger models than the ECM-Project, if absolute accuracy of the composition space is not required. A comparison of the computation time of the different models is shown in table 5.8.

The resulting composition spaces of the two-organism genome-scale model are depicted in Figure 5.6. The FVA-method computation was run with 40 calculation steps. Still, minor differences between the composition spaces calculated by the two methods exist.

Table 5.8.: Comparison of computation time of composition space for different communities. Flux variability analysis was conducted with 20 iteration steps.

Model	Method	Computation time $[s]$
two-organism toy model	ECM-Project	19.0
	Flux variablity analysis	12.8
three-organism toy model	ECM-Project	65.7
	Flux variablity analysis	83.2
three-organism genome-scale model	ECM-Project	2206.8
	Flux variablity analysis	84.1



(a) FVA-Method

(b) ECM-Project method

Figure 5.6.: Computed composition spaces for a three-membered genome-scale model consisting of *Desulfovibrio vulgaris* (DV), *Methanococcus maripaludis* (MM), and *Methanosarcina barkeri* (MB). 5.6a: Composition spaces calculated by the FVA-method. 5.6b: Composition spaces calculated by the ECM-Project method.

6. Conclusion and Outlook

Within this project, the python package PyCoMo [37] for the construction of metabolic models of communities was created. For this, SBML files of single organism metabolic models can be selected from one of multiple databases. PyCoMo conducts quality control on these models, and matches duplicate metabolites between the different models. Following that, a community metabolic model is created, which can exist in two different modes. The reactions within the model can either be constrained as in the single models, or an equivalent, bound-free model can be created by the addition of artificial reactions and metabolites.

The bound-free model contains one abundance reaction for each member in the community. The flux through this reaction corresponds to the relative abundance of the member in the community, and can therefore exist in a range between 0 and 1. By conducting flux variability analysis on the community model, the minima and maxima of the flux through the abundance reactions can be computed. This in turn enables the creation of a community composition space, which characterises each feasible composition of the community at a given growth rate.

The alternative model without the additional artificial reactions can be used with the ECM-Project. By creating a flux cone of the metabolic model and projecting this cone onto the abundances of the members in the community via mplrs, the vertices of the community composition space can be computed. This method offers exact results, and is suitable for small models with a couple hundred reactions. Larger networks with over 2000 reactions have been tested, but take way too much time. This is due to the fact that the computation time scales exponentially with the reaction number.

In contrast to that, the computation of the community composition Fspace via flux variability analysis of the equivalent bound-free model is also suitable for genome-scale models with a large amount of reactions.

To improve this project in the future, the focus should be on changing the projection algorithm used in the ECM-Project method. The step that takes the most time in larger models is projecting the flux space and composition space. If the projection algorithm would be modified, the computational workload could be reduced significantly. Instead of using the Fourier-Motzkin elimination method in mplrs, one could explore the ProCEMs (Projected Cone Elementary Modes) concept proposed by Marashi et al. [48].

Bibliography

(1) Wagner, M.; Loy, A.; Nogueira, R.; Purkhold, U.; Lee, N.; Daims, H. *Antonie Van Leeuwenhoek* **2002**, *81*, 665–680.

(2) Pitt, W. W.; Hancher, C. W.; Patton, B. D. *Nuclear and Chemical Waste Management* **1981**, *2*, 57–70.

(3) Daims, H.; Nielsen, J. L.; Nielsen, P. H.; Schleifer, K.-H.; Wagner, M. *Applied and Environmental Microbiology* **2001**, *67*, Publisher: American Society for Microbiology, 5273–5284.

(4) Franche, C.; Lindström, K.; Elmerich, C. *Plant Soil* **2009**, *321*, 35–59.

(5) Weiland, P. *Appl Microbiol Biotechnol* **2010**, *85*, 849–860.

(6) Abate, C.; Callieri, D.; Rodríguez, E.; Garro, O. *Appl Microbiol Biotechnol* **1996**, *45*, 580–583.

(7) Mamma, D.; Koullas, D.; Fountoukidis, G.; Kekos, D.; Macris, B. J.; Koukios, E. *Process Biochemistry* **1996**, *31*, 377–381.

(8) Olanrewaju, O. S.; Glick, B. R.; Babalola, O. O. *World J Microbiol Biotechnol* **2017**, *33*, 197.

(9) Biokraftstoffe im Überblick https://www.bmk.gv.at/themen/energie/energieversorgung/biomasse/alternative-kraftstoffe/ueberblick.html (accessed 10/11/2023).

(10) Bagi, Z.; Ács, N.; Bálint, B.; Horváth, L.; Dobó, K.; Perei, K. R.; Rákhely, G.; Kovács, K. L. *Appl Microbiol Biotechnol* **2007**, *76*, 473–482.

(11) Eng, A.; Borenstein, E. *Curr Opin Biotechnol* **2019**, *58*, 117–128.

