NTNU
Norwegian University of
Science and Technology

Development of an Internally Constrained Flux Balance Analysis Method for *Saccharomyces cerevisiae*

Pål Røynestad

Biotechnology (5 year)
Submission date: May 2016
Supervisor: Eivind Almaas, IBT

Norwegian University of Science and Technology
Department of Biotechnology

Acknowledgements

Looking back at the two years I've spent in the Network Biology group at the Institute of Biotechnology while working on this thesis, I would like to thank everyone who has contributed to making it a pleasant and interesting experience.

First and foremost, however, I would like to thank professor Eivind Alamas for his constant encouragement, guidance, support and relentless positivity. This was particularly valuable to me when the progress on the project stalled and I started to lose hope. I would also like to thank the other members of the Network Biology group for making the weekly group meetings fun to attend.

I'm grateful for all the support I've been given by my friends and family over the five years I've spent in Trondheim. To my parents in particular: thank you for being there whenever I've needed you.

Sammendrag

En av de mest suksessfulle anvendelsene av systembiologi har vært utviklingen av genomskala metabolske modeler (GSM). GSMer er modeler hvor alle kjente metabolske reaksjoner som tar sted i en celle er tatt med, og tillater *in silico* simulering av cellulær oppførsel. Dette er et hurtigvoksende fagfelt hvor nye metoder og modeler stadig blir utviklet og publisert. De mest avanserte GSM'ene inkluderer bade enzymkinetikk og genutrykk, men så langt har dette kun vært tilgjengelig for *Escherichia coli*.

Denne masteroppgaven introduserer en ny metoder som utvider en tidligere publisert GSM av *Saccharomyces cerevisiae* ved å inkorporere enzymkinetikk for å begrense mulige metabolske tilstander. Dette ble gjort ved å forenkle tidligere utgitte metoder for *E. coli*. Denne metoden kalles ”InteRnally Constrained Flux Balance Analysis” (ircFBA). Metoder ble også utviklet for automatisk innhenting og kategorisering av kinetisk data fra offentlig tilgjengelige databaser for å muliggjøre konstruksjonen av ircFBA-modeller for andre GSM'er. Predikerte vekstrater produsert ved hjelp av ircFBA ble påvist til å korrelere godt med eksperimentelle målinger av *S. cerevisiae* i minimalt vekstmedium med glukose som den eneste karbon- og energikilden.

To algoritmer ble også utviklet for å øke ytelsen til ircFBA ved å endre på kinetiske parametere. Disse var basert på to forskjellige aspekter av matematisk programmering. Den første innfører kunstvariabler til ircFBA problemet for å simulere økt fleks i de kinetiske parameterne. Den andre leter etter minimale endringer til de kinetiske parameterne for å oppnå den ønskede vekstøkningen. Den andre algoritmen ble påvist til å kunne oppnå ønskede vekstrater i aerobisk glukose-begrenset vekst ved å endre på kinetiske parameter på til det meste 6 reaksjoner.

Motivasjonen for å utvikle disse algoritmene var for å kunne lage stamme-spesifikke GSMer for *S. cerevisiae* for å kunne modellere stammer som brukes i vinproduksjon. Begge algoritmene var i stand til å kunne endre på vekstprofilen til ircFBA, men kun den andre algoritmen var i stand til å gjøre det på en pålitelig måte som ikke innfører globale endringer til modellen. Metoden har dermed potensialet til å oppnå dette målet.

Abstract

One of the most successful applications of the principles of systems biology has been the development of genome-scale metabolic models (GSMs). GSMs are models where every known metabolic reaction taking place in a cell is included, and allow *in silico* simulation of cellular behavior. This is a rapidly expanding field where new methods and models are constantly being developed and published. The most advanced GSMs incorporate both enzyme kinetics and gene expression, but so far this has been limited to *Escherichia coli*.

This thesis introduces a method that extends a GSM of *Saccharomyces cerevisiae* by incorporating enzyme kinetics to constrain the the permissible metabolic states. This was done by simplifying existing methods developed for *E. coli*. This method is called InteRnally Constrained Flux Balance Analysis (ircFBA), and methods were also developed for the automated retrieval and categorization of kinetic data from publicly accessible databases to enable construction of ircFBA models for other GSMs. The growth rate predictions produced by ircFBA were shown to correlate well with experimental measurements of *S. cerevisiae* in minimal media with glucose as the only carbon and energy source.

Two algorithms were developed to enhance the performance of ircFBA by altering the kinetic parameters. These were based on two different aspects of mathematical programming. The first introduces dummy variables into the ircFBA problem to simulate the effect of altering kinetic parameters during growth rate simulation. The other searches for minimal changes to the kinetic parameters in order to reach a higher growth rate. The first algorithm succeeded in closing growth gaps, but was numerically unstable and difficult to work with. The second algorithm, however, was demonstrated to be able to close the gap between ircFBA and experimental growth rates for aerobic glucose-limited growth by making changes to the kinetic parameters of at most six reactions.

The motivation for developing these algorithms was to use them to create strain-specific GSMs for *S. cerevisiae* in order to model strains used in wine production. Both algorithms were demonstrated to be able to close growth rate gaps, but only the second one could do it reliably and without making global changes to the model. This algorithm has potential for future development to actually achieve the stated ambitions. Future work on the algorithm is currently being planned in co-operation with professor Eivind Almaas.

Table of Contents

Acknowledgements **i**

Sammendrag **ii**

Abstract **iii**

Table of Contents **vii**

List of Tables **x**

List of Figures **xv**

Abbreviations **xvi**

1 Introduction **1**

2 Theory **3**

2.1 Linear programming 3

2.1.1 The Standard Form of a Linear Program 3

2.1.2 Solving LP problems 4

2.1.3 Duality theory & Shadow Prices 5

2.1.4 Quadratic programming 6

2.2 Enzymes 7

2.2.1 Michaelis-Menten kinetics 7

2.2.2 EC numbers 8

2.3 Genome scale metabolic models 8

2.3.1 Model reconstruction 8

2.3.2 The Stoichiometric Matrix 10

2.4 Flux balance analysis 11

2.4.1 Phenotype Phase Plane & the Line of Optimality 12

2.4.2 Flux coupling analysis 14

2.5 FBA with internal constraints 14

2.5.1 FBAwMC 15
2.5.2 MOMENT 15
2.6 *Saccharomyces cerevisiae* 16
2.6.1 Metabolic reconstructions of S. cerevisiae 17
2.7 Hierarchical clustering 18

3 Methods & Software 21
3.1 Software 21
3.1.1 Matlab 21
3.1.2 Gurobi 22
3.1.3 Python 22
3.2 ircFBA 22
3.3 Assembly of an ircFBA model based on iTO977 23
3.3.1 Data gathering 23
3.3.2 Data integration 25
3.3.3 Model formulation and implementation 28
3.4 ircFBA parameter tuning 28
3.4.1 The QP Algorithm: ircFBA performance tuning through Quadratic Programming 29
3.4.2 The SP Algorithm: ircFBA performance tuning through Post-Optimality Analysis 30
3.4.3 Combining turnover number sets 31
3.5 Analysis of randomized kcat distributions 32

4 Results & Discussion 33
4.1 ircFBA Model Assembly & Performance 33
4.1.1 Phenotype prediction 34
4.2 Fitting ircFBA model behavior to experimental measurements 39
4.2.1 Applying the QP algorithm 39
4.2.2 Applying the SP algorithm 44
4.2.3 Combining *kcat* sets 50
4.3 ircFBA with randomized *kcat* sets 53
4.3.1 Correlation of the final *kcat* values with the original *kcat* set . . . 55

5 Conclusion 57

Bibliography 59

Appendix 65

A Implementing an ircFBA model based on iTO977 for use with the COBRA Toolbox 67

B Uptake rates for glucose and oxygen from minimal glucose media 69

C KEGG pathway maps of glycolysis and the TCA cycle in *S. cerevisiae* 71

D Histograms of the relative changes in kcat caused by the QP algorithm 75

E The effect of reverting kcat ratios back to its initial value 79

List of Tables

4.1 Correlations between predictions and the experimental measurements. The ircFBA growth rate row gives the Pearson correlation coefficient with its p-value for ircFBA's growth rate predictions against the experimental growth rates. The other rows are interpreted in the same way. 34

4.2 Pearson correlations between predictions and the experimental measurements. Correlations were not computed for the whole range of experimental values. The range was instead split into two, and correlations were computed for each range. Low glucose uptake rate is defined as less than 4 mmol gDW^{-1} h^{-1}, while high is greater than than 4. The ircFBA growth rate row gives the Pearson correlation coefficient with its p-value for ircFBA's growth rate predictions against the experimental growth rates. The other rows are interpreted in the same way. 36

4.3 Pearson correlations between ircFBA predictions and the experimental measurements. The ircFBA models were optimized with the QP algorithm. Row ircFBA$^{0.15}$ gives correlations for the the growth rate and oxygen uptake rate predictions made after fitting the base ircFBA model to the target growth rate of 0.15. The other rows have the same interpretation. . 40

4.4 The magnitude of the changes made to the base $kcat$ set when the QP algorithm is run. $|\frac{kcat^F}{kcat^E}|$ gives average relative change in the $kcat$ set. $|\sum(g_i > 0)|$ is the mean number of enzymatic reactions active in the intermediary steps of the QP algorithm. ircFBA$^{\mu}$ is the ircFBA model that was fitted to a target growth rate of μ. 44

4.5 Pearson correlations between ircFBA predictions and the experimental measurements. The ircFBA models were optimized with the SP algorithm. Row ircFBA$^{0.15}$ gives correlations for the the growth rate and oxygen uptake rate predictions made after fitting the base ircFBA model to the target growth rate of 0.15. The other rows have the same interpretation. 46

4.6 The final kcat-ratios obtained by running the SP algorithm with $M = 1000$ independently to reach the target growth rates from the data set in appendix A. The subscript for the column label μ is the growth rate used as the target when running the SP algorithm. The row labels are names for the enzymatic reactions where $kcat$ was adjusted by the SP algorithm 50

B.1 The experimental data set used to evaluate and enhance the ircFBA model. Each triad of values comes from aerobic minimal media with glucose as the sole energy and carbon source. The data was taken from [1] 70

List of Figures

2.1 The effect of increasing the right-hand side of a $\leq$ constraint in a LP maximization problem. The breakpoints in the line occur when the shadow price range is violated, meaning that the increase in growth rate is less than desired. 6

2.2 A stepwise list of the steps involved in producing a metabolic reconstruction. Figure taken from [2]. 9

2.3 A graphical depiction of a toy stoichiometric matrix. The metabolites A and B are imported with exchange reactions with reactions V_1 and V_2. A pathway consisting of two reactions V_3 and V_4 leads accumulation of D, which is then exported out of the system with reaction V_5. Figure adapted from [3]. 11

2.4 FBA phenotype phase plane of the iTO977 metabolic reconstruction of *S. cerevisiae*. Each point in the plane gives the growth rate when glucose and oxygen is fixed. The red line is the line of optimality, and shows the area of the phenotype phase plane where the carbon source is fully oxidized. . 13

2.5 A) The ratio of the shadow prices for the glucose and oxygen exchange reactions for the yeast metabolic reconstruction iTO977. A positive value means that the shadow prices have opposite signs, meaning that one of them is constraining the growth rate by being too high. A negative value means that both substrates are constraining the growth. B) The boundaries for the different phases. P_1 is the futile phase, while P_2 is limited by both metabolites. 14

2.6 The ethanol and acetate fermentation pathway in *S. cerevisiae*. Although not shown on the figure, the conversion of acetaldehyde to acetate reduces NAD(+) to NADH. Taken from [4]. 17

2.7 The phylogeny of the iTO977 model. iTO977 was sourced based on two previously published models, Yeast 1.0 and iIN800. Figure taken from [1]. 18

2.8 A dendrogram for a matrix of randomly generated numbers.No real clustering pattern can be seen. 19

3.1 The structure of the compound thesaurus. The outer layer of the thesaurus have links to the main entry. The thesaurus itself consists of synonyms for most of the compounds in the iTO977 model. 25

3.2 The workflow of matching a BRENDA database entry with a metabolite in the iTO977 model. Each BRENDA substrate points to a list of synonyms. The synonyms are searched against the synonyms layer of the iTO977 metabolite thesaurus. When a match is found, the BRENDA substrate is defined as being the same compound as the iTO977 metabolite. 26

3.3 The structure of the EC table. Each EC number has an associated list of organisms, and each organism has an associated list of substrates, and each substrate has a list of turnover numbers. 27

3.4 The representation of reactions in the reaction hash table. Each reaction has several fields, like the EC numbers associated with it, the reactants and the products. 27

4.1 A) The turnover number lists sorted in rising order. Max iTO977 is the distribution obtained by choosing the largest $kcat$ in the $kcat$ list for any given reaction, while median iTO977 is from choosing the median. Median *E. coli* is from MOMENT [5]. B) A histogram of max iTO977 on a log scale. The $kcat$ set varies across 12 orders of magnitude, but has a bell shaped distribution. 34

4.2 FBA (iTO977) and optimized MOMENT line of optimality plotted along with experimental measurements from *S. cerevisiae*. 35

4.3 α as defined by equation 2.27 for the ircFBA model. The leftmost plot shows the raw α, while the plot to the right only shows the sign. Ten distinct phases can be seen in the leftmost plot, but these correspond to two main phases demarcated by the sign of α. The positive phase exhibits futile metabolise ($\alpha \geq 0$). LOO ircFBA separates the second major phase into two parts. To the left of LOO ircFBA, growth is limited by both glucose and oxygen. To the left, both inhibit growth. 36

4.4 A) The respiratory quotient, RQ, as a function of the ircFBA growth rate. A RQ close to 1 indicates near full oxidation of glucose, while a RQ over 1 is a sign of fermentation. The dip below 1 is caused by the excretion of pyruvate. B) Excretion profile for ircFBA. Ethanol, acetate and pyruvate are all excreted at high rates. 38

4.5 The sign of fluxes in the TCA cycle and electron transport chain pathway as a function of the growth rate in an ircFBA model of *S. cerevisiae*. The reaction names are listed on the y-axis. Yellow squares indicate positive flux, green squares no flux and blue squares negative flux. NADH and $FADH_2$ producing fluxes (IDH1, KGD1 and $SDH3_2$) are turned off at higher growth rates. 38

4.6 Growth rates for ircFBA models fitted to experimental growth rates with the QP algorithm. The in the label μ was the target growth rate used by the QP algorithm for generating the corresponding $kcat$ set. 41

4.7 Oxygen uptake rates for ircFBA models fitted to experimental growth rates with the QP algorithm. The in the label μ was the target growth rate used by the QP algorithm for generating the corresponding $kcat$ set. 42

4.8 d(exp,LOO) is defined as the distance an experimental growth rate, exp, is from the line of optimality (LOO) in a three dimensional Euclidean space where the dimesions are the growth rate, oxygen oxygen uptake rate and glucose uptake rate. A) After separating from the line of optimality, the distance of the experimental data points to the line of optimalityincreases linearly. B) The oxygen uptake rate Pearson correlation coefficient for ircFBA models plotted against the distance the target growth rate was from the line of optimality. 43

4.9 The number of enzyme concentration variables (g_i) greater than zero during intermediary steps of the QP algorithm. The QP lines are the number of non-zero enzyme concentrations within the augmented QP problem solution, while blue line is from growth rate maximization of the normal ircFBA model with the $kcat$ sets produced by the augmented QP problem solution. 45

4.10 Growth rates for ircFBA models fitted to experimental growth rates with the SP algorithm. The in the label ircFBA$^\mu$ was the target growth rate used by the SP algorithm for generating the corresponding $kcat$ set. 47

4.11 Oxygen uptake rates for ircFBA models fitted to experimental growth rates with the SP algorithm. The in the label ircFBA$^\mu$ gives the target growth rate used by the SP algorithm for generating the corresponding $kcat$ set. . 48

4.12 Dendrogram of the kcat-ratio coefficient columns in table 4.6. Bottom-up hierchical clustering was performed with average linkage as the between-group distance metric. The leaf nodes are labeled after the SP target growth rate. $\mu_{0.38}$ is an out-group here. 49

4.13 The final $\frac{kcat^F}{kcat^E}$ ratio as a function of varying M, the number of iterations performed to be performed by the SP algorithm. After an early period fluctuation, each $kcat$ converges to its final value. 50

4.14 A heat map of the relative proximity a reaction is from having its $kcat$ altered during a step of the SP algorithm. The target growth rate was the highest growth rate in the experimental data set (Appendix B). The closer the value is to 1, the closer the reaction is to being altered. All reactions where this relative proximity was at least 0.7 can be seen, and shows that few reactions ever come close to 51

4.15 Linear weights on the columns of table 4.6 when combining the $kcat$ sets to minimize the relative deviations of the ircFBA model's predictions for growth rate and oxygen uptake from the experimental measurements. The subscript for μ indicates the column of the table the weight represents, with μ without a subscript being an additional column added consisting only of ones. 52

4.16 ircFBA growth predictions based on combining the $kcat$ sets from table 4.6. A) The growth rate as a function of the upper bound on the glucose uptake rate. B) The oxygen uptake rate as a function of the glucose uptake rate upper bound. 52

4.17 A) The major excretion product at any given combination of glucose and oxygen uptake for the ircFBA model generated by combining the $kcat$ columns in table 4.6. At least one of acetate, pyruvate or ethanol was excreted in every point plane, except for the part prior the ircFBA and FBA lines of optimality separating. B) The rate of excretion of acetate, pyruvate and ethanol in the ircFBA model generated by combining the columns in table 4.6. The predicted ethanol production start point, 0.36 h^{-1}, occurs at a later stage than experimental evidence suggests [6]. . . . 53

4.18 Growth rate plotted against oxygen and glucose uptake rates for unconstrained ircFBA models after randomizing the $kcat$ set and optimizing with the SP algorithm. The target growth rate was 0.38 h^{-1}. Individual observations seem to fall into two categories: High glucose, low oxygen or low glucose and high oxygen. 54

4.19 A heat map of the correlation matrix for the fluxes from the 459 enzymatic reactions that were switched on in at least one of the 1000 ircFBA models. 55

4.20 A heat map of the correlation matrix for the fluxes from the 459 enzymatic reactions that were switched on in at least one of the 1000 ircFBA models. 56

4.21 The $\log_1 0$ transformed mean of $kcat$, here designated as $kcat_F$ values changed by the SP algorithm after randomization plotted against the $\log_{10}$ transformed original $kcat$ set, $kcat_E$. A linear trend can be seen, confirmed by a Pearson coefficient of 0.37 with a p-value of 1.4×10^{-9} . . . 56

B.1 iTO977 FBA growth rates plotted against the experimental growth rate. The FBA model was constrained with the experimental uptake rates for glucose and oxygen. 70

C.1 A pathway map of glycolysis in *S. cerevisiae* taken from the KEGG PATHWAY database and formated with Keggscape [7]. Metabolites are white squares, and the green boxes are the enzymes catalyzing the reaction. . . 72

C.2 A pathway map of the TCA cycle in *S. cerevisiae* taken from the KEGG PATHWAY database and formated with Keggscape [7]. Metabolites are white squares, and the green boxes are the enzymes catalyzing the reaction. 73

D.1 log_2 of the final ratio, $\frac{kcat_F}{kcat_E}$, for enzyme turnover numbers that were changed when running the QP algorithm to fit for the highest growth in the data set. Changes vary over several orders of magnitude, with the most extreme cases representing 12 doublings. 76

D.2 The number of active genes in intermediary QP algorithm solutions. The LP curve is the number of active reactions when the $kcat$ set was updated according to the preceding QP algorithm step. If no numerical issues were present, these curves would be identical. The large gap, even at the lowest level of feasibility intolerance, shows that the numerical solvers used aren't precise enough to actually solve the QP problem effectively. 77

E.1 The effect of resetting individual kcats back to the their initial value for ircFBA$^{0.38}$. The leftmost plot in the upper row is the original ircFBA$^{0.38}$ curve. 80

E.2 The effect of resetting individual kcats back to the their initial value for ircFBA$^{0.38}$. The leftmost plot in the upper row is the original ircFBA$^{0.38}$ curve. 81

Abbreviations

S. cerevisiae	=	*Saccharomyces cerevisiae*
FBA	=	Flux Balance Analysis
FVA	=	Flux Variability Analysis
MOMENT	=	MetabOlic Modeling with Enzyme kiNeTics
FBAwMC	=	Flux Balance Analysis with Mass Constraints
ircFBA	=	InteRnally Constrained Flux Balance Analysis
LP	=	Linear Programming
QP	=	Quadratic Programming
COBRA	=	COnstraints Based Reconstruction and Analysis
PhPP	=	Phenotype Phase Plane
HC	=	Hierarchical Clustering
BRENDA	=	BRaunschweig ENzyme DAtabase)
ChEBI	=	Chemical Entities of Biological Interest
KEGG	=	Kyoto Encyclopedia of Genes and Genomes
RQ	=	Respiratory Quotient
TCA	=	Tricarboxylic Acid
GSM	=	Genome-Scale Metabolic Model
F2C2	=	Fast Flux Coupling Calculator
$kcat$	=	Turnover number

Chapter 1

Introduction

As aptly stated by François Jacob in 1974 [8]:

> Every object that biology studies is a system of systems.

This view has been echoed in many biological fields such as ecology for a long time, but has only really been in the mainstream of molecular biology since the start of the 2000s when the field of systems biology started to mature [9, 8]. This was precipitated by the increase in availability of high-throughput data sets and the increased capacity for sequencing full genomes. Armed with detailed information about how the components of biological systems function and interact, systems biology integrates this with genomic data to create detailed models of biological systems where emergent properties can be studied.

One of the most successful applicants of systems biology has been the development of genome-scale metabolic models. These are models where every known chemical reaction is represented in a metabolic network. This enables predicting metabolic states by looking at the flow of metabolites across the network by simulating growth [3]. Furthermore, by adding layers of additional complexity to these models in the form of enzyme kinetics and gene regulation, the predictive capabilities of these models have been compounded [10].

S. cerevisiae has a key role in molecular biology, where it has been a workhorse for genetics research [11], but also in industry where a myriad of more or less well-characterized strains are used for the production of alcoholic beverages and baked goods [12]. Being able to model industrially relevant strains of *S. cerevisiae* would be helpful, but since metabolic models are only available for the lab strains, which are known to be phenotypically quite dissimilar from from non-lab strains [13], this can be difficult to do. However, in the cases where the metabolic states that the non-lab strain is capable of reaching is contained within an existing metabolic model, simulating the cellular metabolism should be possible. Metabolic models typically require information about the uptake and excretion rates of various metabolites to be able to make accurate predictions, however [14]. Metabolic models incorporating other aspects of cellular metabolism than the metabolic network itself, such as enzyme kinetics, should then allow for modelling of different strains without measuring uptake rates if the constraints are strain specific [5, 15]. Unfortunately, no such methods are available for *S. cerevisiae*.

This thesis aims to deliver on this, by taking an already published metabolic model of *S. cerevisiae* and adding enzyme kinetics and mass constraints to it. Furthermore, methods for tailoring the model to new kinetic coefficients to produce more reliable growth rate predictions are developed, with the ultimate goal of one day being able to use it to customize the metabolic model to represent any strain where enough growth phenotyping has been done.

Chapter 2

Theory

2.1 Linear programming

Linear programming (LP) is a form of mathematical programming based on optimizing linear functions subject to linear constraints. As in the broader field of mathematical programming, the function being optimized is known as the objective function. Linear programming has been applied in a broad set of fields, such as economics, logistics, chemistry and modelling of metabolism [14, 16]. An overview of some basic concepts of linear programming will be provided here based on Lundgren [16].

2.1.1 The Standard Form of a Linear Program

A linear programming problem is composed of three parts: an objective function, a set of linear constraints and a set of restrictions on the sign of the variables.

The objective function is simply a linear function representing the value being optimized in the linear programming problem. Linear programming allows for the objective function to be minimized or maximized, although it is typically expressed as a minimization problem in the standard (or canonical) form of the problem. The linear constraints can include both equalities or inequalities, although inequalities have to be converted to equalities in order to satisfy the conditions of the standard form. This can be done by adding additional variables to the problem. The same applies for the restrictions on the sign of the variables. Linear programming allows for both positive, negative or unrestricted signs on the values of variables, but the standard form requires variables to be non-negative. Again, this can be done through suitable variable transformations, or by introducing additional variables.

In general, then, the standard form of a LP problem with n variables and m linear constraints can be expressed as:

$$\min z = \sum_{j=1}^{n} c_j x_j \tag{2.1}$$

$$\text{s.t.} \sum_{j=1}^{n} a_{ij} x_j = b_i, i = 1, ..., m \tag{2.2}$$

$$x_j \geq 0, j = 1, ..., n \tag{2.3}$$

where z is the value of the objective function, c_j is the objective value coefficient for variable x_j, a_{ij} is the coefficient for variable x_j in the ith constraint and b_i is the right-hand side in the ith constraint. Equation 2.1 then specifies the optimization problem, 2.2 gives the set of linear constraints, and 2.3 gives the variable sign destrictions.

Another common representation of the standard form of linear programming can be given using matrix notation. The same general linear programming problem can be stated as:

$$\min z = \mathbf{c}^T \mathbf{x} \tag{2.4}$$

$$\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b} \tag{2.5}$$

$$\mathbf{x} \geq \mathbf{0} \tag{2.6}$$

$\mathbf{A} = [A_{i,j}]$ is the coefficient matrix for the constraints in equation 2.2, $\mathbf{b} = [b_i]$ is a $m \times 1$ vector, $\mathbf{x} = [x_i]$ is a $n \times 1$ vector and $\mathbf{c} = [c_i]$ is a $n \times 1$ vector.. The constraints in equation 2.5 describe the feasible region of the LP problem, and any solution satisfying this equation is known as a feasible solution. Conversely, any solution violating this equation is referred to as an infeasible solution.

