Investigating the effect of cellular objectives on genome-scale metabolic models
Biotechnology specialization project Fall 2012

Jarle Magnus Ribe Pahr

Abstract

The study of microbial metabolism through computational methods is a thriving area of systems biology where Flux Balance Analysis (FBA) is a central methodology. FBA employs an optimization procedure guided by an objective function to explore metabolic performance limits and predict favourable biochemical flux patterns. Various objective functions have historically been proposed for choice as the biologically relevant optimization principle. A previous, highly cited study investigated the effect of various objectives on a small-scale model of central carbon metabolism in the bacterium *E. coli*. However, the approach used therein does not scale well to genome-scale models. In this project, a MATLAB program for analysing the effect of cellular objectives on metabolic models up to the full-genome scale has been developed. Quadratic programming can be used with genome-scale models to rapidly compute flux patterns which have the best fit to experimentally gathered data, as measured by the euclidean distance between the experimental and computed flux vectors. The program framework is considered to be in a preliminary working state and some results are presented - further development would allow rigorous analyses to be performed using a variety of models, objectives and constraints. Some issues in the use of data gathered through Metabolic Flux Analysis (MFA) as a reference for FBA results are pointed out.

Contents

Page

1: Introduction **4**

2: Theory **5**
- 2.1 Metabolic models . . . 5
  - 2.1.1 General concepts . . . 5
  - 2.1.2 Mathematical formulation . . . 7
- 2.2 Flux Balance Analysis . . . 12
  - 2.2.1 General concepts . . . 12
  - 2.2.2 Mathematical formulation . . . 15
  - 2.2.3 Applications . . . 16
  - 2.2.4 Software . . . 23
- 2.3 Metabolic Flux Analysis . . . 24
  - 2.3.1 General concepts . . . 24
  - 2.3.2 Stoichiometric MFA . . . 26
  - 2.3.3 ^{13}C-MFA . . . 26
- 2.4 Network optimization . . . 28

3: Evaluating objective functions **32**

4: Methods **38**
- 4.1 Model setup . . . 38
- 4.2 Experimental data . . . 39
- 4.3 Model analysis . . . 40

5: Results and discussion **42**

6: Conclusion **48**

References **49**

Appendix A: Model details **57**

Appendix B: Experimental data **63**

Appendix C: Software **64**

1 Introduction

During the past two decades, the rise of high-throughput technologies for acquisition and analysis of biological data has enabled the rise of systems biology as an interdisciplinary field of study rooted in molecular biology.[1] Molecular biology has traditionally been a reductive science, studying one or a few components at a time. This has led to a good understanding of many basic molecular components and subsystems. However, high-throughput technologies can now be used to enumerate and analyse all components or all interactions of a specific kind in a cell, massively increasing the potential scale of analysis. With this advance, new subfields such as genomics, transcriptomics, proteomics and metabolomics have arisen to tackle the large-scale analysis of their respective areas of molecular biology. In contrast to the traditional reductive approach of molecular biology, systems biology aims to integrate knowledge from the vast datasets produced by these new subfields and other relevant sources.[2] The desired result of systems biology is a holistic understanding of the biological system under study as a whole.

The concept of networks is central to systems biology. In general, interactions between components of the biological system under consideration are described by networks. The components of the system are represented as "nodes" in the network, and two nodes are linked together if they somehow interact. The definition of interaction as used here is broad. For example, in a metabolic network representing biochemical transformations, two molecules can be said to interact if it is possible to transform one into the other by a chemical reaction. Several types of biological networks are commonly studied. Among these are protein interaction networks, transcriptional regulation networks, and metabolic networks. Specialized software is available for investigating the topological properties of diverse kinds of networks.[3]

It is impossible to separate the different biological networks from each other causally. For example, gene regulation is mediated by protein-protein and protein-DNA interactions among other factors, and the state of the metabolic network is affected by gene expression levels. Synthesis of proteins and other gene regulatory molecules is in turn part of the metabolic activity of the cell. Thus, transcriptional regulation networks, protein interaction networks and metabolic networks all interact. However, meaningful information can still be acquired by considering one network at a time. Recently, efforts have also been made to integrate all the commonly studied networks in a whole-cell model to give a more complete biological picture of the processes in a single cell.[4] This advance has been hailed as "the dawn of virtual cell biology".[5]

A central task in systems biology is the construction of models to enable pre-

dictions to be made about the behaviour of complex biological systems under various circumstances. Metabolic models based on biochemical reaction networks have been subject to much research, and Flux Balance Analysis (FBA) is a widely used method for making predictions from such models. In addition to furthering basic biological research, metabolic modelling and FBA has applications in medicine.[6]

In performing FBA, an optimization strategy is applied to obtain a biologically realistic solution from the large feasible set containing mathematically possible solutions satisfying the constraints of the model. Mathematically, the optimization strategy is described by an objective function. This report is concerned with strategies and procedures for evaluating various objective functions. A collection of MATLAB functions for analyzing genome-scale metabolic models with respect to the effect of various objectives and constraints has been developed, and results obtained through their use are presented and discussed. The functions utilize features of the COBRA MATLAB Toolbox, a widely used tool for Flux Balance Analysis and related methods.

2 Theory

2.1 Metabolic models

2.1.1 General concepts

A metabolic model is a mathematical representation of the possible biochemical reactions in a biological system. Such a model may be limited to a few reactions, or include all known reactions in a system. Genome-scale metabolic models are examples of the latter kind, and are so called because they are based on a combination of genomic information covering the whole genome of the organism together with biochemical knowledge regarding the enzymes encoded by that information.[7] Reactions that are not predicted directly by gene sequence must also be added.[7] Such is the case for reactions which are not catalyzed by an enzyme, and thus are not linked to any gene, and for reactions whose genes a cursory genomic inspection fail to reveal. Software tools and methods are available online for semi-automated construction, curation and quality control of genome-scale models, and the number of available models is large and increasing.[8] A selection of software tools supporting model reconstruction is listed in Table 2. Automatic construction of models has made possible the comparison of metabolic networks among a large number of species, and thereby the identification of their conserved properties.[9]

Using a semi-automated approach, a time requirement as short as four full-time weeks for producing a genome-scale model has been reported while at the same time noting the limitations of automatic procedures.[10] A general protocol for reconstruction of genome-scale models has been described by Thiele and Palsson.[11] For a high-quality, well curated model a time-frame closer to a year is suggested by the latter authors. *Escherichia coli* is one of the best studied microbial organisms, but even the latest model of *E. coli* metabolism, iJO1366 contains 208 "blocked" metabolites, corresponding to gaps in the network.[12] Gaps in metabolic models can be classified as *scope gaps* or *knowledge gaps*. Scope gaps are those gaps which exist because not all types of reactions are included in the model, while knowledge gaps are the result of incomplete knowledge about the biochemistry of the organism in question. It is unclear when a genome-scale metabolic model that is considered "complete" will be available.

A selection of notable genome-scale metabolic models is shown in Table 1. A database of genome-scale metabolic models is maintained at the Genome-Scale Metabolic Network Database (GSMNDB) at `http://synbio.tju.edu.cn/GSMNDB/gsmndb.htm`.

Table 1: Selected genome-scale metabolic models

Model	Organism	Reactions	Metabolites
RECON 1 [13]	Human	3311	2766
AraGEM [14]	*A. thaliana*	1567	1748
AlgaGEM [15]	*C. reinhardtii*	1725	1862
iJE660 [16]	*E. coli*	627	438
iJR904 [17]	*E. coli*	931	625
iAF1260 [18]	*E. coli*	2077	1039
iJO1366 [19]	*E. coli*	2251	1136
iJE303 [20]	*H. influenzae*	488	343
Yeast 5 [21]	*S. cerevisiae*	1102	924

Metabolic models constructed from public databases may be found to be biologically unrealistic for various reasons, and manual curation is necessary to ensure accurateness.[27] Tools are available for finding and removing errors, corresponding for example to thermodynamically or stoichiometrically impossible situations, in completed models.[28] One of the main requirements of a metabolic model is that it is without major gaps. That is, the model

Table 2: Software tools for metabolic model reconstruction

Software	Features
GEMSiRV	Metabolic models simulation, reconstruction and visualization.[22]
MEMOSys	Management, storage and development of metabolic models with version control.[23]
Merlin	Automatic gene annotation, export to metabolic model in SBML format.[24]
rBioNet	MATLAB environment for model reconstruction with quality control measures.[25]
Model SEED	Web-based, automated high-throughput generation, optimization and analysis of metabolic models.[26]
MetaFlux	Generation of FBA models directly from pathway/genome databases through the Pathway Tools system.[10]

should include all major reactions which are possible in the actual biological system. After reactions have been mapped from the genomic information available from the organism, gaps in the model may be identified and eliminated by an automatic routine.[29] Computational methods for gap-filling have been reviewed by Orth and Palsson.[30] Still, models that are in themselves not incorrect may still give rise to thermodynamically prohibited flux distributions when methods such as flux balance analysis is applied. Infeasible solutions may potentially be eliminated by changing the network topology while still maintaining the optimality of the found solution.[31]

The Systems Biology Markup Language (SBML) is a commonly used format for describing metabolic models.[32]

2.1.2 Mathematical formulation

The stoichiometric matrix is at the heart of mathematical treatments of metabolic networks (For an introductory text to linear algebra and matrix operations, see [33]). The stoichiometric matrix, denoted S, relates all the compounds and chemical reactions which is part of the network.[34] In the case of a genome-scale metabolic model, this should include all metabolites and all possible biologically significant chemical reactions in the cell.

The stoichiometric matrix contains as its elements the stoichiometric coefficients for each of the compounds and all the reactions included in the network. Each rows of the matrix corresponds to one compound, and every column corresponds to one reaction. Thus, the element $S_{i,j}$ gives the stoichiometric coefficient of compound no. i in reaction no. j.

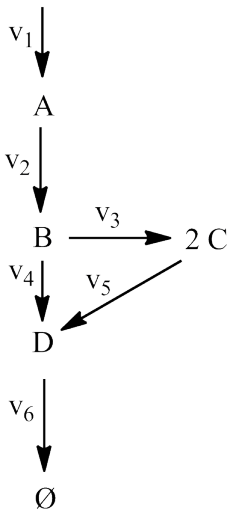

Figure 1: A simple example of a reaction network.

A simple example is given in Figure 1. Consider the reaction network shown. This network contains 4 compounds and 6 reactions. Using conventional chemical notation, we can write the reaction equations as follows:

$$\begin{aligned}
&\xrightarrow{v_1} A \\
&A \xrightarrow{v_2} B \\
&B \xrightarrow{v_3} 2\,C \\
&B \xrightarrow{v_4} D \\
&C \xrightarrow{v_5} D \\
&D \xrightarrow{v_6} \phi
\end{aligned}$$

The stoichiometric matrix then becomes

$$S = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & -1 & 0 & 0 \\ 0 & 0 & 2 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 & 1 & -1 \end{bmatrix}$$

In every row, the number of non-zero elements equals the number of reactions that the metabolite corresponding to that row participates in.

Note that, unlike what is common for chemical equations, in the first and last equations a metabolite is present only on one side of the equation. The equations are thus not balanced with respect to mass. This is acceptable because the equations are written from the point of view of an open system. A cell, the system usually represented by the reaction network, can exchange mass with the environment by both active and passive processes. In the context of a metabolic network, a metabolite "appearing from nothing" usually implies transport of the metabolite in question from the outside to the inside of the cell. It may also be represent production of a metabolite from a large pool of precursor metabolites, when the production rate is low such that the change in the amount of precursors is insignificant. Likewise, when transport of a metabolite out of the cell is modelled, the metabolite effectively 'disappears'. The reactions v_1 and v_6 in the set of equations above could thus be interpreted as transport reactions facilitating the transport of metabolites A and D into and out of a cell, respectively. In the last equation, the symbol ϕ is used to denote the empty set, signifying that the metabolite in question is removed from the system under consideration.

In general, because most metabolites participate in few reactions compared to the total number of reactions, and likewise most reactions involve only a few metabolites, most elements of S are zero. That is, S is a *sparse matrix*. This has importance for the amount of work needed when performing computations using the matrix.

A visual depiction of a stoichiometric matrix and its sparseness is shown in Figure 2.

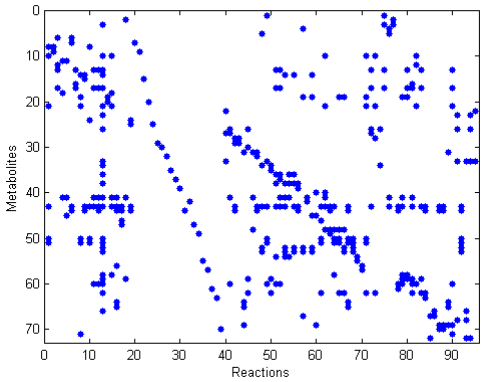

Figure 2: Visual representation of the stoichiometric matrix of an *E. coli* core metabolism model with 72 metabolites and 95 reactions, showing non-zero elements. The stoichiometric matrix S is a sparse matrix, most of its entries being zero.

The stoichiometric matrix only describes what reactions are possible in the

system - it contains no information about the rates at which those reactions proceed. This information can be encoded as a flux vector:

$$\mathrm{v} = (v_1, v_2, ..., v_n) \tag{1}$$

Obtaining a flux vector from a stoichiometric matrix, a set of constraints and a biologically relevant objective function is the goal of Flux Balance Analysis. Also of relevance is a description of the change in concentrations of all metabolites. We write the concentration vector as

$$x = (x_1, x_2, ...x_m) \tag{2}$$

As seen above, in a metabolic network with m metabolites and n reactions, the dimensions of the stoichiometric matrix is m x n.

$$\dim(x) = m \tag{3}$$

$$\dim(v) = n \tag{4}$$

$$\dim(S) = m \times n \tag{5}$$

The stoichiometric matrix can be considered a linear transformation of the flux vector to a vector of concentration time derivatives.[34] Using compact matrix notation, we write

$$\frac{d\mathrm{x}}{dt} = \mathrm{Sv} \tag{6}$$

For each metabolite i, summing the products of the element $S_{i,k}$ of the stoichiometric S multiplied by element k of the flux vector for all values of k, gives the change in concentration for that metabolite with respect to time.

$$\frac{d\mathrm{x}_i}{dt} = \sum_k s_{ik} v_k \tag{7}$$

Extreme pathways and elementary modes: Extreme pathways and elementary modes are two closely related concepts which find use in the analysis of metabolic pathways.

The elementary modes of a network are a set of vectors mathematically derived from the stoichiometric matrix, and have the following properties [35]:

- Each given network has a unique set of elementary modes.
- Each elementary mode consists of the minimum number of reactions that it needs to exist as a functional unit allowing a steady state flux.
- The elementary modes are the set of all paths through the metabolic network consistent with the previous property.

If some reactions in the network are irreversible, the set of flux vectors allowable under the steady-state requirement is reduced to a subset of the null space.[36] A (general) flux mode is defined as a steady-state flux pattern in which the proportions of fluxes are fixed while their absolute magnitudes are indeterminate [36]. Elementary flux modes have the further property of being unique and it can be shown that the set of elementary flux modes is a linearly independent basis for the steady state solution space.[36]

Biologically, elementary modes can be considered as minimal sets of enzymes capable of generating steady state fluxes.[37] Elementary modes can be used to determine maximal yields for biotransformations [36] and to determine the calculability of fluxes when performing Metabolic Flux Analysis (MFA) to determine *in vivo* flux distributions.[38]

The extreme pathways represent the edges of the steady state solution space. Any flux distribution achievable by the metabolic network can thus be represented by a linear combination of one or more extreme pathways.[39] Formally, an extreme pathway is a set of convex basis vectors derived from the stoichiometric matrix with the following properties [35]:

- Each given network has a unique set of extreme pathways.
- Each extreme pathway consists of the minimum number of reactions needed to exist as a. functional unit
- The extreme pathways are the linearly independent subset of elementary nodes.

The set of elementary modes is a superset of the extreme pathways, and the number of extreme pathways is less than or equal to the number of elementary modes.[35]

Any steady state flux vector v can be represented by a non-negative linear combination of extreme pathways or elementary modes.[35] If P is a matrix with the set of extreme pathways or elementary modes contained in its

columns and α is a weighing vector with its elements being weighing factors on the columns in P, the relationship is described by the following equation:

$$P\alpha = v; \quad \alpha_i \geqslant 0 \tag{8}$$

Elementary mode and extreme pathway analysis does not scale well to genome-scale networks, and the concept of elementary flux patterns has been introduced to allow the application of elementary-mode tools to genome-scale networks.[40]

Elementary Flux Mode (EFM) and Extreme Pathway (ExPa) analysis belong to the class of *unbiased* methods which describe all allowable steady state flux distributions.[41] Analysis using extreme pathways can be combined with Flux Balance Analysis in studying metabolic function.[42] For example, change in metabolic behavior can be calculated using the optimization principles of FBA and interpreted by the resulting change in the use of extreme pathways. Elementary flux modes and extreme pathways are not considered further in this report.

2.2 Flux Balance Analysis

2.2.1 General concepts

Flux Balance Analysis (FBA) is one of several methods associated with metabolic models. It is a mathematical method for analysing metabolism, based on linear optimization theory.[43] A linear programming algorithm is used to find a flux profile (a vector specifying the fluxes of all reactions in the network) which optimizes a specified *objective function*. It should be noted that the term "flux balance analysis" does not always refer strictly to this optimization-based process. Written without capital letters, the term flux balance analysis may refer to any procedure that seeks to determine a flux profile under the basic assumption of metabolic steady state, while Flux Balance Analysis or FBA *today* usually refers to the specific method described below. In general, there has been some confusion of terms in the field, so due attention should be paid to the terms used to avoid misunderstanding.

