universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„Benchmarking software tools for the calculation of convex hulls for metabolic modeling applications“

verfasst von / submitted by

Martin Völkl BSc

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

Master of Science (MSc)

Wien, 2022 / Vienna 2022

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066 862
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

Acknowledgment

I would like to thank Univ.-Prof. Dipl.-Ing. Dr. Jürgen Zanghellini and the whole working group for providing me with not only the opportunity but also the knowledge and expertise to finish this project as well as an exceedingly friendly and entertaining working environment. I would also like to thank all other staff and colleagues at the University of Vienna and the Technical University of Vienna to get me to the point of writing this master's thesis. My thanks also go out to all members of my family as well as my friends for igniting and further advancing my interest in the natural sciences in addition to all their emotional support. Lastly, I would like to thank scihub for making all research facilitated and funded by the people also available to the people.

Contents

1 Introduction to systems biology

Systems biology is a relatively new and highly interdisciplinary field combining aspects of biology, computational science, and mathematics. The need for such a holistic approach arises both from the complexity of the biological systems of interest as well as the increasingly large amounts of data generated by modern analysis tools.[1]

A prime example of both would be the processing, analysis, and interpretation of whole genome sequences or the results from mass spectrometry-based omics techniques which rely heavily on advanced computational methods. This new wealth of information is also the basis for the design of mathematical models which can be used to describe the complex processes in biological systems. Such models can then be used to explain observed behaviors or to propose new theories to be tested in the lab.[2]

Among the complex networks facilitating the cellular functions of an organism, the three main ones are:[1]

- Metabolic networks encompass the intake of nutrients and their further conversion into usable energy and building blocks for new cell growth as well as a vast array of intermediates, metabolites, and byproducts. They consist of hundreds of reactions chained together into pathways capable of synthesizing all those substances.
- Signal transduction networks monitor the state of the cell and its surroundings and, based on this information, they change the flow of metabolites, also called flux, through the metabolic network.
- Gene regulatory networks govern the expression of genes based on DNA, RNA, and protein regulators needed for the production of the enzymes needed by the metabolic network.

While some aspects of signal transduction and gene regulatory networks are relevant for the further improvement of many modeling applications, this text is mostly concerned with the metabolic networks of microbes and the tools used to model and analyze them. These computational models can then be used to calculate the intake of nutrients, production of metabolites, and growth of the organism based on the structure of the metabolic network and experimental parameters.

2 Metabolic network models

2.1 Basic metabolic functions

Compared to eukaryotes like animals and plants, many microbes have an exceedingly flexible diet[3] allowing them to keep up many metabolic functions like growth, energy production, or the synthesis of secondary metabolites with a very small suite of substrates,[4] the most important of which are at least one of a variety of possible carbon sources like sugar or alcohol, a nitrogen source like ammonia or amino acids, sources of phosphor and sulphur as well as oxygen in the case of aerobic growth. Microbes can use a variety of different substances to fulfill these nutritional demands. One main example of this flexibility is a microbes' ability to switch between aerobic and anaerobic respiration as a source of energy. The aerobic pathway, utilizing oxygen as an electron acceptor, provides more energy but the anaerobic pathway allows the organism, often through the fermentation of sugars, to thrive even in low oxygen environments.

Another important point of flexibility is the metabolism's ability to use a wide array of carbon sources like glucose, acetate, or ethanol as a growth substrate. This not only makes the organism more resilient to environmental changes but also allows for more optimal biomass production. For example, under optimal conditions, glucose is the preferred growth substrate for E. coli while other carbon sources result in lower growth rates. When the organism has to rely on suboptimal amino acids as a nitrogen source, other carbon sources can achieve higher growth rates than glucose. This means that the organism has the flexibility to switch between different carbon sources to maximize its growth rate in different environmental circumstances.[5]

2.2 Reaction types

The uptake of nutrients and exudation of some metabolites is integrated into the model using exchange reactions.[1] While some of these exchange reactions result in an alteration of the metabolite or nutrient, many exchange reactions do not facilitate any chemical transformation but instead just move the nutrient between compartments. This means that many exchange reactions only represent how much of the nutrient or metabolite is taken up from- or exuded into the environment by the organism. Exchange reactions also allow for the simulation of different environmental conditions by setting upper and lower flux limits, representing how fast the substance exchange with the environment can take place in either direction. An upper bound that the system is not allowed to exceed can simulate nutrient-rich or poor environments while a lower bound can be used to represent a constant demand like ATP consumption for cellular maintenance functions.

The ATP demand reaction is an example of an exchange reaction that technically does not take place in nature in that form but is still needed for modeling. In contrast to other metabolites and byproducts which are exuded into the environment, ATP

is produced for internal consumption by many different processes, but for simulation purposes, it is summarized into just one exchange reaction, which in this case is also often called a demand reaction. This reaction is necessary to simulate the ATP drain required for the maintenance of the system. The same is true for the production of many other substances needed for growth which are combined into one biomass reaction. This biomass or growth reaction takes up all metabolites which make up the cells' building blocks, mostly DNA, RNA, lipids, and enzymes, as educts and produces biomass which is then 'consumed' by a demand reaction similarly to ATP.[1]

While the composition of the biomass function can be more or less approximated with in vitro experiments, it can exhibit some variability in composition mostly depending on the rate of growth and availability of nutrients.[6,7] To account for this, many models include a few different biomass functions to choose from according to the growth conditions.

The growth function is one of the major goals of optimization in nature but for many industrial applications product synthesis is the main goal. Still, it makes sense to enforce at least a minimal growth rate which is needed to replace dying cells in the colony so as to at least keep the population constant.

While those exchange, demand, and biomass reactions are essential for the fulfillment of the cells' functions, they only represent a minority of all the reactions in the metabolic system. The rest are reactions that produce internal metabolites, convert between internal metabolites, and synthesize building blocks and various other more complex molecules.

2.3 Reaction categories

While in the previous section, different reactions were categorized by their metabolic function, another important way to categorize them is by their flux properties, meaning how much substance, if at all, is flowing through the reaction in either direction.

Blocked reactions result from internal metabolites which could be produced but are not used up in any further reaction. Following from this, all reactions leading up to such a dead-end metabolite, which are not part of any other pathway, are also not allowed to carry any flux. Similarly, if a metabolite can only be consumed but there is no way to produce it, then this also constitutes a blocked reaction.[1]

While blocked reactions do not contribute to feasible flux solutions they can be useful to check a model for errors since dead-end metabolites can be the result of a mistake when building the model. In that case, either missing reactions should be added to the model or the blocked reaction should be removed.

Another common occurrence in metabolic networks are *coupled reactions*, also known as enzyme subsets or correlated reaction sets, where the flux values of two or more reactions always correlate with each other at a specific ratio.[8,9] The simplest form of coupled reactions arises from linear chains of reactions without branches. When any of the reactions in the chain has zero flux, so do all the others so as not to lead to a build-up. Notably, the flux values do not have to be of the same value, but they do have to always assume the correct ratio. For example when the reaction $A \rightarrow B$ is followed by $2B \rightarrow C$, the flux ratio of both reactions always has to be 2:1 so that all produced substances are also consumed at the same rate. While this is the most straightforward case, there exist many other topologies which also lead to coupled reactions.

Coupled reactions are very useful for bioengineering tasks. For example, a common strategy to force an organism to produce specific products it normally would not, is

to find a pathway where the production of the desired metabolite is coupled to the growth function and then disable all other pathways for growth by knocking out genes encoding essential enzymes of each pathway. The collection of all genes which have to be knocked out is called a *cut set* and their elimination leaves the organism with no other option to grow other than the chosen pathways which also produce the desired product.[10–15] This approach is illustrated in figure 2.2.

2.4 Flux

Flux is the net amount of metabolites moving through a reaction, meaning that equal forwards and backwards reaction rates result in zero flux and a reaction taking place in reverse has a negative flux value. Flux is dependent on gene expression, translation, post-translational modifications as well as protein metabolite interactions.[16]

While measuring flux for whole reaction pathways is often feasible by measuring substrate uptake and product exudation, it is much harder to get flux values for specific parts of a pathway or single reactions. This means that the highest and lowest possible flux values are generally only known for the major exchange reactions.[1,17]

The current state of the metabolism of an organism is determined by which reactions are active or inactive at any moment and can therefore be described with a flux vector containing the flux values of all reactions in the system. The sum of all possible flux vectors, called the solution space, is contained in a flux cone as shown in figure 2.1. Notably, while the flux cone is unbounded by definition, for the purpose of calculations, arbitrary upper and lower flux bounds of 1000 and -1000 are used. If 'specific' flux bounds are used instead, the formerly unbounded flux cone is transformed into a bounded flux polytope.[18]

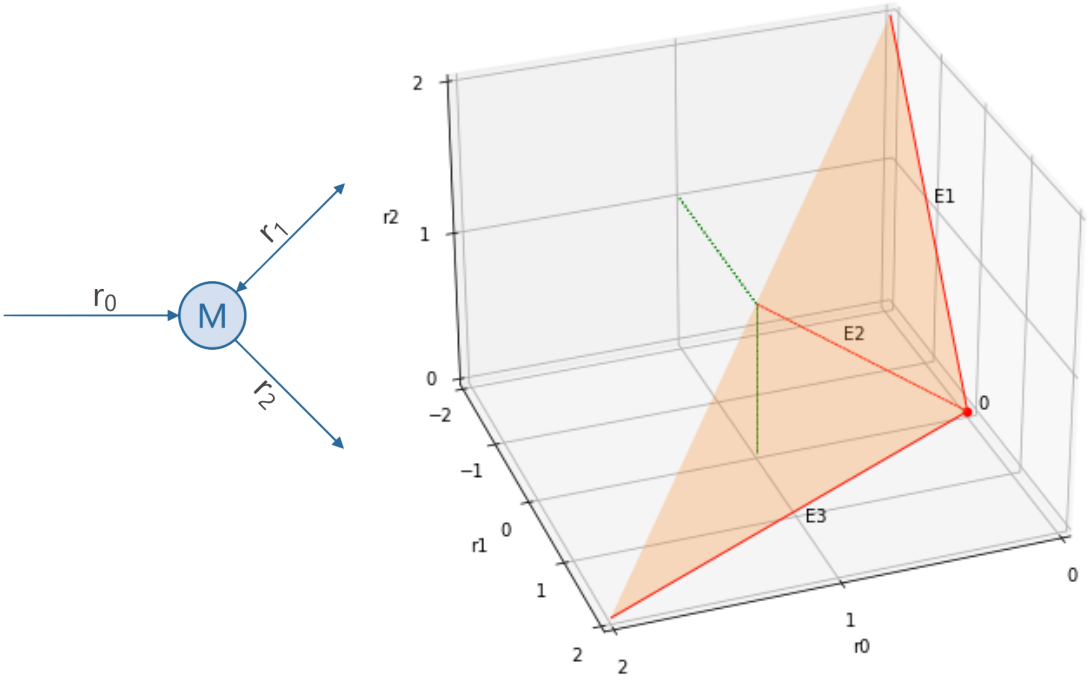

Figure 2.1: A simple toy model consisting of one internal metabolite M, two unidirectional reactions r_0 and r_2 as well as the bidirectional reaction r_1. Next to it is the corresponding solution space in the form of a two-dimensional flux cone. The flux cone is confined on two sides by the rays E1 and E3 in red and is unbounded on what appears to be the third side. E2 represents a flux solution that is a convex combination of E1 and E3. The dotted green lines serve solely as a visual aid to better visualize the orientation of E2 and the flux cone in three-dimensional space. Notably, while this flux cone lies within a two-dimensional plane, the flux cones of more complicated networks can span a much larger number of dimensions.

2.5 Applications of metabolic models

The use of microbes and even mammalian cell lines for the production of varies substances is already economically viable and oftentimes a more sustainable alternative to conventional methods. In its simplest form, this means utilizing wildtype bacteria in bioreactors that mimic known growth conditions, but there is huge potential to improve the output, yield, and scope via the process of metabolic engineering.[1]

While optimizing production rates and yields might sound similar, there is an important distinction to be made. Under nutrient-starved conditions, for example, the organism might want to use all of the existing substrate as efficiently as possible even if this means growing slower or using up high amounts of the more abundant molecules. On the other hand, when nutrients are abundant the organism might want to grow as fast as possible to take advantage of this abundance.

While, depending on the model, optimal rate and optimal yield can be highly correlated, oftentimes they also differ greatly. In addition, some methods of analysis like *elementary flux modes* can only calculate yields, other methods like *flux balance analysis* primarily find rates and some methods like *elementary flux vectors* can do both.[18] In addition, choosing the right tool for the task also depends on the availability of known flux bounds which are needed for rate optimization.

Constraint-based metabolic modeling allows both for the elucidation of possible maximum product rates or yields as well as for suggesting intervention strategies to try to achieve those maxima. One such intervention strategy is the coupling of the organisms' biomass production to a reaction that produces the product of interest and then eliminating all other potential biomass production pathways.[10–13,19]

A simpler example without the need for gene manipulation is used in the production of organic acids. The cell has to balance reducing compounds like NAD(P)H either by reducing oxygen or via the excretion of acids produced in fermentative processes. Therefore, by depriving the organism of oxygen, it is forced to switch to anaerobic metabolism and has to create organic acids. Apart from a guaranteed production of at least a minimal amount of organic acids during each growth cycle, this also results in an additional advantage: while at first, the bacteria have lower growth rates under the new conditions, over the next generations the organism can adapt and increase its biomass production which in conjunction also increases the organic acid output due to them being linked.[20]

When making use of gene editing, the typical strategy is to knock out genes corresponding to unwanted reactions or by overexpressing enzymes which catalyze only the desired reactions.[21,22] Lastly, there is also the option of knocking out or overexpressing metabolites which facilitate regulatory functions.[23]

The common goals for engineered strains are high product yield, high production rate as well as strain stability, meaning that the altered organism has to be resilient to random mutations. When optimizing production rate it is important to note that wildtype bacteria are often quite close to maximum possible biomass production while those pathways which lead to high product synthesis rates often exhibit relatively low, or sometimes even no biomass production at all. These pathways would not be feasible goals for bioengineering since the bacteria would die out very quickly. Therefore the goal is to look for pathways exhibiting high product synthesis rates but which still have moderate growth rates.[1] Figure 2.2 shows an example of how the product formation rate could be increased by knocking out undesired growth pathways.

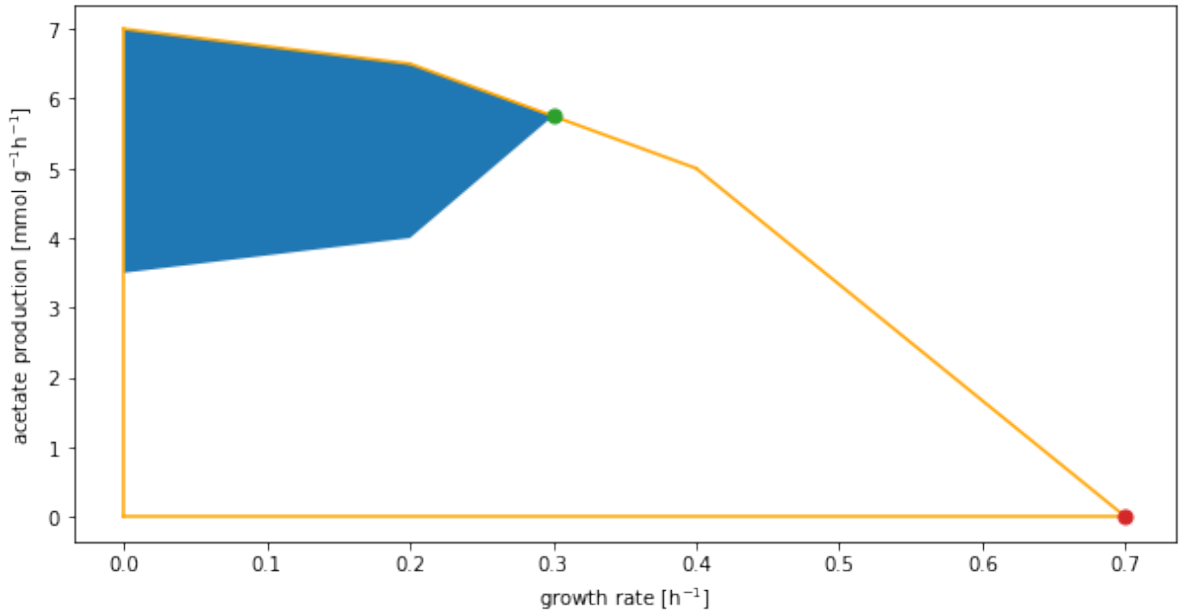

Figure 2.2: The orange hull shows an example production envelope for a wildtype bacterium. Since the organism tends toward high growth rates, it will likely utilize pathways resulting in high growth- and low acetate production rates close to the red dot. The goal of metabolic engineering is then to knock out pathways until the organism is restricted to the blue area. When the bacterium now tries to maximize its growth rate it has no choice but to utilize pathways in the vicinity of the green dot with less than half the growth rate but significantly higher acetate production.

3 Modeling approaches

3.1 Overview

While there are many points of overlap between different modeling approaches, one way to broadly categorize them is by separating them into *graph based* approaches which rely mostly on the topology of the network arising from its stoichiometric matrix, *constraint-based* models which add some additional quantitative restrictions and the rather more distinct group of *kinetic models* based on substance concentrations and rate equations:

- The starting point for analyzing metabolic models is the stoichiometric matrix which consists of the known metabolites and the reactions which facilitate their conversion. Metabolites can be grouped into internal and external metabolites. Internal metabolites are substances like molecular building blocks, intermediates, and energy storage molecules and are located inside the cell. External metabolites consist of nutrients that originate in the environment, as well as products and byproducts which were exuded by the cell. Both the uptake of nutrients and the exudation of products and byproducts is facilitated by exchange reactions.[1] The stoichiometric matrix is the prime tool which shows how all these substances are related to each other and how they can be interconverted. Using principles from graph theory and through pathway analysis, metabolic models based on the relation between metabolites can already give good insights into biological systems and can lead to the development of theories and intervention strategies for further testing in the lab.[1]

- In *Constraint-based models*, which are currently one of the most common and successful methods in systems biology,[21,24–26] the model's accuracy and capabilities can be improved further with the inclusion of constraints. The most important of these is the *steady state condition* which states that, over longer timeframes and under consistent outside conditions, the production of each internal metabolite must be balanced out by an equal consumption of that metabolite.[1] Contrary to that, metabolites which can be exchanged with the environment do not require balanced production and consumption rates. This means that they can exhibit a net inward, outward or no net flux at all. The second important set of constraints, called *reversibility constraints*, involves annotating which reactions are completely reversible and which ones are only feasible in one direction. These restrictions get rid of many biologically infeasible pathways. For some reactions like nutrient and oxygen uptake as well as the synthesis of some metabolites, not only the directionality is known, but concrete rates can be measured. The inclusion of this data leads to the third important type of constraint, *capacity constraints*. This not only sets realistic upper limits for reaction rates but can also be used to force reactions to take place at a certain rate. A common example would be including a minimum value for the rate of ATP production, without which the organism wouldn't be

able to sustain itself. While most metabolic modeling approaches already include the first two constraints and allow for qualitative analysis and some quantitative insight like relative yields, the inclusion of capacity constraints also allows for a wider set of quantitative applications like the calculation of maximum production rates. While those are the most common constraints, many approaches include additional constraints often related to enzymes and genomic regulation which can not only improve how well the model approximates real behavior but interestingly this can also have a positive effect on calculation speed. This is possible because the main limiting factor is the high number of flux solutions, many of which are mathematically possible but can't occur in nature due to a variety of factors that are not taken into account by the model network. Eliminating unrealistic solutions by introducing additional constraints, or limiting the solution space through other means, can therefore reduce calculation times.

- Lastly, a quite distinct approach is that of *kinetic modeling* which includes kinetic parameters to set up a system of equations describing the change of metabolite concentrations over time. While this method has quite a few advantages, it is hindered by the difficulty of measuring the necessary kinetic parameters as well as the complexity of solving the resulting system of differential equations. In addition, more and more methods are developed which are able to integrate these kinetic parameters into the otherwise more convenient constraint-based methods instead. While these methods are still not as accurate as kinetic models, they do constitute the best alternative as long as the problems with kinetic models have not been dealt with.

These different modeling approaches can not only be used to better understand the properties of a biological system but also for the improvement of product synthesis. Metabolic modeling can find a pathway with a higher production rate or a better yield that is only active under certain circumstances or suggest an intervention strategy for changing an organism's DNA to achieve the same goal. This is possible partly because organisms don't necessarily evolve towards maximizing rate or yield but instead often favor systems that are highly robust to mutation by incorporating redundant synthesis pathways.[27]

Apart from developing faster algorithms, there are many ways to further increase the speed and accuracy of metabolic analysis tools. One aspect in need of improvement is the incorporation of enzyme data into metabolic models. Enzymes are produced by the organism to catalyze most of its reactions and while many enzymes for specific reactions are known and their thermodynamic effects can be measured, it proves challenging to include this information in models. One approach is the use of kinetic models which rely on the rate governing effects of the enzymes but come with their own limitations which will be elaborated later.[1]

Alternatively, enzyme data can be used in conjunction with pathway analysis, by relating genomic data to network properties. When for example a synthesis pathway is proposed, the genes corresponding to the enzymes needed for the reactions in the pathway can be knocked out as a test.[25,28]

Usage of enzyme information is made more complicated though by the fact that some reactions can be catalyzed, often with different efficiencies, by various enzymes, called isoenzymes, as well as some enzymes being able to catalyze a variety of different reactions. In addition, enzymes can be encoded in the genome more than once and the subunits of enzyme complexes can be spread out over the different areas of the genome.[1]

3.2 Kinetic models

Kinetic models represent the most direct approach of creating metabolic models. Here the metabolism is modeled by chemically sound reactions, meaning that they not only include stoichiometric relationships but also reaction rates. Since the kinetic parameters of a reaction are related to its thermodynamics, the measurement of thermodynamic parameters can be used to further optimize the kinetic parameters of the model. In a system where all metabolites are connected using such reactions, and initial metabolite concentrations are given, the change in concentrations over time can be calculated. Mathematically speaking, this means solving a set of differential equations.[29]

While in theory, this method would allow for a very accurate representation of biological reality, it is held back both by the available data as well as hardware limitations. Regarding available data, uptake rates of external metabolites and reaction rates of specific enzymes are often known but it is much harder to get such data for internal metabolites which do not leave the cell. For external metabolites, it is sufficient to monitor concentrations in the growth medium of a cell colony while for internal metabolites the concentration inside the cells has to be measured.[1,17]

This is further complicated by the fact that under most circumstances cells obey the steady state condition, meaning production and consumption of metabolites balance each other out and no change in concentration occurs. Furthermore, many kinetic parameters are not static but depend on a host of factors like O_2 concentration, pH, availability of catalysts, and temperature.

Methods utilizing ^{13}C tracers can be used to access some of those harder-to-measure parameters. For this purpose, the bacteria are fed with ^{13}C labeled growth substrate and over time the carbon isotope is incorporated into all metabolites. The time it takes for a metabolite to be saturated with ^{13}C depends on the reaction rates of all reactions comprising a pathway leading towards that metabolite, and therefore the reaction rates can be calculated by measuring the saturation speed of different metabolites throughout the metabolism using nuclear magnetic resonance or mass spectrometry.[30] Another approach is by circumventing the lack of hard data for such parameters and instead approximating them by utilizing computational methods like monte carlo simulations.[29]

Due to these factors, the data acquisition and building of kinetic models is more time-consuming than other approaches and even complete kinetic models are very hardware intensive. In return, they can yield very accurate results and are not limited to systems in steady state.[29]

It is also worthy of note that there are ways to include such kinetic and thermodynamic parameters in stoichiometric models without many of the downsides of kinetic models. Interestingly, for stoichiometric models, this results in an increase in calculation speed due to decreasing the number of possible solutions which have to be calculated. It has to be said though that this approach is still not as accurate as a kinetic model.[31]

3.3 Graph theory for metabolic networks

Building and analyzing metabolic models is heavily reliant on concepts from Graph Theory. In mathematics, a graph describes a set of objects and their relations to each other. This is normally visualized by points or vertices which represent the objects and edges which connect those vertices.

While these regular graphs only connect a single vertice to another single vertice, many reactions in a metabolic network involve multiple products and educts resulting in a directed hypergraph,[32] where a set of vertices (the educts) are connected to another set of vertices (the products). Figure 3.1 shows an example hypergraph based on a smaller toy model.