(12) McFall-Ngai, M. *Nature* **2007**, *445*, Number: 7124 Publisher: Nature Publishing Group, 153–153.

(13) Hooper, L. V.; Littman, D. R.; Macpherson, A. J. *Science* **2012**, *336*, Publisher: American Association for the Advancement of Science, 1268–1273.

(14) Ley, R. E.; Lozupone, C. A.; Hamady, M.; Knight, R.; Gordon, J. I. *Nat Rev Microbiol* **2008**, *6*, 776–788.

(15) Suzuki, K.; Meek, B.; Doi, Y.; Muramatsu, M.; Chiba, T.; Honjo, T.; Fagarasan, S. *Proc Natl Acad Sci U S A* **2004**, *101*, 1981–1986.

(16) Klamt, S.; Hädicke, O.; Kamp, A. v. In *Large-Scale Networks in Engineering and Life Sciences*, Benner, P., Findeisen, R., Flockerzi, D., Reichl, U., Sundmacher, K., Eds.; Modeling and Simulation in Science, Engineering and Technology, 00001; Springer International Publishing: 2014, pp 263–316.

(17) Frioux, C.; Singh, D.; Korcsmaros, T.; Hildebrand, F. *Computational and Structural Biotechnology Journal* **2020**, *18*, Publisher: Elsevier, 1722–1734.

(18) Schellenberger, J.; Park, J. O.; Conrad, T. M.; Palsson, B. Ø. *BMC Bioinformatics* **2010**, *11*, 213.

(19) Barthelmes, J.; Ebeling, C.; Chang, A.; Schomburg, I.; Schomburg, D. *Nucleic Acids Research* **2007**, *35*, D511–D514.

(20) Kanehisa, M.; Goto, S. *Nucleic Acids Research* **2000**, *28*, 27–30.

(21) Machado, D.; Andrejev, S.; Tramontano, M.; Patil, K. R. *Nucleic Acids Research* **2018**, *46*, 7542–7553.

(22) Arkin, A. P. et al. *Nat Biotechnol* **2018**, *36*, Number: 7 Publisher: Nature Publishing Group, 566–569.

(23) DeJongh, M.; Formsma, K.; Boillot, P.; Gould, J.; Rycenga, M.; Best, A. *BMC Bioinformatics* **2007**, *8*, 139.

(24) Prigent, S.; Frioux, C.; Dittami, S. M.; Thiele, S.; Larhlimi, A.; Collet, G.; Gutknecht, F.; Got, J.; Eveillard, D.; Bourdon, J.; Plewniak, F.; Tonon, T.; Siegel, A. *PLOS Computational Biology* **2017**, *13*, Publisher: Public Library of Science, e1005276.

(25) Thiele, I.; Vlassis, N.; Fleming, R. M. T. *Bioinformatics* **2014**, *30*, 2529–2531.

(26) Levy, R.; Carr, R.; Kreimer, A.; Freilich, S.; Borenstein, E. *BMC Bioinformatics* **2015**, *16*, 164.

(27) Ebrahim, A.; Lerman, J. A.; Palsson, B. O.; Hyduke, D. R. *BMC Systems Biology* **2013**, *7*, 74.

(28) O'Brien, E. J.; Monk, J. M.; Palsson, B. O. *Cell* **2015**, *161*, Publisher: Elsevier, 971–987.

(29) Orth, J. D.; Thiele, I.; Palsson, B. Ø. *Nat Biotechnol* **2010**, *28*, Number: 3 Publisher: Nature Publishing Group, 245–248.

(30) Datta, B. N., *Numerical linear algebra and applications*, 2nd ed, OCLC: ocn417442992; Society for Industrial and Applied Mathematics: Philadelphia, 2010; 530 pp.

(31) Klamt, S.; Regensburger, G.; Gerstl, M. P.; Jungreuthmayer, C.; Schuster, S.; Mahadevan, R.; Zanghellini, J.; Müller, S. *PLOS Computational Biology* **2017**, *13*, Publisher: Public Library of Science, e1005409.

(32) Mahadevan, R.; Schilling, C. H. *Metab Eng* **2003**, *5*, 264–276.

(33) Heirendt, L. et al. *Nat Protoc* **2019**, *14*, 639–702.

(34) Heirendt, L.; Thiele, I.; Fleming, R. M. T. *Bioinformatics* **2017**, *33*, 1421–1423.

(35) Jamshidi, N.; Palsson, B. Ø. *Biophys J* **2010**, *98*, 175–185.

(36) Koch, S.; Kohrs, F.; Lahmann, P.; Bissinger, T.; Wendschuh, S.; Benndorf, D.; Reichl, U.; Klamt, S. *PLOS Computational Biology* **2019**, *15*, Publisher: Public Library of Science, e1006759.

(37) Predl, M.; Mießkes, M.; Rattei, T.; Zanghellini, J. *Submitted to Bioinformatics, submission number: BIOINF-2023-1888* **2023**.