Using the stoichiometric matrix as the mathematical representation of the reaction network, the goal of FBA is to find a biologically relevant steady state flux distribution. A calculated flux distribution is a solution of the stated FBA "problem". Mathematically, a steady state is an invariant solution where the variables under scrutiny do not change with time. In this

case, the variables in question are the concentrations of metabolites in the modelled system. The fundamental constraint in FBA is that of mass balance, and at steady state the net change in concentration of all metabolites should be zero.

Keeping in mind the view of the stoichiometric matrix S as a linear transformation of the flux vector to a vector of time derivatives of the concentrations, a mathematical formulation of this requirement is

$$Sv = 0 \tag{9}$$

By solving this equation, the requirement that the change in concentration for all metabolites should be zero can be fulfilled without considering the actual concentrations.

When a cell is growing clearly it is acquiring mass, and as such is not in a static steady state. The problem is solved by including a reaction describing the production of biomass in the model. The stoichiometric coefficients for this reaction are based on experimental determinations of the overall biomass composition of the organism in question. Typically, the right hand side of the biomass reaction equation is empty - the biomass reaction presents a 'drain' for the metabolites used to produce biomass, allowing the flux balance to hold. By appropriate scaling of the stoichiometric coefficients in the biomass reactions, a unit of flux through the biomass reaction can be made equal to a unit of growth rate. This scaling is dependent on the units used to describe the fluxes in the model - note that the stoichiometric matrix represents an inherently dimensionless network. Typically, model fluxes are given units of mmol/gDW$\cdot h$. One mmol is 10^{-3} mole, where 1 mole $= 6.0 \times 10^{23}$ molecules, gDW is the dry weight of cell mass in grams and h is the reaction time in hours. The biomass reaction is scaled so that a flux of one through the biomass reaction equals a growth rate of 1 h^{-1}, or a cell doubling time of one hour.[43]

The prediction of growth rates without much interest in the global flux distribution has been one of the main uses of FBA so far. More recently, the use of FBA as a means of predicting the actual intracellular fluxes has received attention.[44] The use of FBA for this purpose is complicated by the fact that there are typically many solutions of a single FBA problem that are equivalent with respect to biomass production/growth rate - the solutions of the FBA problem are said to be degenerate. Flux vectors that give the same value of the objective function are also called alternate optimal solutions or equivalent phenotypic states.[45] While growth rates are comparatively simple to measure, intracellular fluxes require more advanced methods, and

the availability of experimentally determined values is limited. This poses a problem for the evaluation of the predictive value of FBA and related approaches.

FBA is a *constraint-based* approach to metabolic modelling. Without specific constraints, there is a very large number of steady state solutions of the flux balance equation system. Most of these are biologically unrealistic or meaningless. To achieve a biologically relevant result, the solution-space must be reduced to those solutions which are biologically achievable by applying biologically relevant constrains.These are applied as bounds on the reaction rates at the upper, lower or both ends. Constraints can be set based on thermodynamics, knowledge about enzyme activities, and data from high-throughput experiments in transcriptomics, etc.[46] It is an important point that constraints can only reduce the solution space, not increase it. Identification and incorporation of new constraints will be important for the future of constraint-based modelling by allowing more accurate description of cellular behaviour.[47]

It is important to keep in mind that FBA gives the maximal theoretical performance with respect to any objective as subject only to mass balance and the explicitly stated constraints. In the real biological system, numerous biological limitations apply which are not captured by the model. Other evidence lacking, FBA results should therefore be viewed as performance limits to which a cell may or may not approach.

Thermodynamic considerations: Nigam and Liang presented an algorithm for removing thermodynamically infeasible loops in flux distributions determined by FBA while still maintaining the optimality of the computed solutions.[31] Automatic assignment of thermodynamic constraints is also a possibility.[48] At the same time, inclusion of irreversibility constraints based on a priori knowledge can capture the limits imposed by thermodynamics.[49] However, pre-set definitions of reversibility do not take into account the dependence of reversibility on the intracellular conditions.[50] Thus, curation by hand and algorithmically designated irreversibility of reactions are approaches that can complement each other. Other examples of research in the area include methods for investigating the Gibbs free energy landscape of a metabolic network at steady state and the computation of feasible reaction directions directly from the stoichiometric matrix.[51][52] Thermodynamics-based metabolic flux analysis (TMFA) was described by Henry et al.[53]

Extensions of FBA: In its basic form, FBA does not take into account the transcriptional state and genetic regulation of the model system. The term regulatory FBA (rFBA) is applied to the analysis of combined metabol-

ic/regulatory networks using FBA.[54] Integration of FBA with transcriptional regulatory networks and ordinary differential equations (ODEs) for predicting metabolite concentrations and phenotypes has been called integrated FBA (iFBA) and was found to be an improvement over rFBA or ODEs alone.[55] Furthermore, FBA has been used as one of several modules in whole-cell computational modelling.[4] Lewis, Nagarajan and Palsson list over 100 methods in their review of the "phylogeny" of constraints-based modelling methods.[41]

Alternatives to FBA: One of the advantages of FBA, the low number of parameters required, is also a reason for its limitations. Other techniques for metabolic network analysis which employ more empirically determined parameters include kinetic modelling [56] and Metabolic Control Analysis (MCA).[57][58] However, these are hard to apply at the genome-scale level because of the large number of parameters involved.[59] Energy Balance Analysis [60] is an extension of FBA taking thermodynamic rules into account, while Feasability Analysis (FA), a method inspired by FBA and incorporating kinetic interactions, has been suggested as an approach to understanding regulation of metabolism.[61] A framework based on cybernetic theory has recently been employed to make predictions about the dynamic behavior of mutant strains from limited data gathered from wild-type organisms.[62]

2.2.2 Mathematical formulation

The environmental factors of nutrient availability is accounted for by restricting nutrient uptake rates. This is done by specifying bounds on the transport and/or exchange reactions in the model. Transport reactions are those reactions which model the movement of metabolites into or out of the cell, while exchange reactions model the exchange of metabolites between the 'immediate' extracellular environment, which is considered a part of the model system, and the larger environment which is not part of the system. The same metabolite is modelled as separate species in the intracellular and extracellular compartments - transport is modelled by inter-converting these species.

The definition of extracellular metabolites should not be considered an attempt to actually describe events in the immediate vicinity of the cell, but rather a mathematical abstraction. This abstraction is necessary because a single metabolite may have several transport reactions. By constraining the exchange reaction for a metabolite, the totality of the transport reactions for that metabolite is simultaneously constrained. Exchange reactions are typically defined as going in the positive direction when a metabolite

is removed from the model. When a cell is consuming a metabolite taken up from the environment, the flux through the exchange reaction for that metabolite would then have a negative value. To simulate growth in an environment where a specific nutrient is unavailable, the exchange reaction for that nutrient is constrained to zero, or to positive values only if excretion of the metabolite is possible. Non-uptake reactions can likewise be constrained based on information about their plausible limits. Mathematically, we write

$$LB_i \leqslant v_i \leqslant UB_i \tag{10}$$

where v_i is the reaction rate, LB_i and UB_i represent the lower and upper bounds for each specific reaction.

A solution of an FBA problem is a flux vector describing the reaction rates for all the reactions included in the model.

$$\mathrm{Z} = \mathrm{c}^{\mathrm{T}}\mathrm{v} \tag{11}$$

The most commonly used objective function for genome-scale models optimizes the model with respect to the reaction flux in a reaction corresponding to the production of biomass. This artifical reaction is based on experimentally determined values for the biomass composition of the organism in question, and can be considered a "catch-all" intended to capture the effect of all reactions leading to growth and cell division. Other objective functions have also been proposed and used. For a review of the biomass objective function, see [63].

To summarize, a standard FBA problem can be formulated as follows [64]:

Find a flux vector

$$\mathrm{f} = [v_1, v_2, ..., v_n] \tag{12}$$

that maximizes the objective function:

$$\begin{aligned} \mathrm{Z} &= \mathrm{c}^{\mathrm{T}}\mathrm{v} \\ \text{subject to: } & Sv = b = 0 \\ \text{and: } & LB_i \leqslant v_i \leqslant UB_i \end{aligned}$$

2.2.3 Applications

In this section some common methods used in conjunction with or related to FBA are briefly described. This is intended to give an overview of some of

the applications of metabolic models and flux balance analysis. Most attention has been focused on the problem of *strain design* - identifying genetic modifications that would allow the overproduction of a desired metabolite - and a variety of algorithms has been designed for this purpose.

Flux Variability Analysis (FVA): Often, there exists many solutions to a given FBA optimization problem which all are equally optimal. While the value of the objective function remains constant, the flux through any given reaction may vary between these degenerate solutions. In Flux Variability Analysis, the range of possible values for specific fluxes is determined. Thus, the maximum and minimum values of each flux that allows the optimal objective function value can be determined.[65]

Phenotypic phase planes (PhPP): When performing a regular FBA analysis, the solution obtained is generally valid for only a single set of constraints and does not give an immediate impression of how varying those constraints would change the solution. Phase plane analysis can be used to consider the effect of variations in two constraining reactions, such as uptake fluxes of nutrients.[66]

To make a two-dimensional phenotypic phase plane plot, the flux values of two reactions are used as the axes of the two-dimensional plot, and the FBA algorithm is run a number of times, each run constraining the reaction rates to a single point in the plane.

Using a procedure known as shadow price analysis, the plane can be divided into a finite number of distinct "phases', where the shadow price for the reactions is constant. The shadow price relates the change in availability of a nutrient (or more generally, a change in the constraint on a reaction) to the change in the maximal value of the objective function. Changes in shadow prices can be related to metabolic behavior, which is distinct in each phase, for example giving different excretion products.[66] For the explanation below, it will be assumed that phase plane analysis is applied to the uptake rates of two nutrients.

The definition of the shadow price of a metabolite is the negative of the partial derivative of the objective function with respect to the corresponding element in the right hand side vector:

$$\gamma_i = -\frac{\partial Z}{\partial b_i} \tag{13}$$

More information can be added to the plot by drawing isoclines for the value of the objective function. The isoclines are lines where the maximal objective function value is constant. The slope of the isoclines can be calculated from

the ratio of the shadow prices of the two metabolites used in the plot, and is denoted α.

$$\alpha = -\frac{\gamma_A}{\gamma_B} = \frac{\partial Z/\partial b_A}{\partial Z/\partial b_B} \tag{14}$$

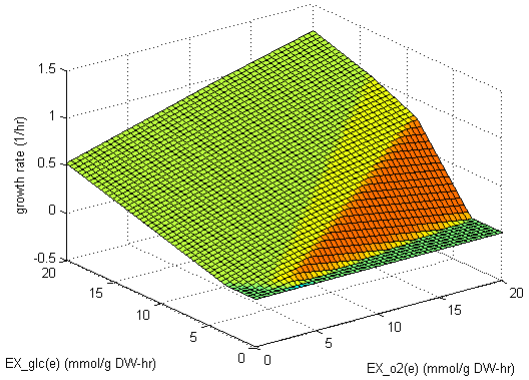

Figure 3: Phase plane plot of oxygen and glucose uptake in a model of *E. coli* core metabolism. Generated using the COBRA Toolbox.

As the shadow prices are constant within each phase, the value of α and the slope of all isoclines drawn through a phase will also be constant. In the case of single substrate limitation, the shadow price of one of the metabolites will be zero, and the slope of the isocline will be either zero or infinite, corresponding to a horizontal or vertical line. A negative value of α implies dual substrate limitation: increasing the availability of either nutrient will increase the objective function. Phases with a positive α value are called *futile* regions because increasing the uptake of one of the nutrients will decrease the objective function. That nutrient can then be considered to exist in excess and has a net negative value for the cell. A phenotype phase plane may also contain infeasible regions where no growth is possible.

A three-dimensional representation can be made with the values of the selected fluxes mapped to the two axes of the horizontal plane, and the growth rate mapped to height in the third dimension. This makes it possible to form an immediate impression of the optimal combination of uptake rates for two nutrients, all other conditions being equal.

Minimization Of Metabolic Adjustment (MOMA): Much of the application of FBA has been aimed towards predicting the phenotypes of gene deletion mutants. In a simple manner, this may be attempted by removing all reactions associated with one or more genes in a model of the "wildtype"

organism before running an FBA optimization. However, it has been noted that while an assumption of metabolic optimality might be justified for wild-type organisms exposed to long-term evolutionary pressure, the assumption might not hold in newly created strains.[67] The minimization of metabolic adjustment (MOMA) algorithm is based on the hypothesis that after a perturbation of the metabolic network by way of one or several gene knock outs, the flux distribution immediately afterwards will tend not towards optimality, but towards a *minimal redistribution* of the fluxes with respect to the wildtype flux distribution.[67]

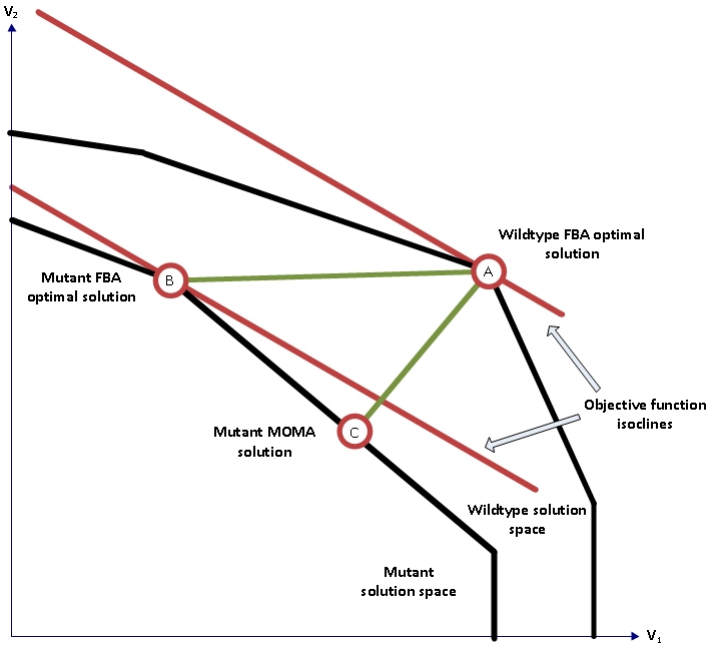

Figure 4: Graphical presentation of the MOMA principle. The point C on the edge of the feasible mutant flux space is closest to the wildtype flux profile at A, and is therefore chosen as the mutant flux prediction, even though point B gives a higher value for the objective function.

In accordance with this hypothesis, the MOMA algorithm searches for a point in the feasible flux space of the mutant strain which minimizes the euclidean distance between the flux distributions for the wildtype and mutant strains. The wild-type flux distribution can be based on experimental data or an FBA solution. If an experimentally determined flux is used, the MOMA result does not depend on a cellular objective as in regular FBA. Mathematically, the mutant flux distribution minimizing the sum

$$D(w,x) = \sqrt{\sum_{i=1}^{N} (w_i - x_i)^2} \tag{15}$$

is sought, where w and x are the wild type and mutant fluxes respectively, summing over all N reactions in the model.

The MOMA algorithm was found to give better predictions of gene essentiality compared to regular FBA as determined by mutant growth experiments.[67].

Regulatory on/off minimization (ROOM) A drawback of the MOMA approach is that large modification of single fluxes incurs a large penalty, but may be necessary for re-routing of fluxes through alternative pathways. This point is adressed by the more recent ROOM algorithm.[68] Like MOMA, ROOM does not attempt to maximize the growth rate or another conventional objective. In ROOM, the aim is to minimize not the total flux change, but rather the number of fluxes that are significantly changed. This requires the solution of a Mixed-Integer Linear Programming (MILP) problem.

The ROOM authors suggest that MOMA is appropriate for predicting transient growth rates, while ROOM and FBA is better suited for determining final growth rates following a perturbation and adapation. [68]

Dynamic FBA (DFBA): Regular FBA is used to calculate a system-wide steady state, while in reality the behavior of the system may change with time. Dynamic FBA is used to simulate such situations.

Two approaches for dynamic FBA were presented by Mahadevan et al, who applied DFBA to model the diauxic batch growth of *E. coli* on glucose and acetate, where the depletion of one substrate at a faster rate than the other leads to a change in the flux profile with time.[69] The dynamic optimization approach (DOA) involves optimizing the system behavior over a complete time course by solving one non-linear programming (NLP) problem. The static optimization approach (SOA) divides the time course into intervals, solving one linear optimization problem for each interval based on the system state at the beginning of the interval to obtain the flux values used for the whole interval. Because of the lower computational complexity, the SOA approach scales better than DOA to large networks.

Gene knockout screening (OptKnock): OptKnock is one of several computational strain optimization methods based on Flux Balance Analysis and was one of the earliest published. The OptKnock algorithm suggests gene deletions which lead to overproduction of a metabolite by coupling production of the metabolite to reactions necessary to growth.[70] The usefulness of the OptKnock method was demonstrated by applying it to production of lactic acid in *E. coli*.[71] One limitation of the OptKnock approach is that the suggested gene deletions may result in a metabolic network where the flux solution giving maximal growth rate and maximal product excretion is accompanied by solutions with equivalent growth rates but lower yields of the

desired products. If mutant strains were selected for growth rate and thus subjected to evolutionary pressure towards the maximum achievable growth rates, the mutants might evolve to their maximal growth rate without over-producing the desired product. To avoid this problem, the objective function can be modified by adding the desired product, in a process called "objective tilting".[72]

OptKnock has been used as a basis and inspiration for further development, resulting in several more recent algorithms such as OptStrain, OptGene, RobustKnock and OptForce

RobustKnock: RobustKnock adresses the problem of alternate optima in use of the OptKnock algorithm by searching for the set of gene knockouts which maximizes the *minimum* production rate of the desired metabolite.[73] In this way, over-optimistic results are avoided.