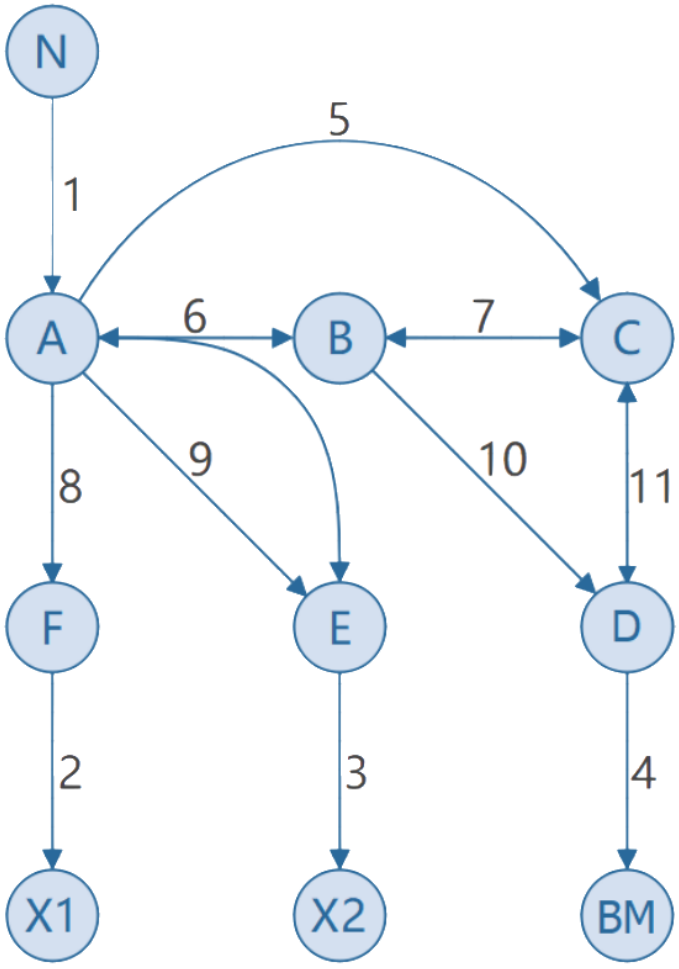

Figure 3.1: A toy model consisting of one nutrient N, two external metabolites X1 and X2, a biomass compound BM, six internal compounds A through F, and 11 reactions.

For further analysis, the metabolic network hypergraph is transformed into a regular graph. One basic way of doing so is by converting the metabolic network hypergraph into a substrate graph, where each metabolite is a vertice and if any reaction exists which has one metabolite as an educt and another as a product, an edge is drawn between those two. Depending on the reversibility of the reaction, this edge can be either directed or undirected, normally represented by arrows with one or two points.[1]

This process comes with a few significant disadvantages. Substrate graphs greatly simplify the complexity inherent in a metabolic model which not only makes it harder to predict real behavior[32,33] but also means that different metabolic models sometimes

share the same substrate graph. Also, just because a reaction for interconversion exists does not mean that it is feasible under specific conditions, for example, if other educts are missing or if the organism has no way to further process or get rid of other products.

A more accurate method is the conversion to a bipartite graph, where there are two types of vertices, one for the metabolites and one for the reactions. Then all educts are directionally connected to their corresponding reaction which in turn is directionally connected to its products. This bipartite graph is a complete and unique representation of its underlying hypergraph.[1] Figure 3.2 shows a comparison of a simple hypergraph and its corresponding substrate- and bipartite graph.

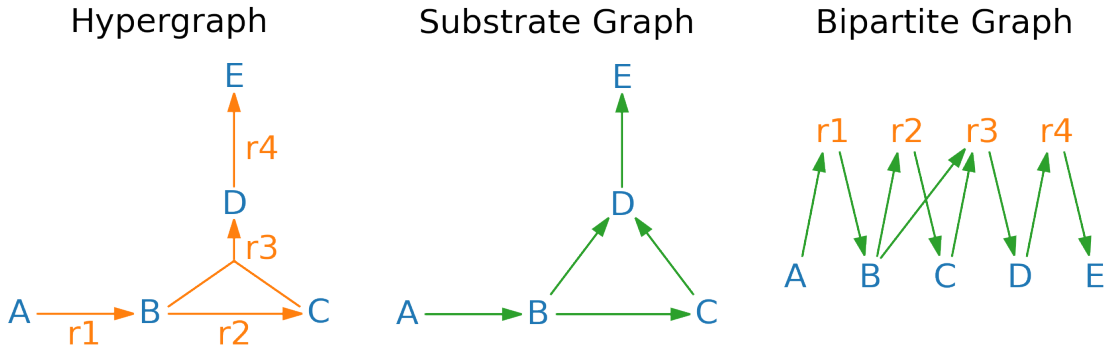

Figure 3.2: A simple hypergraph and its corresponding substrate- and bipartite graph. Metabolites are in blue, reactions in orange, and connecting edges that have no direct biological analog are in green.

Graph Theory provides a few useful concepts for analyzing the structure of metabolic network graphs:

- *Path length* is the number of edges between two vertices. After calculating the shortest path length between any combination of vertices, the average path length of a network as well as the longest of those paths, called the maximum shortest path length, can be used as a general measurement of how well the network is interconnected. Both parameters tend to be relatively low for metabolic networks suggesting high interconnectivity.[1]

- *Connectivity*, also often called degree, is the number of edges a vertice has.

- Following this, *degree distribution* is the distribution of degrees over the whole network. For example, in a randomly generated network, most vertices have a similar number of connecting edges. In contrast, most biological network graphs fall into the category of scale-free networks, where the degree distribution follows a power law. This means that they have a small number of vertices that are highly connected while most other vertices only have very few connecting edges.[34,35] Many real-world systems like economies, traffic systems, the internet as well as metabolic networks exhibit such a scale-free topology when modeled in graph form.[36–38] This is due to the fact that scale-free networks tend to have very short path lengths for almost any combination of two vertices, which is a very favorable property for many networks. For example the quickest path from one point in a transport system or the most direct way to convert one metabolite into another. It has to be noted though, that the shortest path in a network graph is not necessarily the most efficient path for the organism to use in nature. It might have a lower yield, only limited throughput, require substances that are short in supply or it might not be feasible at all.

- *Clustering* describes the likelihood that two neighbors of a node are also neighbors of each other. A high clustering coefficient means that the network contains many tight-knit clusters of vertices. Combined with the concepts of degree distribution and scale-free network topology this leads to some additional observations on metabolic network models. They have sets of metabolites that are very highly clustered and are connected to other such clusters via several central 'hub metabolites' like ATP, NAD(P)H, or pyruvate with very high connectivity and a low clustering coefficient.[1]

- This structure results in metabolic networks exhibiting the so-called *small world property*,[34,35] which means that while many vertices are parts of their own seemingly separate tightly knit clusters, they are non the less only ever separated by a few edges thanks to the central hub nodes connecting the clusters. This not only explains the short average path lengths of metabolic networks but also their resilience to the removal of random nodes through genetic mutation. In a random network without the scale-free and small-world property, removing random vertices quickly results in large parts of the network being disconnected from each other. In comparison, metabolic networks which have those properties can absorb the loss of many nodes while still retaining functionality as long as the central hub nodes remain. This is very important for the organism so as not to lose functionality of vital processes through random gene mutation.

Csete et al[39] suggest a somewhat different way to look at metabolic networks. They propose a bowtie-like structure where the central metabolism consists of ubiquitous cofactors and a few precursor substances which act as building blocks. This center then supports the conversion of a very wide variety of nutrients into a wide variety of products. They call this 'scale rich' instead of scale-free. There is always the question though how well those results from graphs represent the properties of the underlying hypergraph.

3.4 Constraint-based modeling

In contrast to graph-based network analysis which only considers the network topology, constraint-based modeling can produce concrete quantitative results with the inclusion of the previously mentioned constraints based on steady state, reversibility, and capacity.[21,22,40]

Mathematically, this approach utilizes concepts from linear algebra to operate on the stoichiometric matrix S. This matrix consists of n rows for the reactions and m columns for the metabolites. Metabolic models tend to have more reactions than metabolites. The matrix elements S_{ij} describe if, and how much, of a metabolite is needed for each reaction. A positive value for S_{ij} means that the metabolite i is produced by the reaction j and a negative value means it is consumed.[1,41]

It is important to note that this matrix on its own does not describe a specific flux state because it only gives stoichiometric information and does not tell us which reactions are currently active. To get a specific flux state, a flux vector is needed which describes what reactions are active, how much flux they carry, and into which direction. This flux vector has n entries, one for each reaction, a value of zero means a reaction is not active in the described flux state, a positive value to what degree a reaction is active in its stated direction, while a negative value means a reaction carries flux in

the reverse direction. Applying the steady state condition, where internal metabolite concentrations stay constant, now results in the equation: S*v=0.[1]

Mathematically this represents a homogeneous set of linear equations, the solution of which are all possible flux vectors v which obey this constraint. These vectors lie within an n-dimensional flux cone called the solution space, the size of which is one of the main limiting factors in constraint-based analysis which leads to the need for the addition of further constraints to shrink its size.[42,43]

This solution space can be imagined as lying in an n-dimensional coordinate system where each axis corresponds to a reaction and each flux solution is a point whose coordinates describe how active each reaction is in this solution. The solution space then is the geometric shape containing all points compatible with the applied constraints.[1]

This approach also means that while the real unknown flux solution has to be part of the solution space, many other results in the solution space don't actually appear in reality. So while not all solutions in the solution space are 'real' those outside the solution space can at least be ruled out.[1]

Reversibility constraints prescribe which reactions can have either only positive or negative values in the case of irreversible reactions while reversible reactions can still have both signs. Setting a reaction to irreversible greatly reduces the size of the solution space by basically removing half an axis of the n-dimensional space for each irreversible reaction. Mathematically this takes the form of a set of inequalities like: $v_i <= 0$.[1]

Capacity constraints can further cut the axis by setting limits which can not be exceeded either because it was measured that the organism can only support a specific maximum reaction rate, possibly due to enzymatic limitations, or because some external metabolites are so sparse that the cell can not absorb more than a certain amount per timestep.[1]

Mathematically this also takes the form of inequalities but allows for numbers other than 0. For example: $v_i <= 10$. Sometimes, instead of a range of values, the flux value is set to a specific number for processes that must always take place at a certain rate. For example, when a specific growth rate is needed to keep the concentration of a bacteria colony in a bioreactor constant it makes sense to disregard any flux states which can not achieve that growth rate or lead to excessive growth.[1]

4 Specific applications

Even small networks have many possible pathways and networks the size of metabolic models exhibit an exceedingly unwieldy amount of possible pathways. This necessitates methods of simplification and abstraction not only to analyze complex datasets but also to make sense of the results.

Out of the three general approaches for metabolic modeling, analysis based on graph theory which yields qualitative results based on network topology, constraint-based modeling which can also produce quantitative results based on the flux polytope[44] and kinetic modeling which uses rate equations, the first two show considerable overlap and some tools can be used for both.

Therefore, while the focus lies on constraint-based modeling, methods, nomenclature, and insights from graph-based analysis are also of interest. Namely, the tools presented here are: flux balance analysis (FBA), flux variability analysis (FVA), elementary flux modes (EFM), and elementary flux vectors (EFV). These tools rely on concepts from applied mathematics, data science, and informatics like linear algebra, linear programming, combinatorial optimization, graph theory, and statistical analysis methods.[1]

4.1 Flux balance analysis

Like other constraint-based methods, FBA also follows the steady state condition and reversibility constraints but also optimizes an objective function which is most often the production of biomass.[42] Research shows that, at least for small organisms and under substrate limited conditions, it is reasonable to assume that growth is the main priority of cells.[10,12,25] Notably, the property that is optimized here is not the yield of biomass per consumed substrate but instead the overall biomass production rate. Therefore, in contrast with elementary flux vectors,[18] yield optimal solutions are only found in cases where they coincide with rate optimal solutions, which makes FBA unsuitable for some metabolic engineering applications.[18]

Other commonly used optimization functions are the production of ATP[45] or of a product of interest.[46] It is also possible, especially when optimizing product synthesis rate, to include multiple reactions in the optimization function in the form of a linear combination of all reactions of interest. In mathematical terms calculating FBA means solving a linear optimization problem using a variety of computational methods like the simplex algorithm.

4.1.1 Applications

FBA has become the most popular out of the constraint-based analysis methods, resulting in a wealth of different applications:

- *Optimality*: FBA can be used to figure out if an organism can still keep up its maximum production rate of biomass or other products of interest after genetic modification or under different environmental conditions.[10,12,25,47] For example, in bioengineering, it is of interest how the organism behaves when knocking out specific reactions to force the cell into using alternative pathways in order to produce products through normally 'dormant' reactions. This can be tested by performing an FBA, setting the upper and lower flux constraint through that reaction to 0, performing a second FBA, and comparing the results.

- *Essential reactions*: Related to the previous point, if a gene deletion results in zero flux through the objective function, typically the biomass reaction, then an essential reaction was found. This can also be used to improve metabolic models by knocking out reactions that are predicted to be essential and checking if the organism stops functioning in the lab. If the organism can still grow, this suggests that there exist alternative pathways which should be added to the metabolic network reconstruction. It has been shown that the pattern of non-essential reactions can be used to predict which mutants are viable in nature with relatively high certainty,[48,49] information which can also be used to measure the resilience of the organism.

- *Blocked reactions*: Due to the relatively high speed of FBA, it is possible to run multiple calculations where each reaction in turn is set as the objective function and is both minimized and maximized. All reactions where the minimum and maximum are zero, have to be blocked reactions.[8,50] Compared to more basic methods, like the kernel matrix, this process finds some additional blocked reactions since reversibility constraints are also considered.

- *Metabolic engineering*: FBA can be used to calculate the maximum possible yield for any metabolite and this information can be used to come up with intervention strategies to ensure the desired behavior.[21,22,46,51] This is achieved by tools such as OptKnock by solving an optimization problem, with two layered objective functions: the inner layer is the biomass function, and the outer layer is the reaction that produces the desired product. The program then tries out different gene knockouts to maximize product formation rate and then optimizes the biomass production for that specific knockout. This process relies on mixed integer linear programming, which makes it noticeably more hardware intensive than simple FBA calculations, especially with bigger models or a high number of knockouts.[52] To combat this problem, OptGene[53] applies evolutionary optimization while GDLS[54] uses a heuristic local search algorithm.

Avenues for further improvement are the incorporation of heterologous reactions and genes or regulatory systems.[11,15,55] Examples are: OptStrain[56] , OptReg[57] , OptOrf[58] , OptForce[59] , and RobustKnock.[60]

One of the potential flaws of FBA that has to be kept in mind is the fact that results are based on the assumption that the organism tries to optimize biomass production rate which is not universal for all organisms or environmental conditions, especially after gene modification.[61]

The second flaw results from the fact that FBA only finds one of the potentially many rate-optimal solutions which can rely on very different production pathways. Additionally, any linear combination of optimal pathways results in a new optimal pathway, resulting in a technically infinite amount of solutions, each of which could be the one most closely resembling reality.[1] This means that while the predicted growth rate or other objective function might be correct, it allows only for limited insight into the inner workings of the metabolism which can further complicate the process of finding suitable candidates for gene knockout.[62] There are a few ways to find some or all of the missing flux solutions like mixed-integer linear programming,[63] vertex enumeration methods,[64] metabolic pathway analysis, combinations of FBA with FVA or by incorporating omics data.

The third flaw of some methods like optknock is that they only look for knockouts which result in a high maximum product formation rate. Since such knockouts are not guaranteed to result in high minimum production rates, the actual rate of product formation can be much lower than predicted. Other methods like robustknock can overcome this problem.[61]

4.2 Flux variability analysis

FVA is closely related to FBA but, as the name suggests, it does not just calculate the maximum flux of a metabolite but instead the range of all possible flux values.[62] The starting point for this method is also the steady state condition with reversibility and or capacity constraints, but instead of only maximizing an objective function the flux of each reaction of interest is both minimized and then maximized.[63,65,66]

This can be used to partly solve the problem of FBA methods only showing one of many flux vectors which lead to the highest output. FBA starts by maximizing flux through the objective function first, fixing the objective functions flux at that value, and adding it as an additional constraint. After that, all other reactions of interest are in turn minimized and maximized while still forcing maximum output through the initial objective function.[62] Compared to other methods of calculating a bigger area of the solution space, this is not as extensive but makes up for that with quick calculation times, making FVA feasible even for very large networks.[1]

4.3 Elementary flux modes

EFMs can be imagined as smaller versions of metabolic pathways, in fact they are the minimal components that make up a metabolic network. This means that, if any part of the EFM is cut, the whole EFM ceases to function.

Notably, while most EFMs connect two endpoints, loops can also be EFMs. Mathematically speaking, EFMs are rays in the solution space which are unbounded in one direction. Since EFMs are based only on the steady-state principle and reversibility constraints but not on flux constraints, this solution space is an unbounded flux cone instead of a bounded flux polytope. Some of the EFMs lie on the edges of the flux cone while others lie inside the intersection between the cone and the major coordinate planes as can be seen in figure 4.1.[41]

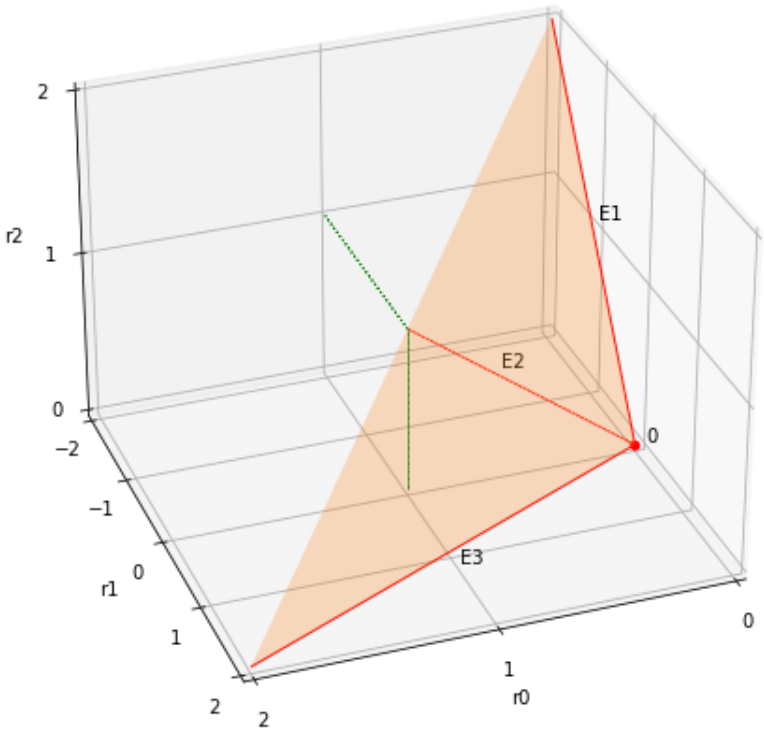

Figure 4.1: A two-dimensional example flux cone in three-dimensional space. The flux cone can be constructed by convex combinations of E1 and E3 while EFM E2 is a specific convex combination of E1 and E3 lying inside the flux cone at its intersection with the plane where r1=0. The origin labeled 0 also lies in this plane and the dotted green lines are used as visual aids to show E2s' orientation in three-dimensional space.

Bigger pathways can be decomposed into several EFMs and in fact, any point in the solution space and therefore any flux state can be arrived at with a linear combination of EFMs. Therefore, compared to some other analytical tools, they are not just an abstraction useful for understanding complex systems, but instead they describe the system completely.[41]

This process of combining EFMs happens without cancellations, which means that no positive and negative fluxes in the linear combination are allowed to cancel each other out. More precisely this means that if an EFM is described by a flux vector v_3 that is made up of flux vectors v_1 and v_2 and if v_3 has zero flux, then v_1 and v_2 also have to carry zero flux. Conversely, if the flux value of v_3 is smaller than zero then either both v_1 and v_2 are also smaller than zero or one of them is zero and the other is smaller than zero. The opposite is true for values higher than zero.[18] In addition to that, if any reaction making up an EFM is irreversible, then the whole EFM also has to be irreversible.[18]

One useful property of EFMs is their interaction with subsystems. If all EFMs of a system are known and some reactions are removed from the system then the EFMs of the resulting subsystem do not have to be calculated anew. The new EFMs are all of those already calculated, which do not include any of the removed reactions.[18]

EFMs can be found by successively removing reactions while checking if the steady state condition still holds until no reaction can be removed anymore.[67] So while finding EFMs is not hard, getting all of them is very time intensive because of the combinatorial explosion of possibilities. For example, a relatively small-scale model of E. coli with

100 reactions already has about 272 million EFMs.[68] Therefore, depending on the size of the system, different strategies for the calculation of EFMs are necessary.

4.3.1 Applications

- *Estimating resilience*: EFMs can be used as a measure of cellular robustness by decomposing a pathway into all its constituent EFMs. If there are many different combinations of EFMs that result in the desired pathway, it is assumed to be more robust since random mutations of one gene can be compensated by all the other options.[69,70]

- *Metabolic engineering*: EFMs are used in quite a few different methods for metabolic engineering, one of the basic examples being minimal metabolic functionality where the complete set of EFMs is split into two parts: one containing all EFMs displaying a high yield of the desired product and one containing all the other EFMs. Since disabling any reaction in an EFM via gene knockout disables the whole EFM, the algorithm goes through the list of reactions and deletes reactions that are not present in the set of high-yield EFMs but which are a part of the other EFMs. After enough iterations, only the EFMs with high product yield remain. While the result consists of multiple EFMs, this method only suggests one set of gene deletions for enabling those EFMs, which is not necessarily the set with the lowest possible number of deletions.[13,14,19,71] Another method called correlation analysis not only selects desired EFMs by their high product yield but instead calculates which reactions in the EFMs correlate with product synthesis. Reactions with positive correlation are suggested as targets for overexpression while those showing negative correlation for knockouts.[23]

- *Miscellaneous applications*: Like many other methods, EFMs can also be used to figure out if the desired product can be produced from a specific substrate, to find the shortest pathway between two metabolites,[18] or to find essential, blocked and coupled reactions.

4.3.2 Possible improvements

There is a wealth of EFM related tools both as implementations in already existing software tools[10,25,47,48,72] but also as standalone implementations like Efmtool[24]or Metatool[48] .

Most of these approaches use the double description method,[73] which uses iterative gaussian combinations to combine existing EFMs into candidates for new EFMs which then have to be verified. This verification process is the limiting factor in the calculation and requires large amounts of RAM.

New implementations try to overcome this problem either by utilizing other mathematical methods altogether, like projection methods,[74,75] or by introducing additional constraints. At first glance, it might seem counterintuitive to attempt to improve calculation times by increasing the complexity of the model with additional constraints. What makes it work though is the fact that the limiting factor for the calculation is not the model itself with its relatively few hundreds or thousands of reactions with a number of constraints at a similar order of magnitude, but instead the sheer size of the solution space which quickly grows to a size of billions or trillions of flux solutions for bigger models. Additional constraints can drastically shrink the size of this solution space to improve calculation speed. Possible additional constraints include:

- *Boolean transcriptional regulation*: Many cell functions are activated and deactivated by the gene regulatory network depending on various internal and external conditions. E. coli for example disables the glyoxylate shunt when glucose is used as the main growth substrate,[76,77] meaning that all EFMs containing reactions from the glyoxylate shunt can be excluded from the solution space in that case. Calculations on an E. coli model have shown that around 99 percent of the 272 million EFMs were not feasible due to regulation. This method is already in use for FBA calculations but in addition to regulatory data not being available for many organisms there is the possibility that boolean constraints are too stringent for limiting the EFM solution space.[78]

- *Thermodynamic constraints*: While metabolic networks are stoichiometrically balanced, they do not contain thermodynamic data, resulting in many EFMs possibly being thermodynamically infeasible. According to Jol et al[79] and Kümmel et al,[80] when testing a yeast model, this was the case for 54 percent of all EFMs. These results corroborate also quite well with data from FBA.[81–83] However, in this case, thermodynamic feasibility was determined after calculating all EFMs. So, while useful as a proof of concept, methods would have to be found which incorporate this testing process into the calculation process itself.

- *Subsystems*: It is also possible to shrink the solution space by calculating EFMs only for subsystems instead of the metabolism as a whole. This comes with some additional problems though: First off, there is no objective way to determine where a subsystem starts and ends, making it a more or less arbitrary choice left to the researcher. Secondly, this process neglects that reactions inside a subsystem can be dependent on reactions from outside the subsystem. Lastly, it was shown that subsystem EFMs are not necessarily part of the EFMs of the system as a whole.[84–86] Kaleta et al[87] tried to solve these problems with the introduction of EFM patterns.

- *Subsets of EFMs*: Similar to only calculating the EFMs of a subsystem of the model, it is also possible to only calculate parts of the solution space. For example, de Figueiredo et al[88] only calculated the shortest EFMs while Pey et al[89] calculated only EFMs containing certain reactions of interest. This method is even feasible for genome-scale metabolic models and can also be applied to EFVs.