2.1.2 Solving LP problems

The region $X = \{\mathbf{x} | \sum_{j=1}^{n} a_{ij} x_j = b_i, i = 1, ..., m\}$ defined by equation 2.2 is known as the feasible region of the problem, and can geometrically be classified as a polyhedron, or more specifically as a polytope when X is bounded. The edges of the polytope are known as extreme points, and the polytope itself can be defined as the set of all convex combinations of its constituent extreme points. The extreme points themselves arise from the intersections of constraints. Furthermore, due to the linearity of LP, any optimal solution necessarily has to be located on surface of the feasible region. Linearity also ensures that the feasible region is convex, which implies that any local optimal solution is also a global optimal solution. This combined with the fundamental theorem of linear programming, which guarantees that an optimal solution can be found in one of the extreme points of the feasible region, provided that the feasible region is both bounded and non-empty, gives the basis for several solution strategies.

Due to the fundamental theorem of linear programming a possible solution strategy is simply to enumerate every single extreme point, evaluate the objective function value in each of them, and choose the best solution. This, however, is an extremely inefficient approach, and several vastly better methods are available. One of these is the simplex method.

The simplex method was developed by George Dantzig in 1947, and remains a popular algorithm for solving LP problems [17]. The simplex method is an algebraic method, but the basic idea behind it is geometric. The method works by starting at a pre-determined extreme point (usually the origin), and moving to adjacent extreme points. For an LP problem with n variables, extreme points are defined as adjacent if they share $n - 1$ constraint boundaries. Adjacent extreme points are important, as they can be used as an optimality test. If no adjacent extreme point has a better objective function value than the extreme point being considered, then that extreme point is a local optimal solution. Due the problem being convex, that implies that a global optimum has been found [17]. Until the optimality test has been met, the simplex method moves along the surface of the feasible region, jumping from extreme point to extreme point. The criteria for choosing the adjacent extreme point is the direction the objective function improves the most in, which means that as long as the objective function can be strictly improved in every iteration, the simplex method will never visit an extreme point more than once, and the problem will be solved in a finite number of steps.

2.1.3 Duality theory & Shadow Prices

The LP problem described by equations 2.1-2.3 or 2.4-2.6 is known as the primal problem. Any primal problem has an associated problem, known as the dual problem. The dual problem has several interesting properties, and is of central importance both to sensitivity analysis and in developing new LP solution methods [16, 17].

In order to state the dual problem in its general form, it's helpful to formulate the primal problem in a slightly different version of the standard form than in equations 2.4-2.6. A slightly different, but wholly legitimate, version of the standard form is formulated as a maximization problem, with the constraints all being $\leq$ inequalities. This gives rise to the general description of the dual problem:

$$\begin{array}{lr}
\textbf{Primal} & \textbf{Dual} \\
\text{max } z = \mathbf{c}^T\mathbf{x} & \text{min } w = \mathbf{b}^T\mathbf{v} \\
\text{s.t. } \mathbf{A}\mathbf{x} \leq \mathbf{b} & \text{s.t. } \mathbf{A}^T\mathbf{v} \geq \mathbf{c} \\
\mathbf{x} \geq \mathbf{0} & \mathbf{v} \geq \mathbf{0}
\end{array}$$

where w is the dual objective function, and **v** is an $n \times 1$ vector where v_i is the dual variable. The dual variable v_i is associated with constraint i (see equation 2.2), and in the context of the primal problem, we have that:

$$v_i = \frac{dZ}{db_i} \tag{2.7}$$

where b_i is the right-hand side of constraint i and Z is the optimal objective function value. When discussing the primal problem, v_i is usually referred to as the shadow price. When interpreted as the derivative of the objective function, the shadow price is only valid for a certain range of Δb_i, and the observed change in Z will always be worse than v_i would suggest the if this range is violated[17, 18]. Figure 2.1 illustrates the potential effect changing the right-hand side in a $\leq$ constraint can have on a maximization LP problem.

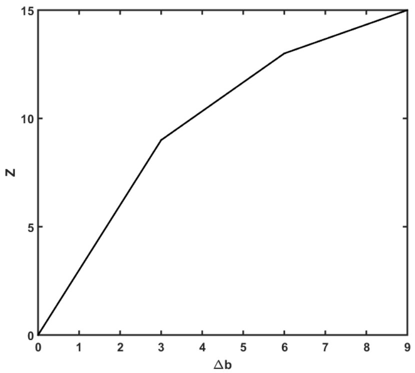


Figure 2.1: The effect of increasing the right-hand side of a $\leq$ constraint in a LP maximization problem. The breakpoints in the line occur when the shadow price range is violated, meaning that the increase in growth rate is less than desired.

2.1.4 Quadratic programming

While linear programming has a wide area of application, there are cases where a linear objective function or constraints are not sufficient. The broader field of nonlinear programming deals with problems of this type. A special case of nonlinear programming is known as quadratic programming (QP), and is related to linear programming in that it has linear constraints, but differs in having a quadratic objective function. The general QP problem can be stated as

$$\min z = \mathbf{c}^T\mathbf{x} + \frac{1}{2}\mathbf{x}^T\mathbf{Q}\mathbf{x} \tag{2.8}$$

$$\text{s.t. } \mathbf{A}\mathbf{x} = \mathbf{b} \tag{2.9}$$

$$\mathbf{x} \geq \mathbf{0} \tag{2.10}$$

where, as with LP, **c** is a $n \times 1$ vector that contains the coefficients for the linear part of the objective function, **A** is the linear constraint coefficient matrix of size $m \times n$, **b** is a $m \times 1$ vector that contains the constraint right-hand sides and **x** is a $n \times 1$ vector where the entries are variable values. QP, however, expands on LP by including the Q matrix, which is a $n \times n$ matrix. This matrix is known as the quadratic matrix[16].

As QP is a form of nonlinear programming, solving a general QP problem is much more difficult than a corresponding LP problem. In fact, the general QP problem has been proven to be NP-hard, meaning that there is no algorithm capable of solving it in polynomial time [19]. There are, however, special cases that can be solved. One of these is if the matrix Q is positive semidefinite, in which case the problem is convex. In this special case, optimal solutions can be found within polynomial time [16, 17]. A test for Q being positive semidefinite can be performed by checking if the inequality

$$\mathbf{x}^T\mathbf{Q}\mathbf{x} \geq \mathbf{0} \tag{2.11}$$

holds for every **x**. If it holds, the problem is convex[17].

2.2 Enzymes

Enzymes are highly specialized proteins capable of catalyzing specific chemical reactions. As life is unsustainable without catalysis of chemical reactions [20], enzymes are critically important to the existence of cellular life. A brief overview of the basics of Michaelis-Menten enzyme kinetics and the classification of enzymes will be provided in this section.

2.2.1 Michaelis-Menten kinetics

The field of enzyme kinetics deals with the measurement and characterization of the reaction rates of enzyme catalyzed reactions. One of the standard models of enzyme kinetics is known as Michaelis-Menten kinetics, and has a central part in the history of biochemistry and has served as a standard when characterizing the kinetic properties of enzymes [20, 21]. The following description of Michaelis-Menten kinetics is taken from Nelson & Cox [20].

Michaelis-Menten kinetics models enzymatic reactions according to:

$$E + S \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} ES \xrightarrow{k_2} E + P \tag{2.12}$$

where E is the enzyme catalyzing the reaction, S is the substrate, ES is the enzyme-substrate complex and P is the product. Here k_1 is the reaction rate for the formation of the enzyme-substrate complex, k_{-1} is the rate for the breakdown of the complex into its original constituent parts and k_2 is the rate of the formation of the product. This model assumes that reverse reaction, i.e. enzyme and product combining to form the enzyme-substrate complex, does not occur at any appreciable rate. This implies that the formation of enzyme and product is the rate limiting step. Additionally, it makes the so-called steady-state assumption, which assumes that rate of ES formation is equal to its rate of breakdown. Using these assumptions, the Michaelis-Menten equation can be derived:

$$V_0 = \frac{k_2[E_t][S]}{K_m + [S]} \tag{2.13}$$

where V_0 is the initial rate of product formation, $[E_t]$ is the total enzyme concentration, i.e. $[E_t] = [ES] + E$, and K_m is the concentration of S where $V_0 = \frac{k_2[E_t]}{2}$. $k_2[E_t]$ is also denoted as V_{max}, as it gives the maximal attainable reaction rate given the current enzyme concentration. The Michaelis-Menten equation can also be used when the final reaction step is not the rate limiting step, or when the number of reaction steps is higher than in equation 2.12. A more general rate constant, $kcat$ (also known as the turnover number), is therefore typically used rather than k_2, giving the Michaelis-Menten equation in its final form:

$$V_0 = \frac{V_{max}[S]}{K_m + [S]} \tag{2.14}$$

where

$$V_{max} = kcat[E_t] \tag{2.15}$$

$kcat$ is also known as the enzyme turnover number.

2.2.2 EC numbers

The most widespread classification system used to classify enzymes is the Enzyme Commission number (EC number) system, which is based on the reaction the enzyme catalyzes. The EC number system is a numerical system, where an enzyme is given a numeric code with four numbers separated by periods. The first number categorizes it into its broadest functional class, while the following numbers categorize the enzyme into increasingly narrower classes [22]. The EC number classification system is extensive, and as of january 2014 5300 valid EC classes were available, with more being added regularly [23].

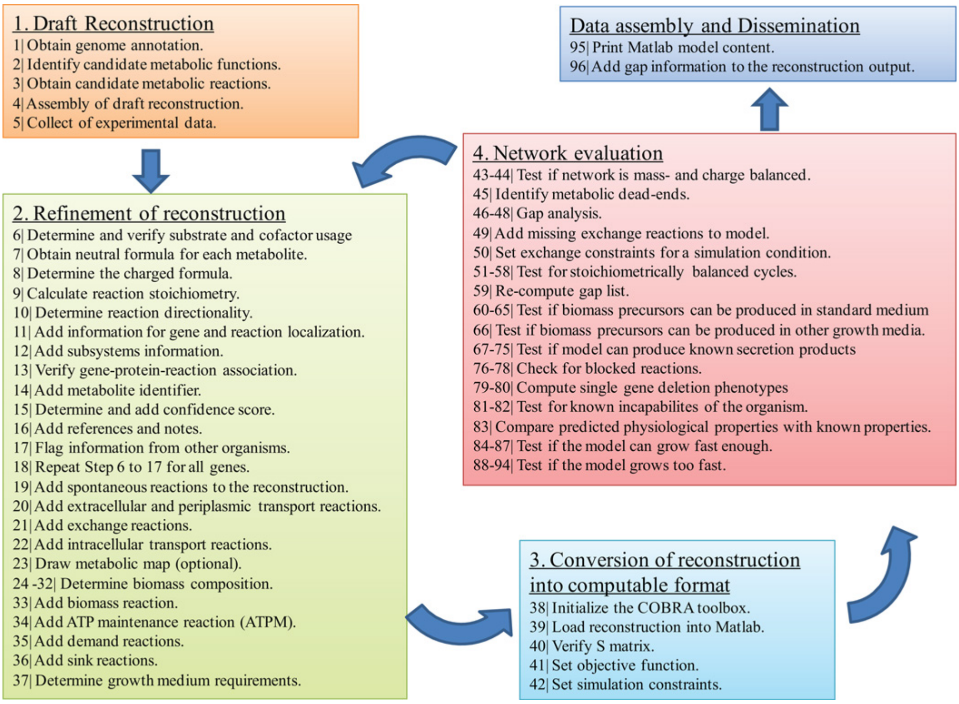
2.3 Genome scale metabolic models

One of the main tasks of the field of biochemistry has been the reconstruction and mapping of cellular metabolism [20]. With the advent of the genomic era and the emergence of systems biology, it has been possible to integrate the extensive data about individual enzymes, chemical reactions, and pathways, assembled through decades of experimental work with genomics data [3, 24]. This has enabled the construction of genome-scale metabolic models (GSMs), which are models encompassing every known chemical reaction taking place in a given organism [3]. GSMs are also typically annotated with the genes involved in the reaction, and gene knockout experiments can be simulated by disabling reactions [14]. These models give a network view of metabolism, where cellular metabolism is seen as an integrated whole rather than as consisting of isolated pathways [25]. GSMs have been applied to solve a broad range of tasks, such as guiding metabolic engineering and aiding in the interpretation of high-throughput data [24].

2.3.1 Model reconstruction

The first genome-scale metabolic reconstructions were primarily based on biochemical data compiled for model organisms. As a result, most of the early published reconstructions were for commonly studied model organisms, such as *Escherichia coli* [3, 26]. However, by utilizing annotated genomes, it has since been possible to create metabolic reconstructions for much less well-studied organisms [26].

The process of creating a GSM is essentially iterative. The first step is to create an initial metabolic model based on the annotated genome, meaning that reactions are added if regions of the genome are annotated as encoding enzymes capable of catalyzing those reactions [26]. The second step is to improve the quality of the initial model to include organism-specific features and to expand on the initial set of reactions. This is typically done by integrating knowledge from other sources, such as biochemical data or organism-specific databases [26, 3]. Reactions needed to be able to simulate growth are added here, such as non-enzymatic reactions. Compartments are also included, where a split between the intracellular and extracellular environment is the most basic one. In order to support having different compartments, transport reactions have to be added [2]. Another important part of this step is the determination of biomass composition, as this is critically important for simulating growth [14, 2]. The third step is to convert the reconstruction to an actual computational model. The final step is a debugging step, where the performance



1. Draft Reconstruction
1| Obtain genome annotation.
2| Identify candidate metabolic functions.
3| Obtain candidate metabolic reactions.
4| Assembly of draft reconstruction.
5| Collect of experimental data.

2. Refinement of reconstruction
6| Determine and verify substrate and cofactor usage
7| Obtain neutral formula for each metabolite.
8| Determine the charged formula.
9| Calculate reaction stoichiometry.
10| Determine reaction directionality.
11| Add information for gene and reaction localization.
12| Add subsystems information.
13| Verify gene-protein-reaction association.
14| Add metabolite identifier.
15| Determine and add confidence score.
16| Add references and notes.
17| Flag information from other organisms.
18| Repeat Step 6 to 17 for all genes.
19| Add spontaneous reactions to the reconstruction.
20| Add extracellular and periplasmic transport reactions.
21| Add exchange reactions.
22| Add intracellular transport reactions.
23| Draw metabolic map (optional).
24 -32| Determine biomass composition.
33| Add biomass reaction.
34| Add ATP maintenance reaction (ATPM).
35| Add demand reactions.
36| Add sink reactions.
37| Determine growth medium requirements.

3. Conversion of reconstruction into computable format
38| Initialize the COBRA toolbox.
39| Load reconstruction into Matlab.
40| Verify S matrix.
41| Set objective function.
42| Set simulation constraints.

4. Network evaluation
43-44| Test if network is mass- and charge balanced.
45| Identify metabolic dead-ends.
46-48| Gap analysis.
49| Add missing exchange reactions to model.
50| Set exchange constraints for a simulation condition.
51-58| Test for stoichiometrically balanced cycles.
59| Re-compute gap list.
60-65| Test if biomass precursors can be produced in standard medium
66| Test if biomass precursors can be produced in other growth media.
67-75| Test if model can produce known secretion products
76-78| Check for blocked reactions.
79-80| Compute single gene deletion phenotypes
81-82| Test for known incapabilites of the organism.
83| Compare predicted physiological properties with known properties.
84-87| Test if the model can grow fast enough.
88-94| Test if the model grows too fast.

Data assembly and Dissemination
95| Print Matlab model content.
96| Add gap information to the reconstruction output.

Figure 2.2: A stepwise list of the steps involved in producing a metabolic reconstruction. Figure taken from [2].

of the computational model is evaluated by validating model behavior. Information from this step is used to go back to step two, where missing reactions (gaps) are identified and filled in. This can be done by identifying incomplete pathways, and filling in the missing reaction(s). Gap-filling is one of the more challenging parts of assembling a metabolic reconstruction, and should preferably only be done when there is some sort of evidence that the gap actually should be filled. For example, filling in a gap in order to enable lysine synthesis should only be done if the organism is known to be able to synthesize lysine [2]. This basic workflow is illustrated in figure 2.2. The key here is that the second step, refinement of reconstruction, is a circular process, where the results from the model validation step is used to guide further manual curation [2, 26].

The Biomass Reaction

The biomass reaction is an important part of any metabolic reconstruction, and the composition of it is generally determined during the initial refinement step of the process outlined above [26]. The biomass reaction is based on the macromolecular makeup of exponentially growing cells. The first step is to identify the fraction of the main macromolecular groups, i.e. DNA, RNA, lipids and protein, and then to calculate the amounts of metabolites in the metabolic reconstruction needed to produce a unit of biomass [27, 26]. More advanced biomass reactions can also be created by incorporating information about the amount of energy needed to produce the building blocks of the biomass reaction, or by adding in micronutrients, like vitamins and cofactors [27].

2.3.2 The Stoichiometric Matrix

The core of a GSM is the stoichiometry of the organism's metabolic machinery, which can be expressed mathematically in several ways. As an example, consider a toy system consisting of the following reactions:

$$\emptyset \xrightarrow{v_1} A \tag{2.16}$$

$$\emptyset \xrightarrow{v_2} B \tag{2.17}$$

$$A + B \xrightarrow{v_3} C \tag{2.18}$$

$$C \xrightarrow{v_4} D \tag{2.19}$$

$$D \xrightarrow{v_5} \emptyset \tag{2.20}$$

This toy system consists of four metabolites (A,B,C,D) and five reactions. v_1, v_2, v_3, v_4 and v_5 denote the the reaction fluxes for the corresponding reaction (r_1, r_2, r_3, r_4 and r_5). In terms of a cell, equations 2.16-2.17 can be interpreted as exchange reactions for metabolites A and B. Exchange reactions are reactions where the cell takes up metabolites from the environments, and these can be used to define the extracellular environment a GSM is opearting in. Meanwhile, equation 2.20 is an excretion reaction where metabolite D is expelled into the environment. A graphical depiction of this system can be seen in figure 2.3, where the nodes are metabolites, the edges are reactions and the border symbolizes the division of the cell from the environment.

The stoichiometry of this system can be represented with matrix notation, where S is a 5×4 matrix:

$$S = \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{array} \begin{array}{c} \begin{array}{cccc} A & B & C & D \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \\ 0 & 0 & 0 & -1 \end{pmatrix} \end{array} \tag{2.21}$$

where $S_{i,j}$ denotes the change in concentration of metabolite j caused by reaction i per unit of flux in reaction r_i.

This matrix notation can be extended to a generalized metabolic reconstruction, where S, the stoichiometric matrix, is a $m \times n$ matrix, where m is the number of metabolites in the reconstruction and n is the number of reactions [3, 14]. The matrix encodes the stoichiometry of the metabolic reconstruction, and can generally be expressed as:

$$S_{m,n} = \begin{pmatrix} s_{1,1} & s_{1,2} & \cdots & s_{1,n} \\ s_{2,1} & s_{2,2} & \cdots & s_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ s_{m,1} & s_{m,2} & \cdots & s_{m,n} \end{pmatrix} \tag{2.22}$$

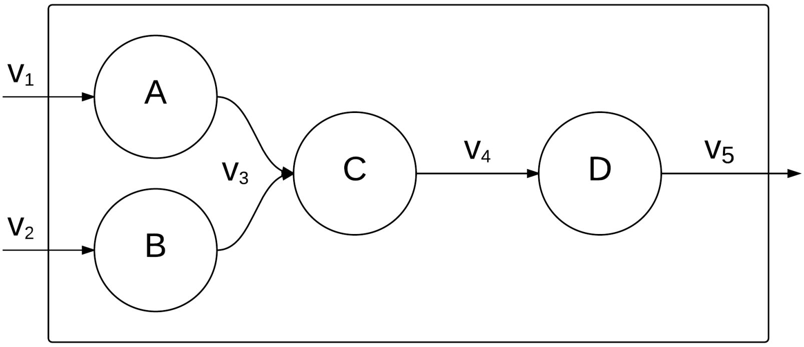


Figure 2.3: A graphical depiction of a toy stoichiometric matrix. The metabolites A and B are imported with exchange reactions with reactions V_1 and V_2. A pathway consisting of two reactions V_3 and V_4 leads accumulation of D, which is then exported out of the system with reaction V_5. Figure adapted from [3].

where $S_{i,j}$ is defined in the same way as in the toy example above.
With this definition of the S matrix, the physiological capabilities of the metabolic reconstruction can be explored. In particular,

$$\frac{d\mathbf{x}}{dt} = \mathbf{Sv} \tag{2.23}$$

where $\mathbf{v} = \begin{pmatrix} v_1 \cdots v_n \end{pmatrix}$, is the flux matrix, and $\mathbf{x} = \begin{pmatrix} x_1 \cdots x_m \end{pmatrix}$ is the vector of concentration. S is then a linear transformation on v, transforming the flux vector into a vector of time derivatives of the concentration [3].

2.4 Flux balance analysis

One of the most successful applications of GSMs has been to perform Flux Balance Analysis (FBA) [24]. FBA is a mathematical framework enabling the simulation of metabolic phenotypes and growth [14]. The aim of FBA is to simulate exponentially growing cells, and makes the assumption that the cell reaches a steady state where there is no net change in any of the metabolite concentrations. This is equivalent to the time derivative of the concentration vector being equal to the null vector, enabling the calculation of metabolic phenotypes through linear programming:

$$\textbf{max/min}\ z = \mathbf{c}^T\mathbf{v} \tag{2.24}$$

$$\mathbf{Sv} = \mathbf{0} \tag{2.25}$$

$$lb_i \geq v_i \leq ub_i, i \in n \tag{2.26}$$

where $\mathbf{S}_{m,n}$ is the stoichiometric matrix for the metabolic reconstruction being optimized, **c** is a $1 \times m$ vector encoding the objective function coefficients, **v** is the flux vector, lb_i and ub_i are the lower and upper bounds of flux i, and z is the value of the objective function.

Typically the flux bounds are set to be essentially unrestricted, except for the exchange reactions where the bounds are used to simulate the growth medium [14].

FBA typically optimizes for growth, and the objective function is therefore normally either the biomass reaction itself or some sort of analogue [14]. The solution of an FBA problem then gives a prediction for the growth rate (through the biomass reaction) as well as predictions for the flow of metabolites across the metabolic network. It should be noted, however, that the flux vector predicted by FBA is generally not unique, and there is typically an infinite number of alternate optimal flux vectors [28]. The exact amount of variation in an optimal flux vector varies and is dependent on the network structure and the environmental conditions (i.e. the flux bounds) [28, 3]. As a consequence of this, the predicted flux vector should be considered critically and not be interpreted as representing the actual internal flux distribution of the organism being simulated [28].

2.4.1 Phenotype Phase Plane & the Line of Optimality

Computing a growth rate and an associated flux vector with FBA is trivial when the exchange reaction fluxes are unbounded. This does not, however, produce results that can be validated experimentally. In order to produce physiologically meaningful growth rate predictions, the bounds of nutrient uptake exchange reactions have to be constrained according to experimentally measured uptake rates for the organism being modelled [3, 14]. If the organism has been allowed to adapt to the environment it's being grown in, a high quality FBA model can usually predict the growth rate with high accuracy [14, 24, 29, 1].

FBA can still be used to examine the metabolic capabilities of the metabolic reconstruction when experimental measurements of uptake rates are unavailable, however. A common method for evaluating the phenotype is by examining the phenotype phase plane (PhPP).

PhPP analysis is a method for evaluating the effect of two separate metabolite uptake rates on the growth rate of the model at the same time. In PhPP analysis, the fluxes for the uptake exchange reactions for the metabolites being examined are locked to predetermined values by setting the lower bound equal to the upper bound. A plane is then generated by examining the growth rate associated with each combination of uptake rates for the two metabolites [3]. An example of a PhPP of *Saccharomyces cerevisiae* can be seen in figure 2.4, where the two exchange reactions being examined are the uptake rates for glucose and oxygen. PhPP analysis useful, as it allows for the identification of phenotypically distinct phases. A phase is defined as an area of the PhPP where the model predicts the same metabolite excretion profile as well as the same relative growth response when the two uptake rates are varied. Phases can be identified by examining the ratios of the shadow prices associated with the two constraints [3]:

$$\alpha = -\frac{\pi_x}{\pi_y} \tag{2.27}$$

where π_x and π_y are the shadow prices associated with exchange reaction x and y respectively. α can be used to interpret the PhPP, as it stays constant within a phenotypic phase. The sign of α is also informative, as it can be used to characterize the effect the metabolites being analyzed are having on the growth rate. A negative sign indicates that both exchange reactions limit the growth rate, whereas a positive sign indicates a futile phase,

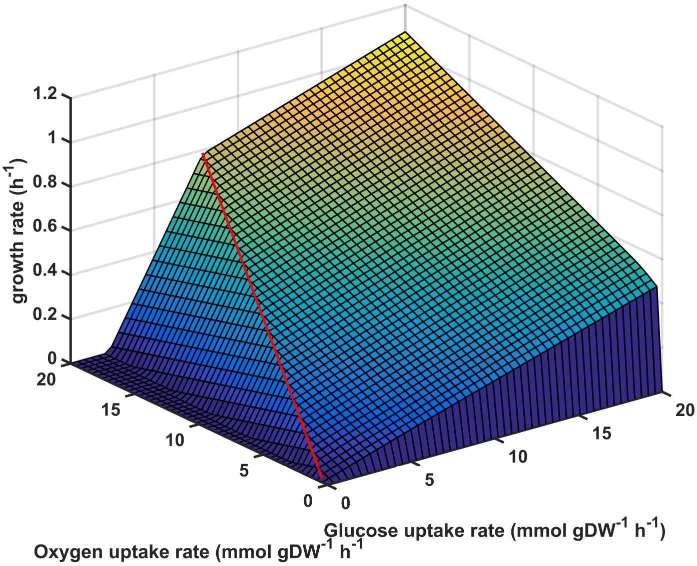


Figure 2.4: FBA phenotype phase plane of the iTO977 metabolic reconstruction of *S. cerevisiae*. Each point in the plane gives the growth rate when glucose and oxygen is fixed. The red line is the line of optimality, and shows the area of the phenotype phase plane where the carbon source is fully oxidized.

where one of the uptake rates limits the growth rate. A phase characterized by a positive α exhibits wasteful metabolism, and phases like these are thought to be unstable in actual organisms. In fact, experimental evidence suggests that organisms exhibiting this sort of wasteful metabolism prior to adaptation to its environment tend to move away from it in the short term adaptation[3, 30, 24] . If either π_x or π_y is zero, α will be either zero or undefined. This represents a state where one of the metabolites has no effect on the growth rate. α fails to identify infeasible phases, but these can easily be identified as regions of the PhPP with zero growth. Finally, if x is the uptake of a carbon source, the line along which the carbon source is fully oxidized can be seen in the PhPP as the line where, for a given value of the carbon uptake rate, the growth rate is maximal. This is known as the line of optimality and represents the optimal usage of the two metabolites being considered [3]. The expectation is then that, if metabolism is optimally maximized for growth, experimental measurements of the uptake rates along with the growth rate should should lie along the line of optimality [30]. Figure 2.5a is a plot of α for the PhPP in figure 2.4. This illustrates that this PhPP is split into two feasible phases. The actual placement of these phases can be seen in 2.5b. The first, P_1, is characterized by a positive α, indicating that one of the metabolites is in excess. In this case, the excess metabolite is oxygen and the phase is characterized by a reduced growth rate due to the need to dissipate excess oxygen. P_2 has a negative α which indicates that the growth rate is limited by both glucose and oxygen. N is an infeasible phase where no growth is observed. The line of optimality splits P_1 and P_2, and can also be seen in figure 2.4 as a red line in the PhPP.