OptReg: OptReg extends Optknock by also considering over-expression and down-regulation of reactions. The regulation of genes is implemented by constraining the corresponding reactions to reaction rates significantly higher or lower than their default values.[74]

OptStrain: OptStrain extends previous strain design methods by considering both reaction additions and deletions, a database of known reactions to suggest gene "knock-ins".[75] The cited database currently appears to be unavailable.

OptGene: OptGene is a Genetic Algorithm (GA) extending the application of the OptKnock approach by using the principle of Darwinian evolution to find the global optimal solution in less computational time.[76] OptGene allows for optimization of a non-linear objective function. However, this method is not guaranteed to converge to a global optimal solution.

OptForce: The OptForce algorithm for strain design applies flux variability analysis to compare the observed flux ranges in a wildtype organism and the computed fluxes in a model of the same organism overproducing a desired metabolite.[77] A list of reactions whose flux must be either increased or decreased to reach the production target is then computed. OptForce has been applied to the overproduction of fatty acids in *E. coli*.[78]

Genetic Design through Local Search (GDSL) The GDSL algorithm uses a random search procedure to explore genetic manipulation strategies with a larger number of simultaneous gene deletions than can feasibly be evaluated in an exhaustive search. This represents a tradeoff between the conflicting algorithmic properties of low complexity and high optimality.[79]

Flux Coupling Analysis: Flux Coupling Analysis (FCA) is the study of

correlation between fluxes. The method was originally described by Burgard et al.[80], and an alternative computational approach was detailed by Larhlimi and Bockmayr.[81] Feasability-based Flux Coupling Analysis (FFCA) is a third implementation.[82]

Flux coupling is of biological interest because functionally related fluxes tend to be coupled to each other. As an example, flux coupling outperforms network distance - the minimum number of nodes that must be passed when moving from one node to another - as a metric for predicting co-regulation of genes.[83] It has been noted that flux coupling analysis is sensitive to missing reactions: two reactions that are uncoupled in a metabolic network may be identified as coupled reactions in an incomplete version of the network.[84] As it is hard to guarantee the completeness of genome-scale metabolic networks, this suggests some caution in the interpretation of FCA results. When subsystems of a network is analyzed, the opposite behavior is observed. That is, coupled reactions in a complete system may be uncoupled in the subsystem.[85]

FCA has found application in improving methods for Metabolic Flux Analysis (MFA), which is described later.[86]

High-Flux Backbones: Almaas et al described an algorithm for uncovering the "High-Flux Backbone" (HFB) of a metabolic network state, describing a connected structure of reactions.[87] In the HFB subnetwork, two metabolites are connected if the reaction that is the largest consuming flux for one of the metabolites is also the largest producing flux for the other metabolite. The structure of the HFB in a network arises from heterogeneous local organization of flux magnitudes, where each metabolite tends to have a dominant producing and consuming reaction, respectively. This was found to be the case for flux distributions obtained through FBA using the genome-scale *E. coli* metabolic network.[87] The HFB is thus a simple way to describe which reactions are important in the network.

FBA as a tool for biological discovery: FBA can be used for discovering previously undescribed reactions which necessarily are not accounted for in metabolic models. Nakahigashi et al. used comparison of growth rates resulting from FBA gene-knockout simulations and *in vivo* double gene knock-out experiments to discover new reactions in central carbon metabolism.[88] A limitation of gene-knockout simulations using FBA, pointed out in the same article, is that the effect of isozyme deletions may not be captured. If a reaction can be catalyzed by two different enzymes, deleting the gene for one of the enzymes will not have an effect for that reaction when performing FBA, while an *in vivo* deletion may result in partial or total loss of reaction

activity.

Recent advances: More recent advances include Comprehensive Polyhedra Enumeration Flux Balance Analysis (CoPE-FBA) as a method for topological characterization of the solution space in a given FBA problem [89] and a hybrid method combining the Bees Algorithm and Flux Balance Analysis (BAFBA) to avoid local minima and find optimal gene deletion sets in knockout studies.[90]

2.2.4 Software

Flux Balance Analysis and related methods are facilitated by general computing software such as MATLAB and specialized software packages. The COnstraint Based Reconstruction and Analysis (COBRA) Toolbox is a popular plugin for MATLAB containing functions enabling easy calculations using FBA and other methods.[91] It can use metabolic models supplied in the SBML format among others. A list of software for constraint-based modelling is shown in Table 3. Computational tools in systems biology, not limited to metabolic networks, have been reviewed by Copeland et al. [92] while mathematical optimization applications in metabolic networks have been reviewed by Zomorrodi et al.[93]

Table 3: Software for constraint-based modelling

Program	Main features
COBRA	FBA, FVA, gene knockout. MOMA. Runs under MATLAB.
FAME	FBA, FVA and network visualization. Web interface.
CellNetAnalyser	Metabolic and signalling network analysis. Network visualization. Runs under MATLAB.
SBRT	Includes 35 methods for stoichiometric analysis.
OptFlux	Simulation of mutant strains, optimization for metabolic engiineering.
FASIMU	Command line interface, batch processing of simulations.
Acorn	Grid computing system for constraint-based simulations
CycSim	In silico knockout experiments and comparison with experimental results. Web interface.
SurreyFBA	Network map visualization, analysis of minimal substrate and product sets

2.3 Metabolic Flux Analysis

2.3.1 General concepts

Metabolic Flux Analysis (MFA), also called metabolomics or fluxomics, is the study and determination of metabolic fluxes *in vivo*.[94] Not all authors adhere to this definition; in some cases Flux Balance Analysis and related computational methods have been included under this term. For the purposes of this report, MFA refers exclusively to the *experimental* study of metabolic fluxes. MFA has significant applications in metabolic engineering, the genetic modification of organisms to enable or increase the production of desired metabolites.[95]

With few exceptions, it is currently infeasible to measure a significant number of *in vivo* reaction rates directly. In MFA, mass balance equations are therefore used together with experimental measurements allowing the calculation of a limited number of fluxes or flux ratios. This data is then used to calculate the remaining fluxes in a network of reactions. This approach has been limited largely to steady state scenarios, as in FBA, but dynamic MFA (DMFA) has recently been introduced as a framework allowing the determination of metabolic flues at non-steady state.[96]. MFA is based on both experimental measurements and computational routines for "deciphering" the actual fluxes from the measured data, as a mathematical model relates the experimental data and the fluxes to be calculated.[97] As with FBA, specialized software is available for the computational work.[98] Most models have been limited to describing fluxes in the central carbon metabolism, covering about 25 to 50 reactions, due to the computational challenges involved in flux mapping.[86] One of the largest models to date covered 350 fluxes and 184 metabolites in *E. coli*.[99]

If a limited number of fluxes is known, the complete flux vector and can be separated to known and unknown fluxes, with a corresponding stoichiometric matrix for each.[100] At metabolic steady state, the following equation then holds:

$$-S_m v_m = S_c v_c \tag{16}$$

Here, m and c refers to known (measured) and unknown (computed) fluxes, respectively. To determine all the fluxes in a network with N fluxes and M metabolites, at least $N - M$ fluxes must be known. Additionally, the stoichiometric matrix of the unknown fluxes must have the mathematical property of *full rank*. If this is the case, the system is called *observable*.[100].

If more fluxes have been calculated than is necessary to determine all the rest of the fluxes, the system is *overdetermined*. Then, in addition to calculating the remaining fluxes, the excess information can be used to increase the accuracy in the estimates of the measured and calculated fluxes, to check for internal consistency in the data set and/or identify the measured fluxes most likely to be in error.[94] If fewer fluxes are measured, the system is underdetermined and the remaining fluxes can be calculated only by applying further constraints or by applying an optimization principle, as in FBA.

Sensitivity to measurement errors: A basic sensitivity analysis may be useful in assessing the trustworthiness of calculated flux values. Ideally, the mathematical system should be *well posed* and the stoichiometric matrix *well conditioned*.[100] Round-off errors during flux calculations may be amplified if the matrix is ill-conditioned. A measure of this sensitivity is the *condition number* of the stoichiometric matrix. The condition number is always larger than 1, and a large condition number means that that the matrix is ill-conditioned. As a rule, measurements should be carried out with the same number of significant digits as the number of digits in the condition number.[100] Based on the current achievable precision in measurements from fermentation experiments, the condition number should be between 1 and 100.

The conditioning number may be useful in an initial consideration of model sensitivity, but does not give any concrete information. The following equation can be used to calculate the sensitivity of each calculated reaction with respect to a measured reaction:

$$\frac{\partial v_c}{\partial v_m} = -(S_c^T)^{-1} S_m^T \tag{17}$$

Here, element (j, i) of the resulting matrix gives the sensitivity of calculated flux j with respect to measured flux i.[100]

Statistical analysis of results: A limitation of much previously published flux data from MFA is a lack of reliable uncertainty estimates. Antoniewicz, Kelleher and Stephanopoulos pointed out the lack of rigorous statistical analysis and published an algorithm for determining confidence interval of the calculated fluxes. The authors pointed out common misconceptions about issues relating to uncertainty estimates in Metabolic Flux Analysis, which in its then current state was described as "a black box whose inner workings are hard to decipher".[101] As an example, it is often assumed that the errors in the measured fluxes are independent of each other, which simplifies the statistical treatment, but this may assumption may not hold, as the fluxes

are not measured directly but calculated from several raw measurements. If several fluxes are calculated using the same raw measurement, this will introduce a correlation in their errors.

2.3.2 Stoichiometric MFA

Early approaches to MFA used primarily the stoichiometry of the biochemical reaction network and externally measurable reaction rates (nutrient uptake and product secretion rates) in order to determine the flux distribution. A number of assumptions about the operation of the network, especially relating to energy metabolism, had to be made and several of these have later been shown to be invalid. Furthermore, the number of externally measurable fluxes, and thus the number of constrains which could be obtained, is small. A number of other limitations also apply - parallel metabolic pathways and bidirectional reactions are among the network features in whose presence stoichiometric MFA fails. For this reason, stoichiometric MFA has largely been superceded by ^{13}C-based methods.[95]

Note that stoichiometric MFA is similar to standard optimization-based flux balance analysis (FBA) in that no experimentally determined constraints are made on the internal fluxes from the outset, while experimental data on externally measured rates may also be used as constraints in performing regular FBA. However, models used in FBA are typically larger than those used in stoichiometric MFA, and thus no longer determined by the same number of fluxes. Stoichiometric MFA can be regarded as a use of flux balance analysis limited to small networks, while the use of an objective becomes necessary to apply the flux balance approach to larger networks in the absence of further constraints. FBA can in theory be performed with constraints derived from ^{13}C-MFA experiments as described below. However, the purpose of MFA is generally to obtain experimental data which constrain the system in such a way as to completely define the flux distribution, avoiding the need to apply a hypothetical objective function.

2.3.3 ^{13}C-MFA

^{13}C-based labelling of metabolites is the current main approach to *in vivo* flux determination. Carbon-labelling experiments (CLE) form the experimental basis of ^{13}C-MFA. In a CLE, an organism is grown in a minimal, chemically defined growth medium containing a single carbon source, typically glucose. The labelled carbon source contains one or more atoms of ^{13}C. ^{13}C is a stable isotope of carbon having one more neutron than ^{12}C, the most

abundant carbon isotope. Labelled and unlabelled molecules are assumed to be chemically identical, but considered different *isotopomers*. The assumption of chemical identity is important for carbon labelling experiments, as it implies that reaction rates are unaffected by the isotopomer of a given metabolite. Isotopomer is a term combining the terms isotope and isomer, the latter denoting different configurations of the same molecule. A metabolite with n carbons can be labelled or unlabelled at each carbon position, and thus can exist as 2^n different isotopomers when a single labelling isotope is used.[95] During a CLE, each metabolite will display an isotopomer distribution characterised by the fraction each isotopomer makes up of the total amount of that metabolite.

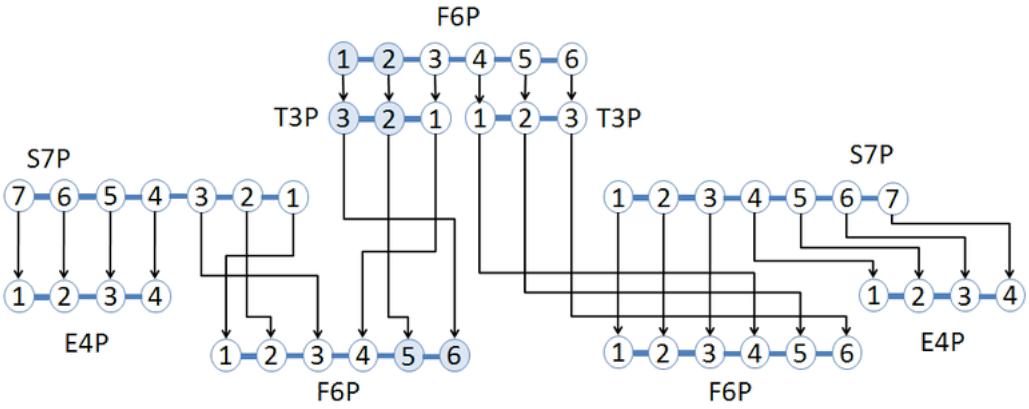

Figure 5: Illustration of carbon labelling patterns exploited in ^{13}C-MFA. From Wikimedia Commons.

Carbon labelling experiments: Carbon labelling experiments present a number of challenges, perhaps explaining the relative small number of studies published in fluxomics compared to the other "omics" fields.[102] Primarily, the metabolic system must be kept at steady state throughout the experiment to allow the isotopomer distribution to also reach steady state. The steady state isotopomer distribution for each metabolite depends both on the ratios of the fluxes and the isotopomer distribution in the carbon source. As a fully labelled or unlabelled carbon source would yield no information, a mixture of labelled and unlabelled carbon substrate is used. The decision of which mixture to use must take into account both the cost of the ^{13}C-labelled carbon source (^{13}C-labelled glucose costs in excess of 100$ per gram) and the information obtainable given the planned measurements. The process of determining whether information about the unknown fluxes will be contained in the measured isotope ratios is called identifiability analysis.[95]

After a time sufficient for the isotopomer distribution to reach equilibrium,

the biomass is analyzed to determine the isotope ratios. Nuclear Magnetic Resonance (NMR) or Mass Spectroscopy (MS) is used for this analysis. An isotopomer balancing model is then used which predicts the steady state isotopomer distribution as a function of the flux distribution. To estimate the flux distribution, a non-linear optimization algorithm is employed which simulates the experiment. Starting from an initial guess, various flux distributions are tried, and the flux distribution which minimizes the difference between the predicted and observed isotopomer distribution is sought.[95] Due to the large number of possible labelling patterns, the isotopomer modelling process is mathematically complex - the evaluation of CLEs has been called one of the most complicated mathematical methods ever applied to biological systems.[95] As an alternative to the comprehensive isotopomer balancing approach, the isotopomer distribution data on a small number of proteinogenic amino acids can be used directly. In this way a smaller number of local flux ratios of central carbon metabolism reactions linked to the production of amino acids can be determined. These flux ratios can then be used as constraints to solve the complete system of fluxes. Both methods can be applied to the same experimental data.[103]

In both methods outline above, the estimated flux distribution is dependent on a model of the possible reactions in the system. This is an important point, as it raises some possible concerns in the use of MFA-derived flux values as a reference for flux distributions predicted using a different model.

A major, often unstated, assumption of MFA is that the flux distributions at steady state are equal for all cells in the reactor system.

2.4 Network optimization

The solution of a standard Flux Balance Analysis problem is found by the application of *linear programming*. A linear programming problem is an optimization problem which can be expressed on the form:

$$\begin{aligned} &\text{maximize } \mathrm{c}^T x \\ &\text{subject to } \mathrm{Ax} \leqslant \mathrm{b} \\ &\text{and} \qquad \mathrm{x} \geqslant 0 \end{aligned}$$

This is the *standard form* of linear programming problems. Here, x represents the vector of variables to be determined, c and b are objective and constraints coefficient vectors, respectively, and A is a matrix of constraint coefficients. In FBA, the vector x is the *flux vector*, denoted v, A is the stoichiometric

matrix S, and b is set equal to zero, constraining the system to steady state. The expression $c^T v$ is the *objective function* whose value should be optimized.

The constrained solution space of a metabolic model is often called the "flux cone". The flux cone has the mathematical property of being *convex*. The consequence of this is that the optimal solution(s) to an FBA problem will always be found at the edge of the flux cone, and application of linear programming is guaranteed to identify an optimal solution, if one exists. Several algorithms are available for solving linear programming problems, but the *simplex* algorithm is typically used. The use of the simplex algorithm requires that the FBA problem is converted to standard form. This conversion, entailing the decomposition of each reversible reactions into two separate reactions, is handled automatically by FBA software such as the COBRA Toolbox.

The time needed for computing the solution to an FBA problem with linear programming is short, on the order of one second even for genome-scale models. This is especially an advantage when evaluating combinations of simultaneous gene-knockouts, a major application of FBA, as the number of combinations grows rapidly with the number of simultaneous knock-outs.