- *Quick decomposition*:[90] Many applications do not need the complete set of EFMs to be calculated. In this method, for example, only the EFMs present in a specific flux state instead of all possible flux states are calculated. Quick decomposition first looks for the reaction in the flux state which has the highest flux value and calculates an EFM containing this reaction. All flux values from this first EFM are subtracted from the flux state in question resulting in a new, smaller flux state on which the process is then repeated. This method is fast enough to apply to genome-scale models but by definition only produces a part of the solution space, which, depending on which original flux state was chosen, might not be representative of biological reality.[91–93]

4.4 Elementary flux vectors

EFVs arise from elementary flux modes with the addition of flux constraints in the form of upper and lower flux bounds.[18] This means that while EFMs are unbounded rays in flux space and serve as the constructors from which the flux cone arises, EFVs are bounded vectors that construct the flux polytope. Interestingly, even though flux bounds are often unknown for reactions involving internal metabolites, those reactions are still indirectly limited due to being part of pathways that contain other reactions for which bounds are known.

Because of their close relation, EFVs and EFMs can be congruent,[18] they share the same behavior towards subnetworks and they can be calculated by transforming the flux polytope into a flux cone and using the already established EFM algorithms.[18] They can be used as constructors for any point in the flux polyhedron by conformal combination but, while they do contain the minimal set of generators as a subset, they are not a minimal set of generators themselves. The number of EFVs can be higher or lower than the number of EFMs.[18]

4.4.1 Applications

Due to their similarity, they share many applications with EFMs and for some applications EFVs have a wider scope.[18] Non support minimal EFVs can contain more reactions than regular EFMs and correspond to larger synthesis pathways or cycles because, while a related EFM might only account for the ability to produce a certain metabolite, the EFV might include additional reactions to achieve a certain flux constraint.

- *Finding specific reaction types*: Like EFMs, EFVs can also be used to find essential, blocked, and coupled reactions. Essential reactions carry flux in all EFVs, blocked reactions do not carry any flux in any EFV and coupled reactions have either no flux at all in all EFVs or the same flux value in all EFVs.[18]

- *Yield optimization*: Since EFMs do not have upper and lower flux bounds, EFMs can only give relative yields. So while EFM analysis can find a pathway that converts the substrate to product very efficiently, the total rate of the reaction might be minuscule. In contrast, EFVs allow for the calculation of maximum reaction rates similar to FBA, but while FBA only finds solutions based on the parameters given by the researcher, which introduces a bias and can lead to missing important pathways, the flux polytope generated from EFVs contains all possible yield optimal solutions. This makes EFVs a more powerful tool for yield optimization than both EFMs and FBA. Lastly, while this is not always the case, those yield optimal solutions can also be rate-optimal solutions.[94]

- *Growth coupling*: In contrast to FVA and other FBA methods, EFVs can give better insights into pathways in which the production of certain products is coupled to biomass production which is quite useful for metabolic engineering. For example, EFV-based flux solutions can provide maximum guaranteed product yields when coupled with biomass synthesis.[95] While the calculation of yields is also possible with EFMs, EFVs can be used for rate calculations.

4.5 EFM and EFV comparison

To compare EFMs and EFVs Klamt et al[18] calculated them both for a modified version of the central metabolism of an E. coli model published by Hadicke et al.[96] Of interest were yield and rate optimal pathways for biomass, acetate, and lysine for growth on glucose as substrate with an uptake rate of 10mmol/gDW/h.

While the growth function already includes a stoichiometric amount of ATP needed for biomass production, an additional minimum ATP production of 8.39mmol/gDW/h was enforced, which the cell needs to maintain metabolic functions not associated with the production of new biomass. These two additional constraints turn the unbounded flux cone into a bounded polytope since more or less all reactions are dependent on glucose either directly as a growth substrate or indirectly for energy production.[18]

Using CellNetAnalyzer,[97] around 300 000 EFMs and 250 000 EFVs, all of which are bounded, were calculated. Around 28 000 thousand of the EFVs are support minimal and for some analysis queries like finding essential reactions and cut sets only this subset is needed while the rest can be discarded.

Almost all essential reactions, except for those needed to fulfill ATP maintenance requirements are the same for EFMs and EFVs. In the flux cone biomass production is 0.105 gDW per mmol glucose, achievable by 11 biomass-yield optimal EFMs, while for the flux polytope it drops to 0.100 gDW per mmol glucose, achievable by 10 biomass-yield optimal EFVs.[18]

The maximum yield of acetate, formate, succinate, lactate and ethanol did not change while lysine production dropped from 0.80 mmol per mmol glucose for the flux cone to 0.76 mmol per mmol glucose for the flux polytope. Notably, all biomass and lysine yield optimal EFMs also correspond to the rate optimal EFVs. For acetate though, 42 of the 198 yield optimal EFMs are not rate optimal EFVs since they have lower substrate uptake rates.[18]

As a next step, the oxygen uptake was constrained to a maximum of 5 mmol/gDW/h which increased the number of EFVs from around 250 000 to 320 000, compared with 300 000 EFMs. Biomass production yield dropped from 0.100 gDW per mmol glucose to 0.080 gDW per mmol glucose while growth rate dropped by an even larger margin from 1 per hour to 0.6 per hour.[18]

In addition, the EFVs for yield and rate optimization respectively differ quite heavily. Yield optimal growth is solely reliant on respiration as an energy source and therefore does not consume all available glucose if it doesn't have enough access to oxygen. In contrast, rate optimal growth managed to use up all the available glucose and covers the additional energy needs via fermentation. This process is less effective though and therefore the yield is lower.[18]

Acetate production behaves similarly but interestingly the number of viable EFVs increases under rate optimization while for biomass the yield optimal path had more EFVs.[18]

When setting oxygen uptake to zero, simulating anaerobic growth, the organism was only able to support around 45 000 EFVs and both yields and rates were lower for biomass, acetate, and lysine. For all three metabolites, there was a single EFV where maximum yield and rate coincide.[18]

5 Project scope

For many scientific and industrial applications an organisms' production envelope, a subspace of the complete solution space is of major interest. In its most basic form, as seen in figure 5.1, this is a two-dimensional, convex shape encompassing all possible production rates of an external metabolite relative to all possible growth rates. Technically this can be extended to any number of external metabolites but even after three dimensions, many existing methods quickly reach their limit.[98] The tools used in this project utilize constraint-based modeling where different kinds of constraints are applied to the stoichiometric matrix as described in the chapter "constraint-based modeling".

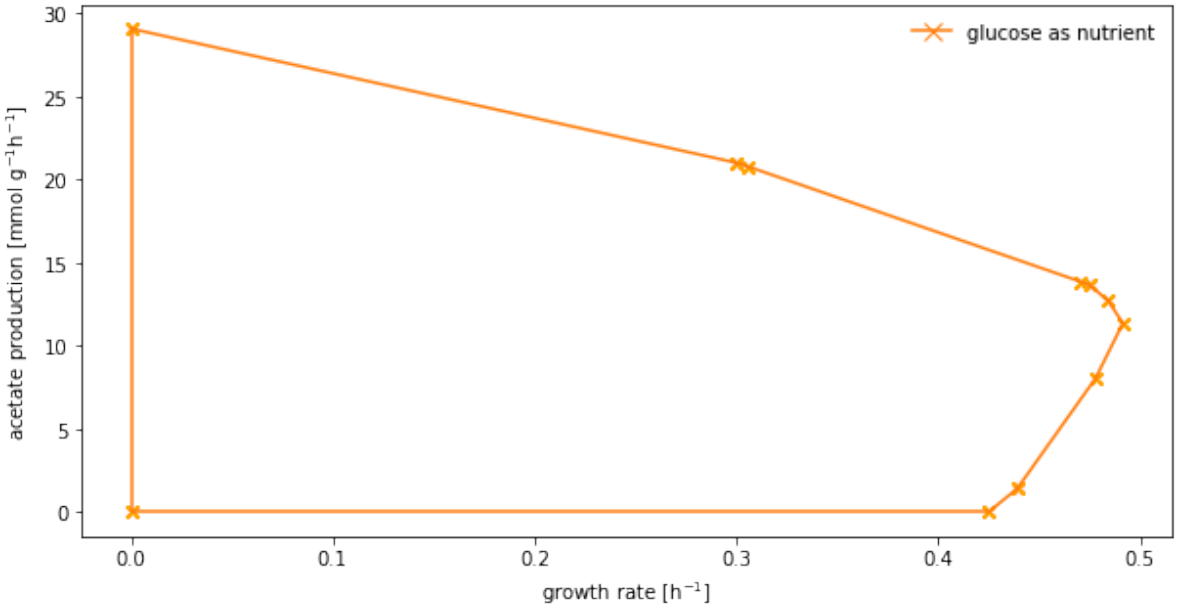

Figure 5.1: Two-dimensional convex hull projected onto the acetate synthesis reaction and the growth reaction with glucose as a growth medium. The hull was calculated using the genome-scale E. coli model

The most established of these methods is that of FVA,[99,100] where all but one of the metabolite fluxes of interest are fixed while the upper and lower limits of the variable fluxes of interest are calculated using FBA. This process is repeated until all variability of that one flux is determined for different fixed values of the other fluxes. This method results in an increase of computation time needed relative to p^{n-1} where p is the number of fixed sampling points at which variability is calculated and n is the number of reactions of interest. As a result of this relationship, calculation times increase at such a rate that calculations with four reactions of interest are already very slow and higher numbers of dimensions are infeasible.

Extending production envelopes to a wider range of metabolites is quite useful when

applying metabolic modeling not just to single organisms but to communities of different organisms instead.[101,102] In such communities, some organisms produce nutrients needed by other organisms and even if multiple species have the capability to produce a specific metabolite, specialization can still take place so that the more efficient producer takes over that role. This process of cross-feeding greatly influences the behavior of the individual organisms.[103]

One attempt to overcome the computational challenges of FVA-based production envelope calculation is the convex hull method (CHM), which is already used in computational geometry to project multidimensional polytopes onto spaces with fewer dimensions.[104] The resulting solution space is a convex hull, a kind of bounded polytope encompassing all points which can be generated by a linear combination of its *extreme points*. These extreme points in turn are the points furthest away from the center, which are not on a line or hyperplane between two or more other extreme points. Due to limits in numerical resolution, the algorithms in use tend to find false 'extreme points' from time to time that lie close to the actual extreme points or directly on a line between two actual extreme points and will be called *false extreme points*. Figure 5.2 visualizes these main concepts based on the convex hull encompassing several arbitrary two-dimensional points. Figure 5.3 shows how a three-dimensional convex hull can be projected onto two dimensions.

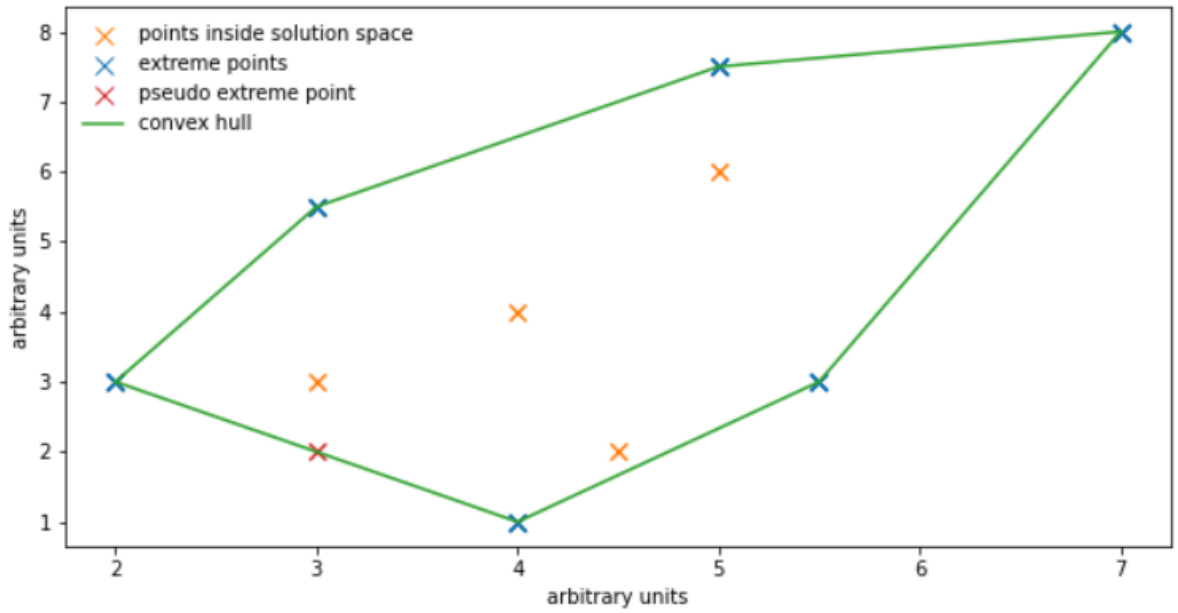

Figure 5.2: A set of arbitrary points showing extreme points in blue, non extreme points in orange, a false extreme point in red, and the boundaries of the encompassing convex hull in green.

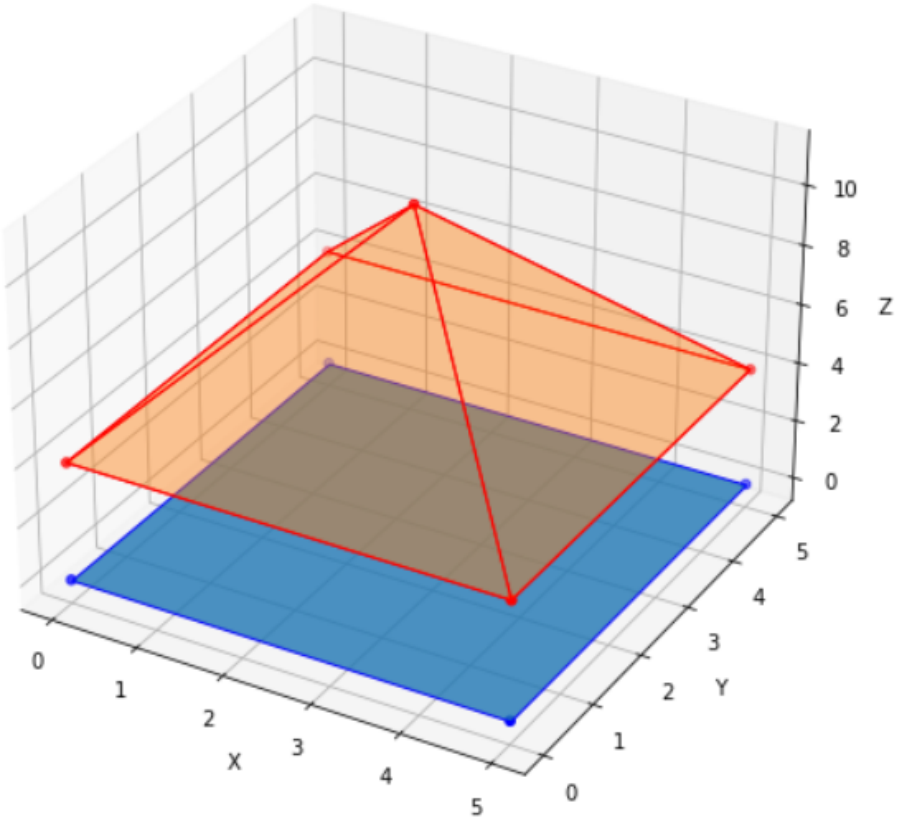

Figure 5.3: A set of three-dimensional points in red and its encompassing convex hull in orange represents an example solution space. Projection of this pyramid-shaped hull onto just the x and y dimensions results in the blue square.

The goal of this project was to quantitatively and qualitatively compare two versions of a CHM tool created by Helena A. Herrmann (available at: https://github.com/HAHerrmann/ConvHull) based on the work of Lassez[104] with a selection of other implementations to test their applicability for the calculation of production envelopes.

6 Materials and methods

6.1 Models in use

6.1.1 Simple toy model

This very basic network is mostly used as a testing ground for software tools under consideration where any results can be quickly checked for correctness and as an explanatory tool to visualize the underlying theoretical concepts. As shown in figure 6.1 it consists of just one internal metabolite and 3 reactions, one of which is bidirectional.

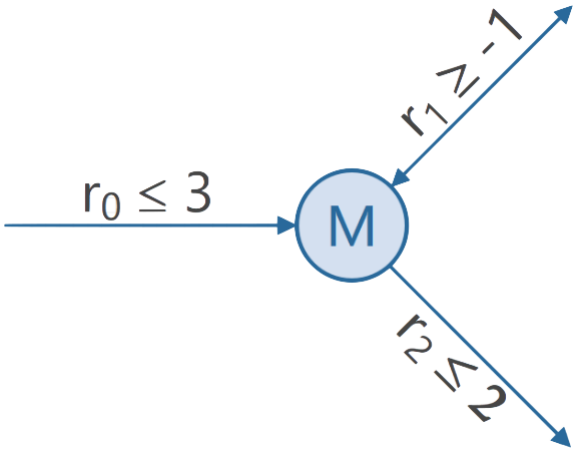

Figure 6.1: The simple toy model consists of one internal metabolite M, two unidirectional reactions r_0 and r_2 as well as the bidirectional reaction r_1. All reactions have flux constraints in their primary direction.

6.1.2 Intermediate toy model

This slightly larger model is meant to include important structural motives which often occur in larger metabolic networks but is still small enough to more easily troubleshoot errors during calculations and to allow for quick calculation times. The network is shown in figure 6.2 and consists of one nutrient *N*, one biomass product *BM*, two other external metabolites X1 and X2, as well as six internal metabolites *A* through *F*, connected by 11 reactions.

In contrast to the simple toy model, it has alternate synthesis pathways (either reactions 5 and 11 or reactions 6 and 10), an infinite loop (reactions 7, 10, and 11), a reaction producing byproducts the cell has to get rid of (reaction 6) and a way to get rid of 'excessive' nutrients (reaction 2 and 8).

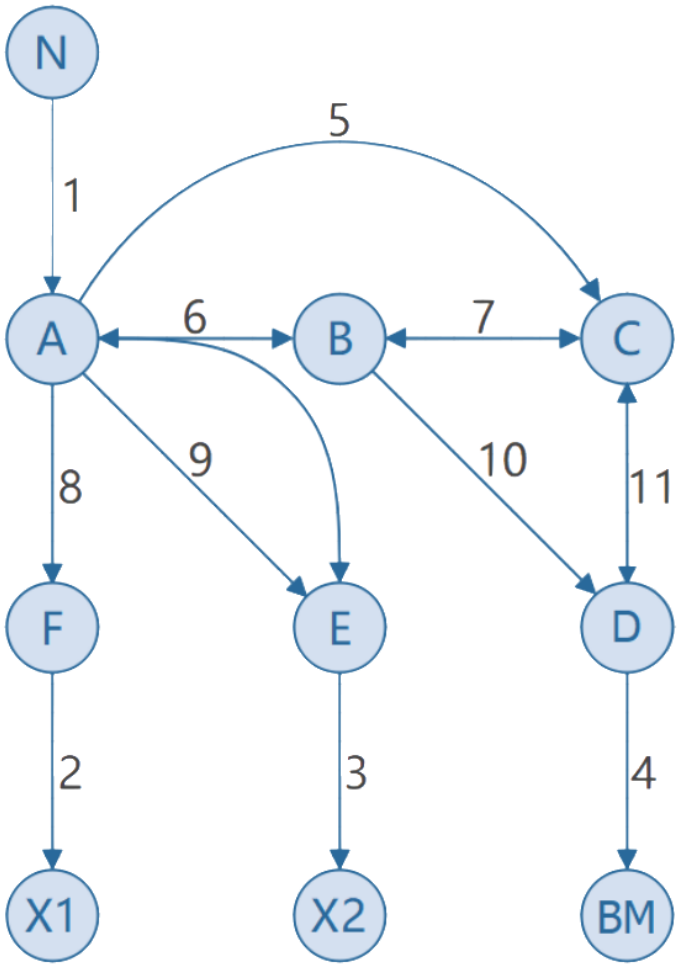

Figure 6.2: The intermediate toy model consists of one nutrient N, two external metabolites X1 and X2, a biomass compound BM, six internal compounds A through F, and 11 reactions.

6.1.3 E. coli core model

This model was designed by Orth et al[105] to serve as a tool for graduate and undergraduate students to learn important concepts of systems biology and constraint-based analysis. It is large enough to exhibit complex properties of organic systems while still being small enough to more easily understand its inner workings and limit calculation times.

Its 95 reactions and 72 metabolites include pathways for glycolysis, fermentation, the nitrogen metabolism, gluconeogenesis, the pentose phosphate shunt, the tricarboxylic acid cycle, the glyoxylate cycle, anaplerotic reactions, the electron transport chain as well as oxidative phosphorylation and the transfer of reducing equivalents. Many of those pathways were simplified by merging chains of reactions.

6.1.4 E. coli genome-scale model

The genome-scale reconstruction iJO1366 of the complete E. coli metabolism was also created by Orth et al[106] as an updated version of an older model of the commonly used E. coli strain *K-12 MG1655*. The original genome-scale model was reconstructed by Edwards and Palsson[107] and has since been updated multiple times.

This version consists of 2583 reactions and 1805 metabolites and includes information on exact reaction stoichiometry, metabolite formulas and charges, location of metabolites in the cytoplasm, periplasm or extracellular space, thermodynamic properties of all reactions, cofactors as well as relations between genes, proteins, and reactions.

6.2 Tools in use

6.2.1 CHM tool

CHM tool has two numerically different implementations based on the same geometric principle. One implementation uses decimal numbers in pythons *double precision* format which enables quick calculation times but is less accurate than the other version which utilizes *exact arithmetic* for calculations, meaning that numbers are represented as fractions. Unlike floating point notation, fractions can represent values with an infinite number of post decimal digits and are implemented in python via the sympy library. The exact version is meant to show the theoretical accuracy of the convex hull method while the double precision version should show its applicability to multiple dimensions and genome-scale models due to its faster calculation times.

The CHM tool algorithm starts with an initialization step where a quick approximation that is a subspace of the solution space is set up, which is then expanded over multiple iteration steps until it arrives at a complete solution. The initialization starts by maximizing the flux value of one of the reactions of interest like in an FBA calculation. Then this flux value is added as a fixed constraint to the model and the second reaction of interest is maximized. When the maximum of all reactions of interest has been calculated, these maxima are the coordinates of the first potential extreme point. The second potential extreme point is found by repeating this procedure but with minimization. The line between those potential extreme points is then optimized to yield a third potential extreme point. The plane spanned by those three potential extreme points can now be optimized again in a manner similar to the line before, and this process is repeated until n+1 potential extreme points are found, where n is the number of reactions of interest.

The resulting polytope is a subspace of the final convex hull, and some of its confining hyperplanes may already be congruent with the convex hulls hyperplanes. In the iteration phase, to arrive at the complete convex hull, the program goes through all combinations of the already calculated potential extreme points and checks if the hyperplanes spanned by these combinations are terminal hyperplanes, meaning they constitute the outer boundary of the preliminary polytope, or if they represent a cross-section through the polytope. The terminal hyperplanes are then pushed outward if possible. If this expansion is possible and a new potential extreme point is found, the program checks if any old potential extreme points are now inside the new polytope and removes them. Through this process, the list of potential extreme points is continuously updated with new points while old points are deleted. The program finishes when no new points can be found. In theory, all of the remaining points would be true extreme points but due to limited numerical accuracy, the algorithm also finds false extreme points which are close to the actual extreme points. This also means that in some cases the actual extreme point is not found at all but instead one or more adjacent false extreme points. Although this constitutes a qualitative error, quantitatively the difference is mostly negligible due to the close proximity of the correct and incorrect points.

6.2.2 lrslib

lrslib is an older convex hull algorithm implemented in the C programming language. It is both accurate and fast but instead of being able to project a polytope on a lower dimensional subspace, it always calculates the whole polytope which makes it infeasible for bigger models. Calculations of the production envelope of the E. coli core model already took between one and two hours. In addition to calculating production envelopes for comparison with the other software tools, lrslib was also used to calculate the volumes of those production envelopes.

6.2.3 chlib

This chm implementation is similar to lrslib but, similar to CHM tool, it is also able to project the result onto a smaller number of specified dimensions. It was chosen both due to this property and to add an additional benchmarking point to the comparisons.

6.3 Parameters for comparison

Due to several factors, the choice of which parameters to use for evaluating the software implementations is not as trivial as might be expected. These are the parameters under consideration and their accompanying advantages and limitations:

- *Runtime*: While calculation speed is one of the most obvious parameters for quantification and high speed is indeed a necessity when working with genome-scale metabolic network models, it can lead to deceptive conclusions. Erroneous solutions are often caused by the programs missing parts of the production envelope as shown in figure 6.3 and since calculation time is more or less proportional to the number of extreme points found, such errors can significantly decrease calculation times. If this is not taken into account, runtime as a parameter would favor the algorithm which, all other things being equal, calculates the most incomplete solution. Runtime is also of little value for very small models where calculation time is both inconsistent and not as important as accuracy.