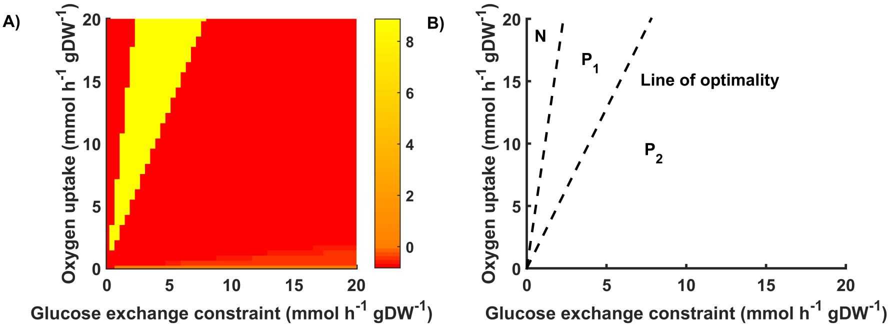


Figure 2.5: A) The ratio of the shadow prices for the glucose and oxygen exchange reactions for the yeast metabolic reconstruction iTO977. A positive value means that the shadow prices have opposite signs, meaning that one of them is constraining the growth rate by being too high. A negative value means that both substrates are constraining the growth. B) The boundaries for the different phases. P_1 is the futile phase, while P_2 is limited by both metabolites.

2.4.2 Flux coupling analysis

The metabolic network obtained from the reconstruction can also be studied on a purely topological level. One interesting question here is to what degree different reactions in the network are coupled together when the steady state assumption ($\mathbf{Sv} = \mathbf{0}$) holds. Flux coupling analysis is a method for answer this question [31].

The aim of flux coupling analysis is to classify the topological relationship between every pair of reactions, v_i and v_j, into one of four categories [31]:

1. Fully coupled: v_i is fully coupled to v_j if, for every valid flux through v_i, there is a fixed corresponding flux through v_j and the other way around.

2. Partially coupled: v_i is fully coupled to v_j if a non-zero flux in v_i necessitates a non-zero flux in v_j and the other way around.

3. Directionally coupled: v_i is partially coupled to v_j if a non-zero flux in v_i necessitates a non-zero flux in v_j, but not the other way around.

Various software packages are available that can perform flux variability analysis [31, 32, 33]

2.5 FBA with internal constraints

Basic FBA with growth rate maximization has become the field standard for applying metabolic reconstructions, but numerous extensions have been developed [24]. Many of these extensions attempt to further constrain the optimal solution flux distribution beyond what is imposed by the network structure of the metabolic reconstruction [34, 5, 35, 15].

One approach has been to add some sort of internal constraint on the flux sum, which has the effect of drastically shrinking the optimal flux distribution [34]. These methods are useful for predicting growth rates without having nutrient uptake rates under some conditions. Two intimately related methods for doing this are known as Flux Balance Analysis with Mass Constraints (FBAwMC) [15] and MetabOlic Modeling with ENzyme kineTics (MOMENT) [5].

2.5.1 FBAwMC

FBAwMC is a basic extension of FBA based on the assumption that the permissible flux through a reaction is based on the abundance of the enzyme catalyzing that reaction, and that any enzyme takes up a certain amount of volume. As the cytoplasm is crowded by a variety of macromolecules, only a certain amount of the volume is available to enzymes. The total volume of the cell is then used to constrain the enzyme concentrations [15, 15]. Mathematically, for a metabolic reconstruction with n reactions and where G is the set of indices for the g enzymatic reactions, this can be done by adding g variables and adding the following constraints to the FBA problem in equations 2.24-2.26 [15]:

$$\sum_{i \in G} \lambda_i E_i \leq \frac{1}{C} \tag{2.28}$$

$$\sum_{i \in G}^{n} a_i v_i \leq 1 \tag{2.29}$$

where λ_i is the molecular volume of enzyme i, E_i is the concentration of enzyme i, C is the cytoplasmic density of the organism being modelled, and $a_i = \frac{C\lambda_i}{b_i}$ is the crowding coefficient of reaction i, where b_i is a kinetic parameter. Equation 2.28 constrains the total enzyme concentration, while equation 2.29 constrains the reaction fluxes by incorporating the molecular crowding coefficient. The flux distribution constraint does not constrain reactions individually, but rather constrains the flux sum in aggregate.

FBAwMC has been demonstrated to be able to predict growth rates for *E. coli* in a limited set of single-substrate limited media without experimental measurements of nutrient uptake rates [31]. It should be noted, however, that the average value of the *E. coli* crowding coefficient was used for every reaction in the model due to a dearth of accurate measurements for *E. coli* enzymes [31].

2.5.2 MOMENT

As an alternative to using molecular crowding coefficients, MOMENT was developed to incorporate kinetic data directly [5]. MOMENT takes advantage of the gene-to-reaction mapping available in metabolic reconstructions, and constrains individual fluxes rather than the flux distribution in aggregate. MOMENT assumes a steady state (equations 2.26-2.25), as in an FBA or FBAwMC problem. The turnover number for a reaction i, $kcat, i$, is incorporated into the model along with a variable representing the concentration of the enzyme catalyzing the reaction, g_i, by using the V_{max} equation (equation 2.15). For any enzyme catalyzed reaction i, it then adds one of the following three constraints to the optimization problem [5]:

$$v_i \leq kcat_i g_i \tag{2.30}$$

$$v_i \leq kcat_i(g_a + g_b) \tag{2.31}$$

$$v_i \leq kcat_i \text{min}(g_a, g_b) \tag{2.32}$$

where $kcat$ is the turnover number of the enzyme catalyzing the reaction. Turnover numbers derived from the organism being studied are used when available, otherwise the turnover number is estimated from other organisms or by the average of all the other turnover numbers in the model. Equation 2.30 is used when the reaction is catalyzed by the gene product of a single gene, 2.31 when the reaction can be catalyzed by two (or more) different gene products and 2.32 when the reaction is catalyzed by a complex consisting of two gene products. MOMENT also allows for more complex gene-to-reaction relationships, and handles this by introducing various auxiliary variables [5]. These constraints have the same role as equation 2.29 in FBAwMC. Turnover numbers can be obtained from databases like BRENDA [36] and SABIO-RK [37]. In order to constrain the enzyme concentrations, MOMENT uses a capacity constraint similar to equation 2.28:

$$\sum_{i \in G} \text{MW}_i g_i \leq C \left[\frac{g_{protein}}{g_{DW}} \right] \tag{2.33}$$

where MW_i is the molecular weight of enzyme i, and C is the fraction of the total dry weight which is composed of protein. Accurate values for the enzyme molecular masses can be obtained from various databases, such as UniProt [38], or estimated from the gene sequence.

MOMENT also differs from FBAwMC in its choice of objective function. Rather than optimizing the growth rate directly, MOMENT is formulated as a nonlinear non-convex optimization problem [5]:

$$\text{max } z = \frac{v_{\text{ATP}}}{\sum_{i=1}^{n} v_i^2} \tag{2.34}$$

where v_{ATP} is the ATP yield. The logic behind this objective function is based on the hypothesis that at optimal metabolic states, ATP production is maximized while enzyme production is kept minimal [39]. In practice, a quadratic programming approximation of this objective function is used when actually optimizing the model [5].

In contrast to FBAwMC, both of the two key model parameters, $kcat$ and MW, can be obtained from publicly accessible databases. As a consequence of this, MOMENT is able to predict growth rates for *E. coli* across a much more diverse set of growth media than FBAwMC [5].

2.6 *Saccharomyces cerevisiae*

Saccharomyces cerevisiae, also known as baker's yeast, has been a workhorse for genetics and biochemistry research since the early twentieth century [40], and is one of the standard model systems for studying eukaryote molecular biology [11]. *S. cerevisiae* is a unicellular fungus that can also grow as long filaments under certain conditions [40]. It has several properties making it an ideal model organism, such as the ability to be grown on inexpensive media and a quick doubling time of around 90 minutes [11]. The S288C strain of *S. cerevisiae* was the first eukaryote genome to be fully sequenced [41], and this strain serves as one of the reference strains for the species. The S288C genome consists of 12 million base pairs, across 16 haploid chromosomes.

In addition to being a model organism for scientific work, *S. cerevisiae* also has significant industrial and cultural significance. This is due to its ability to perform ethanol fermentation, which has been used to produce alcoholic beverages for over 7000 years [40, 11]. Figure 2.6 shows an overview of the ethanol fermentation pathway. An interesting aspect of *S. cerevisiae* metabolism is that fact that fermentation occurs even in the presence of oxygen at high concentrations of glucose. This is known as the Crabtree effect and is an example of respiro-fermentative metabolism, where fermentation and respiration occur at the same time [42].

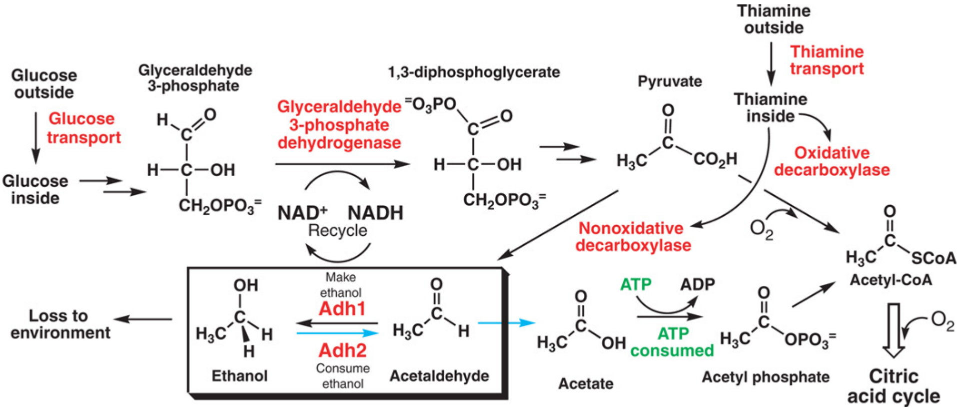


Figure 2.6: The ethanol and acetate fermentation pathway in *S. cerevisiae*. Although not shown on the figure, the conversion of acetaldehyde to acetate reduces NAD(+) to NADH. Taken from [4].

2.6.1 Metabolic reconstructions of S. cerevisiae

While the number of organism with available metabolic reconstructions continues to grow, *S. cerevisiae* remains one of the the only organism with metabolic reconstructions that have gone through multiple revisions and rounds of improvement over a time scale of decades [29]. The biggest currently available model is the consensus model, known as the Yeast 7 model, signifying it as the 7th published iteration of the *S. cerevisiae* consensus model project, and consists of 916 genes, 3493 reactions and 2218 metabolites [43]. An alternative to the Yeast 7 model is the smaller iTO977 model, consisting of 977 genes, 1566 reactions and 1353 metabolites [1]. While the Yeast 7 model has nearly twice as many reactions as iTO977, the majority of the difference comes from Yeast 7 having more duplicate reactions to accommodate the larger number of compartments. While the Yeast 7 model is more complex, it has not been demonstrated to predict growth rates more accurately than iTO977 [29]. iTO977 was based on the original yeast consensus network and iIN800, and further curated and refined to support integration with OMICS data [1]. The phylogeny of iTO977 can be seen in figure 2.7.

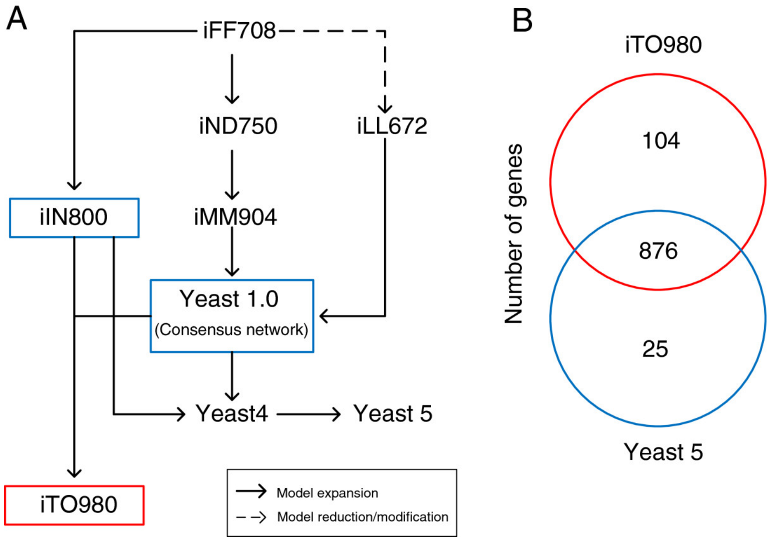


Figure 2.7: The phylogeny of the iTO977 model. iTO977 was sourced based on two previously published models, Yeast 1.0 and iIN800. Figure taken from [1].

2.7 Hierarchical clustering

Hierarchical clustering (HC) is a form of cluster analysis, where the goal is to group objects into clusters, preferably in a hierarchy so that more similar clusters are grouped together [44]. The following description of HC is based on Friedman [44].

HC methods can theoretically be performed on any data set where a numerical dissimilarity data points can be obtained. This is done by considering the pairwise dissimilarity between groups of data points. The aim of HC is then to create a hierarchical representation of the data set, where at the lowest level of the hierarchy, every data point is represented as its own cluster, while at the highest point every data point is included in a single cluster. This can be represented as a rooted tree, where the root is the entire data set, the nodes are clusters and the leaf nodes are the individual data points. The pairwise dissimilarity between clusters increases monotonically at each level compared to the one below it. The tree can then be viewed graphically by making the height of the nodes in a given level of the tree proportional to the pairwise dissimilarity between clusters. Plots of this type are known as dendrograms, and are useful for interpreting the results of HC. A dendrogram for a data set where the data points are vectors of random numbers can be seen in figure 2.8.

This clustering can be done in either a bottom-up or top-down method. The bottom-up method works by starting at the lowermost level, where every data point is its own cluster, and creating the next level by merging the two clusters that are the least dissimilar. On

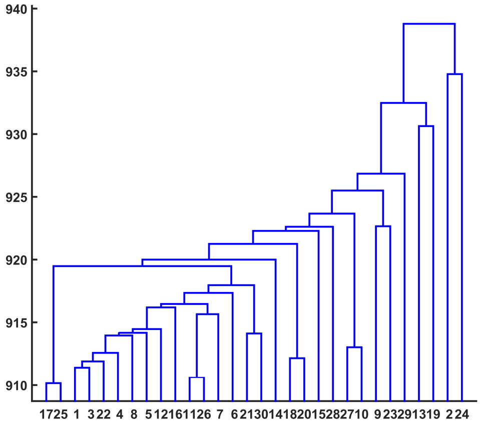


Figure 2.8: A dendrogram for a matrix of randomly generated numbers.No real clustering pattern can be seen.

the other hand, top-down methods are recursive and work by starting at the top and then splitting one cluster to create the lower level. The split cluster is chosen to maximize the between-cluster dissimilarity between the two resulting clusters.

In order to perform either top-town or bottom-up HC, some sort of cluster dissimilarity measure is required. For two clusters G and H, the dissimilarity can be expressed as $d(H,G)$ where $d_{i,j}$ is the dissimilarity measure between the two data points i and j. For bottom-up HC, the aim is to minimize $d(H,G)$. The three most popular methods for doing bottom-up HC are single linkage, complete linkage and average linkage, These methods differ by how $d(H,G)$ is defined:

- Single linkage: $d(H,G) = \min\limits_{i \in H, j \in G} d_{i,j}$
- Complete linkage: $d(H,G) = \max\limits_{i \in H, j \in G} d_{i,j}$
- Average linkage: $d(H,G) = \frac{1}{N_G N_H} \sum_{i \in G} \sum_{j \in H} d_{i,j}$

where N_G and N_H are the number of data points in G and H respectively. Single linkage merges the clusters with the most similar pair of data points (nearest neighbor), complete linkage merges the two clusters where the two most dissimilar data points are most similar, while average linkage merges the two clusters that are the most similar on average. These three methods tend to produce similar results for data sets with a high degree of natural clustering.

The quality of the resulting dendrogram can then be evaluated with the cophenetic correlation coefficient. This is a measure of how well the pairwise dissimilarity, $d_{i,j}$, is correlated with the between-cluster dissimilarity from when clusters containing i and j were first merged in the construction of the dendrogram. The closer this value is to 1, the better the dendrogram maintains the original pairwise distance between the data points

[45]. There is no pre-defined procedure for deciding on where the cut a dendrogram to get the best clustering, but several approaches exist. The most basic approach is to simply make a horizontal cut in the dendrogram to get an arbitrary number of clusters based on what ”looks” right [44]. A less subjective alternative is to use an inconsistency metric. This is a way of quantifying how high a link in a dendrogram is compared to other links at the same level. Links scoring high on the inconsistency metric are then said to be inconsistent, while low scoring links are consistent. The justification for this approach is that high links form when dissimilar clusters are merged during the construction of the tree. As such, measuring the linkage inconsistency and using it to cut the dendrogram avoids the issues associated with just cutting based on the eyeball test [46, 47].

Chapter 3

Methods & Software

3.1 Software

A brief overview of the different types of software used in this project will be given here.

3.1.1 Matlab

MATLAB is a proprietary programming language designed for numerical computation. Data analysis was performed by using the MATLAB 2015b Statistics and Machine Learning Toolbox [48]. The following external MATLAB packages were used for metabolic model simulation and analysis.

COBRA toolbox

The COnstraints Based Reconstruction and Analysis (COBRA) toolbox version 2.0.6 is a MATLAB package for performing constraint-based analysis on metabolic reconstructions [49], was used for all growth simulations for this project. When using COBRA, metabolic models are represented as objects where the fields are the various components of the metabolic reconstruction. When possible, the methods developed in this project were designed to use the existing COBRA data types to ensure compatibility with the standard COBRA methods.

F2C2

Fast Flux Coupling Calculator (F2C2) is a COBRA toolbox-compatible MATLAB package for performing flux coupling analysis [32]. F2C2 takes the stoichiometric matrix as input, and returns a matrix recording the flux coupling relationships as well as a list of blocked reactions. Due to F2C2 not supporting the Gurobi solver, the GLPK solver [50] was used instead.

3.1.2 Gurobi

Gurobi is a proprietary mathematical programming solver [51]. Gurobi has an easily accessible MATLAB interface, and recent versions of the COBRA toolbox [49] has added support for Gurobi. Gurobi 6.0.5 was used to solve all LP and QP problems in this project.

3.1.3 Python

All automated accession of external databases and processing of the the resulting data was done with written scripts in Python. The version of Python used was Python 2.5 [52].

3.2 ircFBA

FBAwMC and MOMENT are both able to make predictions for the maximal growth rates for *E. coli* in limited growth media with glucose as the only carbon and energy source [5, 34], but so far this approach has not been extended to organisms other than *E. coli*. Internally Constrained FBA (ircFBA) was developed to be an alternative to MOMENT [5] and FBAwMC [15] that could be easily applied to existing metabolic reconstructions for organisms other than *E. coli*. ircFBA uses the mathematical description of the MOMENT problem as a basis, but employs linear programming to optimize for growth, simplifies the representation of complexes and makes the further assumption that an individual enzyme molecule can only carry flux in a single reaction. It does, however, use the same enzyme concentration constraint as MOMENT.

Instead of defining the concentration of a complex, G_c, consisting of two enzymes, G_a and G_b, as $\min{(G_a, G_b)}$, ircFBA simply introduces G_c as its own variable. The molecular weight of the complex is then defined as

$$MW_c = MW_a + MW_b \tag{3.1}$$

where MW_i is the molecular weight of enzyme i. For a reaction catalyzed by isozymes, i.e. by either G_a or G_b, ircFBA again uses a single variable. The molecular weight corresponding to this variable is then:

$$MW_{ab} = \min{(MW_a, MW_b)}. \tag{3.2}$$

where MW_{ab} is the molecular weight corresponding to the new variable. If either G_a or G_b is a complex, equation 3.1 is first applied. For a metabolic model with m metabolites, n reactions, p enzymatic reactions, and G is the set of indices for the enzymatic reactions, N is the set of indices for the non-enzymatic reactions, ircFBA adds p enzyme concentration variables, $\mathbf{g} = [g_1 \cdots g_G]$. The full ircFBA problem can now be stated as:

$$\max z = v_{\text{biomass}} \tag{3.3}$$

$$\sum_{j=1}^{n} S_{i,j} v_j = 0, i = 1, 2, ...n \tag{3.4}$$

$$\text{lb}_j \geq v_j \leq \text{ub}_j, j \in N \tag{3.5}$$

$$-kcat_i g_i \geq v_i \leq kcat_i g_i, i \in G \tag{3.6}$$

$$\sum_{i \in G} \text{MW}_i g_i \leq C \left[\frac{g_{protein}}{g_{DW}} \right] \tag{3.7}$$

$$g_i \geq 0, i \in G \tag{3.8}$$

where v_{biomass} is biomass reaction flux, S is the stoichiometric matrix, lb_j and ub_j are the lower and upper bounds on v_j, $kcat_i$ is the turnover number for reaction i, MW_i is the molecular weight of the enzyme catalyzing reaction i, and C is the fraction of the dry weight of the organism devoted to protein. Since every enzyme associated reaction has its own protein concentration variable in ircFBA, the concentration of a multifunctional enzyme can be calculated as the sum of the variables representing it. However, these are all independent in ircFBA, and an individual g_i simply represents the concentration of enzyme devoted to catalyze reaction i. The justification for this is equation 2.15. If the enzyme capacity constraint (equation 3.7) limits the growth rate, any enzymatic flux value, v_i, will satisfy $|v_i| = kcat_i g_i$. If not, a lower g_i would allow for an increase in the growth rate by allocating more protein concentration to another enzyme. The enzymes will therefore always be saturated. In MOMENT, however, g_i represents the concentration of protein i as a whole. A consequence of this is that a unit of concentration of enzyme can produce a flux higher than its v_{max} value. The exact details of MOMENT's implementation are not entirely clear from the publication [5], and while the article states that an implementation is available online, the files are missing from the website. Additionally, attempts at contacting the authors to resolve this issue were unsuccessful. It's therefore possible that MOMENT deals with multifunctional enzymes in ways not apparent in the article.

3.3 Assembly of an ircFBA model based on iTO977

As no FBAwMC or MOMENT models have been published for *S. cerevisiae*, an ircFBA model based on the iTO977 metabolic model was assembled and implemented for simulation with the COBRA toolbox. To do this, data from multiple databases were gathered and integrated. The process for creating an ircFBA model was split into two phases. The first phase was a data gathering phase, where EC numbers, molecular weights and turnover numbers for the enzymatic reactions in iTO977 were obtained and processed. The second phase involved data integration, where the actual parameter values were chosen.

3.3.1 Data gathering

ircFBA requires two sets of parameters: the molecular weights of the enzymes and turnover numbers. Molecular weights can be obtained fairly trivially from various databases, while turnover number retrieval is more complicated. EC numbers are available for most enzymatic reactions in the iTO977 model, but a significant portion lacked any annotation of this sort. In this section, the approaches used to obtain these parameters will be described.

Molecular Weight retrieval

The fungiDB *S. cerevisiae* S288C gene information table is a comprehensive data file of functional annotation for every single known gene from this organism [53]. The gene IDs for every gene in the iTO977 SBML file were extracted, and used to find the correct molecular weight, in units of kDa, in the gene information table. As the molecular weights in the fungiDB gene information table are based on experimentally determined molecular weights of the actual gene products, these values are more accurate than if weights were estimated based on the genome sequence.

EC number retrieval

For enzymatic reactions in the iTO977 SBML file without EC number annotation, a list of potential EC numbers were obtained from two sources. First, the gene IDs were used to find possible EC numbers in the fungiDB gene information table. The gene information table contains two fields for EC numbers: manually annotated EC numbers, and EC numbers from OrthoMCL. OrthoMCL is a method for finding orthologs for genes from eukaryote genomes [54]. The OrthoMCL EC numbers are therefore derived on sequence similarity to genes from other genomes that have had the EC number in question assigned. These EC numbers are therefore less certain. The UniProt IDs for the proteins were also extracted from the gene information table, and additional EC numbers were extracted from UniProt [38]. The resulting list of EC numbers were then stored in a CSV file, with the OrthoMCL EC numbers kept separate from the more reliable manually assigned EC numbers.

Turnover number retrieval

Turnover numbers were then retrieved from the BRaunschweig ENzyme DAtabase (BRENDA), database compiling enzyme and metabolic data from the scientific literature [36]. BRENDA was accessed programmatically through its SOAP web service with the SOAPpy Python package [55]. A list of every EC number in the iTO977 model was extracted from the SBML file and combined with the EC numbers from fungiDB and UniProt. SOAPpy was used to extract every single turnover number in BRENDA associated with one of these EC numbers. The BRENDA results were in the form of structured strings, containing not just the turnover number itself, but additional meta data such as the source organism, the enzyme substrate, and information about the experimental conditions. The BRENDA results also contain a reference to the compound database pubchem [56] in order to disambiguate the substrate name. A data file containing all the BRENDA database hits was created and stored locally.

3.3.2 Data integration

To integrate the data from the data gathering stage, a method for connecting a BRENDA database hit with a reaction in the iTO977 model had to be developed. This was done by matching the metabolite name in the model against the enzyme substrate name. A multi-step approach for achieving this was developed.