(38) Koch, S.; Benndorf, D.; Fronk, K.; Reichl, U.; Klamt, S. *Biotechnology for Biofuels* **2016**, *9*, 17.

(39) Grünbaum, B., *Convex Polytopes*; Kaibel, V., Klee, V., Ziegler, G. M., Eds.; Graduate Texts in Mathematics, Vol. 221; Springer New York: New York, NY, 2003.

(40) Avis, D. User's Guide for lrs, User's Guide for lrs - Version 3.2 a http://cgm.cs.mcgill.ca/~avis/C/old/USERGUIDE32a.html (accessed 08/28/2023).

(41) Lauritzen, N. In *Undergraduate Convexity*; WORLD SCIENTIFIC: 2013, pp 1–15.

(42) Chakraborty, M.; Chowdhury, S.; Chakraborty, J.; Mehera, R.; Pal, R. K. *Complex Intell. Syst.* **2019**, *5*, 265–281.

(43) Avis, D.; Fukuda, K. *Discrete Comput Geom* **1992**, *8*, 295–313.

(44) Bland, R. G. *Mathematics of Operations Research* **1977**, *2*, Publisher: INFORMS, 103–107.

(45) Avis, D.; Jordan, C. mplrs: A scalable parallel vertex/facet enumeration code, 2017.

(46) Walker, D.; Dongarra, J. *Supercomputer* **1995**, *12*.

(47) Kenefake, D.; Armingol, E.; Lewis, N. E.; Pistikopoulos, E. N. *BMC Bioinformatics* **2022**, *23*, 550.

(48) Marashi, S.-A.; David, L.; Bockmayr, A. *Algorithms for Molecular Biology* **2012**, *7*, 17.

A. Appendix

A.1. Abstract

Microbial communities are groups of individual microorganisms that can interact in various ways. For instance, they can compete for limited resources, or cooperate through symbiosis. These interactions shape the structure as well as the function of a community. Microbial communities exist all around us and play important roles in ecology, agriculture, and human health. They are also employed in a variety of industrial applications, such as wastewater treatment and the production of biofuels.

The behaviour of a microbial community is significantly influenced by the composition and abundance of its microbial members. Due to that, efforts have been made to develop techniques to characterise the makeup of such communities. One such method, found in the field of systems biology, involves the integration of metabolic models of individual microorganisms into a larger community model. Employing constraint-based modeling, the characteristics of these models can be analysed and hypotheses for their behaviour in vivo can be formulated.

The aim of this thesis is to establish a methodology for the construction of community metabolic models from individual metabolic models. Furthermore, a computational method should be created which is able to determine the composition space of a metabolic community model under specified constraints. The composition space itself describes the area that encompasses every feasible composition of the members within the community.

A.2. Zusammenfassung

Mikrobielle Gemeinschaften sind Gruppen einzelner Mikroorganismen, die auf vielfältige Weise interagieren können. Zum Beispiel können sie um begrenzte Ressourcen konkurrieren oder durch Symbiose zusammenarbeiten. Diese Interaktionen prägen die Struktur und die Funktion der mikrobiellen Gemeinschaft. Mikrobielle Gemeinschaften existieren überall um uns herum und spielen wichtige Rollen in der Ökologie, Landwirtschaft und menschlichen Gesundheit. Sie finden auch in verschiedenen industriellen Anwendungen Verwendung, wie etwa in der Abwasserbehandlung und der Herstellung von Biokraftstoffen.

Das Verhalten einer mikrobiellen Gemeinschaft wird maßgeblich von der Zusammensetzung und der Anzahl ihrer mikrobiellen Mitglieder beeinflusst. Aus diesem Grund wurden Bemühungen unternommen, Techniken zur Charakterisierung der Zusammensetzung solcher Gemeinschaften zu entwickeln. Eine solche Methode, die im Bereich der Systembiologie zu finden ist, umfasst die Integration von Stoffwechselmodellen einzelner Mikroorganismen in ein umfassenderes Gemeinschaftsmodell. Mithilfe von Restriktions-basierter Modellierung

können die Eigenschaften dieser Modelle analysiert und Hypothesen zu ihrem Verhalten in vivo formuliert werden.

Das Ziel dieser Arbeit besteht darin, eine universelle Methodik zur Erstellung von stoffwechselbasierten Gemeinschaftsmodellen aus individuellen stoffwechselbasierten Modellen zu etablieren. Darüber hinaus soll eine computergestützte Methode entwickelt werden, die in der Lage ist, den Zusammensetzungsraum eines stoffwechselbasierten Gemeinschaftsmodells unter festgelegten Beschränkungen zu bestimmen. Der Zusammensetzungsraum selbst beschreibt den Bereich, der jede mögliche Zusammensetzung der Mitglieder innerhalb der Gemeinschaft umfasst.