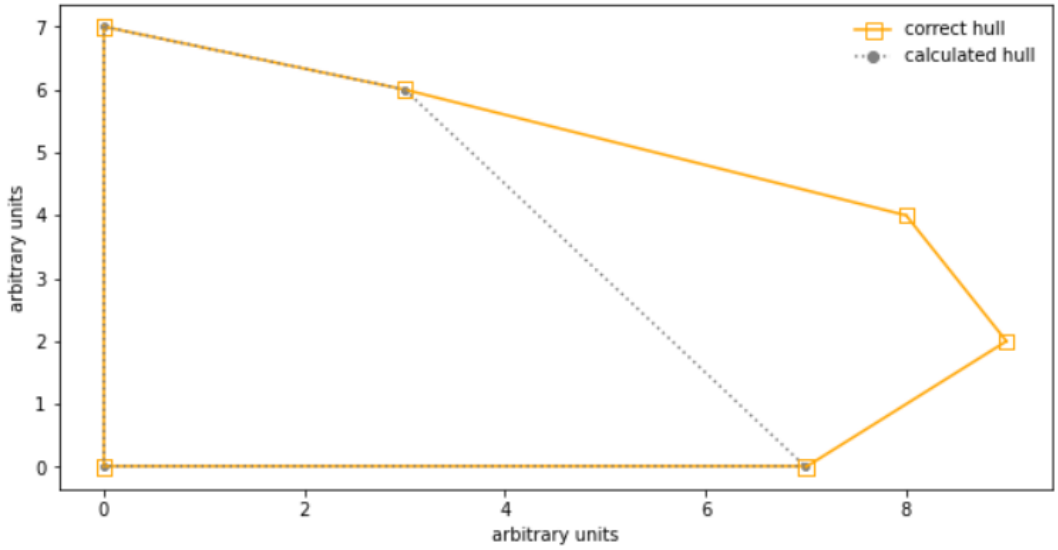

Figure 6.3: Example of a correct convex hull in orange and an erroneously calculated approximation in grey

- *Number of extreme points*: Ironically, while the extreme points are the direct result of the calculation, their number has only marginal comparative power. One use is to quickly find candidates for erroneous results if they have a significantly lower number of extreme points like in figure 6.3. On the other hand, a high number of extreme points is not necessarily a sign of accuracy if those points are close together like in figure 6.4 which has more extreme points than figure 6.3 but the calculated hull is very similar. Finding such similar points is often the result of limited numerical precision which can lead the program to find multiple false extreme points when in reality, these points should be just one. Lastly, even if the number of extreme points is the same for two results, the coordinates of those points can be quite different.

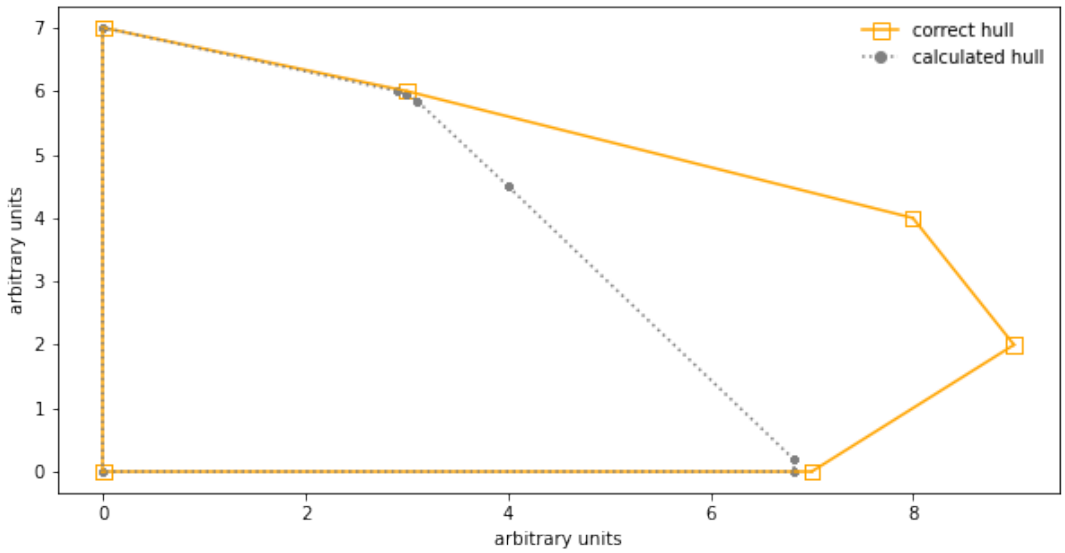

Figure 6.4: Example of a correct convex hull in orange and an erroneously calculated approximation in grey

- *Distance between points*: As an attempt to quantify how 'similar' two convex hulls are it seems quite useful to calculate the distance between a point of one solution and its closest counterpart in the other. While generally, this measurement is more useful than just the number of extreme points, a few, mostly statistical, considerations have to be taken into account. Since bigger models can have a large number of extreme points, an average distance is the only option for comparisons, but this introduces a choice between the different kinds of averages. It is not objectively apparent if median, arithmetic, or some differently weighted average is the better choice. It is also important to note that the distance between all points in convex hull A to those in convex hull B is not the same as from B to A. These problems become apparent when comparing figures 6.3, 6.4 and 6.5. The extreme points of the correct hull have a distance of zero to their closest counterparts of the calculated hull in figure 6.5 while the same is not true the other way around since the false extreme point has no equivalent in the calculated hull. Contrary to that, figure 6.3 shows a large distance between correct and calculated hull extreme points but zero distance between calculated and correct. While the calculated hull in figure 6.4 has a few extreme points which are not exactly congruent like those in figure 6.3 and therefore results in a larger distance from calculated to correct than in figure 6.3, the average distance between correct and calculated is smaller than in figure 6.5 since the multiple small inaccuracies outweigh the larger outlier. The usefulness but also the complexity of the calculation could be further improved by using the distance from a point in one solution to the hull of the other solution as a parameter.

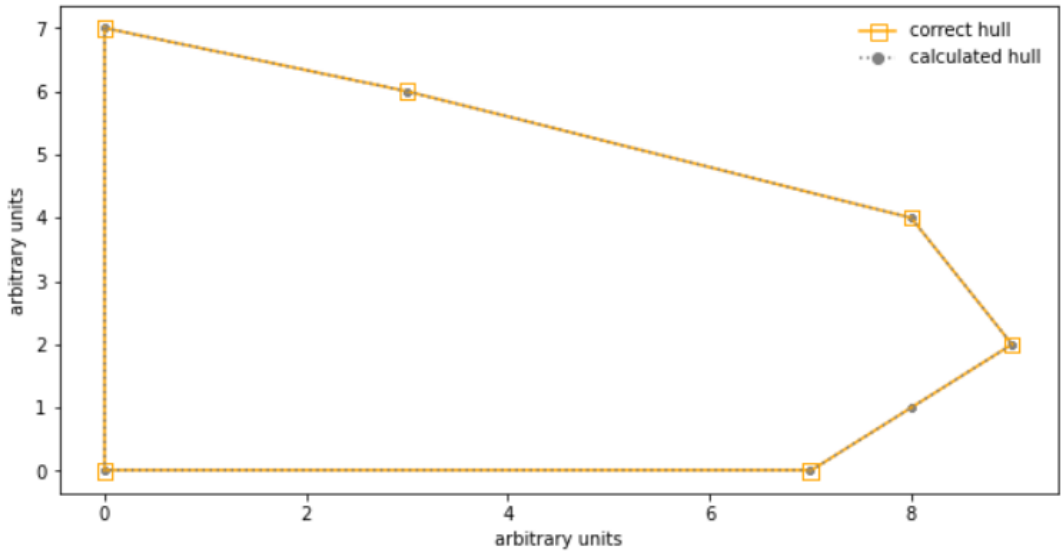

Figure 6.5: Example of a correct convex hull in orange and an erroneously calculated approximation in grey

- *Volume*: Calculating the volume enclosed in a convex hull solves many of the previous problems. It is just one number, so no more or less arbitrary statistical average has to be chosen, and far away outliers have a bigger impact than false extreme points. The only two problems are that volume is not affected by false extreme points which lie more or less exactly on a line between two other extreme points and that calculating volumes in four or more dimensions is a nontrivial task which is not possible in many computational environments. Here, lrslib was used for the volume calculations.

7 Results and discussion

7.1 Comparison between tools

Convex hulls were calculated for the simple and intermediate toy model as well as for the E. coli core model using chlib, lrslib, and both versions of CHM tool to gauge the tools' accuracy as well as calculation speed. Since this comparison spans many dozen convex hull projections which generally show good congruence between the different software tools, only a few are selected for each model to show general trends as well as outliers.

7.1.1 Simple toy model

Two-dimensional hulls were calculated for various permutations of the three reactions ([0,1], [0,2], [1,2], [2,1]) using the four tools. All calculated extreme points were exactly the same with just one exception shown in figure 7.1. There the chlib tool calculated an additional false extreme point which should have been filtered out during the iteration phase. While this constitutes a quantitative difference, the hull is qualitatively the same since the redundant point lies exactly on an outer boundary line.

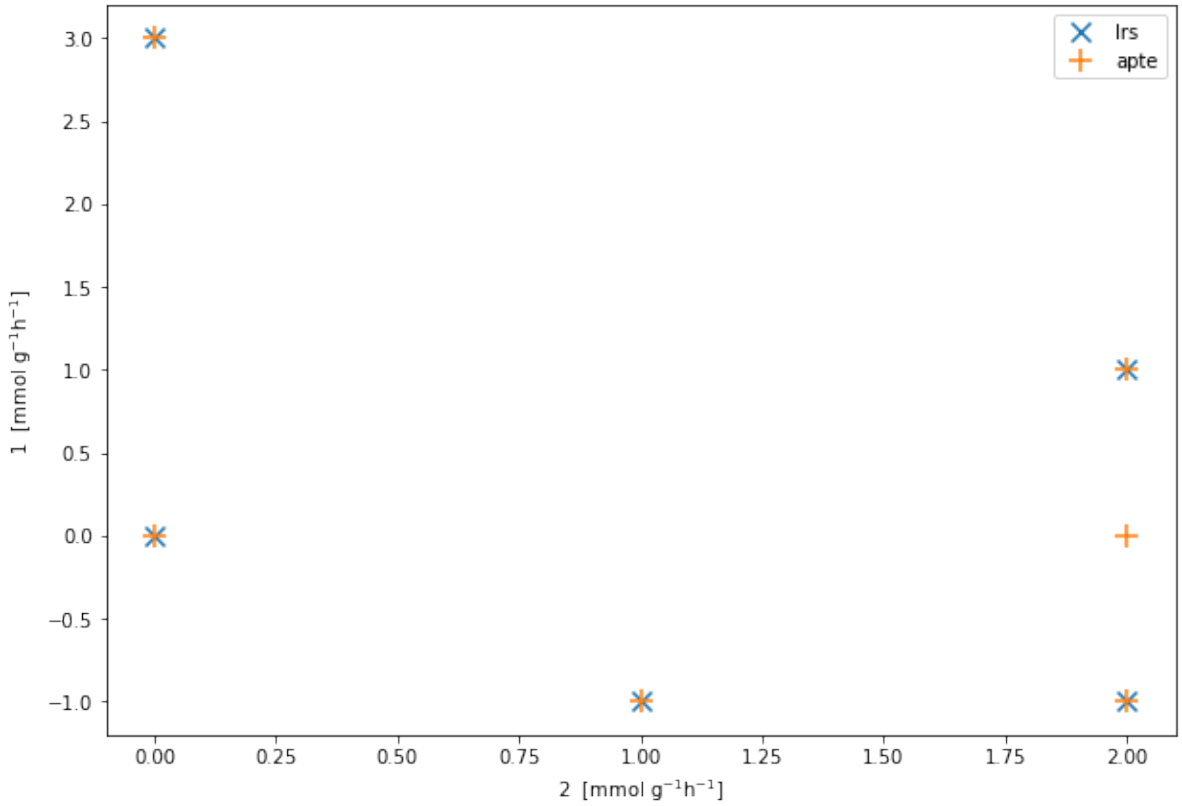

Figure 7.1: Production envelope of flux through reaction 1 vs flux for reaction 2 of the simple toy model calculated with lrslib (blue crosses) and chlib (orange plus signs).

7.1.2 Intermediate toy model

Similar to the simple toy model, two-dimensional hulls were calculated for all permutations of all external reactions ([0,1], [0,2], [0,3], [1,0], [1,2], [1,3],[2,0], [2,1], [2,3], [3,0], [3,1], [3,2]) using the four tools. In the same fashion as for the simple toy model but this time occurring multiple times, chlib found redundant extreme points. One example is shown in figure 7.2.

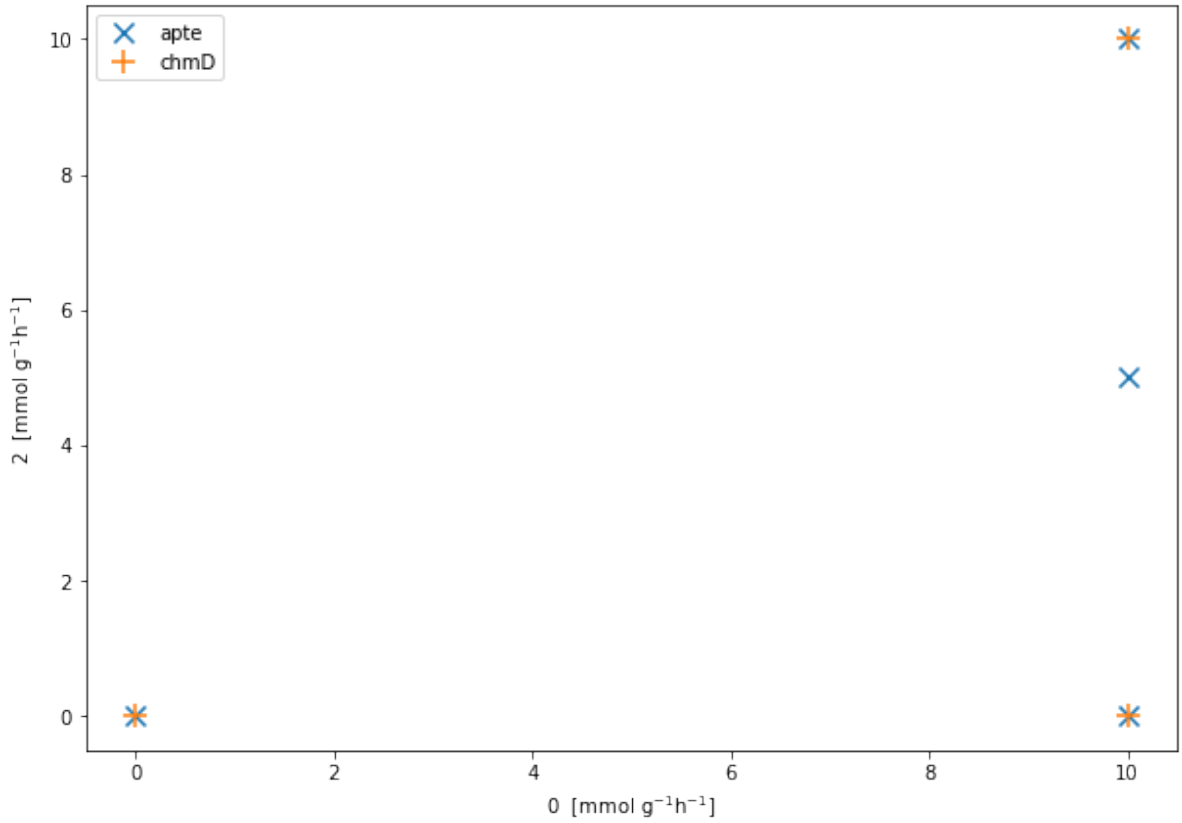

Figure 7.2: Production envelope of flux through reaction 2 vs flux for reaction 0 of the intermediate toy model calculated with chlib (blue crosses) and the double precision version of CHM tool (orange plus signs).

7.1.3 E. coli core model

For this model, in addition to all permutations of two dimensions, hulls were also calculated for all permutations of three dimensions using the four different algorithms. The growth function, acetate production, and succinate production were chosen as the dimensions for projection. Since the two-dimensional extreme points are contained within the set of three-dimensional extreme points, the following images will show the three-dimensional extreme points projected onto the plane spanned by two of its constituent dimensions in a similar manner to how the whole solution space is already projected onto the three dimensions of interest. This process results in extreme points which seem to lie inside the two-dimensional projection of the three-dimensional hull. Figure 7.3 shows how those seemingly incorrect extreme points are actually correct when considering all three dimensions.

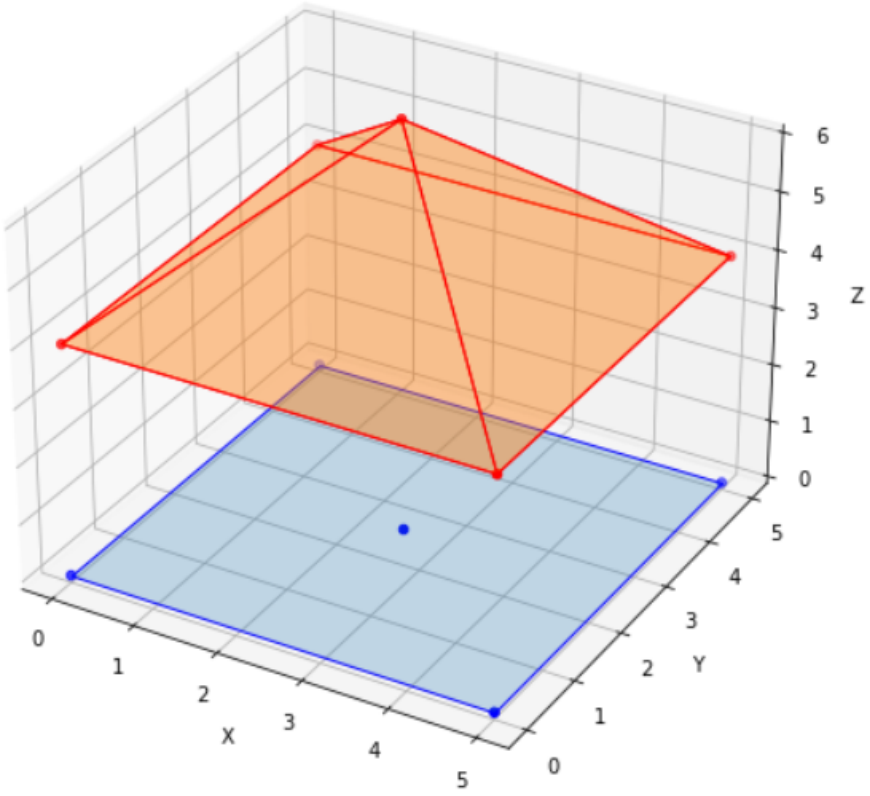

Figure 7.3: The orange pyramid with red vertices represents an example projection of an n-dimensional solution space into three dimensions. For visualization purposes, the three-dimensional hull is then further projected onto two dimensions resulting in the blue square. The tip of the pyramid is a correct extreme point in three dimensions but its two-dimensional projection looks like it lies within the boundaries of the convex hull similar to a false extreme point.

Figure 7.4 compares the results from the double and exact version of CHM tool leading to the following observations:

- (a) shows that the maximum growth rate is only achieved at one specific acetate production rate which is relatively close to the maximum rate of acetate production. However, this maximum growth rate is not much higher than the maximum growth rate achievable without any acetate production.
- Following from the last observation, this model suggests that for most of its variability, increasing acetate production can lead to only moderate increases in growth rate until the maximum growth rate is reached. After this point, further increases in acetate production drastically lower the organisms' ability to grow. This and the previous observation will be confirmed qualitatively but not quantitatively in a later calculation on the complete E. coli metabolic model.
- In comparison (b) shows that lower succinate production allows for continuously more biomass production.
- (c) and (d) show a similar trend between succinate and acetate, but there is significant variability in the level of succinate production which still allows for maximum acetate production.
- While the double precision version finds some of the exact extreme points with very high precision, a significant amount of them are also missed completely. Still, the general shape of the hull is captured very well for the dimensions biomass and acetate and still quite well for succinate and biomass.
- Only five of the extreme points which are only found by the exact version are close to a boundary of the hull segment which lies in the biomass acetate plane, while the rest lie close to a boundary in the biomass succinate plane. Interestingly, four out of those five are close to both boundaries while the other one is the only extreme point that is close to a biomass acetate boundary but not to a biomass succinate boundary.
- The last observation suggests that the double precision version was quite accurate for determining extreme points arising from acetate production but faced more difficulties when searching for extreme points caused by succinate production.
- All of the hard-to-find extreme points occur at growth rates that are close to or above half the maximum growth rate.
- (c) shows that for one permutation of the three chosen dimensions, all but one of the major extreme points in the plane spanned by acetate and succinate production was found by the double precision version. In contrast, this extreme point was found in (d) which used a different permutation. This shows how significant the order of dimensions can be for the double precision version, since (d) found the exact outer hull while (c) misses a significant portion just because it is missing a single extreme point.
- The previous observation reinforces the inadequacy of the number of extreme points found and missed as a parameter for judging the quality of an algorithm. The double precision hulls in (a), (b) and (d) capture the shape of the hull very well even though quite a few extreme points are missing while (c) shows how massive of an impact missing just one extreme point can have.

- More thorough inspection of images (c) and (d), which is corroborated multiple times in the many calculations made with the double precision version also suggests that the question of missing extreme points is more complicated than is apparent at first sight. All extreme points except for the one in the lower right are either found by both permutations or missed by both and occur close to a border. This category of missed extreme points is expected due to the limited accuracy. in contrast, outliers like the one in the lower right only happen occasionally and can lie farther away from the boundaries, which suggests that they arise through a different mechanism and should be seen as errors in the program as opposed to expected inaccuracies.

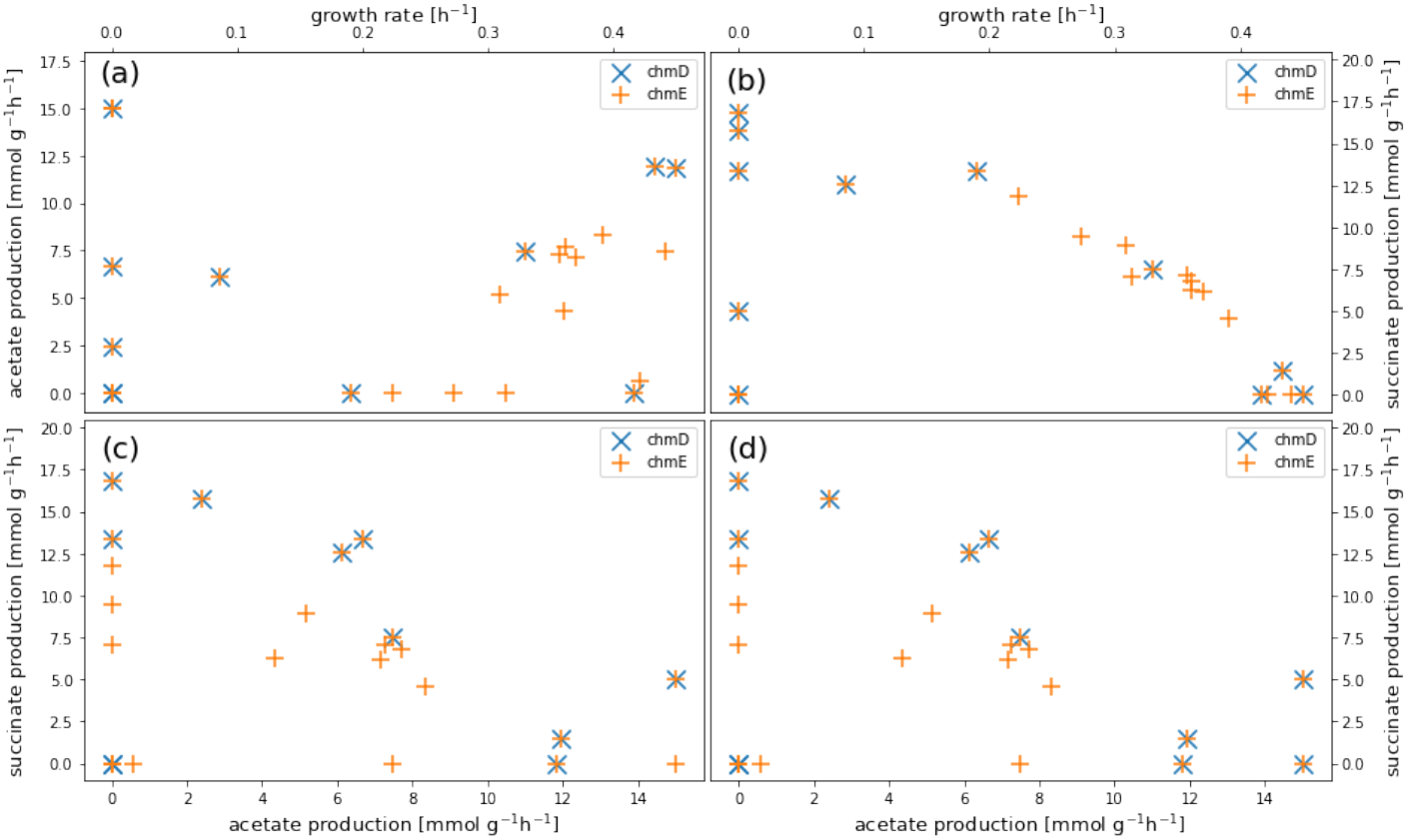

Figure 7.4: two-dimensional projections of the three-dimensional production envelope of the E. coli core model calculated with the double precision version of CHM tool (blue crosses) and the exact version of CHM tool (orange plus signs). (a) shows flux through the acetate synthesis reaction vs cell growth. (b) shows flux through the succinate synthesis reaction vs cell growth. (c) and (d) show flux through the succinate synthesis reaction vs flux through the acetate synthesis reaction for two different permutations of dimensions.