Compound thesaurus assembly

In order to match substrates from a BRENDA entry against a chemical species in the iTO977 model, two metabolite thesauruses were designed.The first thesaurus served as a list of different synonyms for the metabolites included in the iTO977 model, while the second one gives a list of synonyms for substrates derived from BRENDA entries. Both thesauruses were implemented as hash tables in Python, with the keys being synonyms for a metabolite, and the value being the main identifier. The ChEBI name was chosen as the main identifier when applicable. The basic layout of the hash table is illustrated in figure 3.1.

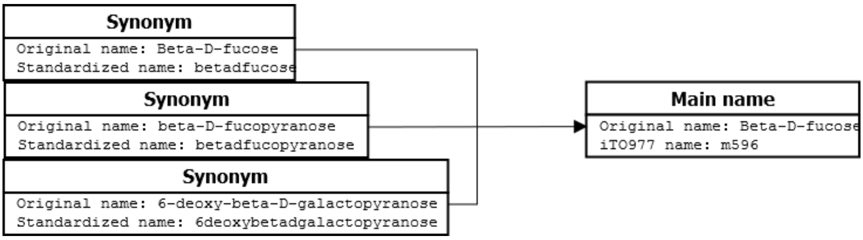

Synonym
Original name: Beta-D-fucose
Standardized name: betadfucose
Synonym
Original name: beta-D-fucopyranose
Standardized name: betadfucopyranose
Synonym
Original name: 6-deoxy-beta-D-galactopyranose
Standardized name: 6deoxybetadgalactopyranose
Main name
Original name: Beta-D-fucose
iTO977 name: m596

Figure 3.1: The structure of the compound thesaurus. The outer layer of the thesaurus have links to the main entry. The thesaurus itself consists of synonyms for most of the compounds in the iTO977 model.

Synonyms for non-model specific metabolite in the iTO977 model were gathered from ChEBI [57] and the KEGG COMPOUND [7] database by using the appropriate database IDs when provided in the model's annotation. While the majority of metabolites in the iTO977 were annotated with IDs for compound databases, several non-model specific metabolite lacked any sort of database reference. Database references were added to these metabolites by looking them up manually in ChEBI or KEGG COMPOUNDS when applicable. Additionally, the cheEBI Python API was used to expanded the thesaurus by considering the conjugate base or acid of a metabolite to be identical to the metabolite in question. This was done by taking advantage of the cheEBI ontology, where a compound is linked to its conjugate acid and base. Thus, the thesaurus entry for a given species was expanded with the synonyms of any conjugate acid or base. Furthermore, every entry in the thesaurus was standardized by removing any capitalization, parentheses, brackets and any other special character, thus representing every species as a lowercase string consisting only of letters and numbers.
A similar thesaurus was created for the substrates from every BRENDA entry. This was done by examining the attached pubchem ID. In the cases where a KEGG or ChEBI ID

was listed as a synonym for a substrate in its pubchem entry, the synonyms reported by ChEBI and KEGG were added along with the other pubchem synonyms.

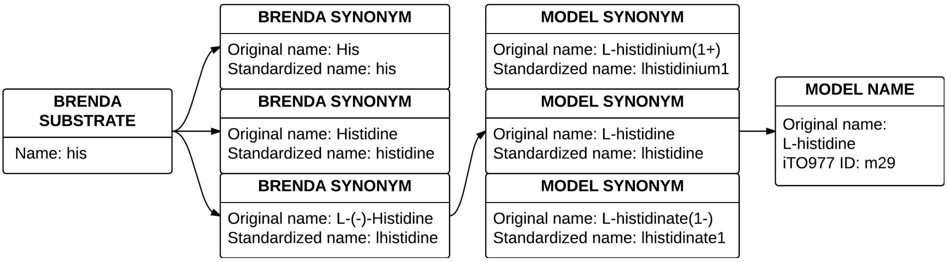

BRENDA SUBSTRATE
Name: his
BRENDA SYNONYM
Original name: His
Standardized name: his
BRENDA SYNONYM
Original name: Histidine
Standardized name: histidine
BRENDA SYNONYM
Original name: L-(-)-Histidine
Standardized name: lhistidine
MODEL SYNONYM
Original name: L-histidinium(1+)
Standardized name: lhistidinium1
MODEL SYNONYM
Original name: L-histidine
Standardized name: lhistidine
MODEL SYNONYM
Original name: L-histidinate(1-)
Standardized name: lhistidinate1
MODEL NAME
Original name:
L-histidine
iTO977 ID: m29

Figure 3.2: The workflow of matching a BRENDA database entry with a metabolite in the iTO977 model. Each BRENDA substrate points to a list of synonyms. The synonyms are searched against the synonyms layer of the iTO977 metabolite thesaurus. When a match is found, the BRENDA substrate is defined as being the same compound as the iTO977 metabolite.

Finally, the thesauruses were USED to connect BRENDA substrateS with metabolites in the iTO977 model. Starting with the BRENDA substrate name thesaurus, the synonyms for a given BRENDA entry were used to search the iTO977 metabolite thesaurus. If a synonym for the BRENDA entry substrate appeared in the iTO977 model thesaurus, a connection was made. If none of the BRENDA entry substrate synonyms appeared in the iTO977 model thesaurus, a second attempt was made where instead of looking for identical standardized names, the Levenshtein distance was computed with the Python package python-levenshtein [58]. The Levenshtein distance between two strings is the minimal number of changes needed to transform one string into the other [59]. When the Levenshtein distance between a BRENDA and iTO977 metabolite synonym was exactly 1, connections were made after manual inspection. Figure 3.2 shows how a BRENDA entry with the substrate His was connected with the iTO977 metabolite L-histidine.

Turnover number matching

The BRENDA entries that were successfully connected to iTO977 metabolites were processed and structured into a three layers deep hash table, referred to as the EC table from here on out, of hash tables in Python. The bottom layer was the EC number, the middle layer the organism, and the final layer was the substrate. The main names from the iTO977 metabolite thesaurus were used instead of the the BRENDA substrate name as the keys for the substrate hash table. The substrate hash table then maps metabolites names to a vectors of turnover numbers. This scheme can be seen in figure 3.3.

The metabolites involved in enzymatic reactions in the iTO977 model were extracted from the SBML file and ordered into a two levels deep hash table. The hash table was designed with the iTO977 reaction names as keys for the first level, with the second level pointing to the reactants, products, reversibility status and EC numbers associated with the reaction, as seen in figure 3.4.

A Python script was developed to assign turnover numbers to reactions based on the following procedure:

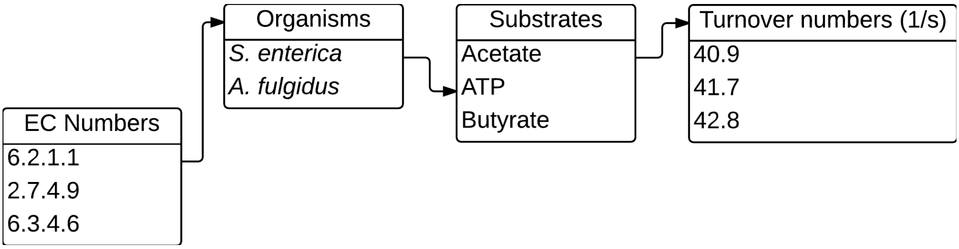


Figure 3.3: The structure of the EC table. Each EC number has an associated list of organisms, and each organism has an associated list of substrates, and each substrate has a list of turnover numbers.

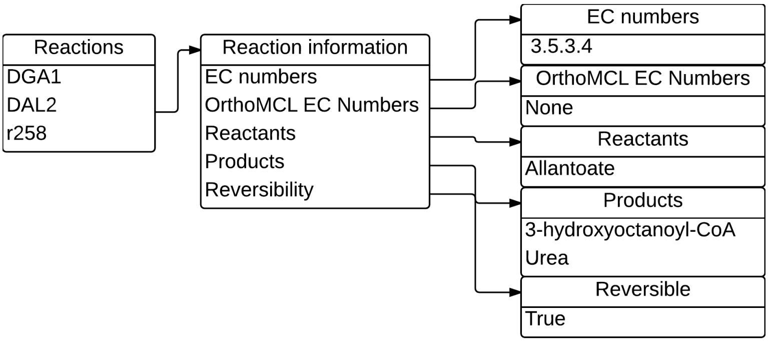


Figure 3.4: The representation of reactions in the reaction hash table. Each reaction has several fields, like the EC numbers associated with it, the reactants and the products.

1. Choose an enzymatic reaction from the iTO977 model.

2. Access the first level of the EC table with all the EC numbers associated with the reaction. Compile a list of the organisms associated with any these EC numbers.

3. For each EC number, iterate over the associated organisms in the EC table. For each organism, iterate over the substrates and enumerate the number of unique reactants and, if the reaction is reversible, products from the reaction matching a substrate. We define this as the substrate matching number. If *S. cerevisiae* is in the organisms list, and the the substrate matching number for this organism is greater than zero, discard the other organisms, and use the substrate matching number from *S. cerevisiae* as substrate matching number for this EC number. If not, compile a list of all the matching substrates for all the organisms and use the length of this list as the corresponding number.

4. If none of the EC numbers associated with the reaction produced a substrate matching number greater than zero, repeat step 2 except with the OrthoMCL EC numbers instead of the EC numbers. If this fails, skip this reaction and go to step 7.

5. If any of the EC numbers produced a substrate matching number greater than zero for *S. cerevisiae*, discard the ones that did not.

6. Determine the EC number with the greatest substrate matching number. Iterate over the organisms (or just *S. cerevisiae* if the other organisms were discarded in step 2). For each organism, and each substrate matching a reactant or product (if the reaction is reversible), add all the associated turnover numbers corresponding to matching substrates to a list. This list is the turnover numbers list for the reaction in question.

7. Choose a new enzymatic reaction and go back to step 2. The procedure is over when all enzymatic reactions have been examined. Record the turnover number lists a CSV file along with the reaction name.

3.3.3 Model formulation and implementation

The CSV file containing the turnover number lists was imported into Python, and the turnover number for the reactions contained in the file was defined to be the largest turnover numbers associated with the reaction. A MATLAB script assembling a COBRA toolbox compatible ircFBA model for iTO977 can be found in appendix A. After implementation in MATLAB, reactions where no turnover numbers were discovered in the previous step were assigned the median of all the turnover numbers in the model.

3.4 ircFBA parameter tuning

Two distinct algorithms were developed to tune the behavior of the ircFBA model to better fit experimental data. Both algorithms tweak the turnover numbers to enable to model to reach a target growth rate in a pre-determined growth medium. The motivation for the creation of these algorithms was to enable finding turnover number sets that fit different strains of *S. cerevisiae* better than the initial one. As the actual value of a turnover number depends on environmental factors, and experimental characterization of enzymes is typically done *in vitro*, the *in vivo* value is expected to be different than what is found in the literature. Furthermore, turnover numbers derived from other organisms were used when building the ircFBA model. As such, making small changes to the turnover numbers should be a reasonable method for improving the quality of an ircFBA model.

Using the notation in section 3.2 and equations 3.3-3.8, we also define the target growth rate as μ_T, and the growth medium as:

$$0 \leq v_i \leq V_i, i \in E \tag{3.9}$$

where E is the set of indices for the exchange reactions the growth medium is defined by and V_i is the experimental uptake rate of the metabolite.

3.4.1 The QP Algorithm: ircFBA performance tuning through Quadratic Programming

By adding a linear variable, r_i, to the right-hand sides of the the enzyme flux constraint:

$$-(kcat_i g_i + r_i) \leq v_i \leq kcat_i g_i + r_i, i \in G \tag{3.10}$$

$$r_i \geq 0, i \in G \tag{3.11}$$

$$r_i \leq H g_i, i \in G \tag{3.12}$$

where H is an arbitrarily large number, the feasible region of the ircFBA model is reverted back to the base FBA feasible region. Equation 3.12 ensures that r_i is greater than zero only if g_i is also greater than zero, enabling $kcat$ to be updated according to:

$$kcat_i^F = kcat_i + \frac{r_i}{g_i}, i \in \{i \in G | g_i > 0\} \tag{3.13}$$

to give a new set of kcat values that would have been able to support the FBA phenotype produced from solving an ircFBA augmented with equations 3.10-3.12. This can be used as the basis for a $kcat$-tuning algorithm by using quadratic programming.

Replace equation 3.6 and the objective function from the base ircFBA problem with

$$\min z = \sum_{i \in G} s_i^2 \tag{3.14}$$

$$-(kcat_i g_i + s_i) \leq v_i \leq kcat_i g_i + s_i, i \in G \tag{3.15}$$

$$s_i \leq kcat_i g_i, i \in G \tag{3.16}$$

$$s_i \geq 0, i \in G \tag{3.17}$$

where s_i is the tuning factor for enzymatic reaction i. When representing this problem with the standard quadratic programming notation(equations 2.8-2.10) the resulting Q matrix, containing the coefficients of the objective function, is diagonal. As any diagonal element of Q will be either 0 or 1 in this case, equation 2.11 will always hold. This problem then meets the conditions for convexity, and can be solved in polynomial time. If the growth rate is specified by setting a constraint on the growth rate reaction, a minimal set of tuning factors will be calculated supporting this growth rate. We refer to this as the augmented ircFBA QP problem. The algorithm is then:

0. Optimize a normal ircFBA problem, with the environment set according to equation 3.9. Set μ_0 to the optimal growth rate. Choose a value for M, the number of iterations to perform. Assign θ the value 0.

1. Set $v_{growth} = (1 - \theta)\mu_0 + \theta\mu_T$, where v_{growth} is the growth rate reaction.

2. Solve the augmented ircFBA QP problem.

3. Update $kcat_i$ according to:

$$kcat_i = kcat_i + \frac{s_i}{g_i}, i \in \{i \in G | g_i > 0\} \tag{3.18}$$

4. Increase the value of θ by $\frac{1}{M}$.

5. If $\theta > 1$, the algorithm has finished. Otherwise, return to step 1.

The algorithm was implemented in MATLAB with the Gurobi QP solver.

3.4.2 The SP Algorithm: ircFBA performance tuning through Post-Optimality Analysis

The Shadow Price (SP) algorithm was developed as a heuristic for finding a set of minimal changes to the turnover number set in order to support the desired growth rate. Furthermore, it does not use an augmented version of ircFBA, but rather tunes the growth rate with post-optimality analysis.

The constraint, $v_i \leq kcat_i g_i$, has an associated shadow price y_i.To increase the growth rate of the ircFBA problem by ΔZ, $\Delta b_i = \frac{\Delta Z}{y_i}$ can be added to the right hand side of the constraint. This, however, only applies if the shadow price for y_1 is valid over the range $[0, \Delta b]$. However, if Δb_i is sufficiently small, the actual change observed in the objective function should be close to ΔZ. If $g_i > 0$ this is equivalent to increasing $kcat_i$ by $\frac{\Delta b_i}{y_i}$. The ircFBA problem is then the normal problem as defined by equations 3.3-3.8, with μ_T as the target growth rate, and the growth medium defined by 3.9. The algorithm can then be formulated as:

0. Solve the ircFBA problem with the initial set of $kcat$ values, and then a regular FBA problem with growth as the objective and bounds on the exchange reactions according to 3.9. Record the shadow prices for the ircFBA problem in a $|G| \times 2$ matrix, $\mathbf{Y}$, where $Y_{i,1}$ and $Y_{i,2}$ contain the shadow prices for the $\leq$ and $\geq$ constraints respectively for the enzymatic reaction flux v_i in equation 3.6. Set μ_0 to be the optimal ircFBA growth rate, and μ_{FBA} to be the FBA optimal growth rate. We define $\mu_1 = \min\{\mu_{FBA}, \mu_T\}$. Choose the growth rate step size according to $\Delta Z = \frac{|\mu_0 - \mu_1|}{M}$. Set $\mu_{OPT} = \mu_0$

1. If $\frac{|\mu_T - \mu_{OPT}|}{\mu_T} \leq \epsilon$, where ϵ is the error tolerance, the algorithm has completed. Otherwise, proceed to the next step.

2. Set $\Delta\mu^* = \min\{\Delta Z, |\mu_T - \mu_{OPT}|\}$

3. Determine how much the right or left-hand sides of equation 3.6 have to change to satisfy a change in the objective function equal to ΔZ according to:

$$\Delta b_i = \frac{\Delta\mu^*}{\max\{|Y_{1,i}|, |Y_{2,i}|\}}, i \in \{i \in G | g_i > 0\} \tag{3.19}$$

4. Calculate the change in $kcat_i$ needed to obtain this change:

$$\Delta kcat_i = \frac{\Delta b_i}{g_i}, i \in \{i \in G | g_i > 0\} \tag{3.20}$$

5. Determine the minimal ratio, t, one $kcat$ value has to be increased by in order to achieve a growth rate increase of ΔZ:

$$t = \min\{\frac{\Delta kcat_i}{kcat_i}\}, i \in \{i \in G | g_i > 0\} \tag{3.21}$$

6. Let i be the index corresponding to the minimal ratio, t. Update $kcat_i$ according to:

$$kcat_i = kcat_i + \Delta kcat_i \tag{3.22}$$

7. Re-resolve the ircFBA problem, with the new kcat-value. Set μ_{OPT} to be the optimal growth rate and return to step 1.

The algorithm was implemented in MATLAB using the COBRA toolbox FBA solver with $\epsilon = 10^{-8}$.

3.4.3 Combining turnover number sets

As the current implementation of the QP and shadow price algorithms do not support solving for multiple growth mediums, or conditions, at the same time, a method was developed to enable the ircFBA model to be fitted to multiple mediums.

Using either the SP or QP algorithm, the iTO977 ircFBA model was fitted to each growth medium. We define $kcat^F_{i,j}$ as the final value for turnover number $kcat_i$ in condition j, $kcat^E_i$ as the original (empirical) $kcat$ value and S as the set of reactions where $kcat$ was altered from $kcat^E$ in any of the conditions. With n growth mediums, **A** is defined as a $|S| \times n$ matrix, where

$$\mathbf{A}_{i,j} = \frac{kcat^F_{i,j}}{kcat^E_i}, i = 1, 2, ..., |S|, j = 1, 2, .., n \tag{3.23}$$

A vector $\mathbf{x} = [x_1, \cdots, x_n]$ is then defined to give a new set of turnovers:

$$kcat_i = kcat^E_i \sum_{j=1}^{n} A_{i,j} x_j, i \in S \tag{3.24}$$

$$x_i \geq 0, i = 1, \cdots, n \tag{3.25}$$

This was used to create a MATLAB function that takes **x** as input and calculates the corresponding $kcat$ values. As all the growth conditions used in this project were derived from single carbon source minimal media with glucose as the only limiting metabolite, with oxygen in excess, the growth conditions were defined by setting the exchange reaction for oxygen uptake to be unbounded, and with the difference between the conditions only being the glucose uptake exchange reaction upper bound. The function then optimizes the ircFBA model using this new $kcat$ set for the n growth conditions individually. The following sum is returned by the the function:

$$W = \sum_{s=1}^{n} \left(\frac{\mu_s - \mu^E_s}{\mu^E_s} \right)^2 + \sum_{s=1}^{n} \left(\frac{v_{o,s} - v^E_{o,s}}{v^E_{o,s}} \right)^2 \tag{3.26}$$

where μ_s is the ircFBA growth rate for condition s, μ_s^E is the experimentally determined growth rate for condition s, $v_{o,2}$ is the ircFBA oxygen uptake rate for condition s and $v_{o,s}^E$ is the experimental oxygen uptake for condition s. If uptake or secretion rates for metabolites other than glucose and oxygen had been provided in the data set, additional terms could have been added to equation 3.26. The MATLAB nonlinear programming function fmincon from the MATLAB optimization toolbox was used to minimize W using the interior-point optimization algorithm. fmincon does not guarantee global optimality, or even convergence. If fmincon fails, other heuristics like simulated annealing or genetic algorithms can be used. These heuristics have implementations available in the MATLAB optimization toolbox.

3.5 Analysis of randomized kcat distributions

The growth curves and phenotype exhibited by an ircFBA model is determined both by the underlying metabolic network as well as the $kcat$ values and enzyme molecular weights. By randomizing the $kcat$ values and then optimizing the $kcat$ set using either the QP or SP algorithm, a broader picture of the sort of phenotypes the ircFBA model can produce should emerge. .

Different $kcat$-sets were then generated by by randomly drawing a turnover from the turnover number CSV file from section 3.3.2. After assigning each enzymatic reaction a $kcat$, the ircFBA model was optimized for growth. If the random $kcat$ set was unable to produce non-zero growth, this random $kcat$ was scrapped. Otherwise, it was retained.

1000 unique $kcat$ sets were generated with this method. The shadow price algorithm (as detailed in section 3.4.2) was used to tune the $kcat$ values to match the optimal growth rates to the experimental growth rate for *S. cerevisiae* in minimal media with both glucose and oxygen in excess. The final $kcat$ values for each of the 1000 sets were then recorded, along with the flux distributions and gene concentrations for the final ircFBA problem.

The T fluxes from enzymatic reactions with a non-zero flux in at least one of the final optimal ircFBA solutions were then ordered into a $T \times 1000$ matrix, **F**, where $F_{i,j}$ is the flux for the ith observation of enzymatic reaction j. The pairwise linear correlation matrix, **C**, was computed for **F** by using the MATLAB function corr. corr is available in the MATLAB Statistics & Machine Learning toolbox. $C_{i,j}$ is the Pearson's linear correlation coefficient between flux i and j, and was kept for further analysis if the associated p-value was less than $\frac{0.01}{T^2}$ in order to account for multiple testing.

Chapter 4

Results & Discussion

The workflow of this project was divided into three main parts, and the main results from each will be provided here. The first part was the development and characterization of an ircFBA model of *S. cerevisiae*. The basic properties of the model parameters and its ability to predict realistic phenotypes were investigated. The second part was the development of two distinct algorithms using two different approaches from mathematical programming to tune the performance of the ircFBA model by tweaking the turnover numbers ($kcat$). The ability of these methods to reliably converge and efficiently solve the problem was investigated, as well as the quality of the predictions. In the final part, the $kcat$ set was replaced by drawing new $kcat$ values randomly and examining properties of the resulting ircFBA models.

4.1 ircFBA Model Assembly & Performance

An ircFBA model was assembled and implemented in in MATLAB as detailed in section 3.3. Of the 930 gene associated enzymatic reactions in the iTO977 model, 547 were assigned unique turnover numbers. For reactions where multiple turnover numbers were discovered in the data integration stage, the maximal turnover number was used. The median of these 547 turnover numbers were used for the remaining reactions. A histogram, figure 4.1b, of these 547 turnovers shows a bell-shaped distribution, which was also reported for the published *E. coli* MOMENT model [5]. Interestingly, if the turnover numbers are plotted in rising order , the distribution of turnover numbers from the iTO977 ircFBA model is similar to that of the *E. coli* MOMENT model. This can be seen in figure 4.1a. As the largest turnover number associated with each reaction was used to set the ircFBA turnover number, and the *E. coli* MOMENT model used the median, this is surprising. The reason for this isn't clear, but could simply represent a difference in the reactions contained in both metabolic reconstructions and the actual metabolic machinery of the organisms, as *E. coli* is a prokaryote and *S. cerevisiae* is an eukaryote. Furthermore, the MOMENT model incorporated data from SABIO-RK, which is a manually curated database where organism-specific turnovers can be found more reliably than in BRENDA [37].

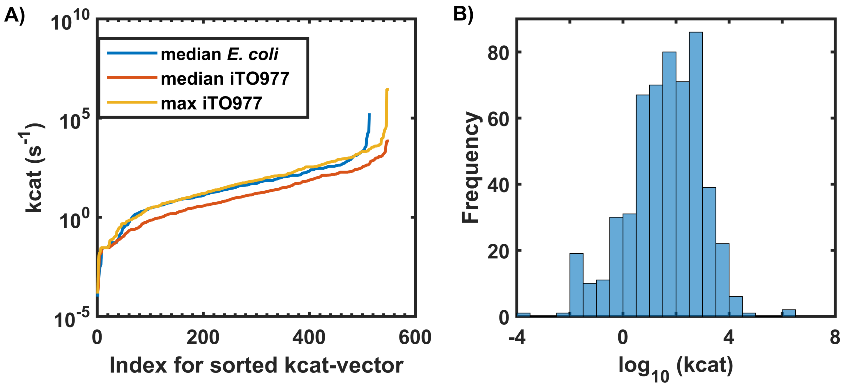


Figure 4.1: A) The turnover number lists sorted in rising order. Max iTO977 is the distribution obtained by choosing the largest $kcat$ in the $kcat$ list for any given reaction, while median iTO977 is from choosing the median. Median *E. coli* is from MOMENT [5]. B) A histogram of max iTO977 on a log scale. The $kcat$ set varies across 12 orders of magnitude, but has a bell shaped distribution.

Table 4.1: Correlations between predictions and the experimental measurements. The ircFBA growth rate row gives the Pearson correlation coefficient with its p-value for ircFBA's growth rate predictions against the experimental growth rates. The other rows are interpreted in the same way.

Prediction	r	p-value
ircFBA growth rate	0.9830	6.18×10^{-8}
ircFBA O_2 flux	0.4019	0.23
FBA growth rate	0.8160	0.002
FBA O_2 flux	6.0824×10^{-4}	0.99

4.1.1 Phenotype prediction

The ircFBA model's ability to predict growth rates in minimal media with glucose as the only limiting metabolite was investigated by comparing the predicted growth rates with experimental measurements derived from chemostat cultures of *S. cerevisiae* CEN.PK [1], one of the main reference strains [60]. This data consists of 11 different growth rates, with associated oxygen and carbon uptake rates, and can be found in its raw form in Appendix B. The highest growth rate in this data set, $0.38\ h^{-1}$, is the saturation growth rate for *S. Cerevisiae* CEN.PK growing in fully aerobic minimal media with glucose as the only carbon and energy source [6]. Oxygen was in excess in these experiments [1], so growth rates were computed by setting the oxygen uptake rate to be unconstrained and the glucose uptake rate to have the experimental uptake rate as its upper bound.