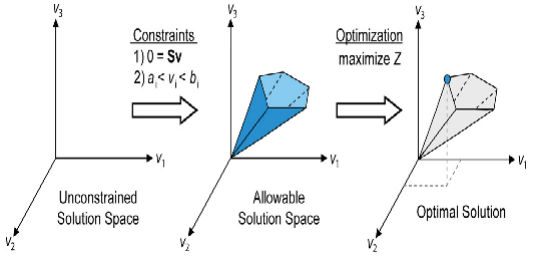

Figure 6: A visual representation of the "flux cone" explored in FBA. The optimal solution to a standard FBA problem is found at one edge of the solution space and the solution is found through the application of linear programming. Figure reproduced from [43].

Non-linear objective functions requiring the use of other optimization methods have also been proposed for use in FBA, but for these it is generally not possible to guarantee that a global optimal solution will be found.

Cellular objectives: To have predictive value, a model should ideally reproduce experimentally acquired data as closely as possible. To achieve this the parameters used in the model are selected to match the physical realities and environmental circumstances. These parameters will generally be based on abstraction or simplification of the actual process to be modelled. The objective function specified in Flux Balance Analysis can be considered one such simplification. In using linear programming, the objective function vector assigns an objective function coefficient to each reaction:

$$c = [c_1, c_2, ..., c_n] \tag{18}$$

The value of the objection function is then the sum of each reaction multiplied by the objective coefficient of that reaction. Clearly, the cell itself is not directly evaluating the value of any objective function and directing its biochemical reactions to maximize it. Rather, it is our assumption that the flux states maximizing an appropriate objective function present an advantageous mode of operation for the cell, and that cells which fail to achieve these states will be selected against in the course of evolution. As evolutionary pressures depend on environmental circumstances, different objective functions may be appropriate for different models. Selecting and evaluating the effect of the objective functions thus becomes important for ensuring validity of the modelling process.

The objective function optimization approach is simple and convenient, but should not be used uncritically. Not all cells show optimal metabolic behaviour.[104] The biomass objective function is most appropriate when the organism is under evolutionary pressure to reproduce quickly. In such cases adaptive evolution may lead to growth rates close to the maximum predicted by FBA. [105] Other objective functions may be suited for simulation of growth in some environments but not other [44] - data-mining algorithms may be used to attempt to identify suitable cellular objectives.[106] The important question of "what is the optimal operation of metabolic networks?" was reviewed by Nielsen. [107]

Implementation of a biological objective as an objective function requires both identification of the objective, and a precise description of it.[50] One issue with using a pre-determined composition for the biomass reaction is that the composition may vary with the environmental circumstances and growth conditions - computational and experimental fluxes may be better reconciled if appropriate modifications are applied.[108] In any case, if objective functions are to be central in metabolic analysis in the future, the ability to select and evaluate them will be important.

The biomass objective function: The starting point for the formulation of the biomass reaction is the empirically determined values for the macromolecular content of the cell in question, together with the amount of the different building blocks making up the different classes of molecules.[63] For example, the total protein content of a type of cell can be measured, and the average amino acid composition of the proteins determined. Similar procedures can be applied to the other major classes of molecules making up the

cell. This information is used to describe the amount of the various metabolites required for producing biomass. Energy requirements for the production of said metabolites can also be taken into account, when known.

The biomass objective vector is generally perpendicular to one of the surfaces limiting the solution space of the FBA problem and therefore biomass-maximizing flux states are most often degenerate.[109] Calculations on a network simulating growth of *E. coli* in minimal glucose medium found a 26-dimensional space of growth-maximizing solutions.[109] There are several issues inherent in the use of a static biomass production/pseudo growth reaction as commonly employed in FBA, related to the fact that the biomass composition varies and that biomass precursors metabolites typically are not included in the biomass model reaction. Metabolite dilution flux balance analysis (MD-FBA) has been introduced as a method to address the second issue.[110]

Lately, strategies of simultaneous optimization of multiple objectives have been considered. Multi-objective optimization may be more useful in exploring the potential of an organism to perform a specific task.[46] They may also be useful in descriptions of mutualism between two organisms where Flux Balance Analysis has been extended to modelling of microbial communities. [46][111].

Quadratic minimization of distance: Flux distributions may be compared by using the *euclidean distance* as a measurement of their deviation from each other. The euclidean distance between two vectors x and y is defined by the following equation:

$$D(x,y) = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2} \tag{19}$$

Several flux distributions may give the same value for the same objective function in an FBA problem, while the euclidean distance to an experimentally determined flux distribution may differ. It may therefore be of interest to perform optimization not only with respect to the objective function value, but also the distance to an experimentally determined flux distribution. This optimization can be posed as a quadratic programming (QP) problem, with the objective being minimization of the sum

$$\sum_i x_i^2 - 2y_i x_i + y_i^2 \tag{20}$$

Using vector and matrix notation, this sum can be written as:

$$x^T F x + C x + \alpha \tag{21}$$

here, F is a coefficient matrix, $C = -2Y$ is a vector and α is a number equal to $\sum_i y_i^2$. F is a diagonal matrix, with $F_{i,i} = 1$, all other elements being equal to zero. This applies to minimization of the unweighed sum. By multiplying each element in the matrix F and the vector C with a weighing factor, it is possible to minimize a weighed sum of the individual flux differences or exclude some fluxes from the analysis.

3 Evaluating objective functions

The problem of choosing an appropriate objective function contains at least two questions: What objective function gives the best solution? And: What defines the "best" solution? Thus we must ask: How can we evaluate objective functions, and how should objective functions be rated? What are the most relevant criteria that can be used? Some possible considerations are listed below:

- If seeking solutions optimizing an objective function, can we trust that the cells investigated in experiments are behaving optimally? How much do we have to relax the optimality requirement to recreate the experimental flux?
- Sensitivity to objective function coefficients. This is especially relevant for the biomass objective function. The biomass composition changes with growth rate and growth conditions. How does changing the biomass and/or objective function coefficients affect results?
- Robustness analysis/FVA: How much are fluxes allowed to change while maintaining the optimal solution for a given objective function? How much does the objective value change if the fluxes change?

The above items are only a few of many possible angles from which the topic of objective functions in metabolic analysis can be approached. Some notable publications in the area are briefly described below:

ObjFind: Burgard and Maranas introduced an optimization-based framework (ObjFind) for inferring and testing hypothesized objective functions.[112] The method is based on calculating "coefficients of importance" (CoI) for

every considered reaction, which measure how much the flux of each reaction contributes to the objective function which would minimize the distance between computed fluxes and experimental fluxes. The CoIs are empirical estimates of the objective function coefficients, and scaled so that they sum to 1 and a high CoI for a given flux implies that the experimental data is consistent with an hypothesis of maximization of that flux as part of the objective. The vector of all CoI values thus suggests an objective function for the system based only on the model and experimental data, without any a priori assumptions about biological relevance.

Mathematically, ObjFind attempts to find a set of coefficients c_j which is consistent with the experimental fluxes $v^{\exp}$ being an optimal solution of the maximization of the sum

$$\sum_j c_j v_j \tag{22}$$

That is, if the values of c_j are used to define the objective function, an optimal solution set of values c_j^* exists such that the distance:

$$\sum_j (v_j^* - v_{\exp})^2 \tag{23}$$

is minimized while adhering to the FBA constraints.

Using a model of *E. coli* central carbon metabolism with 68 reactions and 48 metabolites together with experimental data from growth of *E. coli* on glucose, Burgard and Maranas found the maximal CoI values for the biomass production reaction to be 0.58 and 0.68 for aerobic and anaerobic growth respectively. This suggests that biomass production is important in determining the observed flux distributions. Burgard and Maranas noted that while in their study the range of allowable CoI values for a given flux distribution was narrow, many flux distributions are consistent with a set of CoI values. In that case, the identified objective function has several optimal solutions. As ObjFind requires solving a bi-level optimization problem, the algorithm may be computationally expensive, which becomes a larger concern when working with genome-scale models.

Bayesian-based selection: Knorr, Jain and Srivasta introduced a framework based on bayesian probability for selecting the most probable objective function among a set of candidates by comparing their predictions with experimental flux values.[113] For a genome-scale model of *E. coli* growing on

succinate, the procedure suggested minimization of redox potential production as the best objective among five candidates including maximation of growth rate. A limitation of this approach is that it only attempts to answer which of the objective functions chosen which appears most relevant, and does not give an independent evaluation of any one objective function.

Schuetz. et al - "Systematic evaluation of objective functions for predicting intracellular fluxes in Escherichia coli:"

One of the most cited articles dealing with evaluation of objective functions was published by Schuetz, Kuepfer and Sauer in 2007.[44] In their article, the results of applying 11 objective functions in combination with 8 adjustable constraints to a model of *E. coli* core carbon metabolism were evaluated. The model contained 98 reactions and 60 metabolites. Experimental data for six conditions were used when comparing the results obtained applying different objective functions and constraints. Table 4 shows a list of the objective functions used in the study, while Figure 7 shows the main reactions included in the model.

To measure the overall agreement between a set of experimental and computed split ratios, the euclidean distance between the computed and experimental solutions was computed, weighed by the experimental uncertainty of each split ratio. This measure was termed *predictive fidelity*, and denoted D^S. Somewhat counter-intuitively, a low value for the predictive fidelity indicates good agreement between the FBA results and experiments, contrasting with the common usage of the term "high fidelity".

The predictive fidelity was calculated based on computed *split ratios*. Ten split ratios were defined, based on a calculability analysis showing that those ten split ratios would define the system of fluxes completely. For the cases where the FBA problem had multiple optimal solutions, a range of values for D^S was determined. The minimal and maximal predictive fidelity for each scenario was found through an optimization algorithm described by the following equations:

Table 4: List of objective functions considered in the study by Schuetz et al.

Objective function	Definition
Maximum biomass yield	$\max \frac{v_{biomass}}{v_{glu\cos e}}$
Maximum ATP yield	$\max \frac{v_{ATP}}{v_{glu\cos e}}$
Minimization of overall flux	$\min \sum_{i=1}^{n} v_i^2$
Maximal ATP yield per flux unit	$\max \frac{v_{ATP}}{\sum_{i=1}^{n} v_i^2}$
Minimum glucose consumption	$\min \frac{v_{glu\cos e}}{v_{biomass}}$
Minimum number of reaction steps	$\min \sum_{i=1}^{n} y_i^2,\ y_i \in \{0,1\}$
Maximal ATP yield per reaction step	$\max \frac{v_{ATP}}{\sum_{i=1}^{n} y_i^2},\ y_i \in \{0,1\}$
Minimal redox potential production	$\min \frac{\sum^{n} v_{NADH}}{v_{glu\cos e}}$
Minimal ATP production	$\min \frac{\sum^{n} v_{ATP}}{v_{glu\cos e}}$
Maximal ATP production	$\max \frac{\sum^{n} v_{ATP}}{v_{glu\cos e}}$
Maximal biomass per flux unit	$\max \frac{v_{Biomass}}{\sum_{i=1}^{n} v_i^2}$

$$
\begin{aligned}
&\max / \min\ \ D^S \\
&\text{subject to:} \\
&S \cdot \overrightarrow{v} = 0 \\
&v_j^{lb} \leqslant v_j \leqslant v_j^{ub} \\
&D^S = \varepsilon^T W \varepsilon \\
&\varepsilon_i = R_i^{comp} - R_i^{\exp} \\
&W_{i,i} = \frac{1}{\sigma_i^{\exp}} \left(\sum_i \frac{1}{\sigma_i^{\exp}} \right)^{-1} \\
&R_i = f(v)
\end{aligned}
$$

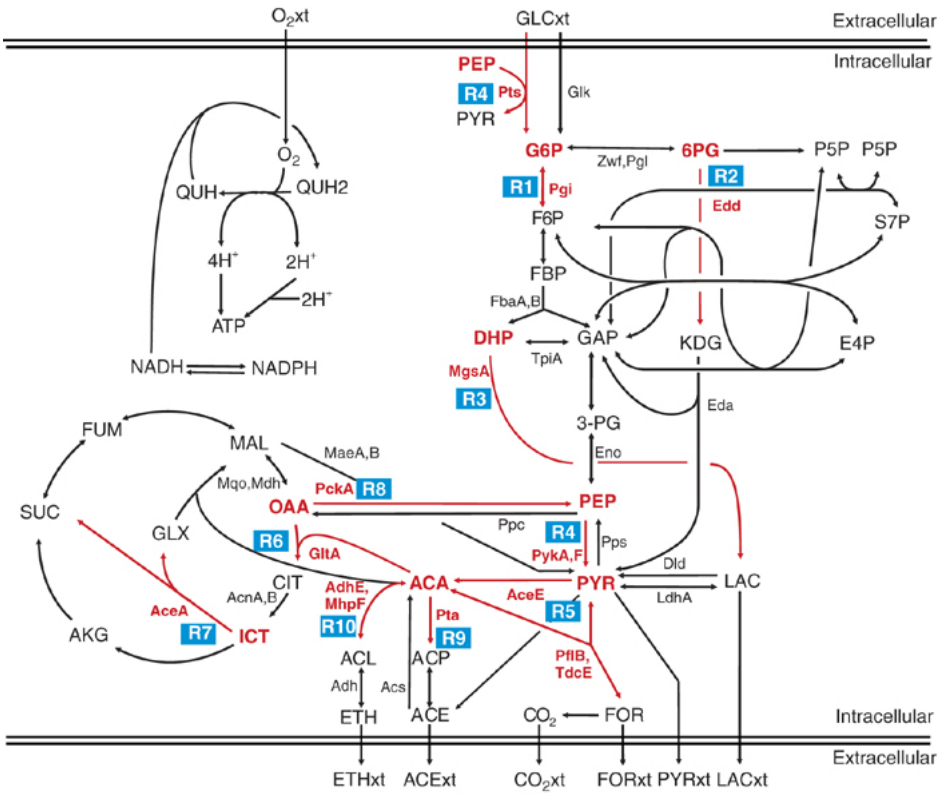

Figure 7: The central carbon metabolism of *E. coli*, as modelled in the study by Schuetz et al. Not all reactions in the network are shown in the figure. The labels R1 to R10 show the placement in the network of the split ratios used in their analysis. Figure reproduced from [44].

Here, ε is the difference between the computed and experimentally determined split ratios, and W is a weighing matrix incorporating the experimental uncertainties represented by $\sigma_i^{\exp}$. In the last equation, R_i is split ratio no. i. The reactions being used in computing the split ratios were as shown in Table 5. However, this table may be misleading. The definition of the split ratios as explained in the article is the sum of one or more fluxes consuming a metabolite divided by the sum of the fluxes producing that metabolite. However, many reactions are reversible and can be either producing or consuming. This is not reflected in Table 5. For example, the split ratio R1 is defined there as:

$$R_1 = \text{pgi}/(\text{Glk}+\text{Pts}+\text{Zwf}+\text{Pgi}) \tag{24}$$

Here, *pgi* is a reaction consuming the metabolite in question, Glucose 6-phosphate (G6P), while *Pts* and *Glk* are producing reactions, and *zwf* is a reversible reaction either consuming or producing G6P. However, when

the *zwf* flux is positive (consuming G6P), it should not be included in the denominator. Neither should pgi, as it is also a consuming reaction, and the calculation simplifies to $R_1 = \text{pgi}/(\text{glk+pts})$. In the model by Schuetz et al., the zwf reaction is directly linked to the pgl reaction, which is irreversible. In practice, *zwf* is therefore also irreversible. In short, a reaction should only be present in the denominator of the split ratio if it is a producing reaction for that metabolite (in MATLAB this can be achieved by using the function *subplus*, which returns the value of a number if it is positive, and else zero).

It is possible to question some of the choices made in the definition and use of the split ratios. Consider for example split ratio R3. It is described as the flux through reaction *mgsA* divided by the flux through reactions *FbaA/FbaB* and *TpiA*. However, when considering the experimental data (shown as absolute fluxes in Table B.1) from which the reference experimental split ratios are calculated, it is worth noting that there is no reaction corresponding neither directly to reaction *TpiA*, nor to reaction *FbaA/FbaB*. Rather, the two are combined into experimental reaction #6. Additionally, a corresponding reaction for *MgsA* is not included in the experimental data set at all. Rather, it is assigned as experimental reaction #27 by Schuetz et al., with a flux relative to the glucose uptake flux (experimental reaction #1) equal to 0±10%.

Put one way, the value of split ratio R3 being zero is not a result from the reference experiment, rather it was an *assumption* (or simplification) of the experiment. It then becomes an interesting question if a split ratio deviating from that assumption should be considered contrary to the experimental result. This is strictly speaking not an issue with the split ratios themselves, but rather with the Metabolic Flux Analysis that produced the data. However, the use of split ratios does somewhat hide the facts of the case, making the calculations less transparent.

The use of split ratios also is potentially troublesome with respect to optimization. Schuetz et al. used the general non-linear solver *fmincon* in MATLAB to minimize the euclidean distance between the experimental and computed split ratio vectors, but this required generation of random starting points and multiple runs to check for convergence, as the approach is not guaranteed to identify the global minimum point. For larger models, solving time may quickly become prohibitive.

As seen, there are several potential issues both regarding the definition and optimization of split ratios. However, the euclidean distance measure is easy to interpret, and can be minimized using quadratic programming if calculated on basis of the raw fluxes. For these reasons, it was decided to focus on the

Table 5: Definition and explanation of split ratios used in the study by Schuetz et al. Reproduced from [44].