Figure 7.5 shows a comparison of the three-dimensional convex hull calculated with the exact version of CHM tool and the full 11-dimensional hull calculated with lrslib. The exact version calculates the same hull shape and all extreme points found are equal to those found with lrslib. The many additional extreme points found by lrslib are expected due to the much higher number of dimensions, which is also the reason for lrslibs slow calculation speed. From a biological perspective, this image suggests that there are many growth-supporting phenotypes which do not exhibit any acetate production and also quite a few possibilities to produce acetate without any cell growth. Figures 7.6 and 7.7 show the other sections of the three-dimensional hull, confirming CHM tools accuracy.

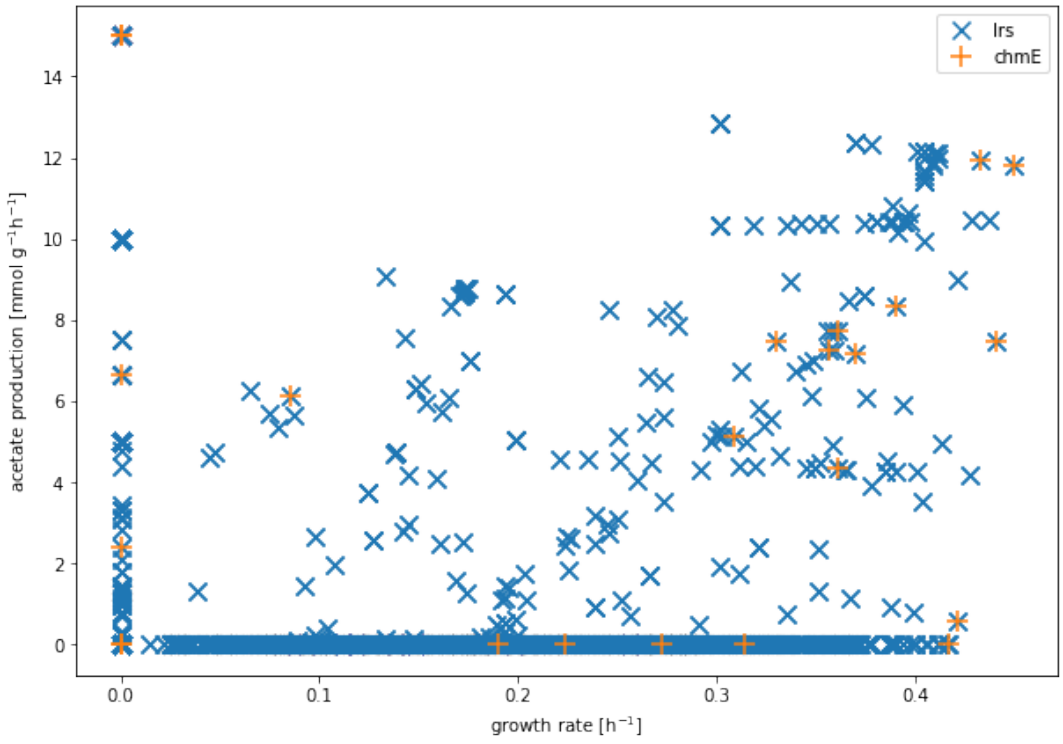

Figure 7.5: Production envelope of flux through the acetate synthesis reaction vs flux through the biomass reaction of the E. coli core model calculated with lrslib (blue crosses) and the exact version of CHM tool (orange plus signs).

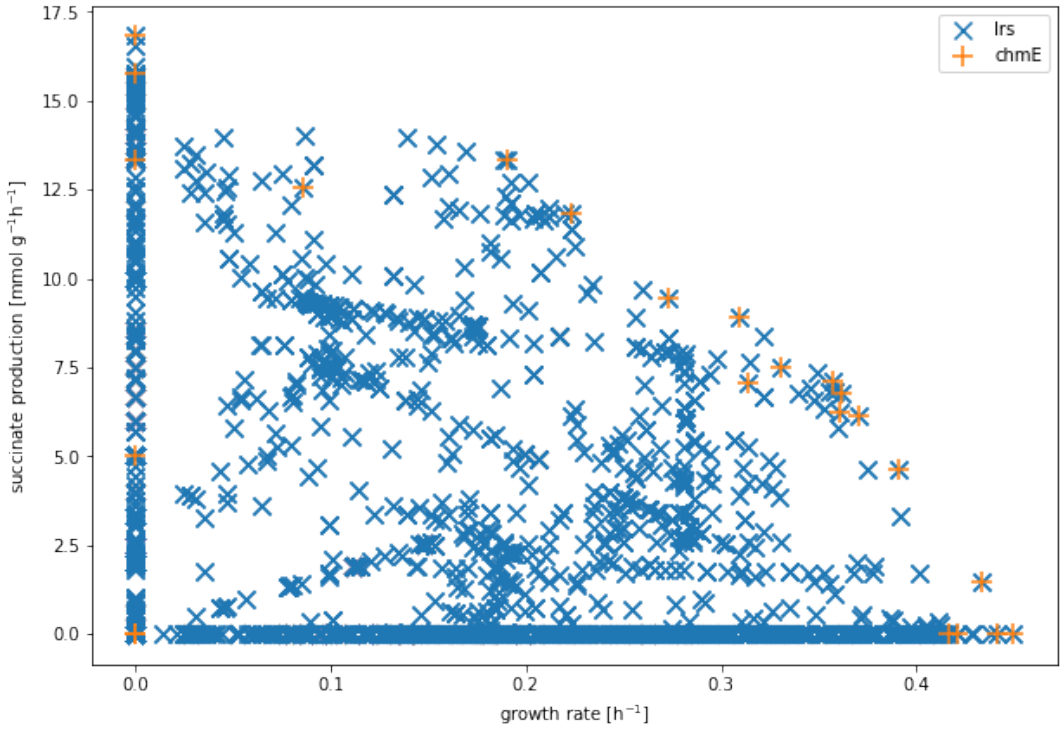

Figure 7.6: Production envelope of flux through the succinate synthesis reaction vs flux through the biomass reaction of the E. coli core model calculated with lrslib (blue crosses) and the exact version of CHM tool (orange plus signs).

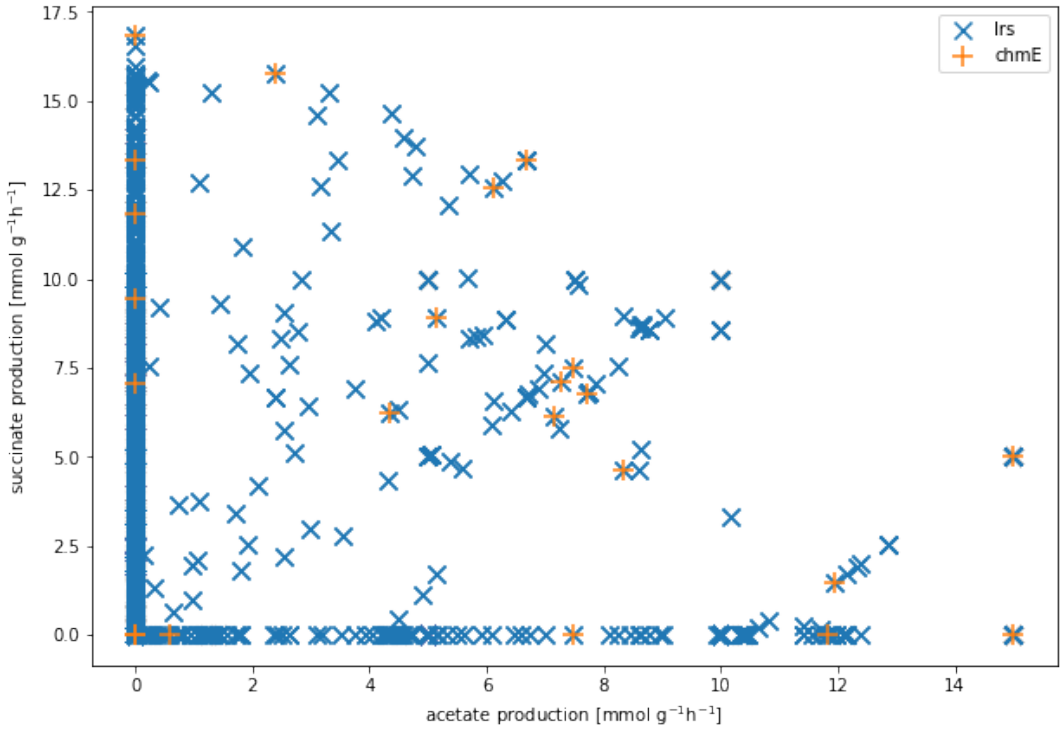

Figure 7.7: Production envelope of flux through the succinate synthesis reaction vs flux through the acetate reaction of the E. coli core model calculated with lrslib (blue crosses) and the exact version of CHM tool (orange plus signs).

Figures 7.8, 7.9 and 7.10 show the same lrslib results as before but compared to the chlib solutions. It is immediately apparent that chlib produced a huge unexplained outlier in biomass production in addition to finding far less extreme points than CHM tool and some of those having low congruence with the lrslib extreme points. While the chlib hull in figure 7.10 is closer to the lrslib hull, it is still missing quite a few of the outer extreme points.

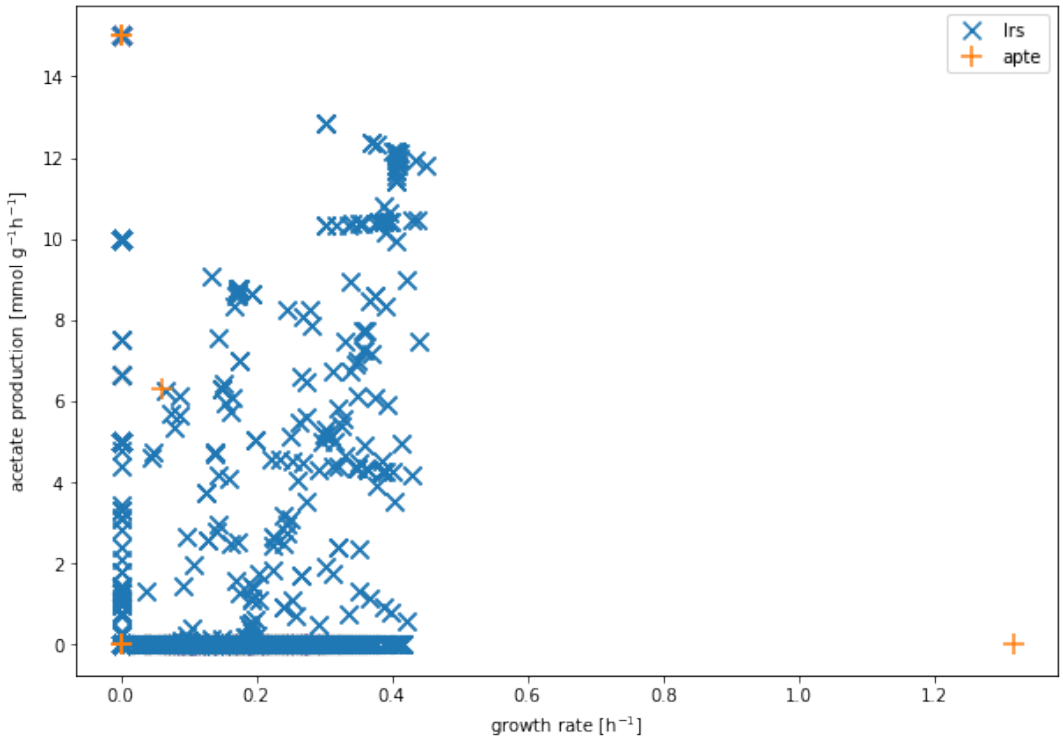

Figure 7.8: Production envelope of flux through the acetate synthesis reaction vs flux through the biomass reaction of the E. coli core model calculated with lrslib (blue crosses) and chlib (orange plus signs).

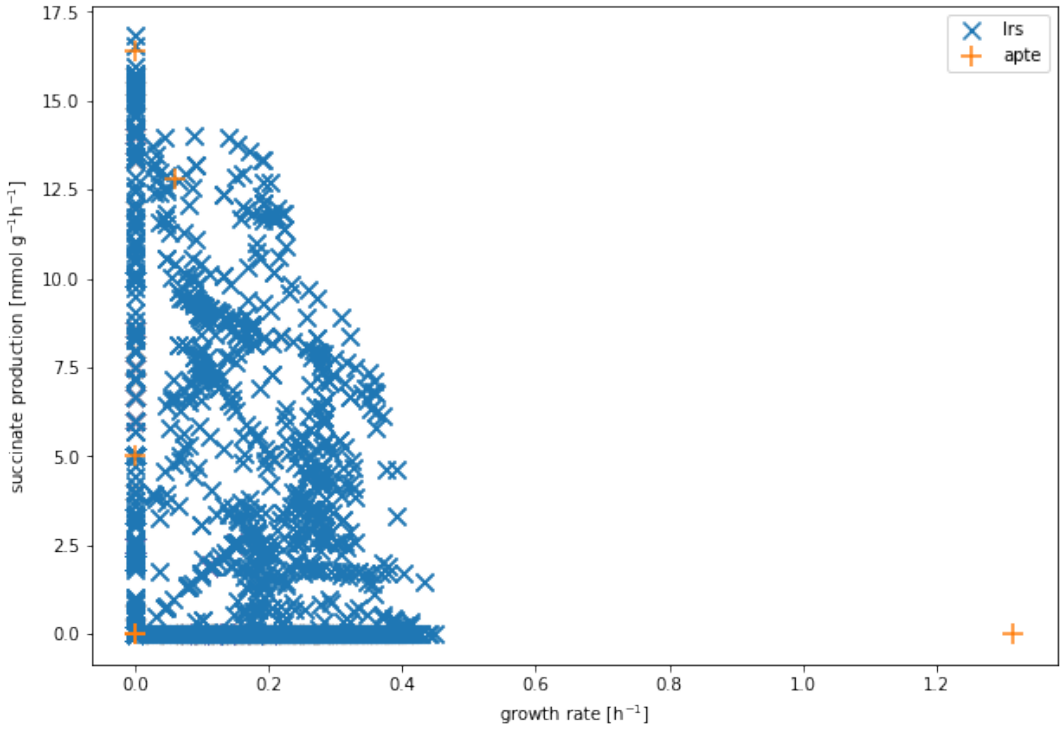

Figure 7.9: Production envelope of flux through the succinate synthesis reaction vs flux through the biomass reaction of the E. coli core model calculated with lrslib (blue crosses) and chlib (orange plus signs).

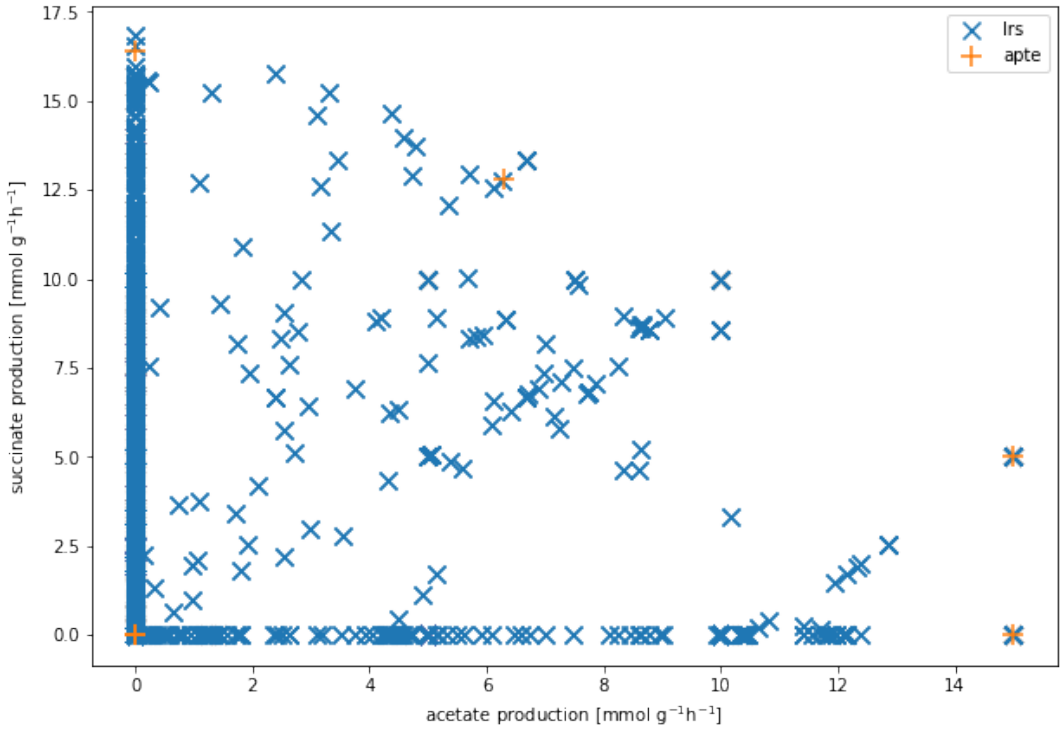

Figure 7.10: Production envelope of flux through the succinate synthesis reaction vs flux through the acetate reaction of the E. coli core model calculated with lrslib (blue crosses) and chlib (orange plus signs).

7.1.4 Conclusion

The calculations on the three different models suggest that lrslib is least prone to errors and should be used as a benchmark for accuracy, chlib generally has the same level of accuracy for smaller models but sometimes produces very significant unexpected outliers. In bigger models, this flaw is even more significant and the accuracy drops further. The CHM tool exact version is comparable to lrslib and does not have the same error problem as chlib and the double precision version of CHM tool produces qualitatively good results at much higher calculation speeds.

7.2 Applying CHM tool to a genome-scale model

The well-established E. coli model iJO1366 from Orth et al[106] was used to test CHM tools' suitability for common metabolic modeling applications as well as for the calculation of a variety of parameters for benchmarking purposes. Environmental parameters were set to low oxygen availability, glucose as the main growth substrate, and an enforced ATP maintenance demand.

7.2.1 Runtime and extreme points

CHM tools double precision and exact versions were used to calculate convex hulls projected onto two to five dimensions. The chosen dimensions were the biomass reaction as well as reactions for the production of acetate, succinate, lactate, and formate. Figure 7.11 shows that for both versions there is only a moderate, approximately linear increase in the number of extreme points when increasing the number of dimensions. In contrast, Figure 7.12 shows that the increase in runtime needed for the calculation is approximately exponential.

Another important distinction is that the double precision variant is significantly faster while still finding a similar number of extreme points. Especially at higher numbers of dimensions the double precision version is faster by multiple orders of magnitude, only needing 110 seconds to calculate the 4-dimensional hull, while the exact version took 30 hours. This difference in calculation speed is the reason that a 5-dimensional hull was not calculated with the exact version due to the excessive time needed.

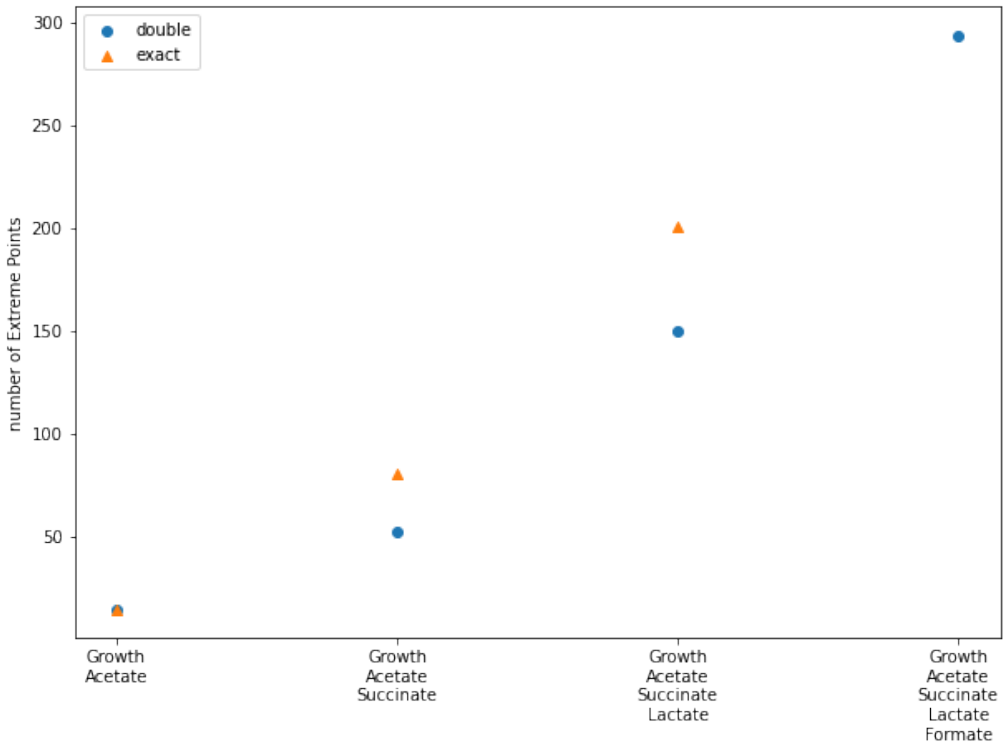

Figure 7.11: Number of extreme points versus the number of dimensions for the hull. While double precision calculations were feasible in five dimensions, the exact version was only used for up to 4 dimensions due to slower calculation speeds. Extreme points were calculated using the genome-scale E. coli model.

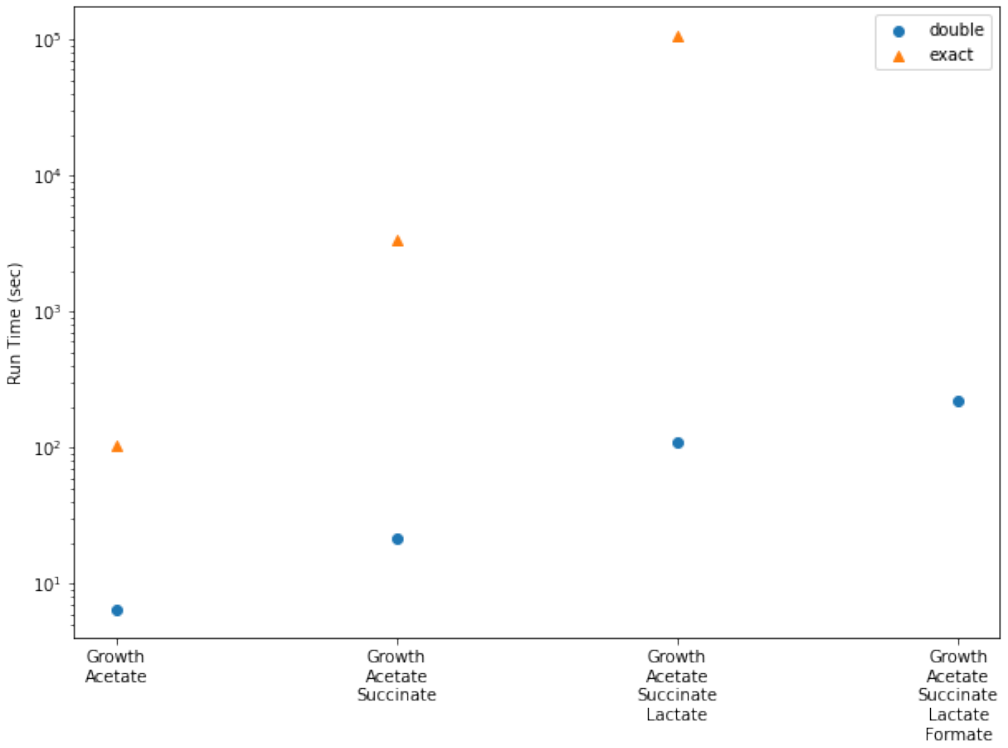

Figure 7.12: Calculation runtime versus the number of dimensions for the hull. While double precision calculations were feasible in five dimensions, the exact version was only used for up to 4 dimensions due to slower calculation speeds. Extreme points were calculated using the genome-scale E. coli model.