The Pearson's Correlation Coefficient, r, was calculated for the growth and oxygen uptake rate predictions generated both by ircFBA and FBA. These can be found in table 4.1. The ircFBA growth rate predictions correlate better with the experimental than FBA does, with $r = 0.9830$ against $r = 0.8160$ for FBA. However, both correlate well and have highly significant p-values. This is not the case for the oxygen uptake rate, where both methods fail to produce significant p-values. Looking at the plots (figure 4.2) of the

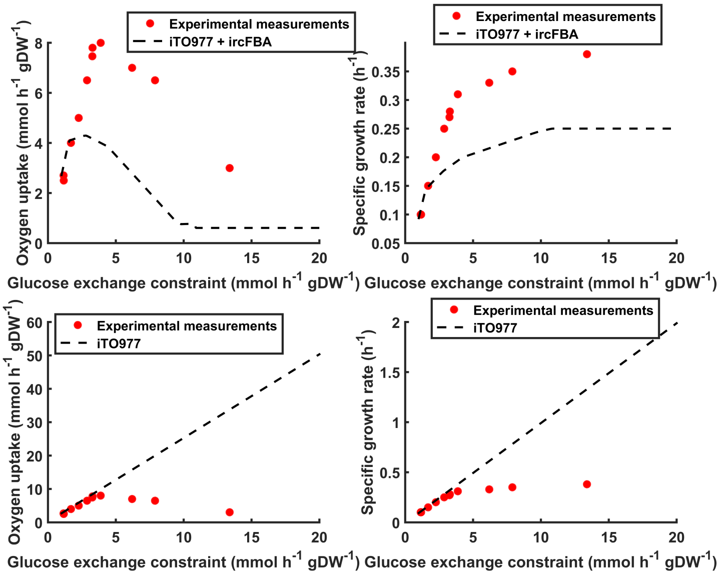


Figure 4.2: FBA (iTO977) and optimized MOMENT line of optimality plotted along with experimental measurements from *S. cerevisiae*.

predictions along with the the experimental measurements, the reason for this is apparent. Up to a glucose uptake rate of around 4 mmol gDW^{-1} h^{-1} the experimental values lie along the FBA line of optimality. After this point, the experimental oxygen uptake rate starts to drop off. The reduced oxygen uptake indicates a switch to respirofermentative metabolism [20], meaning that glucose isn't fully oxidized. Without an upper bound on the oxygen uptake rate, FBA will therefore predict unrealistically high growth rates. This can be seen in table 4.2, where r for the correlation between FBA and experimental measurements for both oxygen and growth close to 1, with highly significant p-values, for low glucose uptake rates. However, with an unbounded oxygen uptake rate, FBA predicts that the growth rate increases linearly with the glucose uptake rate, and it therefore fails to predict the later growth rates. ircFBA's enzyme capacity constraint prevents this from happening, and the growth rate starts to saturate. As a result, ircFBA's growth predictions correlate better over the whole range. It should be noted, however, that FBA predicts growth rate with a high degree of precision if upper bounds are placed on both the glucose and oxygen uptake rates (see appendix B).

The changes in the slope of the growth rate curve for ircFBA in figure 4.2 indicates the metabolic network entering new phases. *S. cerevisiae* CEN.PK growing in glucose-limited growth media with excess oxygen switches from full respiration over to partial ethanol fermentation at a growth rate of 0.3 h^{-1} [6], which is not something the FBA model is able to capture. The α from the ircFBA PPhP, as defined in equation 2.27, can be seen in figure 4.3a. Here π_x and π_y are the glucose and oxygen exchange reaction shadow prices respectively. In contrast to the simple FBA PPhP in figure 2.5 where only two viable phases could be seen, ircFBA has no less than 10 distinct phases. However, they can be

Table 4.2: Pearson correlations between predictions and the experimental measurements. Correlations were not computed for the whole range of experimental values. The range was instead split into two, and correlations were computed for each range. Low glucose uptake rate is defined as less than 4 mmol gDW^{-1} h^{-1}, while high is greater than than 4. The ircFBA growth rate row gives the Pearson correlation coefficient with its p-value for ircFBA's growth rate predictions against the experimental growth rates. The other rows are interpreted in the same way.

Prediction	Low glucose uptake rate		High glucose uptake rate	
	r	p-value	r	p-value
ircFBA growth rate	0.98	2.5×10^{-5}	1	0.0025
ircFBA O_2 flux	0.75	0.03	0.9344	0.23
FBA growth rate	0.99	1.22×10^{-7}	0.98	0.11
FBA O_2 flux	0.99	2.43×10^{-6}	-0.99	0.0711

divided into the two continuous primary phases in figure 4.3B. Here any $\alpha > 0$ has been set to equal 1, and any $\alpha < 0$ has been set to -1. The $\alpha > 0$ phase is a futile phase, where oxygen is in excess. The second primary phase, where $\alpha < 0$, is a little more complicated. Rather than $\alpha < 0$ indicating that both metabolites are limiting the growth, this only applies to the left of the ircFBA line of optimality. The phase to the right must then be characterized as having dual substrate *inhibited* growth, meaning that the growth rate has saturated past this point. The experimental growth rate data set implies that the *S. cerevisiae* growth rate really does saturate in this manner, with the growth rate increasing more slowly at the highest glucose uptake rates. Maximum specific growth rate estimates vary from 0.38 h^{-1} [6] to 0.41 h^{-1} [60] for *S. cerevisiae* CEN.PK, however, which is significantly higher than the 0.25 h^{-1} predicted by ircFBA.

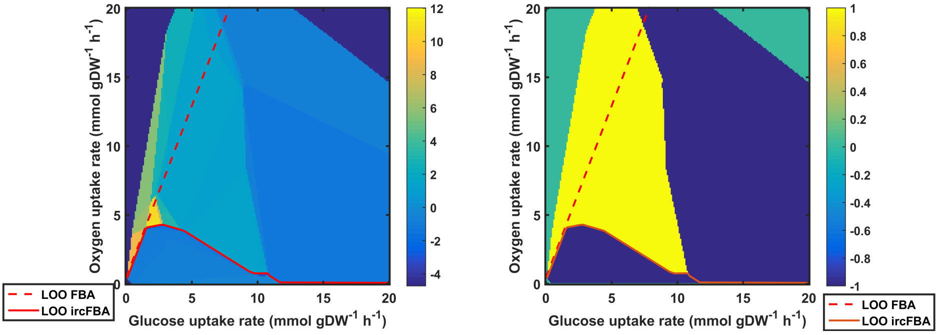


Figure 4.3: α as defined by equation 2.27 for the ircFBA model. The leftmost plot shows the raw α, while the plot to the right only shows the sign. Ten distinct phases can be seen in the leftmost plot, but these correspond to two main phases demarcated by the sign of α. The positive phase exhibits futile metabolise ($\alpha \geq 0$). LOO ircFBA separates the second major phase into two parts. To the left of LOO ircFBA, growth is limited by both glucose and oxygen. To the left, both inhibit growth.

The respiratory quotient,RQ,is defined as the ratio of carbon dioxide excretion to oxygen uptake [6], and was plotted for the ircFBA model as $\frac{v_{CO2}}{v_{O2}}$ as a function of the growth rate in figure 4.4a. With glucose as the sole energy and carbon source, a RQ of 1 implies fully respiratory metabolism with no fermentation, while a RQ above 1 indicates fermentation. The plot has three main phases:

1. At low growth rates, RQ is 1, and glucose is metabolized optimally. In this phase, ircFBA predicts a growth range along the FBA line of optimality. This phase is fully in agreement with experimental data, although ircFBA exits it at a lower glucose uptake rate than the actual organism does.

2. In a narrow range, the RQ is less than 1. This is a surprising phenotypic state that is not observed experimentally during oxygenated glucose limited growth [61, 62]. In the ircFBA model, this is caused by the excretion of pyruvate (see figure 4.4b). The reason for this is that the more typical fermentation products, acetate and ethanol, are derived from pyruvate through nonoxidative decarboxylation (see figure 2.6). Unlike ethanol excretion, pyruvate excretion does not recycle the NAD(+) from glycolysis. NADH produced in glycolysis can hence be used for oxidative phosphorylation, which leads to oxygen consumption. Pyruvate excretion therefore lowers RQ below one by increasing the oxygen uptake rate without releasing CO2. On the other hand, NAD(+) is reduced to NADH when acetate is produced. This offsets the gain in CO_2 excretion by providing more reducing power for the electron transport chain. Figure 4.5 shows the flux direction of key reactions in the tricarboxylic acid (TCA) cycle and oxidative phosphorylation. The reaction labels in 4.5 refer to the same reactions as in figure C.1, except CIT2, SDH4 and MDH3 labeled as CIT1, $\mathrm{SDH3}_1$ and MDH1 instead. ATP1 is ATP production from oxidative phosphorylation, with COX1 and RIP1 being electron transport chain reactions. In the iTO977 model, the cardinal direction of the TCA cycle is with positive fluxes, except for $\mathrm{SDH3}_1$ where the cardinal direction is designated by a negative flux. The drop in RQ coincides with LSC2 and KGD1 being fully turned off, which means that there is no net flux through the traditional TCA cycle. Instead, the TCA cycle goes as normal until 2-oxogluterate is produced, but the flux then flows entirely to amino acid synthesis pathways (see figure C.1). In order to run this partial TCA cycle, fumarate is imported from the cytosol.

3. At higher growth rates, RQ goes above 1 and saturates at 31.7. The phase is dominated by ethanol excretion, which starts at a growth rate of 0.2 h^{-1}. At very highest growth rates, no ATP is produced from oxidative phosphorylation. The electron transport chain remains active, however, serving only as a mechanism to oxidize NADH into NAD(+).

   Figure 4.5 shows the the fluxes for IDH1 and $\mathrm{SDH3}_1$ being turned off as the growth rate crosses over 0.2 h^{-1}.

Aside from CO_2, the only excreted carbon compounds were acetate, pyruvate and ethanol. While detailed fluxes for the acetate, pyruvate and ethanol excretion rates in the experimental data set being used are not available, a limited amount of information can be found in the original publications [6, 63]. The lowest growth rate ethanol was detected for was at 0.3 h^{-1} where the excretion flux was non-zero but very small. It then increases linearly until it saturates at a growth rate of 0.38 h^{-1} with an excretion flux of 16.2 gDW^{-1}

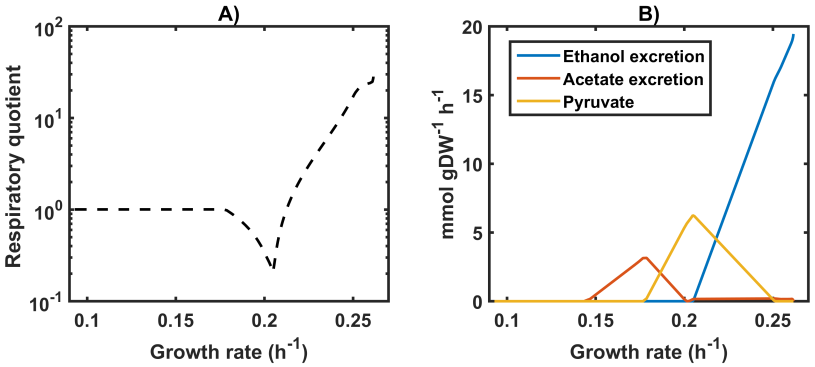


Figure 4.4: A) The respiratory quotient, RQ, as a function of the ircFBA growth rate. A RQ close to 1 indicates near full oxidation of glucose, while a RQ over 1 is a sign of fermentation. The dip below 1 is caused by the excretion of pyruvate. B) Excretion profile for ircFBA. Ethanol, acetate and pyruvate are all excreted at high rates.

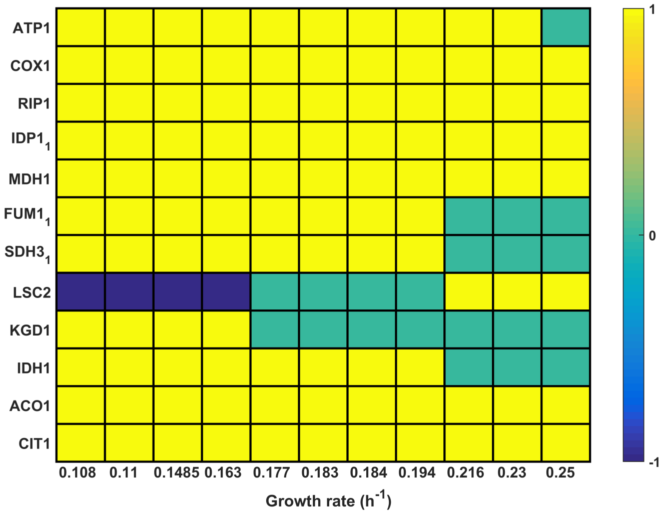


Figure 4.5: The sign of fluxes in the TCA cycle and electron transport chain pathway as a function of the growth rate in an ircFBA model of *S. cerevisiae*. The reaction names are listed on the y-axis. Yellow squares indicate positive flux, green squares no flux and blue squares negative flux. NADH and FADH$_2$ producing fluxes (IDH1, KGD1 and SDH3$_2$) are turned off at higher growth rates.

h^{-1}. ircFBA, however, predicts a start of ethanol production at 0.2 h^{-1}. Acetate and ethanol excretion fluxes, however, reached peak values of 0.5 mmol gDW^{-1} h^{-1} and 0.7 mmol gDW^{-1} h^{-1} respectively. This is 15.9% and 22.2% of the peak excretion rates predicted by ircFBA. This can be partially explained by ircFBA predicting lower growth rates in phases off the FBA line of optimality, with fermentation commencing much earlier.

At the saturation growth rate, the experimental RQ ratio is 8.4, which is smaller than the peak RQ for ircFBA by a factor of 3.77.

Measurements of TCA flux values in *S. cerevisiae* CEN.PK growing in respirofermentative conditions in minimal media, at the same pH and temperature as the data set used in this study, have demonstrated that the net flux through the TCA cycle is negatively correlated with both the growth and glucose uptake rates. The net flux bottomed out at 0.03 mmol gDW^{-1} h^{-1} [61], which, while still a non-zero flux, points to the reduction of TCA fluxes predicted by ircFBA not being unreasonable. The same goes for oxidative ATP production under the same conditions. While ethanol fermentation is the dominant pathway for glucose metabolism at high respirofermentative growth rates, oxidative ATP production never shuts off entirely [64]. Therefore, while predicting a switch from respiration to respiro-fermentation, the ircFBA model's ability to predict meaningful phenotypes with its initial $kcat$-set is fairly limited. One of the main reasons for this is that ircFBA underestimates growth rates significantly after diverging from the FBA line of optimality.

4.2 Fitting ircFBA model behavior to experimental measurements

Two methods were developed to incorporate experimental data to tweak the ircFBA $kcat$ values in order to make more accurate phenotype predictions. Section 3.4 gives the derivations of these methods. Both methods work by tuning the $kcat$ set to reach a target growth rate.

4.2.1 Applying the QP algorithm

The QP algorithm was applied to the 9 data points in appendix B where the experimental growth rate was higher than the ircFBA growth rates. Beyond the augmented QP constraints, the only added constraints were the glucose uptake rates. M, the number of QP iterations to perform, was set to be 40. Due to equation 3.16, 40 QP iterations would allow a maximum relative change in any $kcat$ in the ircFBA model to be 2^{40}. Persistent numerical issues, however, caused problems as the Gurobi QP solver often stalled or claimed that the problem was infeasible. To overcome this issue, the Gurobi feasibility tolerance was lowered from 10^{-9} to 10^{-7}. After making these changes, the algorithm started to behave as expected.

Growth and oxygen uptake rates from the ircFBA models with $kcat$ sets derived from applying the QP algorithm individually to the experimental growth rates were plotted in figures 4.6 and 4.7. Each plot shows the results of varying the glucose uptake rate in the ircFBA model when allowing the oxygen uptake rate to be unconstrained. For example, the leftmost plot in the top row is the result of setting a target growth rate of 0.15 h^{-1}, with a glucose uptake rate of 1.69 mmol gDW^{-1} h^{-1}, in the QP-algorithm, and then plotting the ircFBA growth rate as a function of the glucose uptake rate. To distinguish ircFBA models with $kcat$ sets obtained by running the QP algorithm on different experimental data points, we define ircFBA$^{\mu}$, where μ is the target growth rate from the QP algorithm, as the ircFBA model with this $kcat$ set.

Table 4.3: Pearson correlations between ircFBA predictions and the experimental measurements. The ircFBA models were optimized with the QP algorithm. Row ircFBA$^{0.15}$ gives correlations for the the growth rate and oxygen uptake rate predictions made after fitting the base ircFBA model to the target growth rate of 0.15. The other rows have the same interpretation.

Prediction	Growth rate		Oxygen uptake rate	
	r	p-value	r	p-value
ircFBA$^{0.15}$	0.98	2.0×10^{-7}	0.46	0.16
ircFBA$^{0.20}$	0.97	1.0×10^{-6}	0.80	0.0027
ircFBA$^{0.25}$	0.99	2.0×10^{-9}	0.98	3.7×10^{-8}
ircFBA$^{0.27}$	0.99	1.5×10^{-8}	0.92	4.8×10^{-5}
ircFBA$^{0.28}$	0.99	8.0×10^{-9}	0.95	6.1×10^{-6}
ircFBA$^{0.31}$	0.99	4.0×10^{-10}	0.88	3.7×10^{-4}
ircFBA$^{0.33}$	0.99	2.1×10^{-9}	0.96	3.0×10^{-6}
ircFBA$^{0.35}$	0.99	1.98×10^{-9}	0.29	0.40
ircFBA$^{0.38}$	0.99	3.33×10^{-9}	0.18	0.60

The growth rates in figure 4.6 behave exactly as expected, with the effect of the algorithm being to raise the ircFBA model growth rate to the experimental growth rate at the appropriate glucose uptake rate. This allows ircFBA to saturate at a higher growth rate, and therefore match the experimental data set better than the base ircFBA model. This is confirmed by table 4.3 where the Pearson correlation coefficients, r, along with the p-values, can be found for the correlation between the experimental growth rates and the ircFBA models with one of the new $kcat$ sets. The r coefficients are 0.98 and 0.97 for ircFBA$^{0.15}$ and ircFBA$^{0.20}$, and at least 0.99 for the other ircFBA models. Furthermore, all the p-values are highly significant ($p < 2.0 \times 10^{-7}$). The oxygen uptake rate predictions in figure 4.7 are less clear, however. Table 4.1 shows that ircFBA$^{0.2}$ to ircFBA$^{0.33}$ correlate well with the the experimental uptake rates (r from 0.80 to 0.96 with p-values of 3.7×10^{-4}-3.7×10^{-8}). With no constraints being set on the oxygen uptake rate during the operation of the QP algorithm, mixed results are not unexpected here.

If the experimental growth rates are viewed as points in a three dimensional space, with the dimensions being the growth rate, glucose uptake rate and oxygen uptake rate, the euclidean distance from the experimental points to the iTO977 FBA line of optimality (the red line in figure 2.4) can be computed. The results of this was plotted in figure 4.8a, and gives a quantitative measure of how much the experimental data has separated from the line of optimality. Up to a growth rate of $0.3h^{-1}$, the experimental growth rates have a distance from the line of optimality of close to zero, and then starts to diverge linearly as a function of growth rate. When comparing the results for the oxygen uptake rate in table 4.3 with the distance from the line of optimality, it can be seen that the oxygen uptake rate predictions made by ircFBA models fitted to growth rates that are far from the line of optimality (growth rates of greater than 0.33 h^{-1}) result in poor r coefficients and insignificant p-values. On the other hand, the growth rates close to the line of optimality produce r coefficients close to 1, with highly significant p-values. Figure 4.8b shows the r coefficients plotted against the distance the corresponding experimental data point is from the line of optimality. A negative trend can be seen, and calculating the Pearson correlation

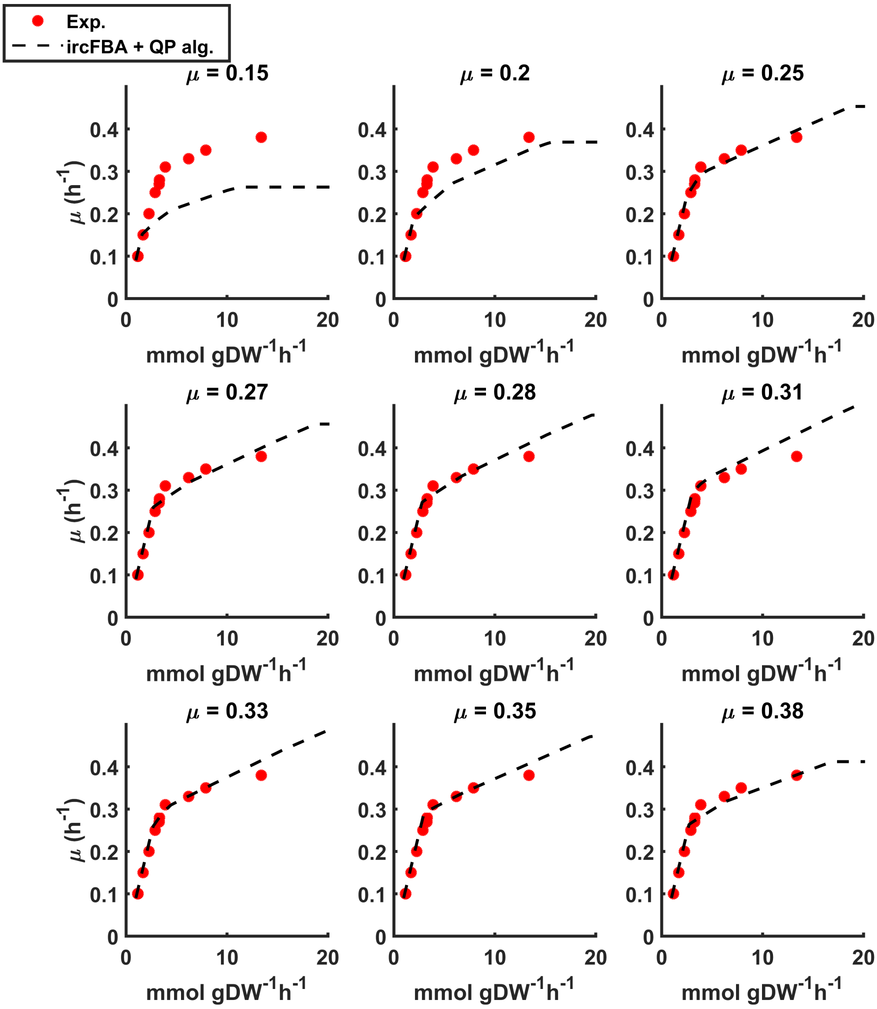


Figure 4.6: Growth rates for ircFBA models fitted to experimental growth rates with the QP algorithm. The in the label μ was the target growth rate used by the QP algorithm for generating the corresponding $kcat$ set.

coefficient gives it at -0.75 with a p-value of 0.0215. As the QP algorithm simply makes the growth rate reach a desired level in the pre-determined growth medium, the explanation of this is fairly simple. When the QP algorithm was used with the growth conditions corresponding to an experimental data points close to the FBA line of optimality, the ircFBA model will have its growth rate raised to the experimental growth rate at that glucose uptake rate. Due to this pair of parameters being close to the line of optimality, the oxygen uptake rate will also have to be close to the line of optimality, as a higher or lower rate

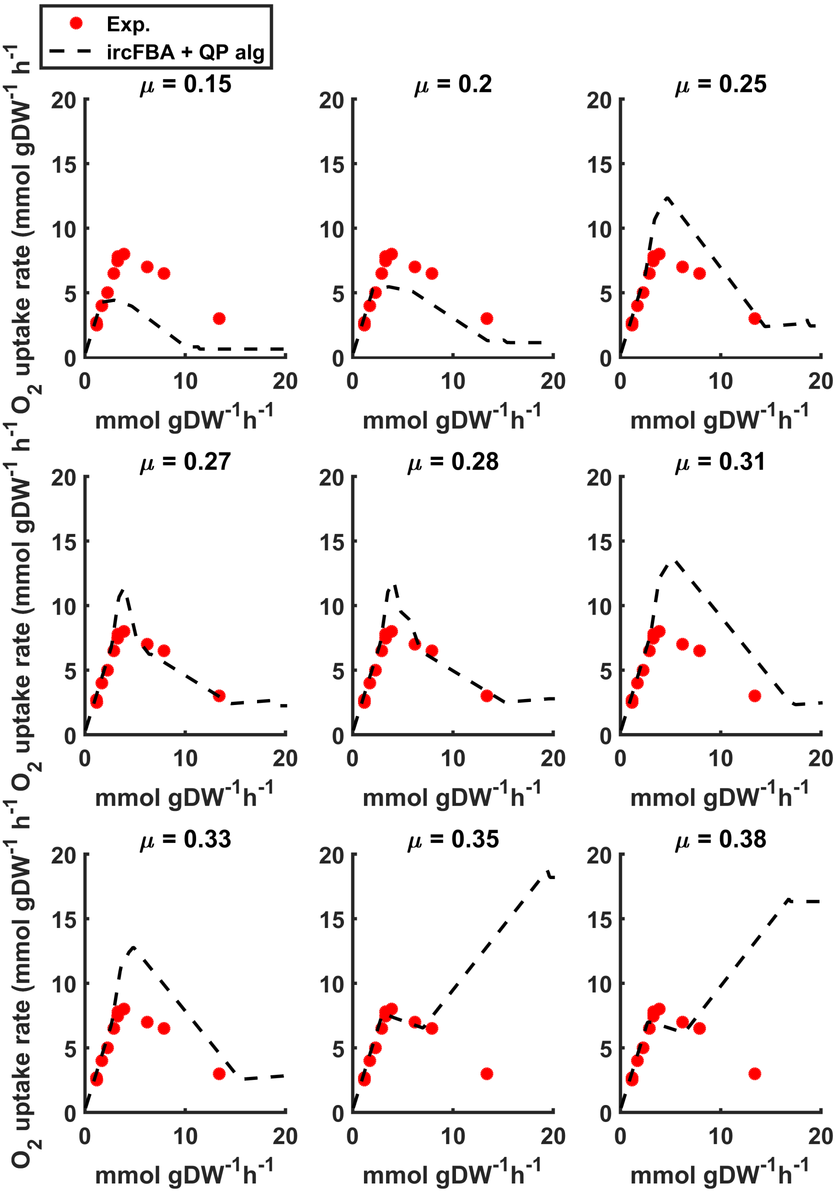


Figure 4.7: Oxygen uptake rates for ircFBA models fitted to experimental growth rates with the QP algorithm. The in the label μ was the target growth rate used by the QP algorithm for generating the corresponding $kcat$ set.

would both lead to a reduced growth rate. The resulting ircFBA problem from running the QP algorithm will then give growth curves that separate from the FBA line of optimality at a later stage than the base ircFBA model, and the oxygen uptake flux will correlate well with the experimental oxygen uptake rate up until *at least* the point of separation.