Definition	Explanation
$R_1 = \frac{v_{Pgi}}{v_{Glk}+v_{Pts}+v_{Zwf}+v_{Pgi}}$	Flux to glycolysis
$R_2 = \frac{v_{Edd}}{v_{Pgl}}$	Flux to Entner-Doudoroff pathway
$R_3 = \frac{v_{MgsA}}{v_{FbaA}+v_{FbaB}+v_{TpiA}}$	Flux to methylglyoxal pathway
$R_4 = \frac{v_{PykA}+v_{PykF}+v_{Pts}}{v_{Eno}+v_{Pps}+v_{PckA}}$	Flux of PEP to pyruvate
$R_5 = \frac{v_{AcE}+v_{flB}+v_{TdcE}}{v_{PykA}+v_{PykF}+v_{MaeB}+v_{Dld}+v_{LdhA}+v_{ldhA}+v_{Eda}+v_{Pts}+v_{PflB}+v_{TdcE}}$	Flux of pyruvate to acetyl-CoA
$R_6 = \frac{v_{GltA}+v_{PrpC}}{v_{AcE}+v_{PflB}+v_{TdcE}+v_{Acs}+v_{AdhE}+v_{MhpF}}$	Flux to citric acid cycle
$R_7 = \frac{v_{AceA}}{v_{AcnA}+v_{AcnB}}$	Flux to glyxocylate shunt
$R_8 = \frac{v_{PckA}}{v_{Mdh}+v_{Mqo}+v_{Ppc}}$	Flux of oxcaloacetate to PEP
$R_9 = \frac{v_{Pta}}{v_{AcE}+v_{PflB}+v_{TdcE}+v_{Acs}+v_{AdhE}+v_{MhpF}}$	Acetate secretion
$R_{10} = \frac{v_{AdhE}+v_{MhpF}}{v_{AcE}+v_{PflB}+v_{TdcE}+v_{Acs}+v_{AdhE}+v_{MhpF}}$	Ethanol secretion

euclidean distance between raw flux vectors as the main metric in this project.

4 Methods

4.1 Model setup

Three metabolic models were chosen for use in the study. The metabolic model of *E. coli* central carbon metabolism used by Schuetz et al. and the *E. coli* genome-scale metabolic model iJO1366 was downloaded from the supplementary materials sections of the online versions of the respective papers.[44][19] The *E. coli* core model was downloaded from `http://systemsbiology.ucsd.edu/Downloads/EcoliCore`.

The .xml file containing the model by Schuetz et al. was edited by by changing metabolite and reaction identifiers to the format expected by the COBRA toolbox, to allow the model to be processed by the program. In addition, identical duplicate reactions were removed to simplify the computational work and reduce the potential for flux loops in the solutions. In the following, the edited version of the model used in the calculations will be referred to as the "Schuetz Revised" or SCHUETZR model. The *E. coli* core model was edited to add reactions present in the SCHUETZR model and the experimental data set but missing in that model. In the following, the edited version of the *E. coli* core model will be referred to as the *E. coli* Core Expanded or "ECME" model. A full list of changes made to the two models

is shown in Appendix A. In addition to the above changes, for those experimental reactions which had several corresponding model reactions, token metabolites and reactions were added so that the flux through each 'token' reaction would give the sum of the model reactions. This was done to allow those sums to be used directly during optimization. The addition of token metabolites and reactions does not otherwise affect the behaviour of the model. The iJO1366 model was used unchanged except for the addition of token reactions. Being so changed it is referred to as iJO1366b. iJO1366 contains two biomass reactions using either a "wildtype" or "core" biomass composition. The "core" biomass reaction is the default defined in the model file and was used for all calculations.

Pairing of model and experimental reactions: As there was not always a one to one correspondence between the available model and experimental reactions, all three models were reviewed to decide which model reactions should be considered equivalent to a given experimental reaction. A list of all 28 experimental reactions and their corresponding reactions in the three models is shown in Appendix A.

4.2 Experimental data

The experimental data, also used by Schuetz et al.[44], was originally published by Perrenoud and Sauer.[114] The experimental data was collected for exponential growth of *E. coli* strain BW25113, a derivate of *E. coli* K-12, on glucose. The intracellular fluxes were determined by ^{13}C-based MFA using a stoichiometric model published by Fischer et al.[115] The model consisted of a reaction network with 25 fluxes to be determined and 21 metabolite balances. The latter included three experimentally determined rates - glucose uptake, acetate production and biomass production. The resulting underdetermined system having four degrees of freedom was solved by using seven experimentally determined flux ratios as constraints. The biomass composition used was based on data published by Emmerling et al.[116]

Calculation of absolute experimental fluxes: The study by Schuetz et al. [44] used ratios between fluxes when comparing experimental and computational results, and the experimental fluxes were reported relative to the glucose uptake flux. However, when using quadratic programming it is necessary to use absolute flux values. As the experimental data cited by Schuetz et al. includes the growth rate, and biomass production as a experimental flux, it is possible to scale the other fluxes using these values in the following equation.

$$v_{abs} = \frac{v_{rel} \times \mu}{v_{biomass}} \tag{25}$$

The coefficients of the biomass reaction are scaled so that a flux of one unit through the biomass reaction equals the growth rate in h^{-1} when the fluxes are expressed in units of mmol/gDW·h.[43] From the set of relative fluxes, including the biomass flux, and the reported growth rate, the values of all the fluxes in mmol/gDW·h can then be calculated. However, the data as published by Perrenoud and Sauer is given as absolute fluxes, and so using the data as presented by Schuetz et al. would introduce a needless decrease in accuracy. The data for anaerobic batch culture used by Schuetz et al. is given with reference to the same article, but only the data for aerobic batch culture is available in the supplementary information for that article. Therefore, the results presented are limited to analysis using the data for aerobic batch culture.

4.3 Model analysis

MATLAB workflow: To facilitate the analytical procedures, a collection of MATLAB functions and scripts was produced. The workflow of the main functions utilized is shown in Figure 8.

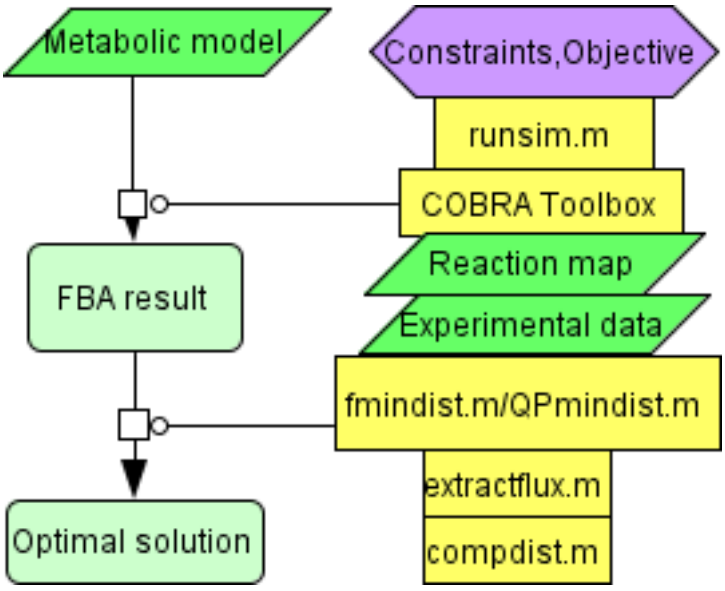

Figure 8: Graphical presentation of the MATLAB workflow of model analysis.

The function runsim.m was created to perform rapid analysis using various combinations of models, objectives and constraints by specifying these through the input parameters. A list of the MATLAB functions used and a closer description of their operation is shown in Appendix C. All computations were carried out using MATLAB R2012a and COBRA Toolbox v. 2.04 running under Windows 7 on a 64-bit Intel Core 2.1 Ghz processor. The necessary software is detailed in Appendix C.

Optimization solvers: Initial solutions to FBA problems were computed using the COBRA Toolbox called through *runsim*. For minimization of distance problems, the MATLAB function *fmincon* and the Gurobi 5 optimization solver (used through MATLAB) were used. When *fmincon* was used, the initial FBA solution was used as the starting point in all cases. Unless otherwise noted, solutions of quadratic minimization of distance problems were produced using Gurobi.

Objective functions: Four linear objectives are currently supported in *runsim*, and have been used in the analyses presented under Results and Discussion. These are maximization of biomass flux (`max BM`), maximization of ATP producing fluxes (`max ATP`), minimization of glucose uptake (`min glucose`) and minimization of ATP producing fluxes (`min ATP`). Adenosine triphosphate (ATP) is a high-energy molecule vital to cell growth and maintenance. Maximization of ATP producing fluxes would imply high availability of energy for cellular processes including growth, while minimization of ATP producing fluxes suggest an energy-conserving mode. Note that when glucose is the limiting nutrient, maximization of biomass flux (growth rate) is equivalent to maximization of biomass yield. Likewise, minimization of glucose uptake is equivalent to maximization of biomass yield when the biomass production is fixed. The definitions of the objectives are given in Appendix A.

Model constraints: The model constraints implemented in *runsim* are the same as used by Schuetz et al.[44] The constraints P/O ratio = 1,glucose/O_2 uptake = 3/2 and global reaction bounds (All reactions $< 200\%$ of glucose uptake rate) are supported only for the SCHUETZR model.

Weighing of fluxes: All distances presented and all optimizations use the unweighed fluxes. The option to use fluxes weighted by their experimental uncertainty is implemented in *runsim* but untested.

5 Results and discussion

Software: Much of the project time was spent developing the MATLAB framework for model analysis based around the top-level function *runsim*. It was a goal that the framework should be extendable to analysis using other models, objectives and constraints than used here. To facilitate this, the programs should load all problem-specific data - specifying the model(s), objectives, constraints etc. - from external files. In accordance with this principle, the experimental data and reaction maps relating the model and experimental reactions are loaded from external .mat files. Another design principle was modularity. The different operations being carried out through *runsim* are divided between separate function files, most of which share a common input format and can be used alone. This allows parameters to be easily passed from one function to the next, and makes it easy to switch between using one function as a stand-alone or running a complete analysis through *runsim*. There is still a fair amount of hard-coded relations and model-specific operations in the code, but with the basic framework and logical structure in place a moderate effort should be sufficient to achieve the generalization of the programs to handle any model.

Definitions: In this section, the optimality of identified solutions are considered both with respect to their objective function values and their fit to experimental data. A definition of terms is thus needed to avoid confusion about the different references to optimality. In the following, "objective-optimal" will refer to optimality of the solution with respect to the objective function while a "distance-optimal" solution is a solution with minimal distance (however defined) to the experimental data.

The global distance-optimal solution in the solution space of an FBA problem must necessarily be equally or more distance-optimal than the maximally distance-optimal solution among the objective-optimal subset of solutions. Of special interest are those objective-optimal solutions which are distance-optimal within the set of objective-optimal solutions. As any objective is unlikely to describe or capture the metabolic behaviour of any cell perfectly, it is reasonable to expect the distance-optimal solution to be non-optimal with respect to the objective. For this reason, the distance-optimal solutions achievable when the requirement for optimality is relaxed are also considered.

Performance of *fmincon* and Gurobi solvers: The MATLAB function *fmincon* is a general non-linear optimization solver function which can use several different optimization algorithms. *fmincon* attempts find the input variables that minimizes the value of a specified function. However, it can not be guaranteed that *fmincon* will find the global optimal solution. An

initial starting point must be supplied, and the result may depend on this starting point. Using the initially computed FBA result as a starting point for *fmincon*, *fmincon* and Gurobi generally produced equivalent solutions. However, even allowing a run-time of several hours, *fmincon* was not able to compute distance-optimal solutions for the iJO1366 genome-scale model. In comparison, Gurobi provided solutions to most quadratic optimization problems with a run-time of a few seconds.

Feasibility analysis: To determine if the flux distribution described by the experimental data was achievable by each model in their unconstrained states, a flux vector minimizing the distance to the experimental data was sought, not considering any objective and subject only to the constraints of the model. A non-zero minimal distance would indicate that the experimental flux state was outside the solution space of the model. Optimization was performed using both *fmincon* and Gurobi 5 for the SCHUETZR and ECME models, and with Gurobi only for iJO1366b. The result for model iJO1366b is shown in Figure 9. For SCHUETZR, ECME and iJO1366b, the minimal distances found were 0.43, 4.98 and 0.22 mmol/g·h, respectively. For iJO1366, only the biomass reaction (experimental reaction #25) was outside the bounds defined by the experimental uncertainty. The flux through the biomass reaction was 0.46 mmol/g·h in the distance-minimizing solution, versus an experimentally determined value of 0.64±0.01. In the SCHUETZR model, the biomass flux was closer (0.65) to the experimental, while the fluxes through experimental reactions 10 and 11 deviated by 0.25 and -0.30 mmol/g·h, giving in each case a deviation relative to the experimental uncertainty equalling 1.24 and -1.46, respectively. The ECME model showed largest inconsistencies, with 16 of 28 fluxes outside the uncertainty limits, and a minimal euclidean distance of 4.98 mmol/g·h.

Calculation of best-fitting objective-optimal flux distributions: The above analysis was repeated, now with requirement to objective-optimality. The distance-optimal solution among the objective-optimal solutions for model iJO1366 using the optimization of biomass objective and default constraints is shown in Figure 10. 21 of 28 reactions are now outside the experimental uncertainty bounds. The euclidean distance to the experimental flux distribution is 12.39 mmol/g·h. The flux through the biomass reaction is 0.76 and the glucose uptake rate, having been constrained to the experimental range, is 7.80 mmol/g·h. Most fluxes are above their experimentally determined levels. This can be interpreted as a general scaling-up of the flux pattern, due to the requirement of maximal flux through the biomass reaction for the solution to be optimal with respect to the objective. The biomass flux in turn is limited by the glucose uptake rate.

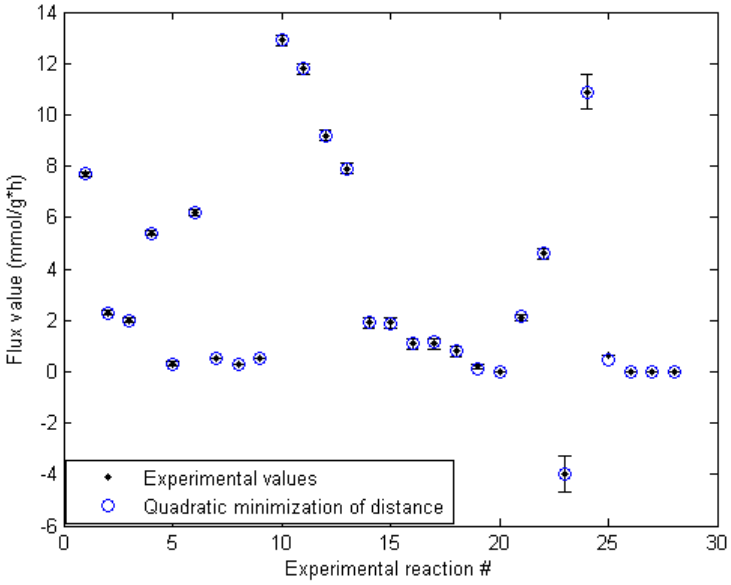

Figure 9: Results of quadratic minimization of distance to experimental flux values for the model iJO1366b. Default model constraints apply, but no requirement with respect to the value of the objective function is made. Only the biomass flux is outside the bounds defined by the experimental uncertainty.

Minimal distance vs. requirement for optimality: To evaluate to what degree the hypothesis of optimization of the selected objective agrees with the experimental data, the minimal achievable distance to the experimental data was calculated while varying the requirement for optimality with respect to the objective. This was performed using the script `optreqanalysis.m`. Results using the maximization of biomass and maximization of ATP production objectives on all three models with default constraints are shown in Figure 11. For the SCHUETZR and iJO1366b models, the minimal achievable distance approaches zero with the optimality requirement, as expected. For the ECME model, the minimal distance shows a higher baseline, pointing to a basic inconsistency as found in the feasibility analysis. The minimal distance for the ATP and biomass objectives in the ECME model differ even at a relative optimality requirement at zero, due to the flux through the biomass reaction being constrained to the experimentally determined range when using the ATP objective. It is worth noting that above an optimality requirement of 0.5, the ranking of the objectives with respect to the minimal distance does not change. This suggests that reducing the relative optimality requirement slightly from unity does not affect the ranking of objectives by this measure. A thorough analysis covering more objectives would reveal if this is a generally valid observation.

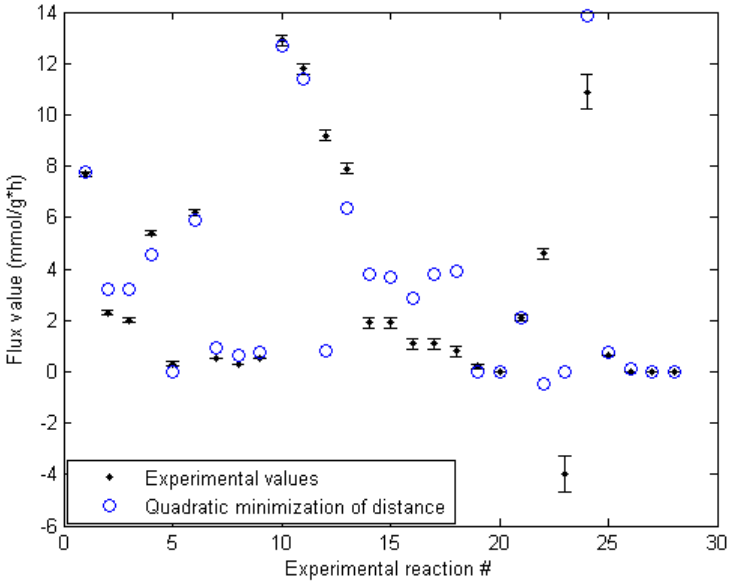

Figure 10: Results of quadratic minimization of distance to experimental flux values while maintaining objective-optimality, for model iJO1366b. Maximization of biomass objective and default model constraints apply. 21 of 28 fluxes are outside the bounds defined by the experimental uncertainty, and the euclidean distance is 12.39 mmol/g·h.