7.2.2 Nutrients and extreme points

CHM tool was used to calculate the convex hulls projected onto two to four dimensions for a variety of growth media, and the number of extreme points in those hulls is shown in figure 7.13 and the zoomed-in version in figure 7.14. All growth media contain glucose but seven of them also use glycerin, pyruvate, glutamate, fumarate, malate, or citrate as additional growth substrates. While there are a few inconsistencies, some general trends can be observed:

- A higher number of dimensions also results in a higher number of extreme points. This has to be the case for correct solutions since any 2D extreme point is necessarily also present in 3D. At the very least an increase in dimensions can lead to the number of extreme points staying the same if the new reaction always has zero flux, but it can't get any lower.

- Similarly, at the same number of dimensions, the exact version finds an equal or higher number of extreme points than the double precision version. This is a good sign as the exact version is expected to be more accurate and should therefore not miss any extreme points which are found by the double precision version.

- This last trend is less universal, but generally, those nutrient combinations which lead to a higher number of extreme points found in lower dimensions also lead to a higher number with more dimensions.

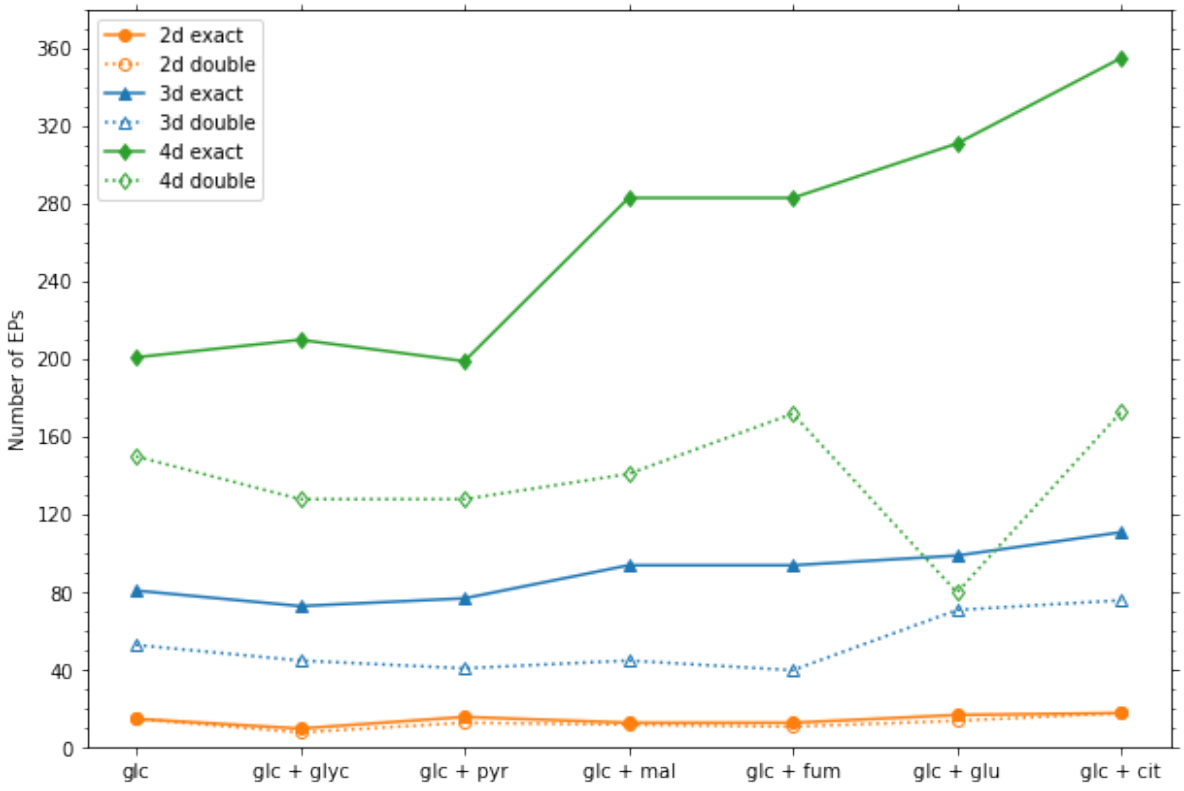

Figure 7.13: Number of extreme points dependent on nutrient combinations and number of dimensions for the convex hull projection. Extreme points were calculated using the genome-scale E. coli model.

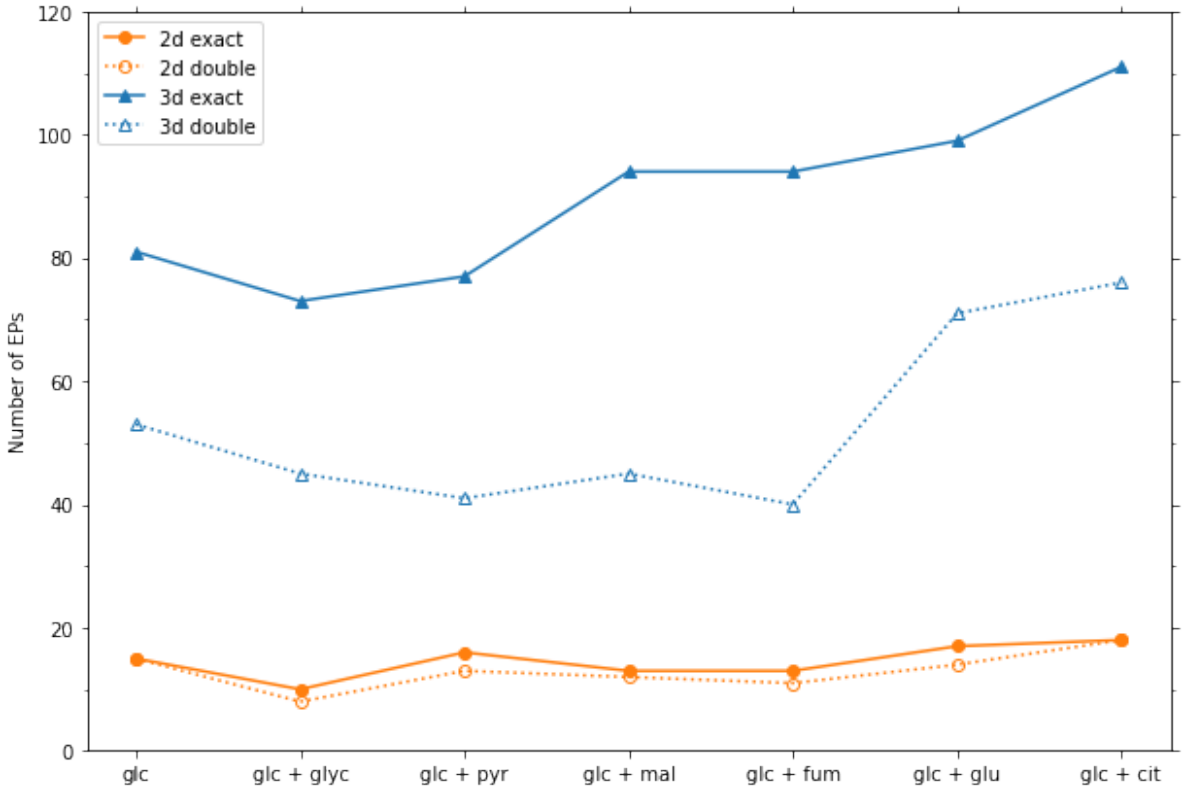

Figure 7.14: Zoomed in view of the 2D and 3D results from figure 7.13 for better visibility

For the exact version, the order of the dimensions does not matter but for the double precision version in 3D and 4D, the number of extreme points depends on the order in which the dimensions are provided. Figure 7.15 was calculated by initializing CHM tool not only with dimensions in ascending order ([1,2,3]) but in addition with all other permutations ([1, 2, 3], [1, 3, 2], [2, 1, 3], [2, 3, 1], [3, 1, 2], [3, 2, 1]). While this dependence on permutations is sub-optimal behavior, the previously observed trends still mostly hold true. Only one double precision hull for growth on glucose and citrate has fewer extreme points in four dimensions than the two-dimensional hulls.

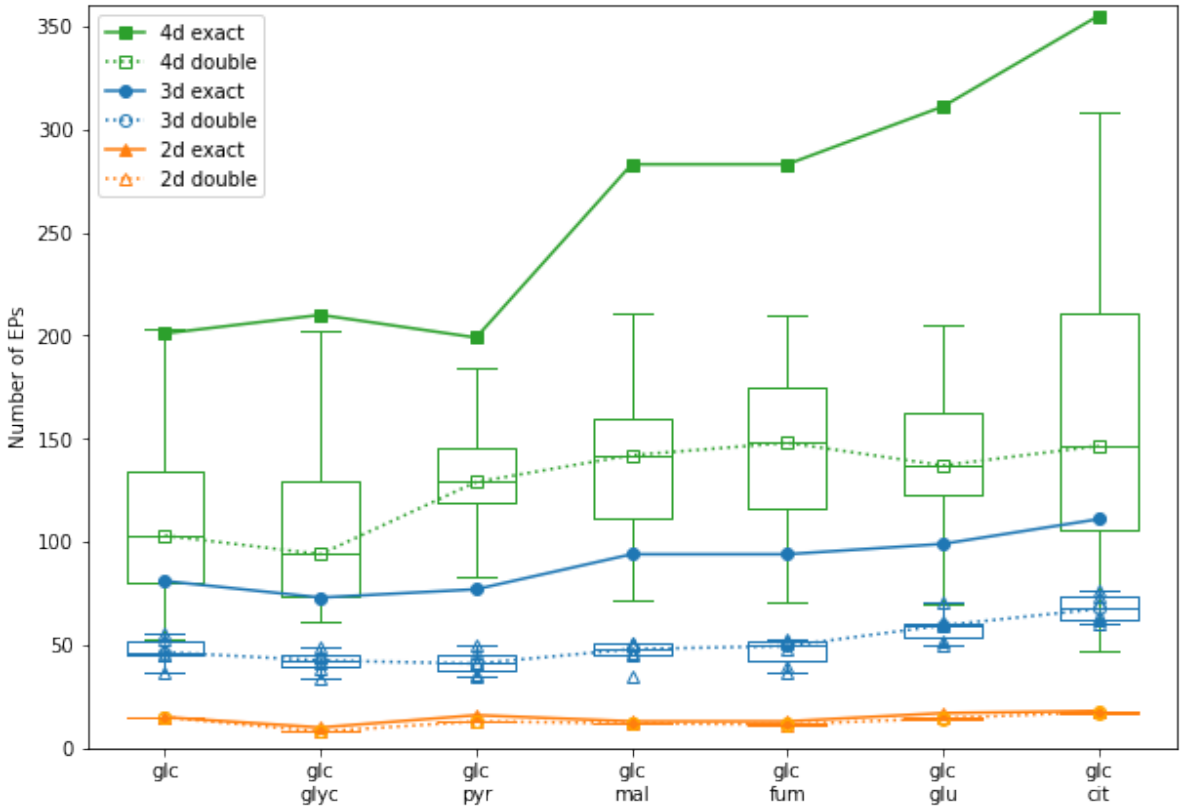

Figure 7.15: Number of extreme points dependent on nutrient combinations and number and permutation of dimensions. Extreme points were calculated using the genome-scale E. coli model.

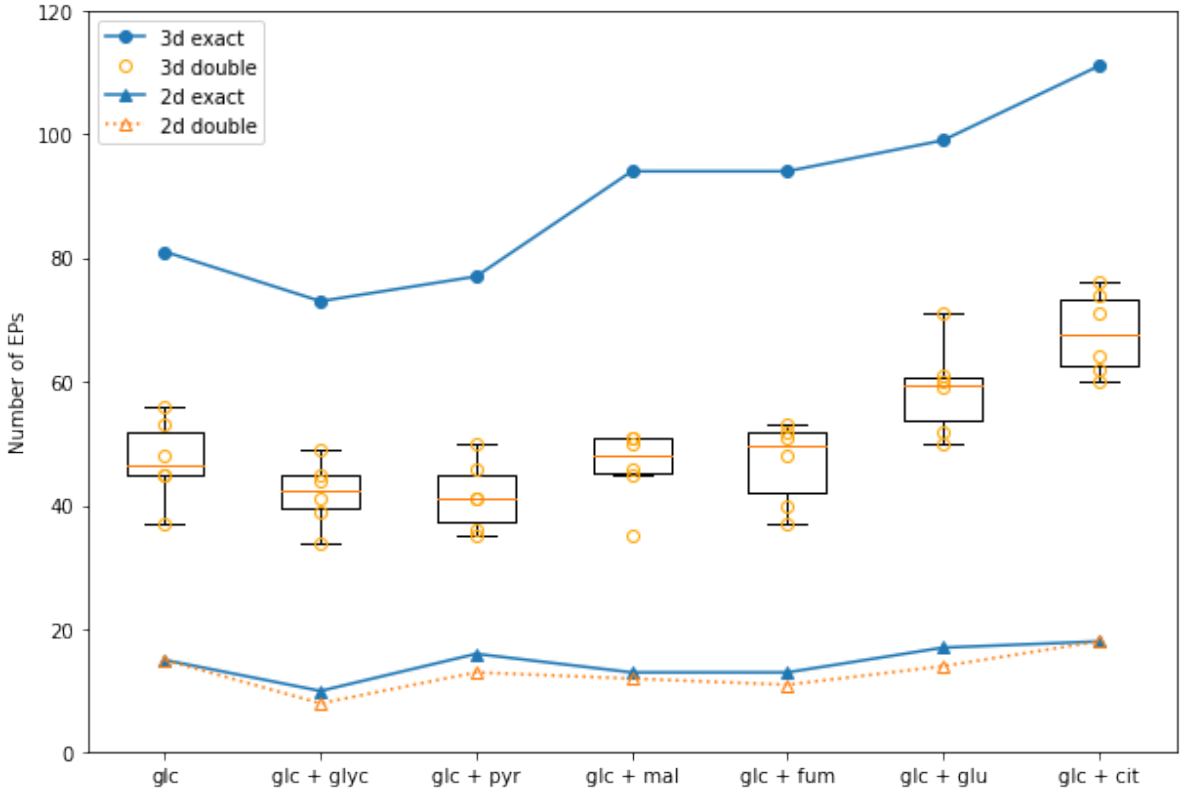

Figure 7.16: Zoomed in view of the 2D and 3D results from figure 7.15 for better visibility

7.2.3 Nutrients and runtime

Figures 7.17 and 7.18 show the associated runtimes for the previous calculations, with some similar trends, but also a few new ones:

- Generally, a higher number of extreme points tends to necessitate longer calculation times, which also means that previous correlations with dimensionality and nutrient combinations are still broadly observed.
- While going from two to three dimensions only leads to a moderate jump in the number of extreme points, The differences in calculation time are much more significant.
- With such high runtimes, the speed advantage of the double precision version over the exact version becomes quite significant. Notably, for most permutations of four dimensions, the double precision implementation is still faster than the exact version in just two dimensions.

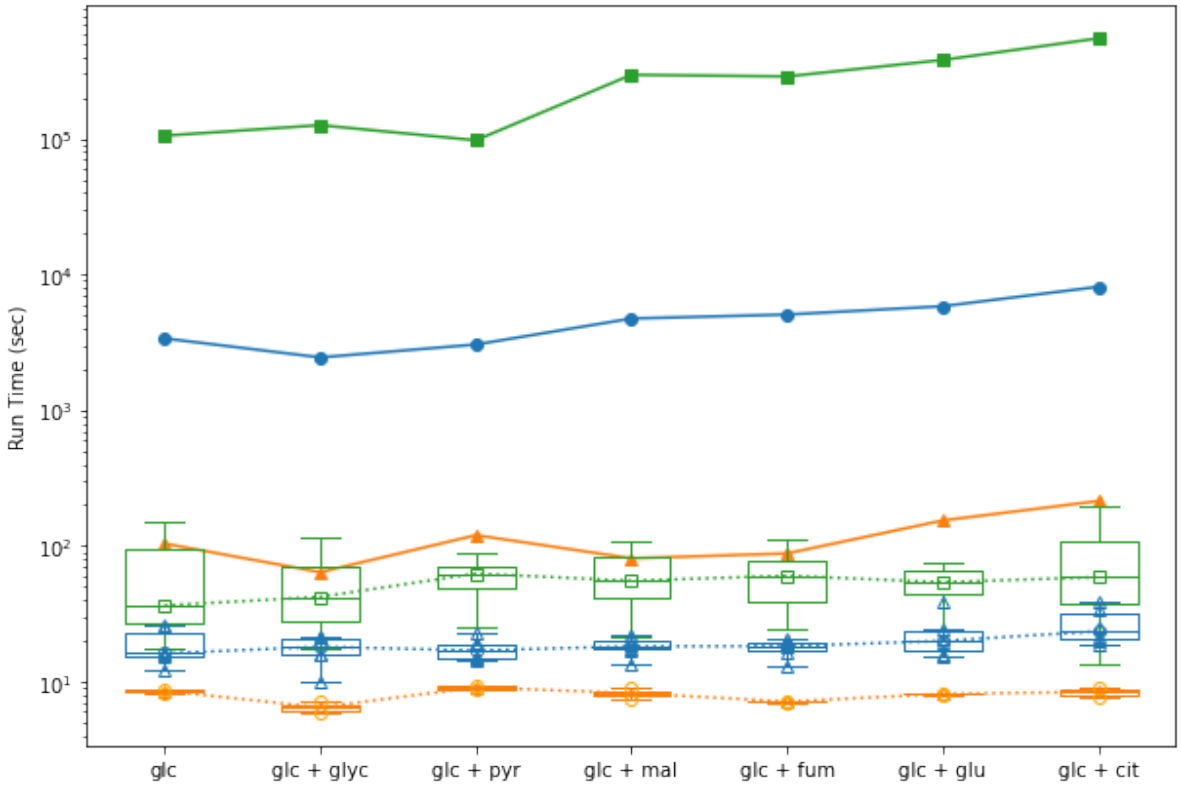

Figure 7.17: Calculation runtime dependent on nutrient combinations and number and permutation of dimensions. Extreme points were calculated using the genome-scale E. coli model.

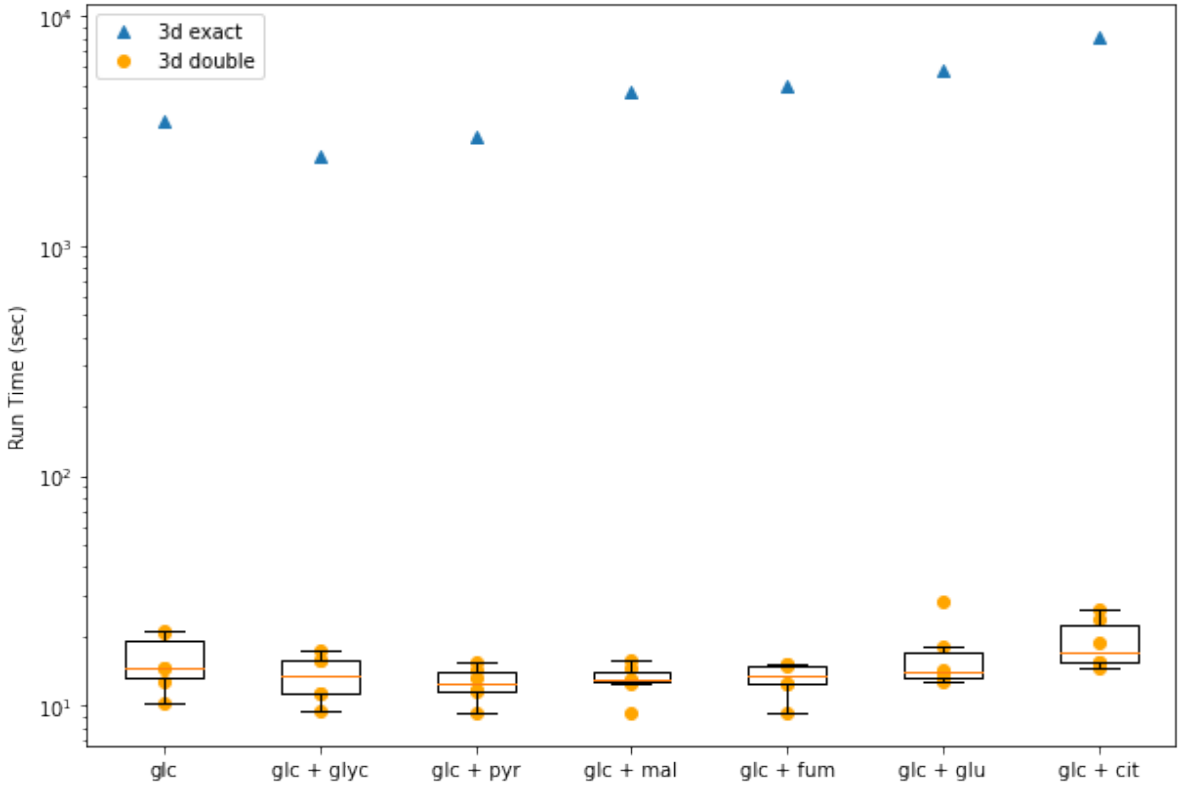

Figure 7.18: Zoomed in view, showing only the 3D results from figure 7.17

7.2.4 Hull volumes

As a final measure of comparison, the hull volumes for both versions of CHM tool were calculated using lrslib. Figures 7.19 and 7.20 show the ratio between the volumes calculated with both versions respectively, where a ratio of one means that both hulls have exactly the same size. It can be observed that, as expected, the double precision hulls are never bigger than the exact hulls and the average difference in size is less than 1% with only a few outliers in three dimensions which differ by around 4% , a few bigger outliers in four dimensions which are around 10% smaller and just one extreme outlier which is 40% smaller.

Interestingly, large outliers often have the same volume as other large outliers, due to them missing the same extreme points. This corroborates the observation made for figure 7.4 (c) and (d) that there are expected numerical inaccuracies due to the limited number of post decimal places that miss extreme points close to the convex hulls boundaries and lead to many small discrepancies in volume and another distinct source of unexpected errors which completely misses extreme points far away from any other extreme points or boundaries and lead to the double precision version missing significant parts of the hull and therefor more drastically lowering the volume.

These two separate sources for discrepancies also explain why there is no correlation between the average error margin and the occurrence of large outliers. This phenomenon can be observed in figure 7.19 where the permutations of glucose and glutamate have the second lowest error margin but also the largest outlier.

In summary, it can be surmised that for 2 dimensions the double precision hull is an almost exact replica of the exact version results, with an average difference of only 0.000226%. At higher dimensionalities, the difference can be higher for some permutations of the chosen dimensions but on average it is still very low when multiple permutations are calculated, which is quite feasible due to the huge increase in calculation speed over the exact version.

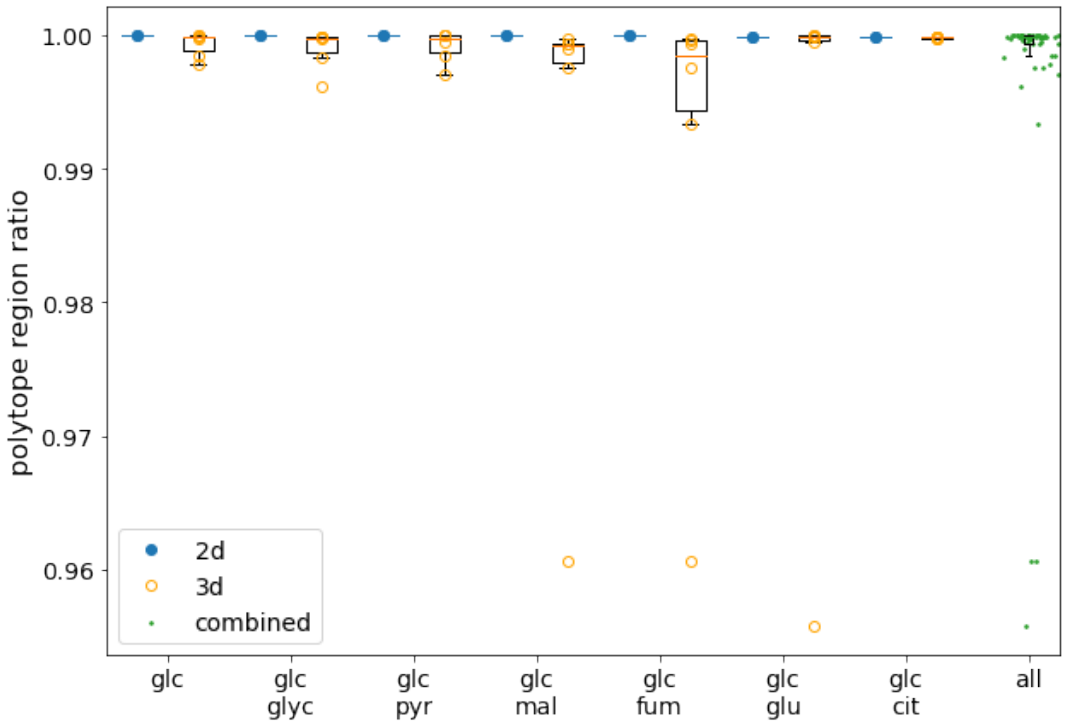

Figure 7.19: Ratio between exact and double precision hull size dependent on nutrient combinations for different permutations of two and three dimensions. A value of one corresponds to no difference in volume, while lower values result from the double precision hull having less volume due to missing or inaccurately positioned extreme points. Extreme points were calculated using the genome-scale E. coli model.