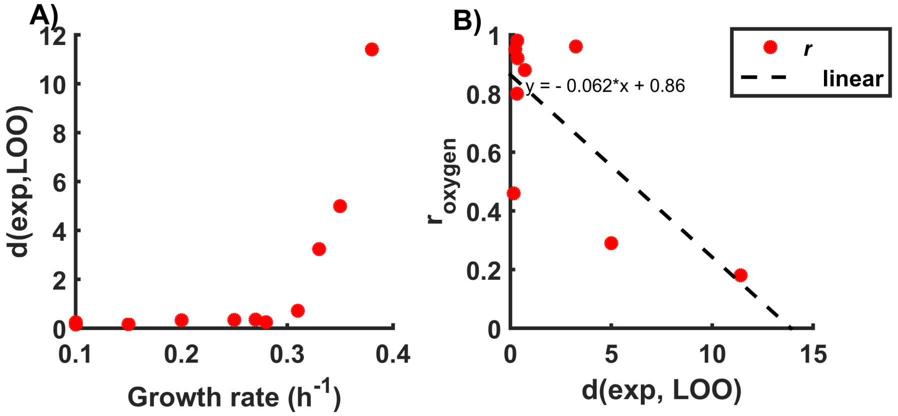


Figure 4.8: d(exp,LOO) is defined as the distance an experimental growth rate, exp, is from the line of optimality (LOO) in a three dimensional Euclidean space where the dimesions are the growth rate, oxygen oxygen uptake rate and glucose uptake rate. A) After separating from the line of optimality, the distance of the experimental data points to the line of optimalityincreases linearly. B) The oxygen uptake rate Pearson correlation coefficient for ircFBA models plotted against the distance the target growth rate was from the line of optimality.

The impact of the QP algorithm on the $kcat$ sets produced was investigated by looking at the fractions $\frac{kcat^F}{kcat^E}$ where $kcat^F$ is the $kcat$ at the end of the QP algorithm, and $kcat^E$ is the starting value. Table 4.4 summarizes this, and also shows the total number of $kcat$ values that the QP algorithm changed as well as the mean number of active enzymes in any of the intermediate QP solutions. Histograms of the fraction $\log_2(\frac{kcat^F}{kcat^E})$ for each of the 9 experimental conditions can be found in appendix D. The motivation for the development of the QP algorithm was to make modest changes to the $kcat$ set, as the guiding assumption was that the quality of the $kcat$ set was limited by numerous factors. Table 4.4 shows that the QP algorithm fails to achieve this. Sweeping changes are made across the board, and more individual $kcat$ changes are made than the average number of active genes in any of the intermediary QP solutions. Due to the objective being optimized being a square sum of variables, every active enzyme in a particular QP solution being altered is not a surprising result. However, the mean $\frac{kcat^F}{kcat^E}$ was surprisingly high for every ircFBA model, except for ircFBA$^{0.15}$ where only minor changes had to be made to reach the QP-algorithm target growth rate. This points to the QP algorithm essentially inflating the $kcat$ set, and this can be indeed seen in the histograms in figure 4.4 in appendix D. The difference between the mean number of active genes in a QP solution and the total number of changes made is also surprising, and suggests that $kcat$ values are changed during intermediary QP solutions that are not needed in future solutions.

The QP algorithm solutions were then re-examined, and for each QP solution com-

Table 4.4: The magnitude of the changes made to the base $kcat$ set when the QP algorithm is run. $|\frac{kcat^F}{kcat^E}|$ gives average relative change in the $kcat$ set. $|\sum(g_i > 0)|$ is the mean number of enzymatic reactions active in the intermediary steps of the QP algorithm. ircFBA$^{\mu}$ is the ircFBA model that was fitted to a target growth rate of μ.

Prediction	Number of $kcat$ values changed	$\lvert\sum(g_i > 0)\rvert$	$\lvert\frac{kcat^F}{kcat^E}\rvert$
ircFBA$^{0.15}$	282	169.6	1.0145
ircFBA$^{0.20}$	285	161.4	13.9
ircFBA$^{0.25}$	282	146.2	41.3
ircFBA$^{0.27}$	285	147.7	41.9
ircFBA$^{0.28}$	285	144.4	45.3
ircFBA$^{0.31}$	284	142.5	40.75
ircFBA$^{0.33}$	286	154.5	36.42
ircFBA$^{0.35}$	289	157.1	35.42
ircFBA$^{0.38}$	288	176.6	25.9

puted by Gurobi, a standard ircFBA problem was solved using the new $kcat$ values. This approach revealed that the QP problems have significant feasibility issues, as the number of active enzymes in any QP problem was significantly lower than it should have been. Figure 4.9 summarizes this for the 9 different growth conditions. If performing as designed, the QP and LP curves should be indistinguishable. This is clearly not the case, however, as a significant gap of around a 100 active genes or more can be seen in every plot. Due to Gurobi's feasibility tolerance being set to 10^{-7}, a constraint violation of this magnitude is permitted in the QP problem. This enables enzymatic reactions that have low flux in an ircFBA LP solution to be carried out without any enzyme. Attempts were made at using a lower feasibility tolerance, but this resulted in Gurobi being unable to solve the problem in most cases. In the cases where Gurobi was able to solve the problem with lower feasibility tolerance, the gaps between the LP and QP curves were smaller, which can be seen in figure D.1 in appendix D. Despite the QP problems being convex, actually solving it computationally is challenging. This appears to be because of the variation in the $kcat$ set, which ranges across 12 orders of magnitude. Gurobi typically deals with differences in scale by scaling the rows and columns of the coefficient matrix, but with massive differences in scale, numerical instability can become a problem due to finite numerical precision leading to significant round errors[51], which can cause the algorithms being used by Gurobi to fail [51]. These issues appear to be intractable, so while the growth rate and oxygen uptake rate predictions produced by applying the QP algorithm seem promising, the QP algorithm is too unstable and unpredictable to be useful.

4.2.2 Applying the SP algorithm

The base ircFBA model was fitted to the 9 experimental growth rates with the SP algorithm in the same way as was done for the QP algorithm in section 4.2. This was done by setting M, the number of iterations to perform, to be 1000. A high M is necessary for the SP algorithm, as the shadow price ranges aren't calculated or applied. Again, no upper bounds were placed on the oxygen uptake rates. Analogous to the naming conven-

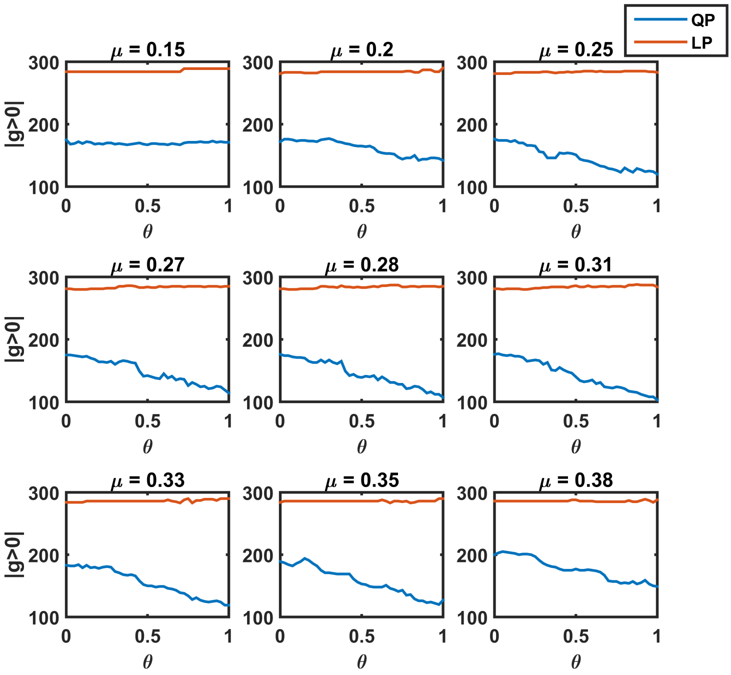


Figure 4.9: The number of enzyme concentration variables (g_i) greater than zero during intermediary steps of the QP algorithm. The QP lines are the number of non-zero enzyme concentrations within the augmented QP problem solution, while blue line is from growth rate maximization of the normal ircFBA model with the $kcat$ sets produced by the augmented QP problem solution.

tion developed for the QP ircFBA models, an ircFBA model with a $kcat$ set derived from applying the SP algorithm to a growth condition with a growth rate of μ will be referred to as ircFBA$^{\mu}$. Growth rate and oxygen uptake rate plots for the ircFBA models after optimization with the SP algorithm can be found in figures 4.10 and 4.11. Looking at figure 4.10, the same basic behavior can be observed as in figure 4.6, but with some key differences. Almost irrespective of the growth rate the ircFBA was generated at, the QP ircFBA models all, with the exception of ircFBA$^{0.15}$ and ircFBA$^{0.2}$, either fail to saturate within the glucose uptake rate range, or saturate at higher growth rates than the experimental data set suggest it should. Conversely, the ircFBA models all saturate within the glucose uptake range, and at consistently lower levels than the QP ircFBA models. As table 4.5 shows, the increased saturation growth rates lead to significantly improved correlations between the ircFBA models and the experimental growth rates, with r, the Pearson correlation coefficient, being at least 0.97 for every ircFBA model. This holds true for the oxygen uptake rate correlations as well, and every ircFBA model aside from ircFBA$^{0.15}$ had a statistically significant r of at least 0.82. Phenotypically, ircFBA$^{0.2}$ to ircFBA$^{0.35}$ can be grouped together. They all saturate at higher oxygen uptake rates than the experimental measurements would suggest, and at very similar levels. ircFBA$^{0.38}$, on the other

Table 4.5: Pearson correlations between ircFBA predictions and the experimental measurements. The ircFBA models were optimized with the SP algorithm. Row ircFBA$^{0.15}$ gives correlations for the the growth rate and oxygen uptake rate predictions made after fitting the base ircFBA model to the target growth rate of 0.15. The other rows have the same interpretation.

Prediction	Growth rate		Oxygen uptake rate	
	r	p-value	r	p-value
ircFBA$^{0.15}$	0.98	5.5×10^{-8}	0.41	0.21
ircFBA$^{0.20}$	0.97	8.9×10^{-8}	0.82	0.0022
ircFBA$^{0.25}$	0.98	2.0×10^{-9}	0.84	0.001
ircFBA$^{0.27}$	0.98	2.9×10^{-8}	0.85	8.9×10^{-4}
ircFBA$^{0.28}$	0.99	2.6×10^{-8}	0.86	7.3×10^{-4}
ircFBA$^{0.31}$	0.99	1.2×10^{-8}	0.88	3.5×10^{-4}
ircFBA$^{0.33}$	0.99	2.8×10^{-9}	0.84	0.001
ircFBA$^{0.35}$	0.99	3.1×10^{-9}	0.82	0.0018
ircFBA$^{0.38}$	0.99	3.8×10^{-8}	0.95	9.14×10^{-6}

hand, saturates at a lower oxygen uptake rate, but correlates better with the experimental oxygen uptake rates than any of the other ircFBA models.

In effect, the SP algorithm increases the ircFBA model's capacity for aerobic respiration. The QP algorithm does this as well, but it also inflates the $kcat$ set in general, which makes interpreting QP algorithm $kcat$ sets difficult. In stark contrast, the SP algorithm only adjusted a total of six different $kcat$ values across the 9 different growth conditions. The ratio $\frac{kcat^F}{kcat^E}$, with $kcat^F$ being the final value after running the SP algorithm and $kcat^E$ being the initial value, for every one of these six enzymatic reactions is available in table 4.6.

In contrast to the QP algorithm where the $kcat$ for every active enzymatic reaction was changed, the SP algorithm limits the changes to the reactions having the biggest effect on the growth rate. The reactions in table 4.6 fall into different categories, but three of them are particularly informative: ATP1, RIP1 and PDC1. ATP1 is the ATP synthase complex, which means that the flux through this reaction is equal to the rate of ATP production through oxidative phosphorylation. RIP1 is a protein in the electron transport chain. When these $kcat$ values for these two reactions are increased, using oxidative phosphorylation will take up less of the total available enzyme fraction and an increase in the oxygen uptake should be seen compared to the base ircFBA model. PDC1 is the pyruvate decarboxylase reaction, and generates acetaldehyde from pyruvate (see figure 2.6), leading to the secretion of either acetate or ethanol depending on the redox state of the cell. The other three reactions have less clear interpretations: ACO1 is a reaction in the TCA cycle, FKS1 produces 1,3-β-D-glucan which is consumed by the biomass reaction, and TDH1 is one of the main enzymes in glycolysis. The biomass reaction *requires* FKS1 to carry flux for the growth to be non-zero. ACO1 is in the TCA cycle, so an increase in the $kcat$ can help drive flux into the TCA cycle. As the TCA cycle generates NADH that can be used to run oxidative phosphorylation, ACO1 can help to drive up the oxygen uptake rate, but as demonstrated in figure 4.5, ACO1 can carry flux even when there is no flux through ATP1.

Looking at the ratios in table 4.6 and comparing it with figure 4.11 is helpful for in-

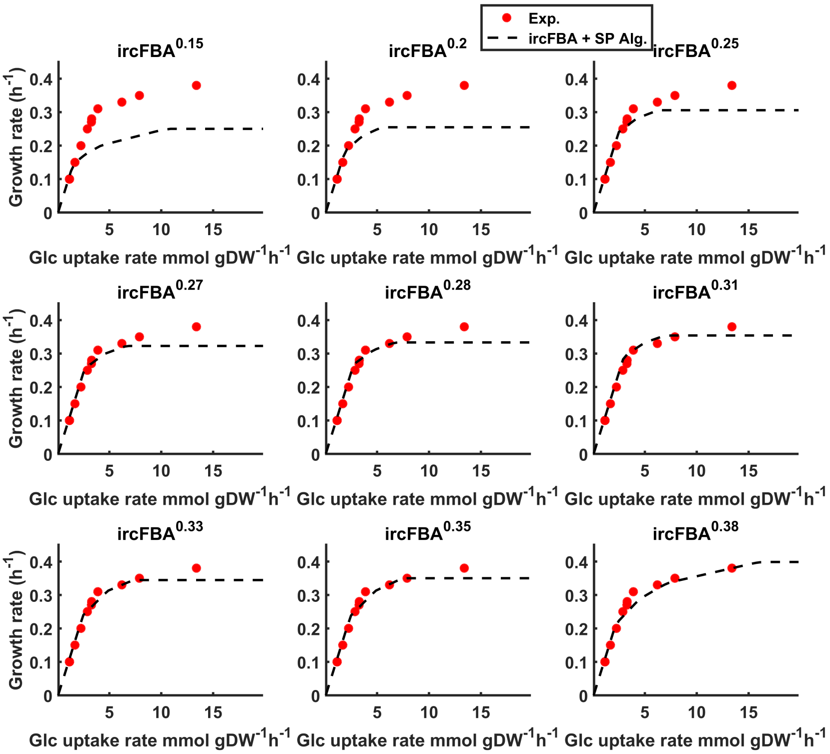


Figure 4.10: Growth rates for ircFBA models fitted to experimental growth rates with the SP algorithm. The in the label ircFBA$^{\mu}$ was the target growth rate used by the SP algorithm for generating the corresponding $kcat$ set.

terpreting why ircFBA$^{0.2}$ to ircFBA$^{0.35}$ have such a high oxygen saturation level. These all have high $\frac{kcat^F}{kcat^E}$ ratios for ATP1 and RIP1, which means that these ircFBA can carry out oxidative phosphorylation with significantly less enzyme investment. On the other hand, ircFBA$^{0.38}$ has increased $kcat$ values for five of the six reactions, but looking at its column in table 4.6 it also appears to be the one most dissimilar from the other ircFBA models. Hierarchical clustering was performed on the columns of the table using a bottom-up approach with average linkage, and the dendrogram can be seen in figure 4.12. The cophenetic coefficient for this dendrogram is 0.93 which suggests that this clustering represents the data well. The column of $\frac{kcat^F}{kcat^E}$ ratios from ircFBA$^{0.38}$ is the last one to be merged into a cluster. The largest source of dissimilarity is the TDH1 ratio, and decreasing it to 1 leads to ircFBA$^{0.38}$ clustering with ircFBA$^{0.15}$ and ircFBA$^{0.20}$ one level lower in the dendrogram. Nonetheless, it was still the last leaf node to be merged into a cluster. The figures in Appendix E show the effect of resetting one of these $\frac{kcat^F}{kcat^E}$ ratios back to 1 in the ircFBA$^{0.38}$ model. While each of the $kcat$ values contribute to the growth rate, none affected the oxygen uptake rate enough to change the basic shape. The phenotype

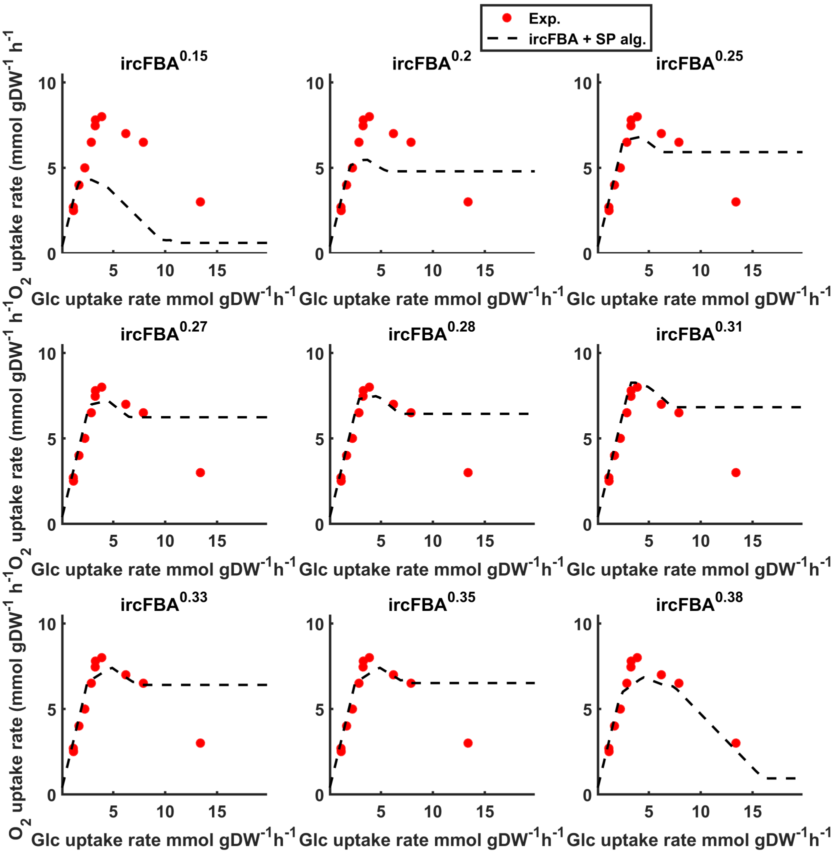


Figure 4.11: Oxygen uptake rates for ircFBA models fitted to experimental growth rates with the SP algorithm. The in the label ircFBA$^{\mu}$ gives the target growth rate used by the SP algorithm for generating the corresponding $kcat$ set.

of ircFBA$^{0.38}$ is therefore not determined by any particular one of these $kcat$ values, but rather by the combination.

Robustness of the SP algorithm

The reliability of the SP algorithm was investigated by varying M, the number of iterations performed, and by looking at the how the $kcat$ set develops. Because the strength of the methods ircFBA was based on is predicting growth rates in single carbon and energy source media when the carbon source is in excess [15, 34, 5], it seems natural that ircFBA$^{0.38}$ should be the best performing ircFBA model. Because of this, 0.38 h^{-1} was chosen as the growth rate to fit the ircFBA model during this robustness test.

The effect of different values of M can be seen in figure 4.13. The SP algorithm was

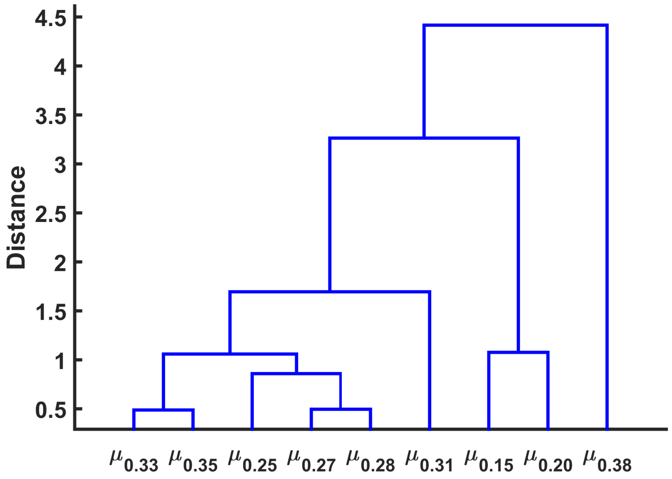


Figure 4.12: Dendrogram of the kcat-ratio coefficient columns in table 4.6. Bottom-up hierchical clustering was performed with average linkage as the between-group distance metric. The leaf nodes are labeled after the SP target growth rate. $\mu_{0.38}$ is an out-group here.

applied to the base ircFBA model with M from 1 to 1000, and the final $\frac{kcat^F}{kcat^E}$ ratios were recorded in each case. This shows that at low values of M, the final $\frac{kcat^F}{kcat^E}$ ratios can vary significantly, but that as M increases, each $kcat$ converges to a stable value. This demonstrates that M has to be careful chosen when applying the SP algorithm to avoid ending getting the final $kcat$ set from the region prior to convergence. Nonetheless, as long as M is large enough, the $kcat$ set converges and the issue can be avoided by simply choosing a big M.

Intriguingly, figure 4.13 shows the set of reactions altered is independent of M under these specific conditions. The SP algorithm works by finding a minimal ratio, t, at each iteration as defined by equation 3.21. The invariance of the $kcat$ set then suggests that no other reactions come sufficiently close to t, even when M is small. Dividing t by $\frac{\Delta kcat}{kcat}$ then gives a measure of how far a reaction in question is from being altered by the SP algorithm in any given iteration. Figure 4.14 shows a plot of this fraction for reactions where $P = \frac{t \times kcat}{\Delta kcat}$ was at least 0.7 in any of the 1000 iterations. This gives a depiction of how the $kcat$ valuess changed while the SP algorithm was running. The only other reaction where P ever reached a level of at least 0.7 was ERG6, and if the target growth rate had been set to be higher, then the trend suggests ERG6 $kcat$ would eventually be altered. This was confirmed by setting the target growth rate to 0.5 h^{-1} and observing that the $ERG6$ kcat was indeed altered.

Unlike the QP algorithm, the Gurobi LP solver had no problems with solving the intermediary LP problems during the SP algorithm, and no value of M caused the problem to fail. The algorithm even works with an M of 1, and simply compensates by making additional iterations until the target growth rate has been reached. This in conjuncture with the other favorable properties demonstrated in this section, underlines the superiority

Table 4.6: The final kcat-ratios obtained by running the SP algorithm with $M = 1000$ independently to reach the target growth rates from the data set in appendix A. The subscript for the column label μ is the growth rate used as the target when running the SP algorithm. The row labels are names for the enzymatic reactions where $kcat$ was adjusted by the SP algorithm

					$\frac{kcat^F}{kcat^E}$				
Reaction	μ_{015}	$\mu_{0.20}$	$\mu_{0.25}$	$\mu_{0.27}$	$\mu_{0.28}$	$\mu_{0.31}$	$\mu_{0.33}$	$\mu_{0.35}$	$\mu_{0.38}$
ACO1	1.0000	1.0000	1.6143	1.6641	1.9162	2.2252	1.0000	1.0000	1.0000
ATP1	1.0168	1.9738	3.4209	3.9068	4.2364	5.0162	4.0940	3.8073	2.2553
FKS1	1.0000	1.0000	1.0858	1.2574	1.3546	1.6281	1.5902	1.6386	2.7593
PDC1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.6179
RIP1	1.0000	1.4933	2.5912	2.9428	3.1962	3.7278	2.9691	2.7357	1.6413
TDH1	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.2750	1.5903	4.6091

of the SP algorithm over the QP algorithm.

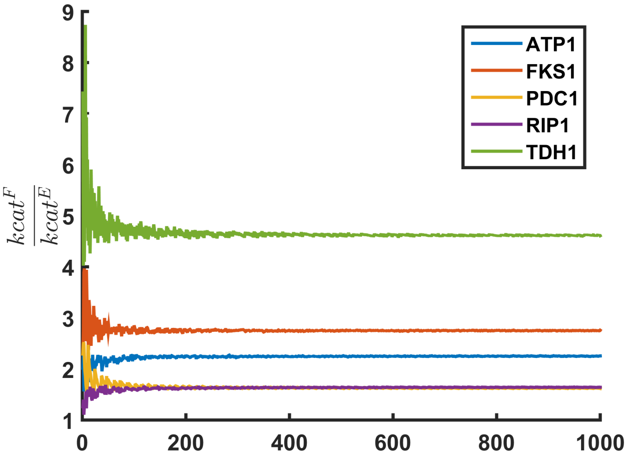


Figure 4.13: The final $\frac{kcat^F}{kcat^E}$ ratio as a function of varying M, the number of iterations performed to be performed by the SP algorithm. After an early period fluctuation, each $kcat$ converges to its final value.

4.2.3 Combining *kcat* sets

Using the approach laid out in section 3.4.3, a linear combination of the columns of table 4.6 minimizing the squared relative deviation of a ircFBA model from the experimental data points was computed. In addition to the 9 columns the table, an additional column of ones was added in order to represent the two lowest growth rates. If this column was weighted with 1, and the others with zero, the $kcat$ set formed by the linear sum would then be the one from the base ircFBA model. The weights can be seen in figure 4.15.