Effect of different constraints: To observe the effect of different constraints, minimization of distance subject to objective-optimality was performed for 20 combinations of objective and constraints with the SCHUETZR model. The results are shown in Figure 12. The tested constraints appear to have little effect, except when using the maximization of ATP objective and in causing a convergence of the results when applying the constraints in combination.

Multi-model comparison of objectives: Minimal achievable distances under objective-optimality requirement for four objectives used with the SCHUETZR central carbon metabolism and iJO1366 genome-scale models are shown in Figure 13. Results for the ECME model are not shown, as use of this model gave highly increased distances, especially for the maximization of ATP production objective. This could possibly indicate an issue with the definition of this objective in that model. The SCHUETZR model gives the shortest distance for three of four objectives, but the between-model variability for SCHUETZR and iJO1366b is on the same order as the effect of changing the objective, except for the maximization of ATP objective. A comparison using several more objectives would be desirable to determine what the general pattern is.

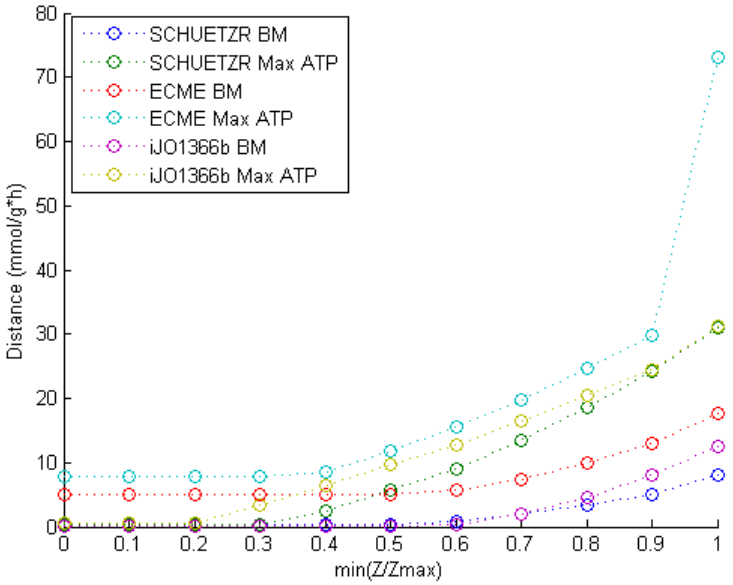

Figure 11: Results of quadratic minimization of distance to experimental flux values, for varying objective-optimality requirements. Note that for the ECME model, a larger minimal distance is found when using the Max ATP objective than when using the Max BM objective, even when the optimality requirement is zero. This is likely because the experimentally determined biomass is applied as a constraint on the model in the former case, possibly increasing the minimum inconsistency.

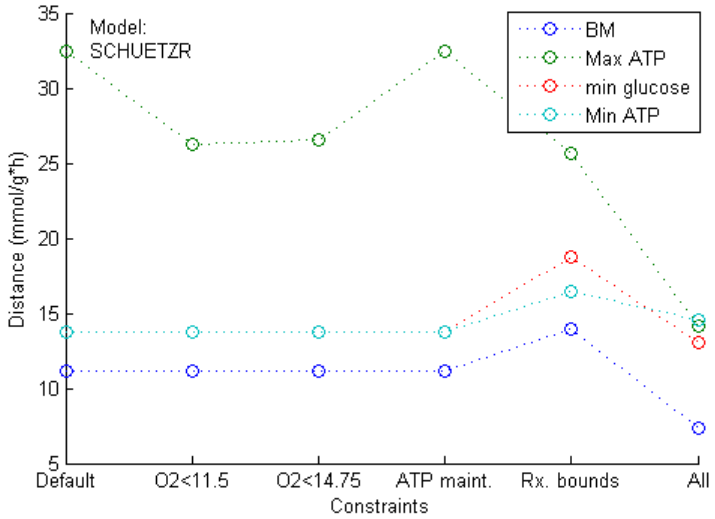

Figure 12: Results of quadratic minimization of distance to experimental flux values, under requirement of objective-optimality, using different model constraints on the SCHUETZR central carbon-metabolism model. Using the maximization of biomass objective with a combined set of constraints gives the best fit to the experimental result. The oxygen uptake and ATP maintenance constraints have no effect except when using the maximization of ATP objective, while the use of the combined set of constraints ("all") causes a convergence of the results for three of the objectives.

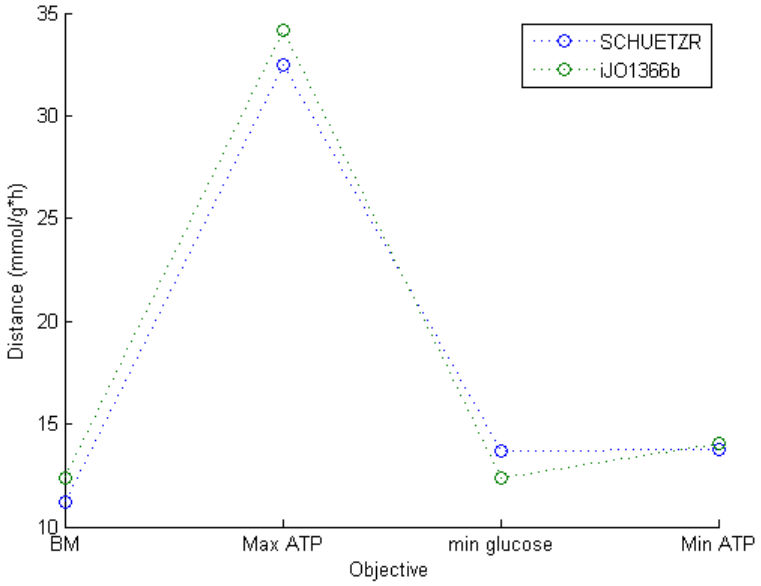

Figure 13: Comparison of minimal distances to experimental fluxes achievable under requirement of objective-optimality for different objectives using the SCHUETZR and iJO1366 models. The SCHUETZR model gives the shortest distance for three of the four objectives shown, but for those same objectives the between-model and between-objective variability is on the same order.

6 Conclusion

A program for investigating the effect of cellular objectives and model constraints on metabolic models has been implemented in MATLAB. The program was applied to analysis using three models of *E. coli* metabolism, including iJO1366, the most recent genome-scale model. Using the euclidean distance between the vectors of corresponding computed and experimentally determined fluxes as the metric, the achievable flux states closest to an experimentally determined flux distribution were determined by application of quadratic programming. Initial results suggest that decreasing slightly the requirement to the optimality of the computed flux solutions with respect to the objective function does not affect the ranking of objectives by the euclidean distance metric. The effect of choice of model constraints appear to be secondary to the choice of objective, as reported by Schuetz et al. For comparison between models using default model constraints, between-model variability was on the order of between-objective variability for three of four investigated objectives. A fourth objective gave generally poorer solutions. Further program development including implementation of more selectable objectives and constraints would allow a fuller analysis to be performed. The experimental data used as reference were gathered through ^{13}C-based metabolic flux analysis. How flux values determined by this method are used should be considered carefully, as these values are based on specific assumptions about the metabolic network structure.

Trondheim, December 20, 2012

Jarle Magnus Ribe Pahr

References

[1] Hiroaki Kitano. Systems biology: A brief overview. *Science*, 295(1662), 2002.

[2] Bernhard Palsson. The challenges of in silico biology. *Nature Biotechnology*, 18:1147–1150, 2000.

[3] Shannon P, Markiel A, Ozier O, et al. Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Res.*, 13(11):2498–2504, Nov 2003.

[4] Jonathan R. Karr, Jayodita C. Sanghvi, Derek N. Macklin, et al. A whole-cell computational model predicts phenotype from genotype. *Cell*, 150(2):389 – 401, 2012.

[5] Peter L. Freddolino and Saeed Tavazoie. The dawn of virtual cell biology. *Cell*, 150(2):248 – 250, 2012.

[6] Neema Jamshidi, Franklin J Miller, Jess Mandel, Timothy Evans, and Michael D Kuo. Individualized therapy of HHT driven by network analysis of metabolomic profiles. *BMC Systems Biology*, 5(200), 2011.

[7] Liming Liu, Rasmus Agren, Sergio Bordel, and Jens Nielsen. Use of genome-scale metabolic models for understanding microbial physiology. *FEBS letters*, 584(12):2556–2564, June 2012.

[8] Marco Terzer, Nathaniel D. Maynard, Markus W. Covert, and Jorg Stelling. Genome-scale metabolic networks. *WIREs Systems Biology and Medicine*, 1(3):285–297, 2009.

[9] Mohammad Tauqeer Alam, Marnix H. Medema, Eriko Takano, and Rainer Breitling. Comparative genome-scale metabolic modeling of actinomycetes: The topology of essential core metabolism. *FEBS Letters*, 585(14):2389 – 2394, 2011.

[10] Mario Latendresse, Markus Krummenacker, Miles Trupp, and Peter D. Karp. Construction and completion of flux balance models from pathway databases. *Bioinformatics*, 28(3):388–396, 2012.

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

[12] JD Orth and B Palsson. Gap-filling analysis of the iJO1366 *Escherichia coli* metabolic network reconstruction for discovery of metabolic functions. *BMC Syst Biol*, 6(30), 2012.

[13] Natalie C. Duarte, Scott A. Becker, Neema Jamshidi, et al. Global reconstruction of the human metabolic network based on genomic and bibliomic data. *Proceedings of the National Academy of Sciences*, 104(6):1777–1782, 2007.

[14] Cristiana Gomes de Oliveira Dal'Molin, Lake-Ee Quek, Robin William Palfreyman, Stevens Michael Brumbley, and Lars Keld Nielsen. AraGEM, a genome-scale reconstruction of the primary metabolic network in *Arabidopsis*. *Plant Physiology*, 152(2):579–589, February 2010.

[15] Cristiana Gomes de Oliveira Dal'Molin, Lake-Ee Quek, Robin Palfreyman, and Lars Nielsen. AlgaGEM - a genome-scale metabolic reconstruction of algae based on the *Chlamydomonas reinhardtii* genome. *BMC Genomics*, 12(Suppl 4):S5, 2011.

[16] J. S. Edwards and B. O. Palsson. The *Escherichia coli* MG1655 *in silico* metabolic genotype: Its definition, characteristics, and capabilities. *Proceedings of the National Academy of Sciences*, 97(10):5528–5533, 2000.

[17] Jennifer Reed, Thuy Vo, Christophe Schilling, and Bernhard Palsson. An expanded genome-scale model of *Escherichia coli* K-12 (iJR904 GSM/GPR). *Genome Biology*, 4(9):R54, 2003.

[18] Adam M Feist, Cristopher S Henry, Jennifer L Reed, et al. A genome-scale metabolic reconstruction for *Escherichia coli* K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Molecular Systems Biology*, 3(121), 2007.

[19] Jeffrey D Orth, Tom M Conrad, Jessica Na, et al. A comprehensive genome-scale reconstruction of *Escherichia coli* metabolism—2011. *Molecular Systems Biology*, 7(535), 2011.

[20] Jeremy S. Edwards and Bernhard O. Palsson. Systems properties of the *Haemophilus influenzae* metabolic genotype. *Journal of Biological Chemistry*, 274(25):17410–17416, 1999.

[21] Benjamin D Heavner, Kieran Smallbone, Brandon Barker, Pedro Mendes, and Larry P Walker. Yeast 5 – an expanded reconstruction of the *Saccharomyces cerevisiae* metabolic network. *BMC Systems Biology*, 6(55), 2012.

[22] Yu-Chieh Liao, Ming-Hsin Tsai, Feng-Chi Chen, and Chao A. Hsiung. Gemsirv: a software platform for genome-scale metabolic model simulation, reconstruction and visualization. *Bioinformatics*, 28(13):1752–8, July 2012.

[23] Stephan Pabinger, Robert Rader, Rasmus Agren, Jens Nielsen, and Zlatko Trajanoski. Memosys: Bioinformatics platform for genome-scale metabolic models. *BMC Syst Biol*, 5(20), 2011.

[24] O. Dias, M. Rocha, E. C. Ferreira, and I. Rocha. Merlin: Metabolic models reconstruction using genome-scale information. In *Proceedings 11th IFAC Conference Computer Applications on Biotechnology (CAB 2010), Leuven, Belgium*, pages 120–125, jul 2010.

[25] Stefan Gretar Thorleifsson and Ines Thiele. rBioNet: A COBRA toolbox extension for reconstructing high-quality biochemical networks. *Bioinformatics*, 27(14):2009–2010, 2011.

[26] Christopher S Henry, Matthew DeJongh, Aaron A Best, et al. High-throughput generation, optimization and analysis of genome-scale metabolic models. *Nature Biotechnology*, 28:977–982, 2010.

[27] M.G. Poolman, B.K. Bonde, A. Gevorgyan, H.H. Patel, and D.A. Fell. Challenges to be faced in the reconstruction of metabolic networks from public databases. *IEE Proc.-Syst. Biol.*, 153(5):379–384, Sep 2006.

[28] A Gevorgyan, MG Poolman, and DA Fell. Detection of stoichiometric inconsistencies in biomolecular models. *Bioinformatics*, 24(19):2245–51, Oct 2008.

[29] Vinay Satish Kumar, Madhukar S Dasika, and Costas D Maranas. Optimization based automated curation of metabolic reconstructions. *BMC Bioinformatics*, 8(212), 2007.

[30] Jeffrey D. Orth and Bernhard Ø Palsson. Systematizing the generation of missing metabolic knowledge. *Biotechnology and Bioengineering*, 107(3):403–412, 2010.

[31] R. Nigam and S. Liang. Algorithm for perturbing thermodynamically infeasible metabolic networks. *Computers in Biology and Medicine*, 37(2):126–133, Feb 2007.

[32] Hucka M, Finney A, Sauro HM, et al. The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–31, Mar 2003.

[33] C. H. Edwards and David E. Penney. *Elementary Linear Algebra*. Prentice Hall, 1988.

[34] Bernhard Ø Palsson. *Systems Biology: Properties of Reconstructed Networks*. Cambridge University Press, 2006.

[35] Jason A. Papin, Joerg Stelling, Nathan D. Price, et al. Comparison of network-based pathway analysis methods. *Trends in Biotechnology*, 22(8):400 – 405, 2004.

[36] S. Schuster, C. Hilgetag, J.H. Woods, and D.A. Fell. Reaction routes in biochemical reaction systems: Algebraic properties, validated calculation procedure and example from nucleotide metabolism. *Journal of Mathematical Biology*, 45:153–181, 2002.

[37] S. Schuster, T. Dandekar, and D.A. Fell. Detection of elementary flux modes in biochemical networks: A promising tool for pathway analysis and metabolic engineering. *Trends in Biotechnology*, 17(2):53–60, 1999. cited By (since 1996) 298.

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

[39] C.H. Schilling, D. Letscher, and B.O. Palsson. Theory for the systemic definition of metabolic pathways and their use in interpreting metabolic function from a pathway-oriented perspective. *Journal of Theoretical Biology*, 203(3):229–248, Apr 2000.

[40] Christoph Kaleta, LuÃs Filipe de Figueiredo, and Stefan Schuster. Can the whole be less than the sum of its parts? Pathway analysis in genome-scale metabolic networks using elementary flux patterns. *Genome Research*, 2009.

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

[42] C.H. Schilling, J.S Edwards, D. Letscher, and B.O. Palsson. Combining pathway analysis with flux balance analysis for the comprehensive study of metabolic systems. *Biotechnology and Bioengineering*, 71(4):286–306, 2000.

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

[44] Robert Schuetz, Lars Kuepfer, and Uwe Sauer. Systematic evaluation of objective functions for predicting intracellular fluxes in *Escherichia coli*. *Molecular Systems Biology*, 3(119), 2007.

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

[46] Jong Myoung Park, Tae Yong Kim, and Sang Yup Lee. Constraints-based genome-scale metabolic simulation for systems metabolic engineering. *Biotechnology Advances*, 27(6):979–988, 2009.

[47] Markus W. Covert, Iman Famili, and Bernhard O. Palsson. Identifying constraints that govern cell behavior: a key to converting conceptual to computational models in biology? *Biotechnology and Bioengineering*, 84(7):763–772, Dec 2003.

[48] Anne Kümmel, Sven Panke, and Matthias Heinemann. Systematic assignment of thermodynamic constraints in metabolic network models. *BMC Bioinformatics*, 7(512), Nov 2006.

[49] Nathan D. Price, Iman Famili, Daniel A. Beard, and Bernhard Ø. Palsson. Extreme pathways and Kirchhoff's second law. *Biophysical Journal*, 83(5):2879–2882, 2002.