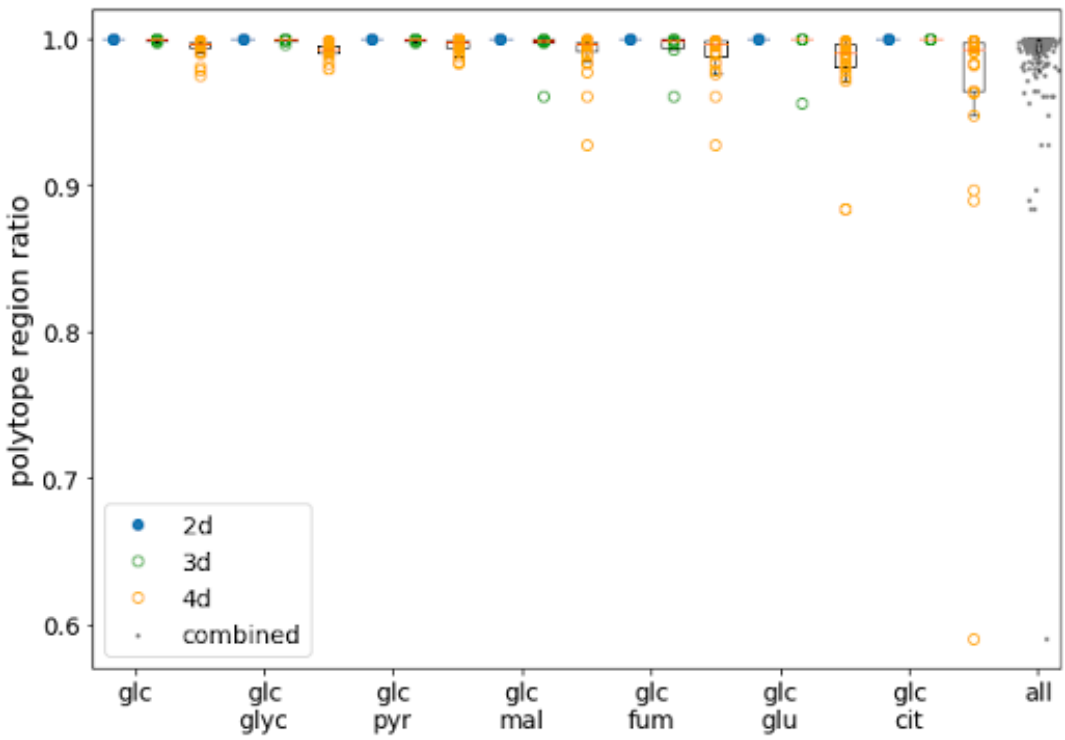

Figure 7.20: Same data as in figure 7.19 but with the addition of convex hull volumes resulting from different permutations of four dimensions.

7.2.5 Example production envelopes

two-dimensional production envelopes were calculated to test CHM tool in a common example use case. The polytope was projected onto the biomass and the acetate production reaction and the calculation was repeated for various growth substrates. Glucose was chosen as the main substrate with a maximum flux rate of 10 mmol/gDW/h and for the other calculations, the model was additionally provided with either glycerin, pyruvate, glutamate, fumarate, malate, or citrate also with a maximum flux rate of 10 mmol/gDW/h.

Figure 7.21 shows the results for the exact and the double precision variant and Figure 7.22 shows a zoomed-in area to show how well both areas overlap. The following observations were made:

- With the chosen parameter set, the exact and double precision variants are highly congruent for all nutrient media. Those extreme points which the double precision version does not find as well as redundant points which are not found by the exact variant, lie very close to the outer hull, therefore only having a negligible impact on accuracy.

- Out of the additional substrates, citrate, and glutamate have the highest impact on acetate production increasing it by about 70% from 30 mmol/gDW/h to 52 mmol/gDW/h compared to between 41 and 46 mmol/gDW/h for the other nutrients. Still, an increase by 70% when doubling the total substrate intake means that all additional substrates are less yield efficient than glucose.

- Similar to acetate production, citrate and glutamate are also the most biomass yield efficient of the additional substrates but with an increase of about 40%, the difference is significantly smaller than the difference for growth rate.

- The general shape for all hulls is similar in that high growth rates are possible for low to moderate acetate production while high acetate production quickly limits the organisms' ability to grow. For all growth media there exists an optimum for biomass production at around half of the maximum possible acetate production.

- The biggest differences in hull shape for the various growth substrates occur around the area of maximum growth rate. Especially noteworthy is that glycine achieves its growth maximum at comparatively low acetate yield, while for glutamate the maximum coincides with high acetate yield. Interestingly, citrate is the only substrate that shows significant variability in acetate production while still maintaining maximum growth rates.

- Figure 7.22 (a) shows that the double precision variant finds extreme points in close proximity to all of the exact precision extreme points but also finds an additional point that is quite far away from any corresponding exact point but still very close to the outer hull.

- Figure 7.22 (b) shows that some areas which seem like just one extreme point actually contain multiple extreme points when zooming further in. It also shows that the double precision variant did not find all of those extreme points and therefore the hulls' outer line is slightly farther inward, resulting in a lower volume of the solution space.

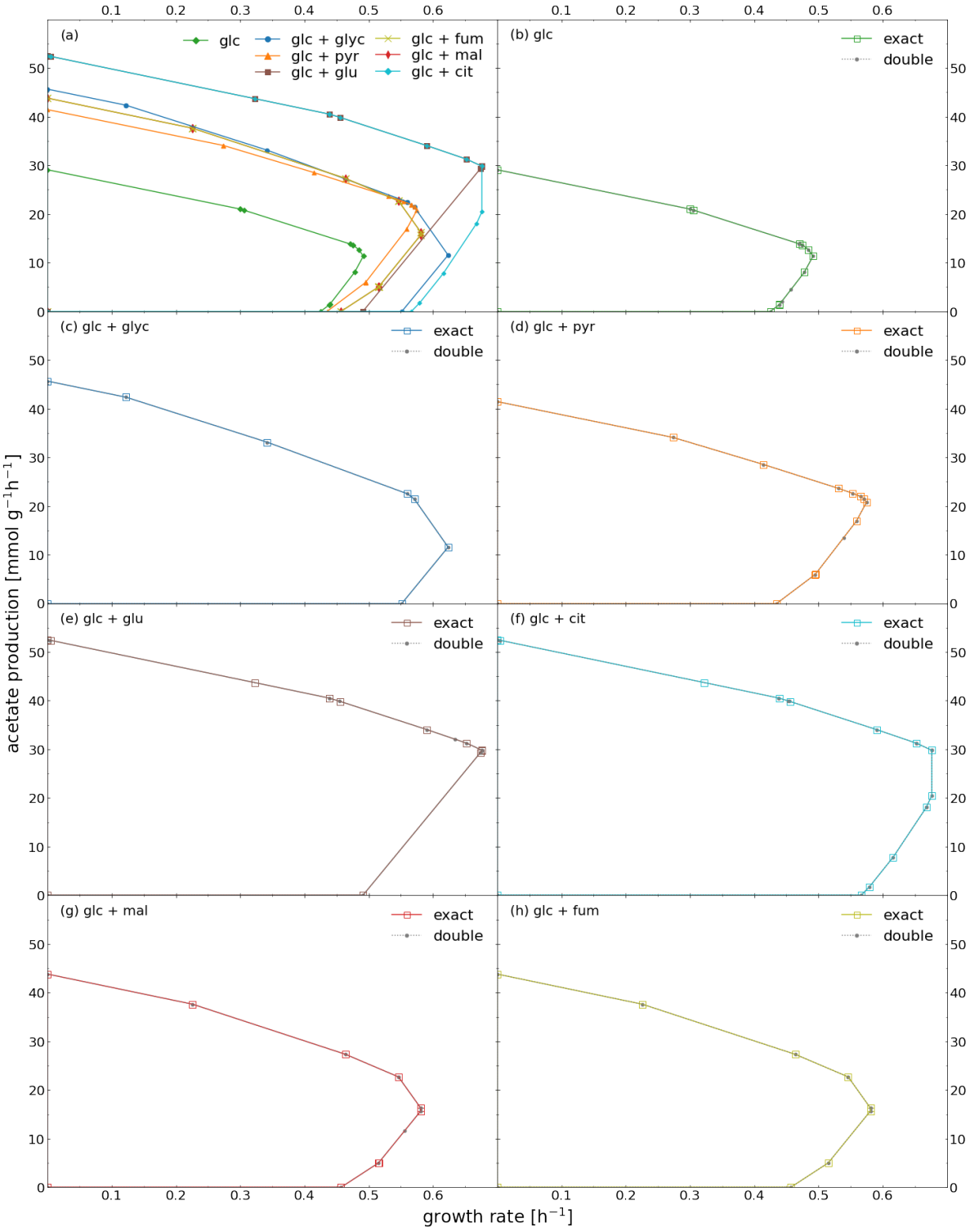

Figure 7.21: two-dimensional convex hulls projected onto acetate and growth rate for various combinations of nutrients. Panel (a) compares results from the exact version while (b) to (h) compare exact version results (full lines with open squares) to those from the double precision version (dotted lines with full circles). Extreme points were calculated using the genome-scale E. coli model.

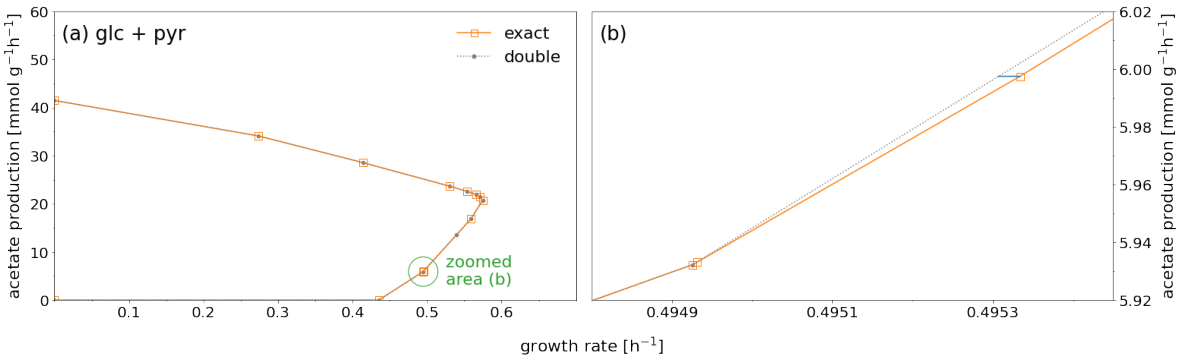

Figure 7.22: two-dimensional convex hull projected onto acetate glucose and pyruvate as nutrients. Panel (a) is identical to figure 7.21 (d), comparing exact version results (full lines with open squares) to those from the double precision version (dotted lines with full circles) and panel (b) shows a highly zoomed-in view of panel (a), containing three exact version extreme points and one from the double precision version.

8 Conclusion

Comparison of the four different convex hull projection tools concluded in the following insights:

- lrslib proved to reliably yield the best results and was the least prone to errors but it is completely infeasible for use on larger networks. This results from its inability to project convex hulls on less than the full set of reactions, which means that runtime scales very badly with model size. As an example, it took lrslib around two hours to calculate the hull of the E. coli core model.

- In addition lrslib provides the additional functionality to calculate multidimensional volumes of convex hulls. Critically though, this function is separate from the calculation of the hull itself which means that it can also be applied to hulls generated with the other tools.

- While chlib boasts quick calculation times and can project hulls onto an arbitrary set of dimensions, it is greatly hampered by its own set of problems. The main limitation is that the codebase is quite old and hard to get to work. Even after lots of troubleshooting, the program is still not quite stable leading to early terminations. While for small models like the two toy models this just means that the calculation might have to be restarted a few times, terminations occur at a much higher frequency with increased model size. In the case of the complete E. coli model, not a single successful calculation was achieved in over 100 000 restarts. A second, less significant error is the admittedly quite infrequent occurrence of very large outliers. While this only occurred in one calculation for the E. coli core model it made the result almost unusable.

- In contrast to the other tools both versions of CHM tool could be used to calculate hulls of larger networks like the genome-scale E. coli model.

- Both versions proved more stable than chlib but still, the program has to be terminated sometimes when entering an endless loop. In contrast to chlib this error did not occur randomly though but instead only for specific permutations. This means that in cases where CHM tool fails, switching around the order in which dimensions are provided often yields solutions. While the mathematical reason that sometimes produces these endless loops could not be determined, it was possible to find the point of its origin in the code and it could be modified to self-terminate when the problem occurs.

- The exact version of CHM tool showed very similar results to lrslib and therefore makes for a suitably accurate but faster alternative for large models. It could be determined that a big contributor to its runtime was its usage of the sympy library compared to the double precision version which uses the numpy library.

Since sympy is not strictly necessary for the calculation, a switch to numpy could yield further increases in calculation speed.

- While the double precision version of CHM tool has somewhat lower accuracy compared to the exact version, this is balanced out by a huge increase in calculation speed, especially when hulls are projected onto more than three or four dimensions.

- Inaccuracies of the double precision version could be distinguished between two categories. An expected source of inaccuracy is the finite number of post-decimal places used for floating point calculations. While the effect was measurable it is neither qualitatively nor quantitatively relevant for expected use cases. Such minor inaccuracies are made even less relevant by the fact that the accuracy of production envelopes ultimately depends on the stoichiometry of the biomass reaction which can only be measured experimentally with limited accuracy anyways in addition to significant variance introduced by environmental conditions. More significant but also less frequent is the second source of errors which can cause somewhat larger sections of the hull to be missed. While this loss can be quite large, the biggest loss in volume encountered during testing was 40%, its very infrequent occurrence means that such results can be filtered out quite easily by calculating the same hull for a few different permutations of dimensions and discarding outliers. This process is enabled due to the very low run times of the double precision version. It could not be determined if this second error source is related to the source of the occasional endless loops.

- Interestingly neither error source seems to result in 'false positive' results but only in missing some of the correct solutions. Arguably it can be seen as preferable for some applications as well as for general research to not completely discover all possible capabilities rather than suggesting biological functionalities which are far removed from nature.

- All in all while both versions of CHM tool still show venues for improvement they are both viable alternatives to the other tested programs, with potential applications which are not feasible with the tested alternatives.

9 Outlook

After this assessment of CHM tools' capabilities and finding it suitable for the calculation of convex hulls of genome-scale models in up to six dimensions, it is ready for applications where its significant increase in speed and scope over other methods is necessary. One such application is the study of nutritional relationships between different species in microbial communities or the extensions of production envelopes to larger model sizes in fields where they are already in use. Additionally, the use of convex hull methods like CHM tool could be introduced to the vast array of applications that are currently tackled by fba methods like flux variability analysis where they provide a similar kind of insight but at a much larger scale.

While the still remaining obstacles do not pose a major barrier to the use of CHM tool, especially in comparison to the limitations of lrslib and chlib, they still provide an avenue to further improve calculation speed for the exact version as well as the double precision versions ability to consistently provide accurate results. Another priority would be finding the source of the infinite loops occasionally encountered mostly in the double precision version but also to a lesser extent in the exact version.

Bibliography

[1] Steffen Klamt, Oliver Hädicke, and Axel von Kamp. Stoichiometric and constraint-based analysis of biochemical reaction networks. In *Large-scale networks in engineering and life sciences*, pages 263–316. Springer, 2014.

[2] Stefan Bornholdt. Less is more in modeling large genetic networks. *Science*, 310(5747):449–451, 2005.

[3] Ya-Jou Chen, Pok Man Leung, Jennifer L Wood, Sean K Bay, Philip Hugenholtz, Adam J Kessler, Guy Shelley, David W Waite, Ashley E Franks, Perran LM Cook, et al. Metabolic flexibility allows bacterial habitat generalists to become dominant in a frequently disturbed ecosystem. *The ISME journal*, 15(10):2986–3004, 2021.

[4] Franck Carbonero, Brian B Oakley, and Kevin J Purdy. Metabolic flexibility as a major predictor of spatial distribution in microbial communities. *PLoS One*, 9(1):e85105, 2014.

[5] Anat Bren, Junyoung O Park, Benjamin D Towbin, Erez Dekel, Joshua D Rabinowitz, and Uri Alon. Glucose becomes one of the worst carbon sources for e. coli on poor nitrogen sources due to suboptimal levels of camp. *Scientific reports*, 6(1):1–10, 2016.

[6] Jean-Christophe Lachance, Colton J Lloyd, Jonathan M Monk, Laurence Yang, Anand V Sastry, Yara Seif, Bernhard O Palsson, Sébastien Rodrigue, Adam M Feist, Zachary A King, et al. Bofdat: Generating biomass objective functions for genome-scale metabolic models from experimental data. *PLoS computational biology*, 15(4):e1006971, 2019.

[7] Matthew Scott, Carl W Gunderson, Eduard M Mateescu, Zhongge Zhang, and Terence Hwa. Interdependence of cell growth and gene expression: origins and consequences. *Science*, 330(6007):1099–1102, 2010.

[8] Anthony P Burgard, Evgeni V Nikolaev, Christophe H Schilling, and Costas D Maranas. Flux coupling analysis of genome-scale metabolic network reconstructions. *Genome research*, 14(2):301–312, 2004.

[9] Thomas Pfeiffer, I Sanchez-Valdenebro, JC Nuno, Francisco Montero, and Stefan Schuster. Metatool: for studying metabolic networks. *Bioinformatics (Oxford, England)*, 15(3):251–257, 1999.

[10] Jeremy S Edwards, Rafael U Ibarra, and Bernhard O Palsson. In silico predictions of escherichia coli metabolic capabilities are consistent with experimental data. *Nature biotechnology*, 19(2):125–130, 2001.

[11] Stephen S Fong, Anthony P Burgard, Christopher D Herring, Eric M Knight, Frederick R Blattner, Costas D Maranas, and Bernhard O Palsson. In silico design and adaptive evolution of escherichia coli for production of lactic acid. *Biotechnology and bioengineering*, 91(5):643–648, 2005.

[12] Rafael U Ibarra, Jeremy S Edwards, and Bernhard O Palsson. Escherichia coli k-12 undergoes adaptive evolution to achieve in silico predicted optimal growth. *Nature*, 420(6912):186–189, 2002.

[13] Cong T Trinh and Friedrich Srienc. Metabolic engineering of escherichia coli for efficient conversion of glycerol to ethanol. *Applied and environmental microbiology*, 75(21):6696–6705, 2009.

[14] Pornkamol Unrean, Cong T Trinh, and Friedrich Srienc. Rational design and construction of an efficient e. coli for production of diapolycopendioic acid. *Metabolic engineering*, 12(2):112–122, 2010.

[15] Harry Yim, Robert Haselbeck, Wei Niu, Catherine Pujol-Baxley, Anthony Burgard, Jeff Boldt, Julia Khandurina, John D Trawick, Robin E Osterhout, Rosary Stephen, et al. Metabolic engineering of escherichia coli for direct production of 1, 4-butanediol. *Nature chemical biology*, 7(7):445–452, 2011.

[16] Jens Nielsen. It is all about metabolicfluxes. *Journal of bacteriology*, 185(24):7031–7035, 2003.

[17] Steffen Klamt, Stefan Schuster, and Ernst Dieter Gilles. Calculability analysis in underdetermined metabolic networks illustrated by a model of the central metabolism in purple nonsulfur bacteria. *Biotechnology and bioengineering*, 77(7):734–751, 2002.

[18] Steffen Klamt, Georg Regensburger, Matthias P Gerstl, Christian Jungreuthmayer, Stefan Schuster, Radhakrishnan Mahadevan, Jürgen Zanghellini, and Stefan Müller. From elementary flux modes to elementary flux vectors: Metabolic pathway analysis with arbitrary linear flux constraints. *PLoS computational biology*, 13(4):e1005409, 2017.

[19] Cong T Trinh, Ross Carlson, Aaron Wlaschin, and Friedrich Srienc. Design, construction and performance of the most efficient biomass producing e. coli bacterium. *Metabolic engineering*, 8(6):628–638, 2006.

[20] Vasiliy A Portnoy, Daniela Bezdan, and Karsten Zengler. Adaptive laboratory evolution—harnessing the power of biology for metabolic engineering. *Current opinion in biotechnology*, 22(4):590–594, 2011.

[21] Nathan E Lewis, Harish Nagarajan, and Bernhard O Palsson. Constraining the metabolic genotype–phenotype relationship using a phylogeny of in silico methods. *Nature Reviews Microbiology*, 10(4):291–305, 2012.

[22] Nathan D Price, Jennifer L Reed, and Bernhard Ø Palsson. Genome-scale models of microbial cells: evaluating the consequences of constraints. *Nature Reviews Microbiology*, 2(11):886–897, 2004.

[23] Guido Melzer, Manely Eslahpazir Esfandabadi, Ezequiel Franco-Lara, and Christoph Wittmann. Flux design: In silico design of cell factories based on correlation of pathway fluxes to desired properties. *BMC systems biology*, 3(1):1–16, 2009.

[24] Maxime Durot, Pierre-Yves Bourguignon, and Vincent Schachter. Genome-scale models of bacterial metabolism: reconstruction and applications. *FEMS microbiology reviews*, 33(1):164–190, 2008.

[25] Adam M Feist, Markus J Herrgård, Ines Thiele, Jennie L Reed, and Bernhard Ø Palsson. Reconstruction of biochemical networks in microorganisms. *Nature Reviews Microbiology*, 7(2):129–143, 2009.

[26] Matthew A Oberhardt, Bernhard Ø Palsson, and Jason A Papin. Applications of genome-scale metabolic reconstructions. *Molecular systems biology*, 5(1):320, 2009.

[27] Jörn Behre, Thomas Wilhelm, Axel von Kamp, Eytan Ruppin, and Stefan Schuster. Structural robustness of metabolic networks with respect to multiple knockouts. *Journal of theoretical biology*, 252(3):433–441, 2008.

[28] Ines Thiele and Bernhard Ø Palsson. A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nature protocols*, 5(1):93–121, 2010.

[29] Mohammad Mazharul Islam, Wheaton Lane Schroeder, and Rajib Saha. Kinetic modeling of metabolism: Present and future. *Current Opinion in Systems Biology*, 26:72–78, 2021.

[30] Wolfgang Wiechert. 13c metabolic flux analysis. *Metabolic engineering*, 3(3):195–206, 2001.

[31] Andreas Hoppe, Sabrina Hoffmann, and Hermann-Georg Holzhütter. Including metabolite concentrations into flux balance analysis: thermodynamic realizability as a constraint on flux distributions in metabolic networks. *BMC systems biology*, 1(1):1–12, 2007.

[32] Steffen Klamt, Utz-Uwe Haus, and Fabian Theis. Hypergraphs and cellular networks. *PLoS computational biology*, 5(5):e1000385, 2009.

[33] Andres Bernal and Edgar Daza. Metabolic networks: beyond the graph. *Current computer-aided drug design*, 7(2):122–132, 2011.

[34] Albert-Laszlo Barabasi and Zoltan N Oltvai. Network biology: understanding the cell's functional organization. *Nature reviews genetics*, 5(2):101–113, 2004.

[35] Hawoong Jeong, Bálint Tombor, Réka Albert, Zoltan N Oltvai, and A-L Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651–654, 2000.

[36] Réka Albert and Albert-László Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

[37] Mark EJ Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[38] Steven H Strogatz. Exploring complex networks. *nature*, 410(6825):268–276, 2001.

[39] Marie Csete and John Doyle. Bow ties, metabolism and disease. *TRENDS in Biotechnology*, 22(9):446–450, 2004.

[40] Nathan D Price, Jason A Papin, Christophe H Schilling, and Bernhard O Palsson. Genome-scale microbial in silico models: the constraints-based approach. *Trends in biotechnology*, 21(4):162–169, 2003.

[41] Jürgen Zanghellini, David E Ruckerbauer, Michael Hanscho, and Christian Jungreuthmayer. Elementary flux modes in a nutshell: properties, calculation and applications. *Biotechnology journal*, 8(9):1009–1016, 2013.

[42] Dimitris Bertsimas and John N Tsitsiklis. *Introduction to linear optimization*, volume 6. Athena Scientific Belmont, MA, 1997.

[43] R Tyrrell Rockafellar. *Convex analysis*, volume 18. Princeton university press, 1970.

[44] Bernhard Palsson et al. Flux-balance analysis. *Nature Biotechnology*, 28(3):245–48, 2003.

[45] Robert Schuetz, Lars Kuepfer, and Uwe Sauer. Systematic evaluation of objective functions for predicting intracellular fluxes in escherichia coli. *Molecular systems biology*, 3(1):119, 2007.

[46] Amit Varma, Brian W Boesch, and Bernhard O Palsson. Biochemical production capabilities of escherichia coli. *Biotechnology and bioengineering*, 42(1):59–73, 1993.