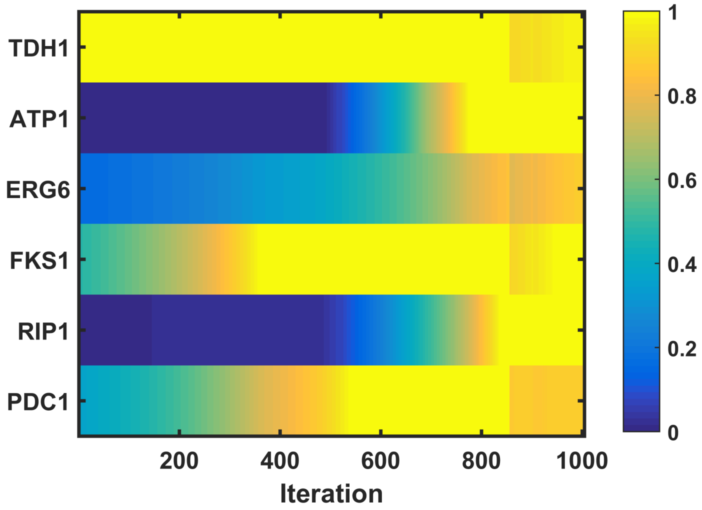


Figure 4.14: A heat map of the relative proximity a reaction is from having its $kcat$ altered during a step of the SP algorithm. The target growth rate was the highest growth rate in the experimental data set (Appendix B). The closer the value is to 1, the closer the reaction is to being altered. All reactions where this relative proximity was at least 0.7 can be seen, and shows that few reactions ever come close to

This shows that the only columns contributing to the linear combination are the ones from ircFBA$^{0.15}$ and ircFBA$^{0.38}$. The growth rates and oxygen uptake rates from ircFBA$^{0.38}$ had the best combination of correlations with the experimental data, which can be seen in table 4.5. As such, ircFBA$^{0.38}$ having a non-zero weight in the linear combination is unsurprising. Figure 4.16 shows the predictions for growth and oxygen uptake rates when using this new $kcat$ set. The ircFBA oxygen uptake rate predictions correlate with the experimental data with a Pearson correlation coefficient, r, of 0.98, with a p-value of 3.0×10^{-7}, while the growth rate predictions correlate with $r = 0.99$ with a value value of 5.4×10^{-10}. The algorithm has therefore succeeded in finding a combination of $kcat$-ratio columns giving a better fit to the experimental data than any of the individual ircFBA models.

In order to investigate the phenotype this $kcat$ set predicts, the metabolite excretion profile was plotted in figure 4.17a. For each combination of a glucose and oxygen uptake rate, the color indicates the largest excretion product. The ircFBA line of optimality is projected down on the floor, and gives the optimal oxygen uptake rate when the glucose uptake rate is fixed. At the lowest growth rates, the ircFBA line of optimality lies on a narrow edge in the phenotype phase plane where no ethanol, pyruvate or acetate is excreted. This region can be identified by observing that it is identical to the FBA line of optimality in this region. A switch then occurs as pyruvate starts to be excreted, followed by acetate, and finally ethanol. As the oxygen uptake rate is unbounded, the ethanol producing phase should start as soon as the line of optimality separates from experimental data [6]. That is, as soon as a breakpoint occurs in the growth rate curve. In the experimental data set

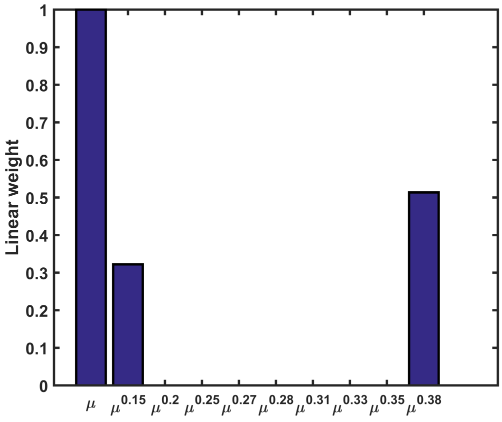


Figure 4.15: Linear weights on the columns of table 4.6 when combining the $kcat$ sets to minimize the relative deviations of the ircFBA model's predictions for growth rate and oxygen uptake from the experimental measurements. The subscript for μ indicates the column of the table the weight represents, with μ without a subscript being an additional column added consisting only of ones.

used to fit the ircFBA model, this should be at a growth rate of between 0.28 to 0.31 h^{-1}. However, figure 4.17b shows that no ethanol is produced until after a growth rate of 0.35 h^{-1}. As was the case for the base ircFBA model, acetate and pyruvate excretion rates are many times greater than what was reported in the article most of the experimental data was sourced from [6, 1]. However, while the ircFBA model's phenotype does not match the real organism's, the algorithm did solve the problem as it was stated. If the SP algorithm, and the method for combining $kcat$ sets, had been applied to a more diverse set of growth conditions it seems reasonable to expect better results. A possible expansion of the ircFBA model to also include transport reactions in the enzyme capacity constraint could also be helpful.

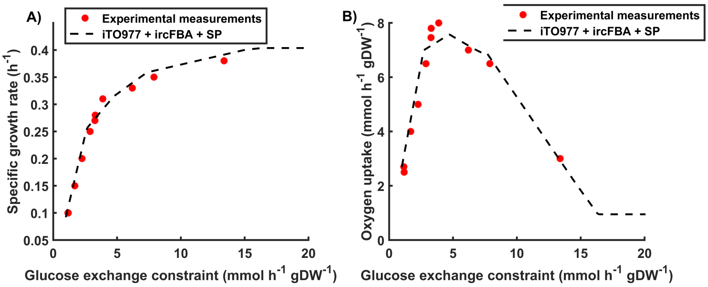


Figure 4.16: ircFBA growth predictions based on combining the $kcat$ sets from table 4.6. A) The growth rate as a function of the upper bound on the glucose uptake rate. B) The oxygen uptake rate as a function of the glucose uptake rate upper bound.

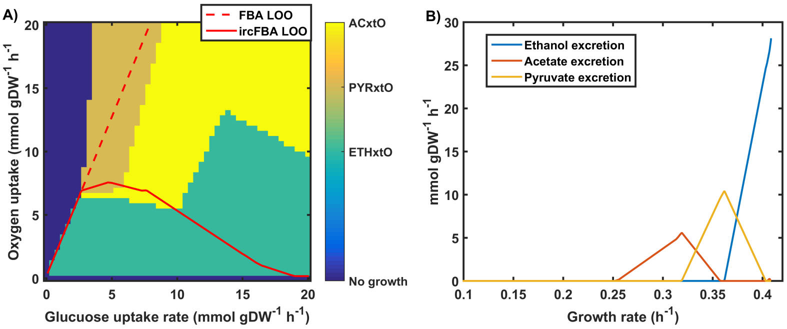


Figure 4.17: A) The major excretion product at any given combination of glucose and oxygen uptake for the ircFBA model generated by combining the $kcat$ columns in table 4.6. At least one of acetate, pyruvate or ethanol was excreted in every point plane, except for the part prior the ircFBA and FBA lines of optimality separating. B) The rate of excretion of acetate, pyruvate and ethanol in the ircFBA model generated by combining the columns in table 4.6. The predicted ethanol production start point, 0.36 h^{-1}, occurs at a later stage than experimental evidence suggests [6].

4.3 ircFBA with randomized *kcat* sets

New $kcat$ sets were created by randomly drawing from the $kcat$ CSV file (see section 3.3.2). $kcat$ sets capable of supporting growth were optimized with the SP algorithm with $M = 1000$ and a target growth rate of 0.38 h^{-1}. Upper bounds were set on both the glucose and oxygen uptake rates following the experimental data set in Appendix B. After the SP algorithm finished, both upper bounds were removed and the unconstrained growth rate along with its flux distribution was recorded.

Figure 4.18 shows the final unconstrained growth rates plotted against the glucose and oxygen uptake rates. Two groups are clearly visible in this plot: a high oxygen group and a low oxygen group. Since the SP algorithm works by optimizing for growth with a set of constraints, no control mechanisms are in place to control for behavior once these constraints are lifted. With the algorithm in its current form, this is an issue that can't be avoided. It should be possible to formulate a different algorithm that avoids increasing $kcat$ values with unwanted side effects instead of just by minimizing a ratio, however. Unfortunately, this falls outside of the scope of this project.

The flux vectors from the 1000 ircFBA solutions with different $kcat$ sets are interesting, as they allow for an examination of the sort of metabolic states attainable by ircFBA. Figure 4.19 shows a heat map of the correlation matrix for enzymatic reactions that carried flux in at least one one the these 1000 ircFBA solutions. The rows and columns of the matrix were sorted by using bottom-up hierarchical clustering with average linkage as the dissimilarity function and an inconsistency metric for determining the clusters. The sparseness of this matrix reveals the great level of freedom and variation in the phenotypes of the 1000 ircFBA models. However, certain fluxes are required for both the low and high oxygen groups in figure 4.18 to sustain a high growth rate. The giant cluster in the middle of figure 4.19 dominates the correlation matrix, and is full of reactions for basic metabolic processes.

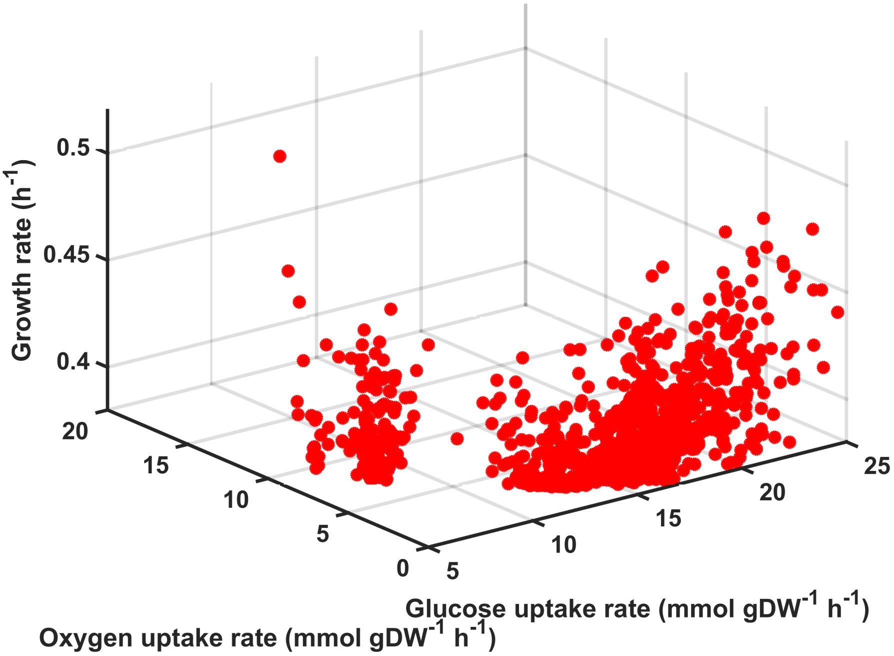


Figure 4.18: Growth rate plotted against oxygen and glucose uptake rates for unconstrained ircFBA models after randomizing the $kcat$ set and optimizing with the SP algorithm. The target growth rate was 0.38 h^{-1}. Individual observations seem to fall into two categories: High glucose, low oxygen or low glucose and high oxygen.

To examine the correlation matrix further, F2C2 was used to find all the flux coupling relationships in the iTO977 metabolic network. Reactions were grouped together based on full and partial coupling. Additionally, reactions that had coupling groups of size 1 were pruned from both its flux coupling group and the flux correlation matrix cluster. Of the 459 enzymatic reactions in figure 4.19, 235 remained at this point. A comparison of the coupling groups with the flux clusters can be done by using the Jaccard index [65]:

$$\frac{F_i \cap V_i}{F_i \cup V_i} \tag{4.1}$$

where F_i is the set of reactions coupled to reaction i, and V_i is the set of reactions clustered with reaction i in figure 4.19. A Jaccard index of 1 means that the sets are identical, while 0 means that they are completely dissimilar. Figure 4.20a is a histogram of the Jaccard index distribution for this comparison. The high degree of similarity between flux clusters and coupling groups is interesting, but has a clear explanation. The flux coupling groups are dependent only on the network structure and will therefore be carried over into the ircFBA models as well. However, re-analyzing these results with different inconsistency cutoffs for the flux clustering leads to some interesting findings. Figure 4.20b shows the mean Jaccard index as a function of the inconsistency cutoff, and it remains robust over

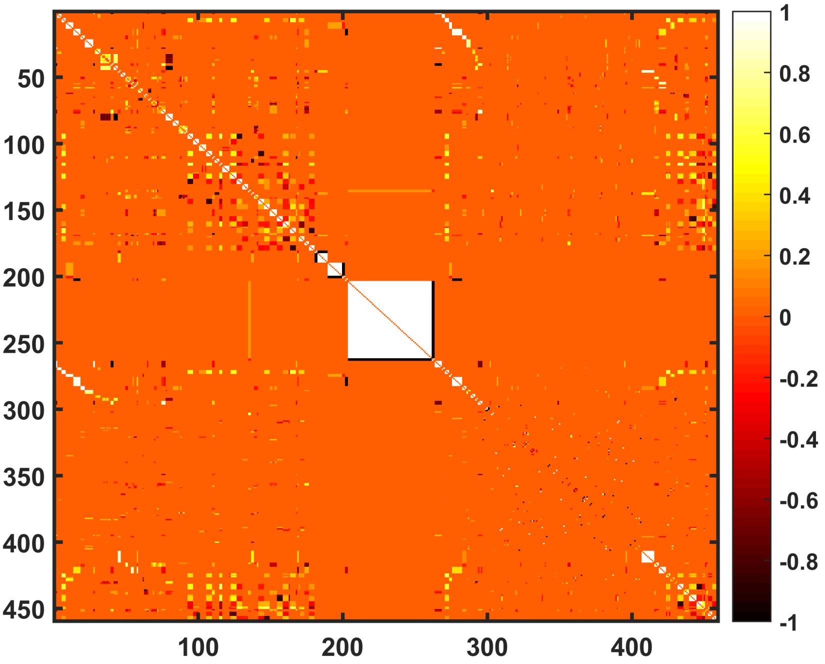


Figure 4.19: A heat map of the correlation matrix for the fluxes from the 459 enzymatic reactions that were switched on in at least one of the 1000 ircFBA models.

most of the range. This suggests that the links in the dendrogram that connect flux coupled reactions are highly consistent. This leads to the clusters retaining integrity when the inconsistency cutoff is very low. Furthermore, since the mean Jaccard index is always significantly different from 1, it shows that there are other ways for fluxes to cluster tightly aside from being coupled. The variation in the phenotypes of the ircFBA models therefore allow flux correlations that are not captured by flux coupling analysis to be discovered.

4.3.1 Correlation of the final *kcat* values with the original *kcat* set

A total of 250 reactions had their $kcat$ values changed by the SP algorithm in at least one of the 1000 ircFBA models. For each of the 250 reactions, all the final values from ircFBA models where the $kcat$ for this reaction was changed during SP algorithm optimization were tabulated. The mean $kcat$ for each calculated, $\log_{10}$ transformed and plotted against the $\log_{10}$ transformed original $kcat$ set in figure 4.21. The Pearson correlation coefficient for these two sets of $kcat$ was 0.37 with a p-value of 1.4×10^{-9}. As these $kcat$ values were obtained by randomizing the $kcat$ set and then optimizing for just one growth rate, this was not an expected result. Whether or not this gives any sort of basis for a method for predicting actual $kcat$ values is unclear however, but further testing with more diverse

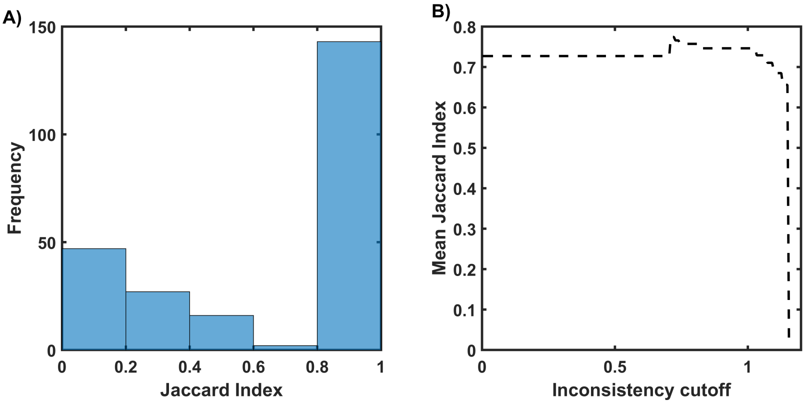


Figure 4.20: A heat map of the correlation matrix for the fluxes from the 459 enzymatic reactions that were switched on in at least one of the 1000 ircFBA models.

growth conditions could lead to an answer.

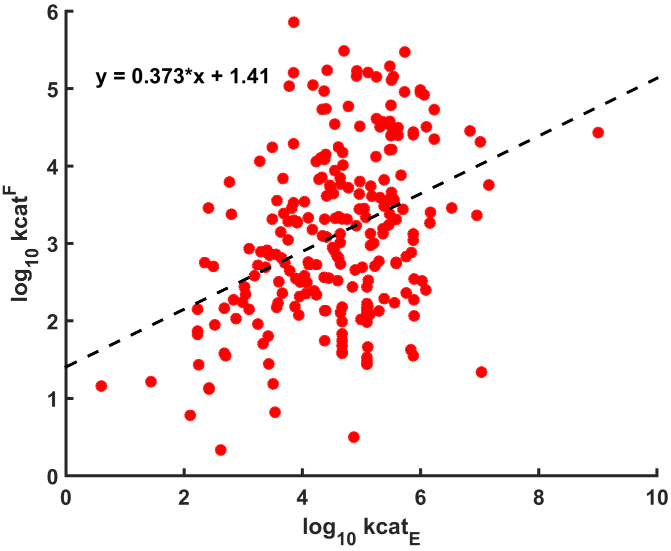


Figure 4.21: The $\log_1 0$ transformed mean of $kcat$, here designated as $kcat_F$ values changed by the SP algorithm after randomization plotted against the $\log_{10}$ transformed original $kcat$ set, $kcat_E$. A linear trend can be seen, confirmed by a Pearson coefficient of 0.37 with a p-value of 1.4×10^{-9}

Chapter 5

Conclusion

A modelling framework named ircFBA was successfully developed to allow incorporation of kinetic data into genome-scale metabolic models. An ircFBA model for *S. cerevisiae* was constructed and implemented in the COBRA toolbox in MATLAB. It was shown to improve on FBA growth rate predictions in the absence of an upper bound on the oxygen uptake by its growth rate saturating. Furthermore, a switch from full respiration over to respiro-fermentation at high levels of glucose was also observed. The growth rates predicted by ircFBA were significantly lower than experimentally observed growth rates, however.

The algorithms developed to close the this growth rate gap showed mixed results. The first algorithm (QP), based on quadratic programming, was able to reach the target growth rate in most cases, but was numerically unstable and prone to failure. The second method was based on using shadow prices from post-optimality analysis. The second algorithm (SP) was a resounding success and was found to only need to change the $kcat$ for a maximum of five reactions to close the growth rate gap to experimental growth rates. Some of the oxygen uptake rate predictions obtained by using the second algorithm were worse than when using the QP algorithm, but could be rectified by finding linear combinations of turnover numbers from different growth rates. This approach was effective for fitting an ircFBA model to a set of experimental data points, but growth rates from more varied experimental conditions should probably be used to really see a benefit. Finally, after randomizing the $kcat$ values and re-optimizing the growth rate with the SP algorithm, it was found that the turnover numbers predicted by ircFBA are correlated with actual turnover numbers to a high degree of statistical significance. This suggests that the method might actually be able to predict reaction $kcat$ values based on the network structure. However, more work is needed to settle this issue.

The favorable performance of the SP algorithm points to its potential applicability for tuning ircFBA model to enable modelling of non-standard strains of *S. cerevisiae*.

Bibliography

[1] Tobias Österlund, Intawat Nookaew, Sergio Bordel, and Jens Nielsen. Mapping condition-dependent regulation of metabolism in yeast through genome-scale modeling. *BMC systems biology*, 7(1):36, 2013.

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

[3] Bernhard Palsson. *Systems Biology: Properties of Reconstructed Networks*. Cambridge University Press, 1st edition, 2006.

[4] J Michael Thomson, Eric A Gaucher, Michelle F Burgan, Danny W De Kee, Tang Li, John P Aris, and Steven A Benner. Resurrecting ancestral alcohol dehydrogenases from yeast. *Nature genetics*, 37(6):630–635, 2005.

[5] Roi Adadi, Benjamin Volkmer, Ron Milo, Matthias Heinemann, and Tomer Shlomi. Prediction of microbial growth rate versus biomass yield by a metabolic network with kinetic parameters. *PLoS Comput Biol*, 8(7):e1002575, 2012.

[6] Karin M Overkamp, Barbara M Bakker, Peter Kötter, Arjen van Tuijl, Simon de Vries, Johannes P van Dijken, and Jack T Pronk. In vivo analysis of the mechanisms for oxidation of cytosolic nadh by saccharomyces cerevisiae mitochondria. *Journal of Bacteriology*, 182(10):2823–2830, 2000.

[7] Kozo Nishida, Keiichiro Ono, Shigehiko Kanaya, and Koichi Takahashi. Keggscape: a cytoscape app for pathway data integration. *F1000Research*, 3, 2014.

[8] Anthony Trewavas. A brief history of systems biology every object that biology studies is a system of systems. francois jacob (1974). *The Plant Cell*, 18(10):2420–2430, 2006.

[9] Bernhard Palsson. *Systems Biology*. Cambridge University Press, 1st edition, 2015.

[10] Zachary A King, Colton J Lloyd, Adam M Feist, and Bernhard O Palsson. Next-generation genome-scale models for metabolic engineering. *Current Opinion in*

Biotechnology, 35:23 – 29, 2015. Chemical biotechnology Pharmaceutical biotechnology.

[11] Andrea A Duina, Mary E Miller, and Jill B Keeney. Budding yeast for budding geneticists: a primer on the saccharomyces cerevisiae model system. *Genetics*, 197(1):33–48, 2014.

[12] JEAN-LUC LEGRAS, Didier Merdinoglu, JEAN CORNUET, and Francis Karst. Bread, beer and wine: Saccharomyces cerevisiae diversity reflects human history. *Molecular ecology*, 16(10):2091–2102, 2007.

[13] Jonas Warringer, Enikö Zörgö, Francisco A Cubillos, Amin Zia, Arne Gjuvsland, Jared T Simpson, Annabelle Forsmark, Richard Durbin, Stig W Omholt, Edward J Louis, et al. Trait variation in yeast is defined by population history. *PLoS Genet*, 7(6):e1002111, 2011.

[14] Jeffrey D Orth, Ines Thiele, and Bernhard Ø Palsson. What is flux balance analysis? *Nature biotechnology*, 28(3):245–248, 2010.

[15] Qasim K Beg, Alexei Vazquez, Jason Ernst, Marcio A de Menezes, Ziv Bar-Joseph, A-L Barabási, and Zoltán N Oltvai. Intracellular crowding defines the mode and sequence of substrate uptake by escherichia coli and constrains its metabolic activity. *Proceedings of the National Academy of Sciences*, 104(31):12663–12668, 2007.

[16] Jan Lundgren, Mikael Ronnqvist, and Peter Varbrand. *Optimization*. Studentlitteratur, Lund, 1 edition, 2010.

[17] Friedrick S Hillier and Gerald J Lieberman. *Introduction to Operations Research*. McGraw-Hill, 9th edition, 2010.

[18] Benjamin Jansen, JJ De Jong, Cornelius Roos, and Tamás Terlaky. Sensitivity analysis in linear programming: just be careful! *European Journal of Operational Research*, 101(1):15–28, 1997.

[19] Panos M Pardalos and Stephen A Vavasis. Quadratic programming with one negative eigenvalue is np-hard. *Journal of Global Optimization*, 1(1):15–22, 1991.

[20] David L Nelson and Michael M Cox. *Lehninger Principles of Biochemistry*. W.H. Freeman and Company, 6th edition, 2013.

[21] Santiago Schnell. Validity of the michaelis–menten equation–steady-state or reactant stationary assumption: that is the question. *FEBS Journal*, 281(2):464–472, 2014.

[22] Gerald P Moss. Recommendations of the nomenclature committee of the international union of biochemistry and molecular biology on the nomenclature and classification of enzymes by the reactions they catalyse. *http://www.chem.qmul.ac.uk/iubmb/enzyme/*
Accessed 06.05.2016.

[23] Ida Schomburg, Antje Chang, and Dietmar Schomburg. Standardization in enzymologydata integration in the world s enzyme information system brenda. *Perspectives in Science*, 1(1):15–23, 2014.

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

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

[26] 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.

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

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

[29] Benjamin D Heavner and Nathan D Price. Comparative analysis of yeast metabolic network models highlights progress, opportunities for metabolic reconstruction. *PLoS Comput Biol*, 11(11):e1004530, 2015.

[30] 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.

[31] 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.

[32] Abdelhalim Larhlimi, Laszlo David, Joachim Selbig, and Alexander Bockmayr. F2c2: a fast tool for the computation of flux coupling in genome-scale metabolic networks. *BMC bioinformatics*, 13(1):57, 2012.

[33] 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, 2011.

[34] Alexei Vazquez, Qasim K Beg, Jason Ernst, Ziv Bar-Joseph, Albert-László Barabási, László G Boros, Zoltán N Oltvai, et al. Impact of the solvent capacity constraint on e. coli metabolism. *BMC systems biology*, 2(1):7, 2008.

[35] 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.

[36] Ida Schomburg, Antje Chang, Christian Ebeling, Marion Gremse, Christian Heldt, Gregor Huhn, and Dietmar Schomburg. Brenda, the enzyme database: updates and major new developments. *Nucleic acids research*, 32(suppl 1):D431–D433, 2004.

[37] Ulrike Wittig, Renate Kania, Martin Golebiewski, Maja Rey, Lei Shi, Lenneke Jong, Enkhjargal Algaa, Andreas Weidemann, Heidrun Sauer-Danzwith, Saqib Mir, et al. Sabio-rkdatabase for biochemical reaction kinetics. *Nucleic acids research*, 40(D1):D790–D796, 2012.