[50] Karthik Raman and Nagasuma Chandra. Flux balance analysis of biological systems: applications and challenges. *Briefings in Bioinformatics*, 10(4):435–449, 2009.

[51] Daniele De Martino, Matteo Figliuzzi, Andrea De Martino, and Enzo Marinari. A scalable algorithm to explore the gibbs energy landscape of genome-scale metabolic networks. *PLoS Comput Biol*, 8(6):e1002562, 2012.

[52] Feng Yanga, Hong Qianb, and Daniel A. Beard. Ab initio prediction of thermodynamically feasible reaction directions from biochemical network stoichiometry. *Metabolic engineering*, 7(4):251–259, 2005.

[53] Christopher S. Henry, Linda J. Broadbelt, and Vassily Hatzimanikatis. Thermodynamics-based metabolic flux analysis. *Biophysical Journal*, 92:1792–1805, Mar 2007.

[54] Markus W. Covert and Bernhard Ø Palsson. Transcriptional regulation in constraints-based metabolic models of *Escherichia coli*. *Journal of Biological Chemistry*, 277(31):28058–28064, 2002.

[55] Markus W. Covert, Nan Xiao, Tiffany J. Chen, and Jonathan R. Karr. Integrating metabolic, transcriptional regulatory and signal transduction models in *Escherichia coli*. *Bioinformatics*, 24(18):2044–2050, 2008.

[56] B Teusink, J Passarge, CA Reijenga, et al. Can yeast glycolysis be understood in terms of in vitro kinetics of the constituent enzymes? testing biochemistry. *European Journal of Biochemistry*, 267(17):5313–5329, SEP 2000.

[57] Douglas B. Kell and Pedro Mendes. Snapshots of systems. In Athel Cornish-Bowden and Maria Luz Cardenas, editors, *Technological and Medical Implications of Metabolic Control Analysis*, volume 74 of *NATO Science Series*, pages 3–25. Springer Netherlands, 2000.

[58] Jan-Hendrik S. Hofmeyr. Metabolic control analysis in a nutshell. In *Proceedings of the 2nd International Conference on Systems Biology*, page 291–300. Omnipress, 2001.

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

[60] DA Beard, SC Liang, and H Qian. Energy balance for analysis of complex metabolic networks. *Biophysical Journal*, 83(1):79–86, JUL 2002.

[61] Emrah Nikerel, Jan Berkhout, Fengyuan Hu, et al. Understanding regulation of metabolism through feasibility analysis. *PlOS One*, 7(7):e39396, 2012.

[62] HS Song and D Ramkrishna. Prediction of dynamic behavior of mutant strains from limited wild-type data. *Metabolic engineering*, 13(2):69–80, Mar 2012.

[63] Adam M Feist and Bernhard O Palsson. The biomass objective function. *Current Opinion in Microbiology*, 13:344–349, 2010.

[64] Amit Varma and Bernhard O. Palsson. Metabolic flux balancing: Basic concepts, scientific and practical use. *Nature Biotechnology*, 12:994–998, 1994.

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

[66] JS Edwards, R Ramakrishna, and BO. Palsson. Characterizing the metabolic phenotype: a phenotype phase plane analysis. *Biotechnol Bioeng.*, 77(1):27–36, Jan 2002.

[67] Daniel Segrè, Dennis Vitkup, and George M. Church. Analysis of optimality in natural and perturbed metabolic networks. *PNAS*, 99(23):15112–15117, Nov 2002.

[68] Tomer Shlomi, Omer Berkman, and Eytan Ruppin. Regulatory on/off minimization of metabolic flux changes after genetic perturbations. *PNAS*, 102(21):7695–7700, 2005.

[69] Radhakrishnan Mahadevan, Jeremy S. Edwards, and Francis J. Doyle III. Dynamic flux balance analysis of diauxic growth in *Escherichia coli*. *Biophysical Journal*, 83(3):1331 – 1340, 2002.

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

[71] Stephen S. Fong, Anthony P. Burgard, Christopher D. Herring, et al. In silico design and adaptive evolution of *Escherichia coli* for production of lactic acid. *Biotechnology and Bioengineering*, 91(5):643–648, 2005.

[72] Adam M. Feist, Daniel C. Zielinski, Jeffrey D. Orth, et al. Model-driven evaluation of the production potential for growth-coupled products of *Escherichia coli*. *Metabolic Engineering*, 12(3):173 – 186, 2010.

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

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

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

[76] Kiran Patil, Isabel Rocha, Jochen Forster, and Jens Nielsen. Evolutionary programming as a platform for in silico metabolic engineering. *BMC Bioinformatics*, 6(1):308, 2005.

[77] Sridhar Ranganathan, Patrick F. Suthers, and Costas D. Maranas. Optforce: An optimization procedure for identifying all genetic manipulations leading to targeted overproductions. *PLoS Comput Biol*, 6(4):e1000744, 04 2010.

[78] Ranganathan S, Wei Tee T, Chowdhury A, et al. An integrated computational and experimental study for overproducing fatty acids in *Escherichia coli*. *Metabolic Engineering*, 2012.

[79] Desmond S Lun, Graham Rockwell, Nicholas J Guido, et al. Large-scale identification of genetic design strategies using local search. *Mol Syst Biol*, 5(296), 2009.

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

[81] Abdelhalim Larhlimi and Alexander Bockmayr. A new approach to flux coupling analysis of metabolic networks. In *Computational Life Sciences II*, volume 4216 of *Lecture Notes in Computer Science*, pages 205–215. Springer Berlin / Heidelberg, 2006.

[82] 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):236, 2011.

[83] Richard A. Notebaart, Bas Teusink, Roland J. Siezen, and Balázs Papp. Coregulation of metabolic genes is better explained by flux coupling than by network distance. *PLoS Comput Biol*, 4(1):e26, 01 2008.

[84] Sayed-Amir Marashi and Alexander Bockmayr. Flux coupling analysis of metabolic networks is sensitive to missing reactions. *Biosystems*, 103(1):57 – 66, 2011.

[85] Sayed-Amir Marashi, Laszlo David, and Alexander Bockmayr. On flux coupling analysis of metabolic subsystems. *Journal of Theoretical Biology*, 302:62 – 69, 2012.

[86] Patrick F. Suthers, Young J. Chang, and Costas D. Maranas. Improved computational performance of MFA using elementary metabolite units and flux coupling. *Metabolic Engineering*, 12(2):123 – 128, 2010.

[87] E. Almaas, B. Kovács, T. Vicsek, and Z. N. Oltvai A.-L. Barabási. Global organization of metabolic fluxes in the bacterium *Escherichia coli*. *Nature*, 427:839–843, 2003.

[88] Kenji Nakahigashi et al. Systematic phenome analysis of *Escherichia coli* multiple-knockout mutants reveals hidden reactions in central carbon metabolism. *Mol Syst Biol*, 5(306), 2009.

[89] Steven M Kelk, Brett G Olivier, Leen Stougie, and Frank J Bruggeman. Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. *Scientific Reports*, 2(580):PMC3419370, 2012.

[90] YeeWen Choon, MohdSaberi Mohamad, Safaai Deris, et al. Identifying gene knockout strategies using a hybrid of bees algorithm and flux balance analysis for in silico optimization of microbial strains. In *Distributed Computing and Artificial Intelligence*, volume 151 of *Advances in Intelligent and Soft Computing*, pages 371–378. Springer Berlin Heidelberg, 2012.

[91] Scott A Becker, Adam M Feist, Monica L Mo, et al. Quantitative prediction of cellular metabolism with constraint-based models: the COBRA toolbox. *Nature Protocols*, 2:727 – 738, 2007.

[92] Wilbert B. Copeland, Bryan A. Bartley, Deepak Chandran, et al. Computational tools for metabolic engineering. *Metabolic Engineering*, 14(3):270 – 280, 2012.

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

[94] Gregory Stephanopoulos. Metabolic fluxes and metabolic engineering. *Metabolic Engineering*, 1(1):1 – 11, 1999.

[95] Wolfgang Wiechert. 13C Metabolic Flux Analysis. *Metabolic Engineering*, 3(3):195 – 206, 2001.

[96] Robert W. Leighty and Maciek R. Antoniewicz. Dynamic metabolic flux analysis (DMFA): A framework for determining fluxes at metabolic non-steady state. *Metabolic Engineering*, 13(6):745 – 755, 2011.

[97] Xueyang Feng, Lawrence Page, Jacob Rubens, et al. Bridging the gap between fluxomics and industrial biotechnology. *J Biomed Biotechnol*, 2010:460717, 2010.

[98] Lake-Ee Quek, Christoph Wittmann, Lars K Nielsen, and Jens O Krömer. OpenFLUX: efficient modelling software for 13C-based metabolic flux analysis. *Microbial Cell Factories*, 8(25), 2009.

[99] Patrick F. Suthers, Anthony P. Burgard, Madhukar S. Dasika, et al. Metabolic flux elucidation for large-scale models using 13C labeled isotopes. *Metabolic Engineering*, 9(5-6):387 – 405, 2007.

[100] Gregory N Stephanopoulos, Aristos A Aristidou, and Jens Nielsen. *Metabolic Engineering: Principles and Methodologies*. Academic Press, 1998.

[101] Maciek R. Antoniewicz, Joanne K. Kelleher, and Gregory Stephanopoulos. Determination of confidence intervals of metabolic fluxes estimated from stable isotope measurements. *Metabolic Engineering*, 8(4):324 – 337, 2006.

[102] Yinjie J. Tang, Hector Garcia Martin, Samuel Myers, et al. Advances in analysis of microbial metabolic fluxes via 13C isotopic labeling. *Mass Spectrometry Reviews*, 28(2):362–375, 2009.

[103] Eliane Fischer and Uwe Sauer. Metabolic flux profiling of *Escherichia coli* mutants in central carbon metabolism using GC-MS. *European Journal of Biochemistry*, 270(5):880–891, 2003.

[104] Fischer E and Sauer U. Large-scale in vivo flux analysis shows rigidity and suboptimal performance of *Bacillus subtilis* metabolism. *Nature genetics*, 37(6):636–40, Jun 2005.

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

[106] K Uygun, HWT Matthew, and Y Huang. Investigation of metabolic objectives in cultured hepatocytes. *Biotechnology and Bioengineering*, 97(3):622–637, Jun 2007.

[107] Jens Nielsen. Principles of optimal metabolic network operation. *Molecular Systems Biology*, 3(126), 2007.

[108] VS Kumar and CD Maranas. Growmatch: An automated method for reconciling in silico/in vivo growth predictions. *PLoS Comput Biol*, 5(3):e1000308, 2009.

[109] Joo Sang Lee, Takashi Nishikawa, and Adilson E Motter. Why optimal states recruit fewer reactions in metabolic networks. *Discrete and Continuous Dynamical Systems - Series A*, 32(8):2937 – 2950, 2012.

[110] Tomer Benyamini, Ori Folger, Eytan Ruppin, and Tomer Shlomi. Flux balance analysis accounting for metabolite dilution. *Genome Biology*, 11(4):R43, 2010.

[111] Ali R. Zomorrodi and Costas D. Maranas. Optcom: A multi-level optimization framework for the metabolic modeling and analysis of microbial communities. *PLOS Computational Biology*, 8(2):e1002363, 2012.

[112] Anthony P. Burgard and Costas D. Maranas. Optimization-based framework for inferring and testing hypothesized metabolic objective functions. *Biotechnology and Bioengineering*, 82(6):670–677, 2003.

[113] Andrea L. Knorr, Rishi Jain, and Ranjan Srivastava. Bayesian-based selection of metabolic objective functions. *Bioinformatics*, 23(3):351–357, 2007.

[114] Annik Perrenoud and Uwe Sauer. Impact of global transcriptional regulation by arca, arcb, cra, crp, cya, fnr, and mlc on glucose catabolism in *Escherichia coli*. *Journal of Bacteriology*, 187:3171–3179, 2005.

[115] Eliane Fischer, Nicola Zamboni, and Uwe Sauer. High-throughput metabolic flux analysis based on gas chromatography–mass spectrometry derived 13C constraints. *Analytical Biochemistry*, 325:308–316, 2004.

[116] Marcel Emmerling, Michael Dauner, Aaron Ponti, et al. Metabolic flux responses to pyruvate kinase knockout in *Escherichia coli*. *Journal of Bacteriology*, 184(1):152–164, 2002.

Appendix A Model details

Model format: The metabolic models used in this project are formulated in the COBRA-compliant SBML format. SBML files are based on .xml markup language and can be viewed with a plain-text editor. The basic features of the COBRA SBML format are explained below. The central elements are species, reactions and compartments. Compartments refer to a physical environment separate from that of other compartments. The most common compartments used in metabolic models are `"cytosol"` and `"extracellular"`. The iJO1366 model also contains a `"periplasm"` compartment. The same metabolite is treated as separate species in each compartment, so a given metabolite may be described by several species. Compartments are defined first in the model file. In the SCHUETZR model, the two compartments are defined as follows:

```
<listOfCompartments>
        <compartment id="C_c" name="cytoplasm" />
        <compartment id="C_e" name="external" />
 </listOfCompartments>
```

Secondly, species are defined between the `<listofspecies>` and `</listofspecies>` tags. In the SCHUETZR model, the following line defines the species glucose in the cytosolic compartment.

```
<species id="M_GLC_c" name="Glucose" compartment="C_c" />
```

Reactions which are defined later in the model file can then reference this species as a reactant or product of the reaction. According to the COBRA-compliant SBML format, all metabolite ID fields must begin with "M_", and all reaction ID fields must begin with "R_". Exchange reactions are those reactions which allow species to enter or leave the system independent from reactions with other species. By convention, the reaction ID of exchange reactions begin with "R_EX". The environmental conditions such as growth media composition, oxygen availability are simulated by setting bounds on the exchange reactions. The essential elements of a model reaction are its reactants, products and upper and lower bounds. Gene associations, enzyme numbers and other additional information can be specified for use in gene-knockout and other analytical procedures but this has not been a concern here. The COBRA-compliant SBML file structure is described in detail at: `http://www.nature.com/protocolexchange/system/uploads/1808/original/Supplementary_Material.pdf`

Validation of the SBML file structures was performed using the SBML validator at `http://sbml.org/Facilities/Validator/index.jsp`. After validation, error correction and re-validation, no errors were found in the final models used. One error was found in the model by Schuetz et al., relating to the definition of a species in the *gnd* reaction. The impact of this, if any, on the results obtained using that model is unknown.

Below is an example of a biotransformation reaction described in SBML. The phosphorylation of Fructose 6-Phosphate (F6P) by ATP producing Fructose diphosphate (FDP) is represented in the SCHUETZR model by the following code:

```
<reaction id="R_pfkAB" name="Phosphofructokinase (pfkA/B)" reversible="false">
    <listOfReactants>
        <speciesReference species="M_F6P_c" stoichiometry="1.0"/>
    <speciesReference species="M_ATP_c" stoichiometry="1.0"/>
    </listOfReactants>
    <listOfProducts>
    <speciesReference species="M_FDP_c" stoichiometry="1.0"/>
    </listOfProducts>
    <kineticLaw>
        <math xmlns="http://www.w3.org/1998/Math/MathML">
        <ci> FLUX_VALUE </ci>
        </math>
        <listOfParameters>
        <parameter id="LOWER_BOUND" value="0" units="mmol_per_gDW_per_hr"/>
        <parameter id="UPPER_BOUND" value="1000" units="mmol_per_gDW_per_hr"/>
        <parameter id="OBJECTIVE_COEFFICIENT" value="0"/>
        <parameter id="FLUX_VALUE" value="0" units="mmol_per_gDW_per_hr"/>
        </listOfParameters>
     </kineticLaw>
</reaction>
```

Note that ATP simply disappears in this reaction. This is an obvious simplification as in reality ADP is produced when ATP gives up one phosphate group. However, the model by Schuetz et al. does not contain ADP as a species. Thus in the SCHUETZR model the ATP synthesis reactions, which in reality consumes ADP and P_i (inorganic phosphate) to produce ATP, likewise show ATP appearing "from nothing". The significance of this simplification for the quality of model predictions has not been investigated here, but for a fully rigorous analysis it would certainly warrant consideration. In the ECME and iJO1366 models, this simplification is not made, and the production and consumption of ADP and other ATP-related compounds are accounted for.