[47] Adam M Feist and Bernhard O Palsson. The biomass objective function. *Current opinion in microbiology*, 13(3):344–349, 2010.

[48] Jeremy S Edwards and Bernhard O Palsson. The escherichia coli mg1655 in silico metabolic genotype: its definition, characteristics, and capabilities. *Proceedings of the National Academy of Sciences*, 97(10):5528–5533, 2000.

[49] Jochen Förster, Iman Famili, Bernhard Ø Palsson, and Jens Nielsen. Large-scale evaluation of in silico gene deletions in saccharomyces cerevisiae. *OMICS A Journal of Integrative Biology*, 7(2):193–202, 2003.

[50] Laszlo David, Sayed-Amir Marashi, Abdelhalim Larhlimi, Bettina Mieth, and Alexander Bockmayr. Ffca: a feasibility-based method for flux coupling analysis of metabolic networks. *BMC bioinformatics*, 12(1):1–7, 2011.

[51] GN Stephanopoulos, AA Aristidou, J Nielsen, GJ Tortora, F BR, and CL Case. Metabolic engineering–principles and methodologies academic press. *San Diego*, 1998.

[52] Anthony P Burgard, Priti Pharkya, and Costas D Maranas. Optknock: a bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. *Biotechnology and bioengineering*, 84(6):647–657, 2003.

[53] Kiran Raosaheb Patil, Isabel Rocha, Jochen Förster, and Jens Nielsen. Evolutionary programming as a platform for in silico metabolic engineering. *BMC bioinformatics*, 6(1):1–12, 2005.

[54] Desmond S Lun, Graham Rockwell, Nicholas J Guido, Michael Baym, Jonathan A Kelner, Bonnie Berger, James E Galagan, and George M Church. Large-scale identification of genetic design strategies using local search. *molecular systems biology*, 5(1):296, 2009.

[55] Ali R Zomorrodi, Patrick F Suthers, Sridhar Ranganathan, and Costas D Maranas. Mathematical optimization applications in metabolic networks. *Metabolic engineering*, 14(6):672–686, 2012.

[56] Priti Pharkya, Anthony P Burgard, and Costas D Maranas. Optstrain: a computational framework for redesign of microbial production systems. *Genome research*, 14(11):2367–2376, 2004.

[57] Priti Pharkya and Costas D Maranas. An optimization framework for identifying reaction activation/inhibition or elimination candidates for overproduction in microbial systems. *Metabolic engineering*, 8(1):1–13, 2006.

[58] Joonhoon Kim and Jennifer L Reed. Optorf: Optimal metabolic and regulatory perturbations for metabolic engineering of microbial strains. *BMC systems biology*, 4(1):1–19, 2010.

[59] Sridhar Ranganathan, Patrick F Suthers, and Costas D Maranas. Optforce: an optimization procedure for identifying all genetic manipulations leading to targeted overproductions. *PLoS computational biology*, 6(4):e1000744, 2010.

[60] Naama Tepper and Tomer Shlomi. Predicting metabolic engineering knockout strategies for chemical production: accounting for competing pathways. *Bioinformatics*, 26(4):536–543, 2010.

[61] Stefan Schuster, Thomas Pfeiffer, and David A Fell. Is maximization of molar yield in metabolic networks favoured by evolution? *Journal of theoretical biology*, 252(3):497–504, 2008.

[62] Radhakrishnan Mahadevan and Chrisophe H Schilling. The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic engineering*, 5(4):264–276, 2003.

[63] Jennifer L Reed and Bernhard Ø Palsson. Genome-scale in silico models of e. coli have multiple equivalent phenotypic states: assessment of correlated reaction subsets that comprise network states. *Genome research*, 14(9):1797–1805, 2004.

[64] Steven M Kelk, Brett G Olivier, Leen Stougie, and Frank J Bruggeman. Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. *Scientific reports*, 2(1):1–7, 2012.

[65] Michael E Bushell, Susana IP Sequeira, Chiraphan Khannapho, Hongjuan Zhao, Keith F Chater, Michael J Butler, Andrzej M Kierzek, and Claudio A Avignone-Rossa. The use of genome scale metabolic flux variability analysis for process feed formulation based on an investigation of the effects of the zwf mutation on antibiotic production in streptomyces coelicolor. *Enzyme and Microbial Technology*, 39(6):1347–1353, 2006.

[66] Oliver Hädicke, Hartmut Grammel, and Steffen Klamt. Metabolic network modeling of redox balancing and biohydrogen production in purple nonsulfur bacteria. *BMC systems biology*, 5(1):1–18, 2011.

[67] Vicente Acuna, Flavio Chierichetti, Vincent Lacroix, Alberto Marchetti-Spaccamela, Marie-France Sagot, and Leen Stougie. Modes and cuts in metabolic networks: Complexity and algorithms. *Biosystems*, 95(1):51–60, 2009.

[68] Christian Jungreuthmayer and Juergen Zanghellini. Designing optimal cell factories: integer programming couples elementary mode analysis with regulation. *BMC systems biology*, 6(1):1–12, 2012.

[69] Jörg Stelling, Steffen Klamt, Katja Bettenbrock, Stefan Schuster, and Ernst Dieter Gilles. Metabolic network structure determines key aspects of functionality and regulation. *Nature*, 420(6912):190–193, 2002.

[70] Jörn Behre, Thomas Wilhelm, Axel von Kamp, Eytan Ruppin, and Stefan Schuster. Structural robustness of metabolic networks with respect to multiple knockouts. *Journal of theoretical biology*, 252(3):433–441, 2008.

[71] Cong T Trinh, Pornkamol Unrean, and Friedrich Srienc. Minimal escherichia coli cell for the most efficient production of ethanol from hexoses and pentoses. *Applied and environmental microbiology*, 74(12):3634–3643, 2008.

[72] Martin Feinberg. Chemical reaction network structure and the stability of complex isothermal reactors—i. the deficiency zero and deficiency one theorems. *Chemical engineering science*, 42(10):2229–2268, 1987.

[73] Komei Fukuda and Alain Prodon. Double description method revisited. In *Franco-Japanese and Franco-Chinese Conference on Combinatorics and Computer Science*, pages 91–111. Springer, 1995.

[74] Christoph Kaleta, Luís Filipe de Figueiredo, and Stefan Schuster. Can the whole be less than the sum of its parts? pathway analysis in genome-scale metabolic networks using elementary flux patterns. *Genome research*, 19(10):1872–1883, 2009.

[75] Sayed-Amir Marashi, Laszlo David, and Alexander Bockmayr. Analysis of metabolic subnetworks by flux cone projection. *Algorithms for Molecular Biology*, 7(1):1–9, 2012.

[76] HL Kornberg. The role and maintenance of the tricarboxylic acid cycle in escherichia coli. In *Biochemical Society symposium*, volume 30, pages 155–171, 1970.

[77] HL Kornberg. The role and control of the glyoxylate cycle in escherichia coli. *Biochemical Journal*, 99(1):1, 1966.

[78] Christian Jungreuthmayer, David E Ruckerbauer, and Jürgen Zanghellini. regefmtool: Speeding up elementary flux mode calculation using transcriptional regulatory rules in the form of three-state logic. *Biosystems*, 113(1):37–39, 2013.

[79] Stefan J Jol, Anne Kümmel, Marco Terzer, Jörg Stelling, and Matthias Heinemann. System-level insights into yeast metabolism by thermodynamic analysis of elementary flux modes. *PLoS computational biology*, 8(3):e1002415, 2012.

[80] Anne Kümmel, Sven Panke, and Matthias Heinemann. Putative regulatory sites unraveled by network-embedded thermodynamic analysis of metabolome data. *Molecular systems biology*, 2(1):2006–0034, 2006.

[81] Yan Zhu, Jiangning Song, Zixiang Xu, Jibin Sun, Yanping Zhang, Yin Li, and Yanhe Ma. Development of thermodynamic optimum searching (tos) to improve the prediction accuracy of flux balance analysis. *Biotechnology and bioengineering*, 110(3):914–923, 2013.

[82] Andreas Hoppe, Sabrina Hoffmann, and Hermann-Georg Holzhütter. Including metabolite concentrations into flux balance analysis: thermodynamic realizability as a constraint on flux distributions in metabolic networks. *BMC systems biology*, 1(1):1–12, 2007.

[83] Christopher S Henry, Linda J Broadbelt, and Vassily Hatzimanikatis. Thermodynamics-based metabolic flux analysis. *Biophysical journal*, 92(5):1792–1805, 2007.

[84] Stefan Schuster and Claus Hilgetag. On elementary flux modes in biochemical reaction systems at steady state. *Journal of Biological Systems*, 2(02):165–182, 1994.

[85] Wynand S Verwoerd. A new computational method to split large biochemical networks into coherent subnets. *BMC systems biology*, 5(1):1–22, 2011.

[86] Stefan Schuster, Thomas Pfeiffer, Ferdinand Moldenhauer, Ina Koch, and Thomas Dandekar. Exploring the pathway structure of metabolism: decomposition into subnetworks and application to mycoplasma pneumoniae. *Bioinformatics*, 18(2):351–361, 2002.

[87] Christoph Kaleta, Luís Filipe de Figueiredo, and Stefan Schuster. Can the whole be less than the sum of its parts? pathway analysis in genome-scale metabolic networks using elementary flux patterns. *Genome research*, 19(10):1872–1883, 2009.

[88] Luis F De Figueiredo, Adam Podhorski, Angel Rubio, Christoph Kaleta, John E Beasley, Stefan Schuster, and Francisco J Planes. Computing the shortest elementary flux modes in genome-scale metabolic networks. *Bioinformatics*, 25(23):3158–3165, 2009.

[89] Jon Pey and Francisco J Planes. Direct calculation of elementary flux modes satisfying several biological constraints in genome-scale metabolic networks. *Bioinformatics*, 30(15):2197–2203, 2014.

[90] Kuhn Ip, Caroline Colijn, and Desmond S Lun. Analysis of complex metabolic behavior through pathway decomposition. *BMC Systems Biology*, 5(1):1–11, 2011.

[91] Siu Hung Joshua Chan and Ping Ji. Decomposing flux distributions into elementary flux modes in genome-scale metabolic networks. *Bioinformatics*, 27(16):2256–2262, 2011.

[92] Jean-Marc Schwartz and Minoru Kanehisa. Quantitative elementary mode analysis of metabolic pathways: the example of yeast glycolysis. *BMC bioinformatics*, 7(1):1–20, 2006.

[93] Jean-Marc Schwartz and Minoru Kanehisa. A quadratic programming approach for decomposing steady-state metabolic flux distributions onto elementary modes. *Bioinformatics*, 21(suppl_2):ii204–ii205, 2005.

[94] IM Stancu-Minasian. Fractional programming applications. In *Fractional Programming*, pages 6–33. Springer, 1997.

[95] Steffen Klamt and Radhakrishnan Mahadevan. On the feasibility of growth-coupled product synthesis in microbial strains. *Metabolic engineering*, 30:166–178, 2015.

[96] Oliver Hädicke and Steffen Klamt. Casop: a computational approach for strain optimization aiming at high productivity. *Journal of biotechnology*, 147(2):88–101, 2010.

[97] Steffen Klamt, Julio Saez-Rodriguez, and Ernst D Gilles. Structural and functional analysis of cellular networks with cellnetanalyzer. *BMC systems biology*, 1(1):1–13, 2007.

[98] Axel von Kamp, Sven Thiele, Oliver Hädicke, and Steffen Klamt. Use of CellNetAnalyzer in biotechnology and metabolic engineering. *Journal of Biotechnology*, 261:221–228, 2017.

[99] Steffen Klamt, Stefan Müller, Georg Regensburger, and Jürgen Zanghellini. A mathematical framework for yield (vs. rate) optimization in constraint-based modeling and applications in metabolic engineering. *Metabolic Engineering*, 47:153–169, 2018.

[100] Steinn Gudmundsson and Ines Thiele. Computationally efficient flux variability analysis. *BMC Bioinformatics*, 11:489, 2010.

[101] Matthew B Biggs, Gregory L Medlock, Glynis L Kolling, and Jason A Papin. Metabolic network modeling of microbial communities. *Wiley Interdisciplinary Reviews: Systems Biology and Medicine*, 7(5):317–334, 2015.

[102] Radhakrishnan Mahadevan and Michael A Henson. Genome-based modeling and design of metabolic interactions in microbial communities. *Computational and structural biotechnology journal*, 3(4):e201210008, 2012.

[103] S Freilich, R Zarecki, O Eilam, ES Segla, CS Henry, M. Kupiec, U. Gophna, R. Sharan, and E. Ruppin. Competitive and cooperative metabolic interactions in bacterial communities. *Nature Communications*, 2(589), 2011.

[104] Catherine Lassez. Quantifier elimination for conjunctions of linear constraints via a convex hull algorithm. *Symbolic and Numerical Computation for Artificial Intelligence*, pages 103–122, 1992.

[105] Jeffrey D Orth, Ronan MT Fleming, and Bernhard Ø Palsson. Reconstruction and use of microbial metabolic networks: the core escherichia coli metabolic model as an educational guide. *EcoSal plus*, 4(1), 2010.

[106] Jeffrey D Orth, Tom M Conrad, Jessica Na, Joshua A Lerman, Hojung Nam, Adam M Feist, and Bernhard Ø Palsson. A comprehensive genome-scale reconstruction of escherichia coli metabolism—2011. *Molecular systems biology*, 7(1):535, 2011.

[107] Jeremy S Edwards and Bernhard O Palsson. The escherichia coli mg1655 in silico metabolic genotype: its definition, characteristics, and capabilities. *Proceedings of the National Academy of Sciences*, 97(10):5528–5533, 2000.

10 Appendix

10.1 Abstract

10.1.1 English

For many scientific and industrial applications, an organisms' production envelope which summarizes its metabolic capabilities is of major interest. In its most basic form, as seen in figure 10.1, this is a two-dimensional, convex shape encompassing all possible production rates of an external metabolite relative to all possible growth rates.

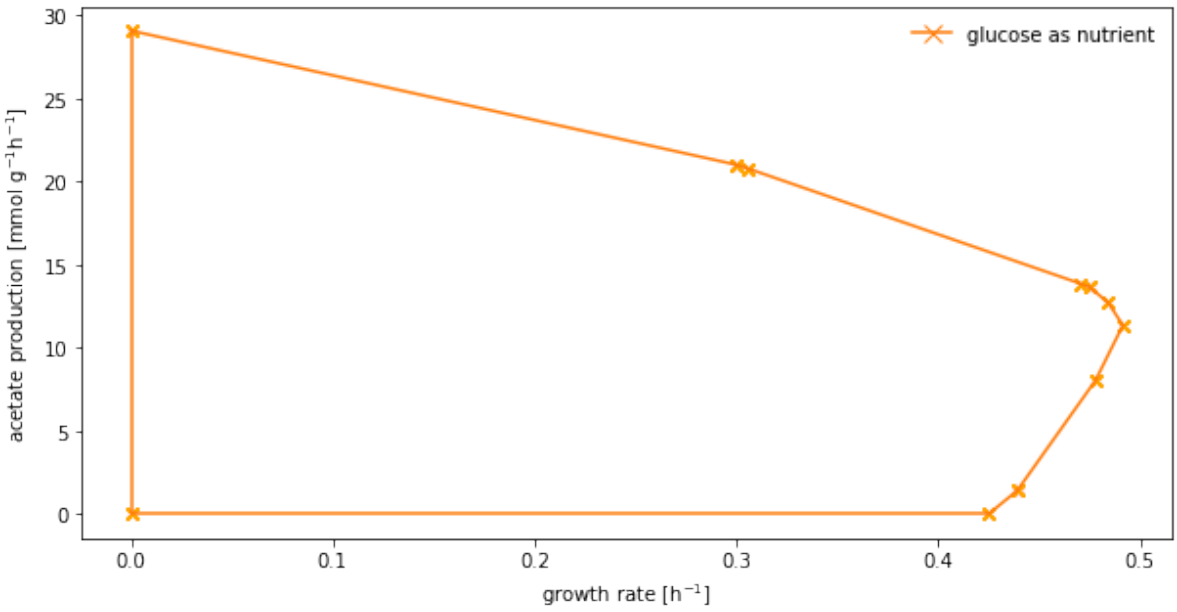

Figure 10.1: two-dimensional convex hull projected onto the acetate synthesis reaction and the growth reaction with glucose as a growth medium. The hull was calculated using the genome-scale E. coli model

Most tools used in this field rely on constraint-based modeling where different kinds of constraints are applied to the stoichiometric matrix as described in the chapter "constraint-based modeling". Technically this could be used to calculate production envelopes encompassing any number of metabolites but even after three dimensions, many existing methods quickly reach their limit.[98]

This scope can be extended to a higher number of metabolites using the convex hull algorithm (CHM), which is already used in computational geometry to project multidimensional polytopes onto spaces with fewer dimensions.[104] The resulting solution space is a convex hull, a kind of bounded polytope encompassing all points which can be generated by a linear combination of its *extreme points*. These extreme points in turn are the points furthest away from the center, which are not on a line or hyperplane

between two or more other extreme points. Figure 10.2 visualizes the main concepts based on a convex hull encompassing several arbitrary two-dimensional points. Figure 10.3 shows how a three-dimensional convex hull can be projected onto two dimensions.

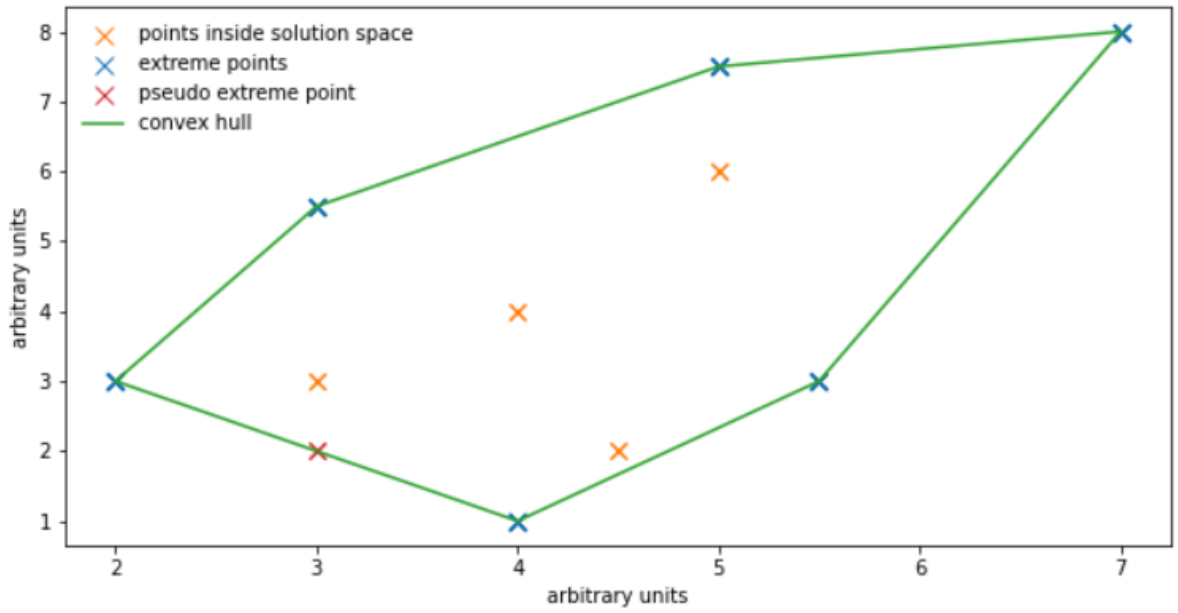

Figure 10.2: A set of arbitrary points showing extreme points in blue, non extreme points in orange, a false extreme point resulting from limited numerical accuracy in red, and the boundaries of the encompassing convex hull in green.

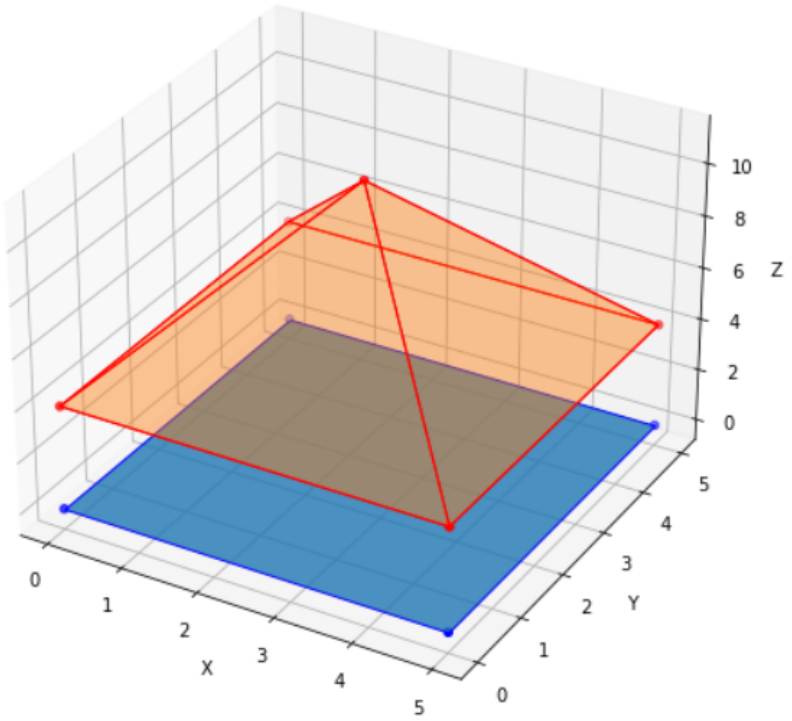

Figure 10.3: A set of three-dimensional points in red and its encompassing convex hull in orange represents an example solution space. Projection of this pyramid-shaped hull onto just the x and y dimensions results in the blue square.

In the context of metabolic modeling, the complete convex hull represents the full range of metabolic capabilities theoretically available to an organism, which is completely infeasible to calculate. In contrast, the projection of the hull onto a subspace with lower dimensionality is possible due to the convex hull method and provides valuable insights into the metabolism.

The goal of this project was to quantitatively and qualitatively compare two versions of a CHM tool created by Helena A. Herrmann (available at: https://github.com/HAHerrmann/ConvHull) based on the work of Lassez[104] with a selection of other implementations to test their applicability for the calculation of production envelopes.

10.1.2 German

Für viele wissenschaftliche und industrielle Anwendungen sind die Produktionskapazitäten von Organismen von großer Wichtigkeit. In ihrer einfachsten Form werden diese durch eine konvexe Hülle dargestellt, die alle potentiellen Produktionsraten für einen externen Metaboliten abhängig von allen möglichen Wachstumsraten wiedergibt. Ein Beispiel für so eine konvexe Hülle ist in figure 10.1 zu sehen.

Die meisten Algorithmen, die in diesem Bereich verwendet werden, basieren darauf, die stöchiometrische Matrix, wie im Kapitel "constraint-based modeling" beschrieben, durch Einschränkungen verschiedener Art zu limitieren, um Produktionskapazitäten zu berechnen. Theoretisch könnten so die Produktionskapazitäten für beliebig viele Metaboliten berechnet werden aber praktisch stoßen bestehende Methoden bei mehr als drei Metaboliten an ihre Grenzen.[98]

Die Anzahl der berücksichtigten Metaboliten kann jedoch durch die Verwendung von convex hull Algorithmen (CHM), die schon länger in der Computerbasierten Geometrie verwendet werden um mehrdimensionale Polytope auf Räume mit weniger Dimensionen zu projizieren[104] , erweitert werden. Der dadurch entstandene Lösungsraum ist eine konvexe Hülle, eine Art von abgegrenztem Polytop das alle Punkte einschließt die durch eine lineare Kombination seiner *Extremalpunkte* gebildet werden können. Diese Extremalpunkte sind alle Punkte mit dem höchsten Abstand zum Zentrum der Punktemenge, welche nicht auf einer Linie oder Hyperebene zwischen zwei oder mehr anderen Extremalpunkten liegen. Figure 10.2 veranschaulicht die wichtigsten Konzepte von konvexen Hüllen an einer Beispielhülle die eine Menge arbiträrer, zweidimensionaler Beispielpunkte einschließt. Figure 10.3 zeigt wie eine dreidimensionale Hülle auf zwei Dimensionen projiziert werden kann.

Beim Modellieren von Metabolismen entspricht die Menge aller potentiellen Produktionskapazitäten einer konvexen Hülle und ist praktisch nicht berechenbar. Durch die Projektion der Hülle auf einen niederdimensionalen Raum mit einem convex hull Algorithmus kann dies jedoch ermöglicht werden.

Das Ziel dieser Arbeit war es, zwei Varianten einer CHM Software, geschrieben von Helena A. Herrmann (verfügbar auf: https://github.com/HAHerrmann/ConvHull) basierend auf der Arbeit von Lassez[104] , qualitativ und quantitativ mit anderen Implementationen zu vergleichen und so ihre Anwendbarkeit für die Berechnung von Produktionskapazitäten zu ermitteln.