[38] UniProt Consortium et al. Uniprot: a hub for protein information. *Nucleic acids research*, page gku989, 2014.

[39] 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.

[40] Snustad Peter D and Michael J Simmons. Genetics. 2012.

[41] Stacia R Engel, Fred S Dietrich, Dianna G Fisk, Gail Binkley, Rama Balakrishnan, Maria C Costanzo, Selina S Dwight, Benjamin C Hitz, Kalpana Karra, Robert S Nash, et al. The reference genome sequence of saccharomyces cerevisiae: then and now. *G3: Genes— Genomes— Genetics*, 4(3):389–398, 2014.

[42] Thomas Pfeiffer and Annabel Morley. An evolutionary perspective on the crabtree effect. *Frontiers in Molecular Biosciences*, 1, 2014.

[43] Hnin W Aung, Susan A Henry, and Larry P Walker. Revising the representation of fatty acid, glycerolipid, and glycerophospholipid metabolism in the consensus model of yeast metabolism. *Industrial Biotechnology*, 9(4):215–228, 2013.

[44] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The Elements of Statistical Learning*. Springer, Berlin, 2 edition, 2008.

[45] Sinan Saraçli, Nurhan Doğan, and İsmet Doğan. Comparison of hierarchical cluster analysis methods by cophenetic correlation. *Journal of Inequalities and Applications*, 2013(1):1–8, 2013.

[46] Muhammad Usman, Russel Pears, and Alvis Cheuk M Fong. A data mining approach to knowledge discovery from multidimensional cube structures. *Knowledge-Based Systems*, 40:36–49, 2013.

[47] Dietmar Cordes, Vic Haughton, John D Carew, Konstantinos Arfanakis, and Ken Maravilla. Hierarchical clustering to measure connectivity in fmri resting-state data. *Magnetic resonance imaging*, 20(4):305–317, 2002.

[48] MATLAB. *Version 8.6.0.267246 (R2015b)*. The MathWorks Inc., Natick, Massachusetts, 2015. `http://www.mathworks.com/`.

[49] Jan Schellenberger, Richard Que, Ronan MT Fleming, Ines Thiele, Jeffrey D Orth, Adam M Feist, Daniel C Zielinski, Aarash Bordbar, Nathan E Lewis, Sorena Rahmanian, et al. Quantitative prediction of cellular metabolism with constraint-based models: the cobra toolbox v2. 0. *Nature protocols*, 6(9):1290–1307, 2011.

[50] Gregory Warnes. Gnu linear programming kit.

[51] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2015.

[52] Python. *Version 2.5*. The Python Software Foundation, 2015. Available at `http://www.python.org/`.

[53] Jason E Stajich, Todd Harris, Brian P Brunk, John Brestelli, Steve Fischer, Omar S Harb, Jessica C Kissinger, Wei Li, Vishal Nayak, Deborah F Pinney, et al. Fungidb: an integrated functional genomics database for fungi. *Nucleic acids research*, 40(D1):D675–D681, 2012.

[54] Li Li, Christian J Stoeckert, and David S Roos. Orthomcl: identification of ortholog groups for eukaryotic genomes. *Genome research*, 13(9):2178–2189, 2003.

[55] Gregory Warnes. Soappy, 2010.

[56] Sunghwan Kim, Paul A Thiessen, Evan E Bolton, Jie Chen, Gang Fu, Asta Gindulyte, Lianyi Han, Jane He, Siqian He, Benjamin A Shoemaker, et al. Pubchem substance and compound databases. *Nucleic acids research*, page gkv951, 2015.

[57] Janna Hastings, Paula de Matos, Adriano Dekker, Marcus Ennis, Bhavana Harsha, Namrata Kale, Venkatesh Muthukrishnan, Gareth Owen, Steve Turner, Mark Williams, et al. The chebi reference database and ontology for biologically relevant chemistry: enhancements for 2013. *Nucleic acids research*, 41(D1):D456–D463, 2013.

[58] Antti Haapala. python-levenshteint. Version 0.12.0. `https://pypi.python.org/pypi/python-Levenshtein/0.12.0`.

[59] Giovanni Pighizzini. How hard is computing the edit distance? *Information and Computation*, 165(1):1–13, 2001.

[60] JP Van Dijken, J Bauer, L Brambilla, P Duboc, JM Francois, C Gancedo, MLF Giuseppin, JJ Heijnen, M Hoare, HC Lange, et al. An interlaboratory comparison of physiological and genetic properties of four saccharomyces cerevisiae strains. *Enzyme and Microbial Technology*, 26(9):706–714, 2000.

[61] Jan Heyland, Jianan Fu, and Lars M Blank. Correlation between tca cycle flux and glucose uptake rate during respiro-fermentative growth of saccharomyces cerevisiae. *Microbiology*, 155(12):3827–3837, 2009.

[62] Felipe F Aceituno, Marcelo Orellana, Jorge Torres, Sebastián Mendoza, Alex W Slater, Francisco Melo, and Eduardo Agosin. Oxygen response of the wine yeast saccharomyces cerevisiae ec1118 grown under carbon-sufficient, nitrogen-limited enological conditions. *Applied and environmental microbiology*, 78(23):8340–8352, 2012.

[63] Johannes P Van Dijken and Jack T Pronk. Effects of pyruvate-decarboxylase overproduction on flux distribution at the pyruvate branch-point in saccharomyces cerevisiae. *uitgevoerde 13C NMR-experinienten een aanwijzing lcveren voor het optreden van*, page 81, 1998.

[64] Avlant Nilsson and Jens Nielsen. Metabolic trade-offs in yeast are caused by f1f0-atp synthase. *Scientific reports*, 6, 2016.

[65] Raimundo Real and Juan M Vargas. The probabilistic basis of jaccard's index of similarity. *Systematic biology*, 45(3):380–385, 1996.

Appendix

Appendix A

Implementing an ircFBA model based on iTO977 for use with the COBRA Toolbox

The script below creates an ircFBA model if the appropriate variables are defined in the workspace.

Listing A.1: A short Matlab script that assembles the ircFBA model for the ito977 model

```
%%Model is the iTO977 model in the COBRA model container
rcFBA_model = model;
%%rxns is the list of enzymatic reactions
enzRxns = length(rxns);
[nMets,nRxns] = size(model.S);
ircFBA_model.csense(1:nMets+2*enzRxns+1,1) = 'E';
ito977_stoichiometric_matrix = model.S;

for i = 1:enzRxns
    key = rxns{i};
 %% turnoverList is a hash table that maps reaction names to the turnover number
    val = turnoverList(key);
    rxnIndex = rxnToIndex(rxns{i}); %%rxnToIndex maps the reaction name to the iTO977 index
    ito977_stoichiometric_matrix(nMets+i,rxnIndex) = 1;
    ito977_stoichiometric_matrix(nMets+i,nRxns+i) = -1*val;
    ircFBA_model.csense(nMets+i) = 'L';


    ito977_stoichiometric_matrix(nMets+enzRxns+i,rxnIndex) = 1;
    ito977_stoichiometric_matrix(nMets+enzRxns+i,nRxns+i) = val;
    ircFBA_model.csense(nMets+enzRxns+i) = 'G';

end
ito977_stoichiometric_matrix(nMets+2*enzRxns+1,1) = 0;

for i = 1:enzRxns
   index = rxnToIndex(rxns{i});
   ito977_stoichiometric_matrix(nMets+2*enzRxns+1,nRxns+i) = rxnToMedianMW(rxns{i})/1000; %% rxnToMedianMW maps reaction names to the MW in units of Daltons.
end

lb = [model.lb;zeros(enzRxns,1)];
ub = [model.ub;ones(enzRxns,1)];
c = zeros(nRxns+enzRxns,1);
c(1559) = 1; %% 1559 is the coefficient of the growth rate function
b = zeros(nMets+2*enzRxns+1,1);
rev = [model.rev;zeros(enzRxns,1)];
ircFBA_model.S = ito977_stoichiometric_matrix;
ircFBA_model.lb = lb;
ircFBA_model.ub = ub;
ircFBA_model.c = c;
ircFBA_model.csense(nMets+2*enzRxns+1) = 'L';
b(nMets+2*enzRxns+1) = 0.4; %% this is the protein fraction constraint
ircFBA_model.b = b;
ircFBA_model.rev = rev;
```

Appendix B

Uptake rates for glucose and oxygen from minimal glucose media

The experimental data set consists of 11 measurements of growth rate along with oxygen and glucose uptake rates. These are listed in table B.1 and were taken from the iTO977 publication [14]. The iTO977 model predicts the growth rates with high accuracy when the upper bounds for the oxygen and glucose uptake rates are set according to the experimental values, which is demonstrated in figure B.1.

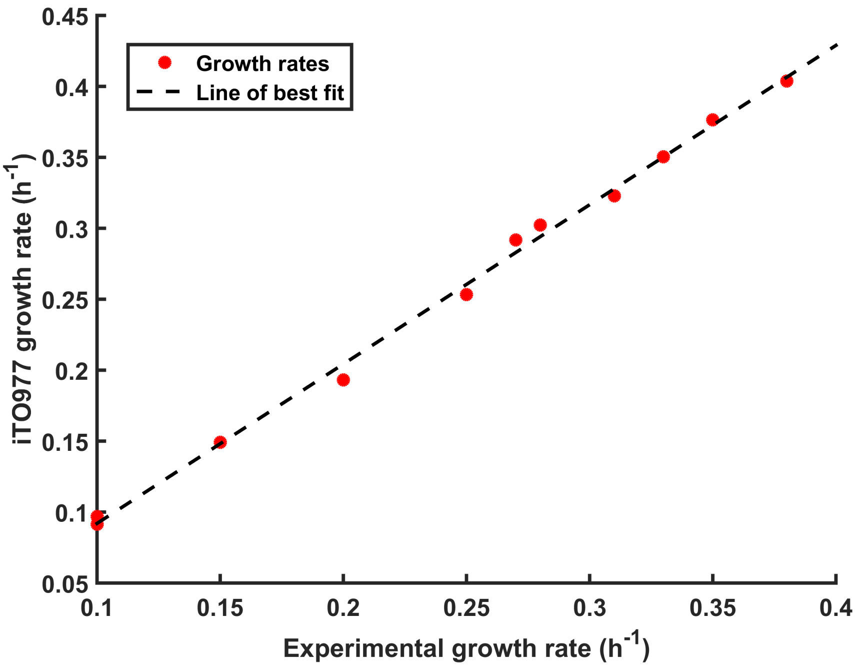


Figure B.1: iTO977 FBA growth rates plotted against the experimental growth rate. The FBA model was constrained with the experimental uptake rates for glucose and oxygen.

Table B.1: The experimental data set used to evaluate and enhance the ircFBA model. Each triad of values comes from aerobic minimal media with glucose as the sole energy and carbon source. The data was taken from [1]

Growth rate	Glucose uptake rate	Oxygen uptake rate
0.1	1.15	2.7
0.1	1.17	2.5
0.15	1.69	4
0.20	2.26	5
0.25	2.88	6.5
0.27	3.27	7.46
0.28	3.29	7.8
0.31	3.88	8
0.33	6.2	7
0.35	7.89	6.5
0.38	13.39	3

Appendix C

KEGG pathway maps of glycolysis and the TCA cycle in *S. cerevisiae*

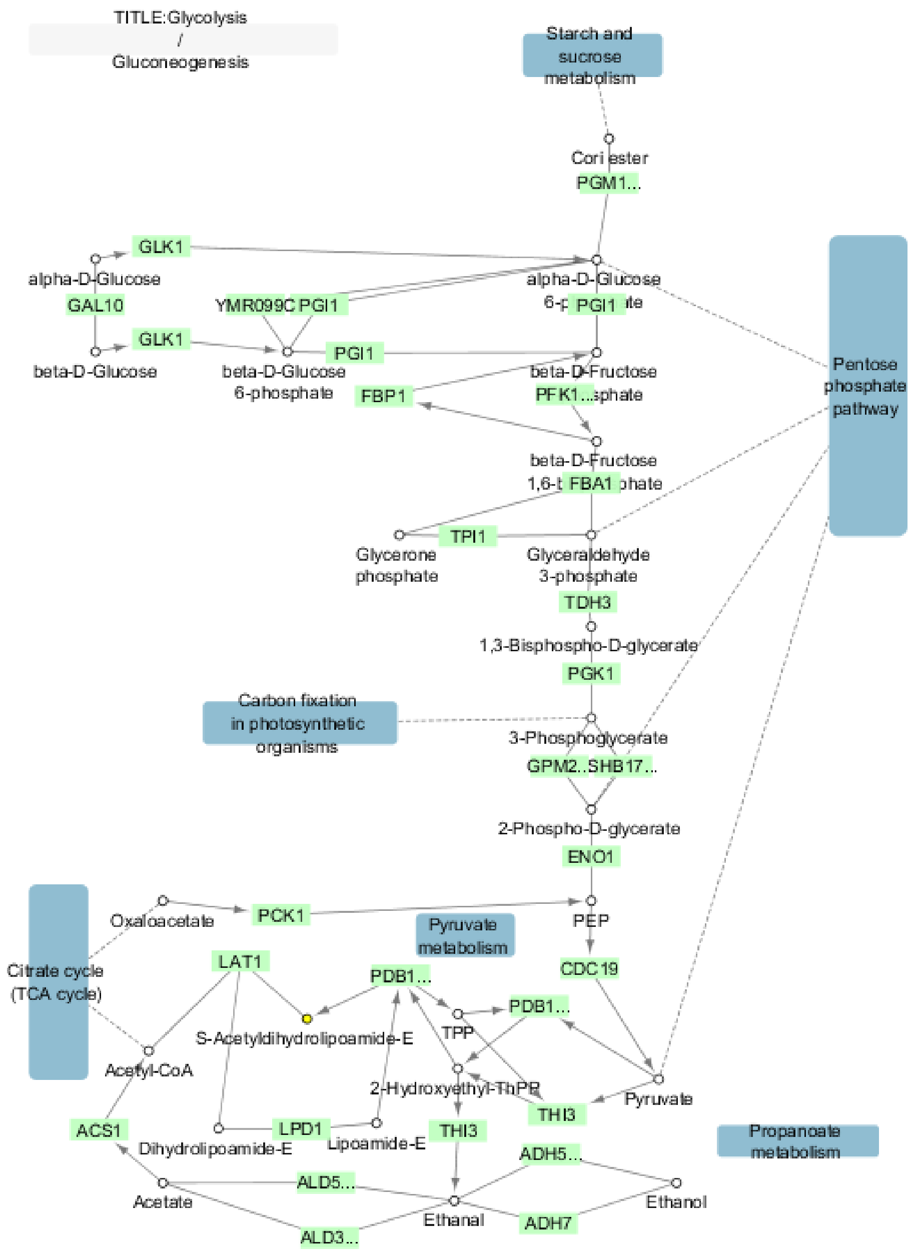


Figure C.1: A pathway map of glycolysis in *S. cerevisiae* taken from the KEGG PATHWAY database and formated with Keggscape [7]. Metabolites are white squares, and the green boxes are the enzymes catalyzing the reaction.

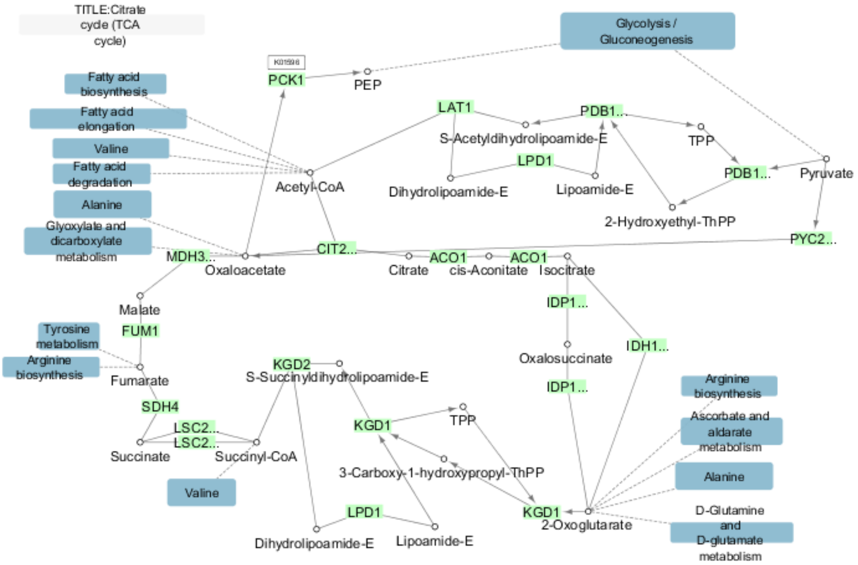


Figure C.2: A pathway map of the TCA cycle in *S. cerevisiae* taken from the KEGG PATHWAY database and formated with Keggscape [7]. Metabolites are white squares, and the green boxes are the enzymes catalyzing the reaction.

Appendix D

Histograms of the relative changes in kcat caused by the QP algorithm

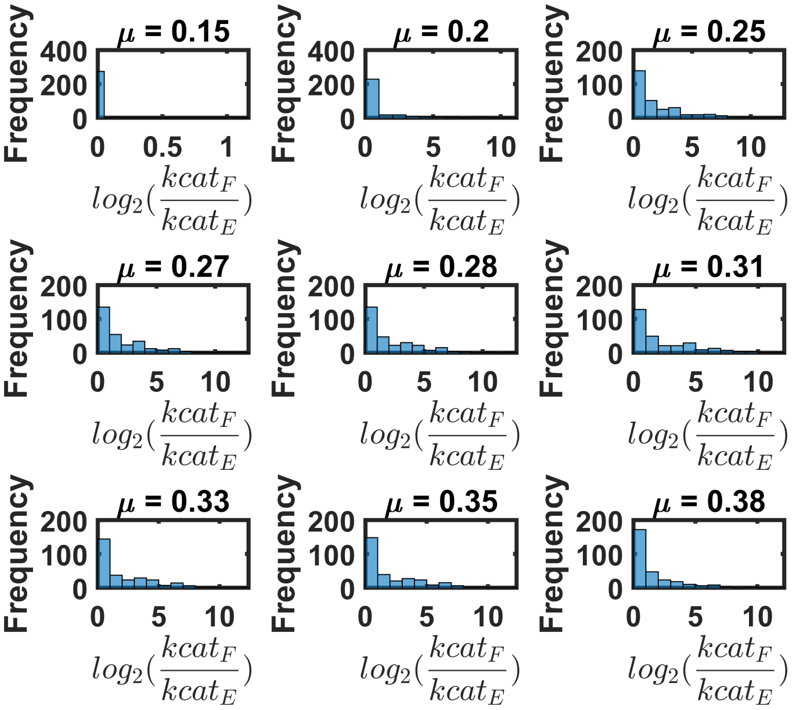


Figure D.1: log_2 of the final ratio, $\frac{kcat_F}{kcat_E}$, for enzyme turnover numbers that were changed when running the QP algorithm to fit for the highest growth in the data set. Changes vary over several orders of magnitude, with the most extreme cases representing 12 doublings.

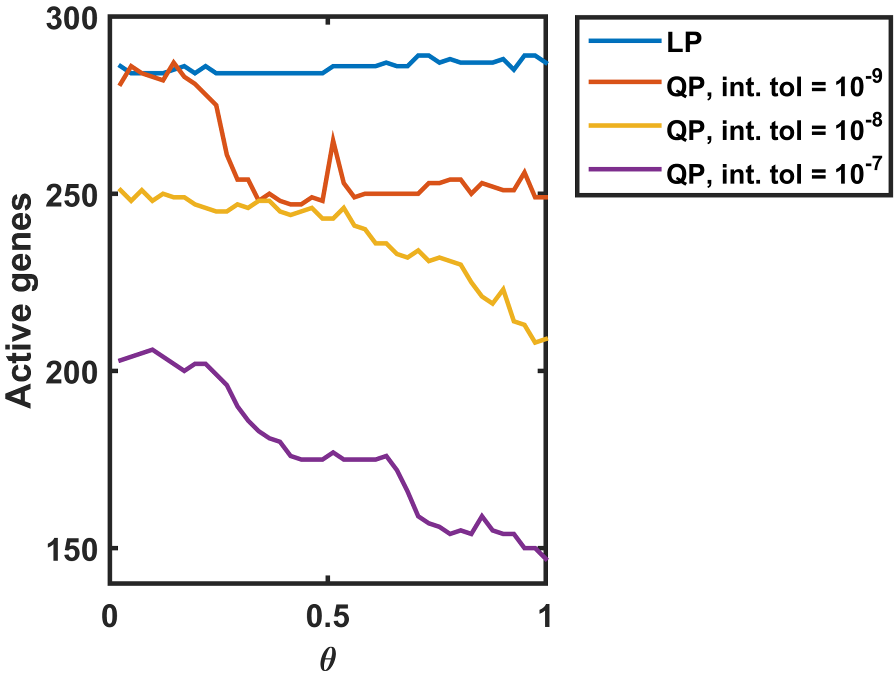


Figure D.2: The number of active genes in intermediary QP algorithm solutions. The LP curve is the number of active reactions when the $kcat$ set was updated according to the preceding QP algorithm step. If no numerical issues were present, these curves would be identical. The large gap, even at the lowest level of feasibility intolerance, shows that the numerical solvers used aren't precise enough to actually solve the QP problem effectively.

Appendix E

The effect of reverting kcat ratios back to its initial value

The ircFBA model obtained by running the SP algorithm on a target growth rate of 0.38 h^{-1} was re-examined by resetting each of the 5 $kcat$ values that were changed by the SP algorithm back to their initial values, one by one.

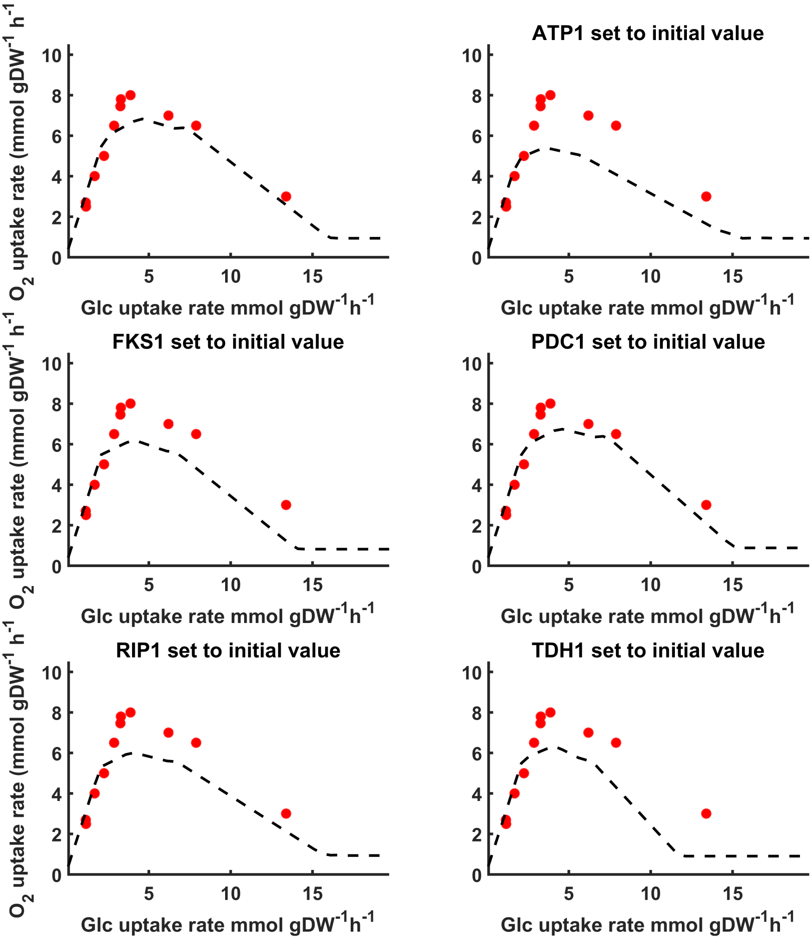


Figure E.1: The effect of resetting individual kcats back to the their initial value for ircFBA$^{0.38}$. The leftmost plot in the upper row is the original ircFBA$^{0.38}$ curve.

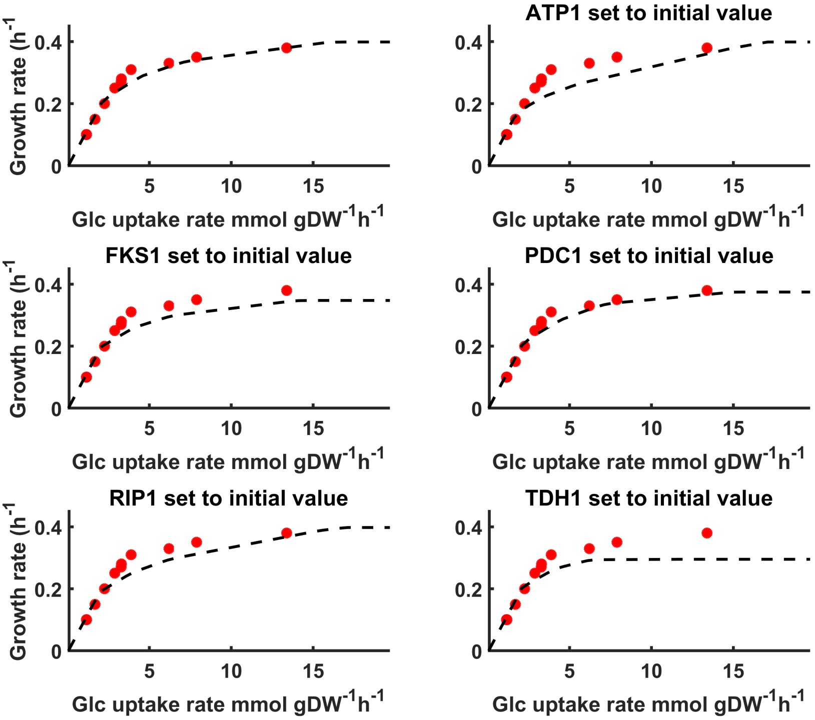


Figure E.2: The effect of resetting individual kcats back to the their initial value for ircFBA$^{0.38}$. The leftmost plot in the upper row is the original ircFBA$^{0.38}$ curve.