Below, an example of a transportation reaction is given. In the SCHUETZR model, the uptake/secretion of lactate is described by the following code. The reaction direction is defined with secretion of lactate giving positive flux values. Note that with the default settings (as defined by the SBML code), uptake of lactate is not allowed:

```
<reaction id="R_lac" name="lactate transport" reversible="false">
        <listOfReactants>
    <speciesReference species="M_LAC_c" stoichiometry="1.0"/>
    </listOfReactants>
    <listOfProducts>
    <speciesReference species="M_LAC_e" stoichiometry="1.0"/>
    </listOfProducts>
        <kineticLaw>
        <listOfParameters>
        <parameter id="LOWER_BOUND" value="0" units="mmol_per_gDW_per_hr"/>
        <parameter id="UPPER_BOUND" value="1000" units="mmol_per_gDW_per_hr"/>
        <parameter id="OBJECTIVE_COEFFICIENT" value="0"/>
        <parameter id="FLUX_VALUE" value="0" units="mmol_per_gDW_per_hr"/>
        </listOfParameters>
    </kineticLaw>
</reaction>
```

Below an example of an exchange reaction is given:

```
<reaction id="R_EX_GLC_e" name="Glucose exchange" reversible="true">
        <notes>
        <html xmlns="http://www.w3.org/1999/xhtml"><p>SUBSYSTEM: Exchange</p></html>
        </notes>
        <listOfReactants>
        <speciesReference species="M_GLC_e"/>
        </listOfReactants>
        <kineticLaw>
        <listOfParameters>
        <parameter id="UPPER_BOUND" value="-10" units="mmol_per_gDW_per_hr"/>
        <parameter id="FLUX_VALUE" value="1000" units="mmol_per_gDW_per_hr"/>
         </listOfParameters>
         </kineticLaw>
</reaction>
```

As a final example, the biomass reaction as defined in the SCHUETZR model is shown below:

```
<reaction id="R_biomass" name="biomass" reversible="false">
        <listOfReactants>
        <speciesReference species="M_ATP_c" stoichiometry="40.2"/>
        <speciesReference species="M_NADPH_c" stoichiometry="15.7"/>
        <speciesReference species="M_G6P_c" stoichiometry="0.33"/>
        <speciesReference species="M_F6P_c" stoichiometry="0.07"/>
        <speciesReference species="M_R5P_c" stoichiometry="0.96"/>
        <speciesReference species="M_E4P_c" stoichiometry="0.36"/>
        <speciesReference species="M_GA3P_c" stoichiometry="0.12"/>
        <speciesReference species="M_3PG_C" stoichiometry="0.86"/>
        <speciesReference species="M_PEP_c" stoichiometry="0.77"/>
        <speciesReference species="M_PYR_c" stoichiometry="2.94"/>
        <speciesReference species="M_ACCOA_c" stoichiometry="2.41"/>
        <speciesReference species="M_OA_c" stoichiometry="1.65"/>
        <speciesReference species="M_AKG_c" stoichiometry="1.28"/>
        </listOfReactants>
        <listOfProducts>
                <speciesReference species="M_BM_e" stoichiometry="1.0"/>
                <speciesReference species="M_NADH_c" stoichiometry="3.0"/>
                <speciesReference species="M_COA_c" stoichiometry="2.41"/>
        </listOfProducts>
        <kineticLaw>
         <math xmlns="http://www.w3.org/1998/Math/MathML">
         <ci> FLUX_VALUE </ci>
```

Table A.1: Definition of objective functions.

Objective (ID)	Model	Reaction(s)
Max BM (1)	SCHUETZR	biomass
	ECME	Biomass_Ecoli_core_w_GAM
	iJO1366b	Ec_biomass_iJO1366_core_53p95M
Max/Min ATP (2/6)	SCHUETZR	pgk,pykAF,sucCD,atp,ackAB_tdcD_purT
	ECME	ATPS4r,Pyk
	iJO1366b	AP5AH,ATPS4rpp,PYK
Min glucose (4)	SCHUETZR	EX_GLC(e)
	ECME	EX_glc(e)
	iJO1366b	EX_glc(e)

Table A.2: Definition of model constraints

Constraint	ID	Definition
P/O = 1	1	nuo = cydAB = 0 (SCHUETZR only)
Glc/O2 = 3/2	2	Coupled Glc/O_2 reaction (SCHUETZR only)
O2 max < 11.5	3	EX_O2(e) > -11.5
O2 max < 14.75	4	EX_O2(e) > -14.75
ATP maintenance	5	'maint' = 7.6 (SCHUETZR) 'ATPM' = 7.6 (ECME/iJO1366b)
Reaction bounds	6	$v_i \leq 2 \cdot v_{glc,max}$ (SCHUETZR only)
All	7	Combination of above. (O_2 max < 11.5)

```
        </math>
        <listOfParameters>
        <parameter id="LOWER_BOUND" value="0" units="mmol_per_gDW_per_hr"/>
        <parameter id="UPPER_BOUND" value="1000" units="mmol_per_gDW_per_hr"/>
        <parameter id="OBJECTIVE_COEFFICIENT" value="1"/>
        <parameter id="FLUX_VALUE" value="0" units="mmol_per_gDW_per_hr"/>
         </listOfParameters>
</kineticLaw>
```

Changes to models: The model used by Schuetz et al. was edited to remove duplicate reactions and to allow processing with the COBRA Toolbox. Table A.3 shows a list of the merged reactions in the SCHUETZR model. The *E. coli* core model was edited to include reactions present in the Schuetz model but missing from that model. Table A.4 shows a list of those reactions. In

addition, an uptake reaction coupling glucose and oxygen uptake to a 2/3 ratio was added to the SCHUETZR model to allow implementation of this ratio as a model constraint. This reaction is by default constrained to zero.

Table A.1 Shows the definition of the objective functions used. Table A.2 shows the definition of the model constraints. Table A.5 shows the experimental reactions and the corresponding model reactions for all three models.

Table A.3: Merged reactions in SCHUETZR model

Original reaction ID	Merged reaction ID	Description
pykA/pykF	R_pykAF	Pyruvate kinase
AcnA/AcnB	R_acnAB	Aconitase half-reaction A
acnA_r2/acnB_r2	R_acnAB_r2	Aconitase half-reaction B
fbaA/fbaB	R_fbaAB	Fructose bis-P aldolase
gpmA/gpmB	R_gpmAB	Phosphoglyceromutase
fbP/glpX	R_fbp_glpX	Fructose-bisphosphatase
rpiA/rpiB	R_rpiAB	Ribose 5-phosphate isomerase
tktA/tktB	R_tktAB	Transketolase
tktA_R2/tktB_R2	R_tktAB_r2	Transketolase
talA/talB	R_talAB	Transaldolase
gltA/prpC	R_gltA_prpC	Citrate synthase
frdABCD/sdhAB	R_frdABCD_sdhAB	Succinate dehydrogenase
fumA/fumB/fumC	R_fumABC	Fumarate hydratase
pflB/tdcE	R_pflB_tdcE	Pyruvate formate lyase
fdhF/fdoGHI/fdnGHI_r2	R_fdhF_fdoGHI	Formate dehydrogenase
adhE/mhpF	R_adhE_mhpF	Acetaldehyde dehydrogenase
ackA/ackB/tdcD/purT	R_ackAB_tdcD_purT	Acetate kinase
adhE_r2/adhP/adhC	R_adhPC_adhE_r2	Alcohol dehydrogenase

Table A.4: Reactions added to *E. coli* core model.

Reaction ID	Description
R_glk	Glucokinase
R_mglABC	Methyl-galactoside transporter
R_edd	Entner-Douderoff dehydratase
R_mgsA	Methylglyoxal synthase
R_Eda	Entner-Douderoff aldolase
R_Acs	Acetyl-CoA synthase
R_Mqo	Malate quinone oxidioreductase

Table A.5: Corresponding experimental and model reactions

Rn. #	Reaction	Reaction ID		
		SCHUETZR	ECME	iJO1366b
1	GLC+ATP→G6P	glk+pts	glk+GLCpts	HEX1+GLCptspp
2	G6P→6PG+NADPH	zwf	G6PDH2r	G6PDH2r
3	6PG→P5P + CO2 + NADPH	gnd	gnd	gnd
4	G6P→F6P	pgi	pgi	PGI
5	6PG → T3P + PYR	edd	edd	edd
6	F6P + ATP → 2T3P	fbaAB	FBA	fba
7	2P5P → S7P + T3P	tktAB	TKT1	TKT1
8	P5P + E4P → F6P + T3P	tktAB_r2	TKT2	TKT2
9	S7P + T3P → E4P + F6P	talAB	talA	tala
10	T3P → PGA + ATP + NADH	gapA	GAPD	GAPD
11	PGA → PEP	gpmAB	pgm	pgm
12	PEP → PYR + ATP	pykAF - pps	pyk-pps	pyk- pps
13	PYR → AcCoA + CO2 + NADH	aceEF	PDH	PDH
14	OAA + AcCoA → ICT	gltA/prpC	CS	CS
15	ICT → OGA + CO2 + NADPH	ICD	ICDHyr	ICDHyr
16	OGA → FUM + CO2 + 1.5ATP + 2NADH	sucAB	AKGDH	akGDH
17	FUM → MAL	fumABC	FUM	FUM
18	MAL → OAA + NADH	mdh+mqo	MDH	MDH 1-3
19	MAL → PYR + CO2 + NADH	maeB+maeA	ME1+ME2	ME1+ME2
20	OAA + ATP → PEP + CO2	pck	ppck	ppck
21	PEP + CO2 → OAA	ppc	ppc	ppc
22	AcCoA → Acetate + ATP + CoA	pta-acs	PTAr - acs	PTAr - ACS
23	NADPH → NADH	udhA-pntAB	NADTRHD,THD2	NADTRHD,THD2PP
24	O2 + 2NADH → 2P/OxATP	O2	O2t	CYTBD(2)pp,CYTBO3_4pp
25	biomass	biomass	Biomass_Ecoli	Ec_biomass(...)
26	ICT + AcCoA → MAL + FUM + NADH	aceA	ICL	ICL
27	DHAP → PYR	mgsA	mgsA	mgsA
28	ETH → ETHxt	eth	EX_etoh_(e)	EX_etoh(e)

Appendix B Experimental data

Table B.1 shows the experimental flux data for growth of *E. coli* strain BW25113 on glucose in aerobic batch culture, published by Perrenoud and Sauer.[114] The fluxes were determined by ^{13}C-based MFA.

Table B.1: Experimental flux data.[114]

Reaction #	Reaction equation	Flux (mmol/g·h)	Error
1	GLC + ATP → G6P	7,7	0,1
2	G6P → 6PG + NADPH	2,3	0,1
3	6PG → P5P + CO2 + NADPH	2,0	0,1
4	G6P → F6P	5,4	0,1
5	6PG → T3P + PYR	0,3	0,1
6	F6P + ATP → 2T3P	6,2	0,1
7	2P5P → S7P + T3P	0,5	0,0
8	P5P + E4P → F6P + T3P	0,3	0,0
9	S7P + T3P → E4P + F6P	0,5	0,0
10	T3P → PGA + ATP + NADH	12,9	0,2
11	PGA → PEP	11,8	0,2
12	PEP → PYR + ATP	9,2	0,2
13	PYR → AcCoA + CO2 + NADH	7,9	0,2
14	OAA + AcCoA → ICT	1,9	0,2
15	ICT → OGA + CO2 + NADPH	1,9	0,2
16	OGA → FUM + CO2 + 1.5·ATP + 2NADH	1,1	0,2
17	FUM → MAL	1,1	0,2
18	MAL → OAA + NADH	0,8	0,2
19	MAL → PYR + CO2 + NADH	0,2	0,1
20	OAA + ATP → PEP + CO2	0,0	0,0
21	PEP + CO2 → OAA	2,1	0,1
22	AcCoA → Acetate + ATP	4,6	0,2
23	NADPH → NADH	-4,0	0,7
24	O2 + 2NADH → 2P/O×ATP	10,9	0,7
25	Biomass production	0,64	0,01

Appendix C Software

All computations were carried out using MATLAB R2012a and COBRA Toolbox v. 2.04 running under Windows 7 on a 64-bit Intel Core 2.1 Ghz processor. MATLAB is a product of The MathWorks, Inc (`https://www.mathworks.com/`). The COBRA Toolbox is a MATLAB plugin available free of charge at `http://opencobra.sourceforge.net`. The COBRA Toolbox requires the installation of LibSBML, available at `http://sbml.org/Software/libSBML`. The Gnu Linear Programming Kit (GLPK) with MATLAB interface (GLPKMEX) and Gurobi 5.01 optimization solvers were used for solving linear and quadratic problems. GLPK is available at `http://www.gnu.org/software/glpk/` and GLPKMEX can be downloaded from `http://sourceforge.net/projects/glpkmex`. Gurobi 5 is available for academic use under a free license at `www.gurobi.com`.

To install the Gurobi 5 MATLAB interface, navigate to the folder "matlab" under the Gurobi 5 installation directory and run the gurobi_setup.m file. The latest version of the COBRA Toolbox, v. 2.05, supports Gurobi 5 natively, using it as the default solver for both LP and QP problems. If COBRA version v. 2.05 and Gurobi 5 are both installed, GLPK is thus not needed. Installation files for COBRA Toolbox v. 2.04 v. 2.05, LibSBML 5.6.0 (MATLAB Windows 64-bit version) and GLPK v. 4.47, are attached with this report. Each time after starting MATLAB, the COBRA Toolbox must be initiated with the command `initCobraToolbox`.

The following section explains the operation and use of the supplied MATLAB files. For each file, the input, output and options are described.

runsim.m: *runsim* is the main function facilitating the study of the effect of combinations of model constraints and objectives on different models. Three models and four objectives are currently supported. *runsim* calls a number of other functions to perform analysis and report the results. The syntax for using the function is:

```
result = runsim(options)
```

`options` is a MATLAB variable structure, where different options are specified in separate fields. Most fields have a default value, and the program will run even if some fields do not exist. This input format is used for many of the functions described. In many cases, several of the options are the same, and one function may pass the options structure received as input for that function as input to another function. The output, here named result,

is another structure containing flux vectors, distance measures and other produced information. The following main options are available:

`model_id`: Default value is 1. Used to specify which model to use. Supported values are 1, 2 and 3, corresponding to SCHUETZR, ECME and iJO1366b models.

`constraints`: Used to specify which set of constraints to use. Default value is 0, signifying default model constraints. Valid values are 0-7 for SCHUETZR. Unless overwritten by the specified constraints, the default model constraints always apply.

`objective`: Used to specify the objective function. Default value is 1, corresponding to maximization of biomass reaction. Currently supported values are 1, 2 (max ATP), 4 (min glucose) and 6 (min ATP).

`optreq`: Objective optimality requirement. Default value is 1. When seeking a distance-optimal solution, only solutions satisfying the objective-optimality requirement specified by optreq are considered. Valid values are 0-1.

`usefmincon`: Use the MATLAB function *fmincon* to seek a distance-optimal solution. Default is 1, except when the iJO1366 model is used.

`usegurobi`: Use the Gurobi quadratic optimization program to seek a distance-optimal solution. Default is 1.

`preloadmodels`: Valid values are 0, 1. Specify whether the COBRA models used should be generated from SBML source files (.xml format) or loaded from a saved MATLAB file. Default is 1 (use preloaded model files).

`makeplots`: If set to 1, a plot comparing experimental and computed flux values will be generated. Default is 0.

`fluxreport`: Default is 1. If set to 1, a report will be printed (on screen and/or to file) with details about the produced flux solution.

`printfile`: If set to 1, the flux report will be written to a file in the working directory, the default file name being fluxreport.txt. Use the parameter options.filename to specify a different file name. If a file with the same name already exists, it will be appended.

Example: To use *runsim* with the ECME model, and leave all other settings at their default values, enter the following at the MATLAB command line:

```
options = struct;
options.model_id = 2;
result = runsim(options)
```

Supporting functions:

The function *extractflux* takes as input a flux vector from FBA result and returns a vector with 28 fluxes corresponding to the experimental reactions. As with runsim.m, the model is specified through the field `options.model_id` in the input parameter options.

Syntax: `fluxes = extractflux(fluxvector,options)`

The function *fluxreport* takes as input a vector of fluxes corresponding to the set of experimental reaction rates, and returns both sets of fluxes, the difference between each corresponding flux, the uncertainty in the experimental fluxes and the difference between the fluxes divided by that uncertainty.

Syntax: `report = fluxreport(fluxresult,expvalues,options)`

The function *compdist* takes as input a complete flux vector containing all the reactions in the model, a model identifier, and an experiment identifier. It then returns a distance between the experimental data and the corresponding model fluxes. The distance measure output can be the euclidean or the manhattan norm, and based on raw flux values or split ratios.

Syntax: `ds = compdist(fluxvector,options,sense)`

The function *fmindist* uses the MATLAB function *fmincon* to seek a solution of a given FBA problem which minimizes (or maximizes) the distance between the computed and experimental flux distribution as reported by *compdist*.

Syntax: `solution = fmindist(result,model,options,sense)`

The function *QPmindist* use the Gurobi 5 optimization solver to find a solution minimizing the euclidean distance between the computed and experimental flux distribution.

Syntax: `solution = QPmindist(model,result,options,sense)`

The function *constrainfluxes* takes as input a COBRA model and constrains one or more reactions with a defined corresponding experimental reaction to limits specified by the experimental value and by a tolerance relative to the experimental uncertainty.

Syntax: `model = constrainfluxes(model,options)`

The function *computesplits* calculates split ratios from experimental or computed flux distributions.

Syntax: `splitratios = computesplits(fluxvector,model_id)`

To view additional information about each function, enter `help function` - where `function` is the name of the function - at the MATLAB command line.

Known bugs: Setting the parameter *verbflag* to 0 currently causes the Gurobi optimization routine to fail. *Fmincon* fails to produce correct solutions for objectives 5 (minimization of redox potential production) and 6 (minimization of ATP production).

Guide to reproducing figures: To reproduce Figure 9, enter the following at the MATLAB command line after installing all necessary software and extracting the function files to the MATLAB working directory:

```
options = struct;
options.model_id=1;
options.optreq = 0;
options.makeplots = 1;
options.usefmincon = 0;
result = runsim(options)
```

To reproduce Figure 10, enter the following:

```
options = struct;
options.model_id=3;
options.makeplots = 1;
options.optreq = 1;
result = runsim(options)
```

To reproduce Figure 11, execute the function `optreqanalysis` by entering the function name at the MATLAB command line. Receiving no input arguments, the function will use default settings to produce the graph.

To reproduce Figure 12, likewise execute the function `bar3Dplot`. Estimated run time is about five minutes.

To reproduce Figure 13, run the script `barplot2.m`.