Università degli Studi di Catania
Dipartimento di Matematica e Informatica

SYNTHETIC AND SYSTEMS BIOLOGY OF GENOME-SCALE MODELS: UNCOVERING MINIMAL METABOLISM AND REGULATORY PRINCIPLES

PhD Thesis

PhD Candidate
Giorgio Jansen

Advisor
Prof. Giuseppe Nicosia
Co-Advisor
Prof. Stephen G. Oliver

PhD Course in Mathematics and Computer Science - XXXII Cycle

Contents

Contents ii

List of Figures iv

List of Tables vii

List of Algorithms viii

1 Introduction 1

2 Metabolic networks and Genome-Scale Models 7

2.1 The Metabolic Networks 7

2.1.1 Medium definition 10

2.1.2 Flux Balance Analysis 11

2.1.3 Flux Variability Analysis 12

2.1.4 Parsimonious Flux Balance Analysis 12

2.2 Sensitivity and Robustness of Strains 13

2.2.1 Sensitivity Analysis 13

2.2.2 Robustness Analysis 14

2.3 Genome inclusion 16

2.4 Redirector 19

2.5 Interactions Network 20

3 Metabolic Engineering and Pathways Design by Computational Systems Biology 22

3.1 Introduction 22

3.2 Multi-Objective Optimization and Analysis 24
3.2.1 Multi-Objective Optimization 24
3.2.2 Adaptation to the Redirector Framework 29
3.2.3 Medium Optimization 30
3.2.4 Robustness Analysis 31
3.3 Ethanol Production 32
3.3.1 Gene KO optimization for Ethanol Production 33
3.3.2 Enzyme Regulation in *S. cerevisiae* 36
3.3.3 Ethanol production without Essential Genes in *E. coli* and *S.cerevisiae.* 38
3.4 Lactate Production 44
3.4.1 Multi Stage Optimization 45
3.5 Future Improvements 49

4 Minimal Metabolism Design by Computational Synthetic Biology 53
4.1 The minimal genome 53
4.1.1 The proposed computational approach 55
4.2 Results for *Saccharomyces cerevisiae* 56
4.2.1 Considering the network of genetic interactions 61
4.2.2 Synthetic Double Deletions Prediction 63
4.3 Extending to other *Saccharomycetales* 65
4.4 Further extension to broad-spectrum models 67
4.4.1 Comparative analysis 68
4.5 Methods 72
4.5.1 Models 72
4.5.2 Minimal Media 72
4.5.3 Algorithm 75
4.5.4 Frequency and Redundancy Analysis 79
4.5.5 BLASTp 80
4.5.6 Pathway-oriented Robustness Analysis 80
4.5.7 Complex Network Analysis 81
4.5.8 Code implementation and machine setting 81
4.6 Future Improvements 82

5 Conclusions **85**

Appendix **88**

Bibliography **102**

List of Figures

2.1 Central metabolism in JCVI-syn3A. Figure taken from [1]. 9
2.2 Affected reactions in the central metabolism of *S.cerevisiae* after a single gene knockout, as predicted by *pFBA*. 18

3.1 Results for optimization of ethanol production and biomass formation in *E. coli*, anaerobic condition. 36
3.2 Normalized Pareto Fronts of models optimizations for ethanol production and biomass formation in various *Prokaryotic* organisms. . 38
3.3 Normalized Pareto Fronts of models optimizations for ethanol production and biomass formation in various *Eukaryotic* organisms. . . 39
3.4 Results for optimization of ethanol production and biomass formation for *S. cerevisiae*. 40
3.5 Results of the Gene Knock-Out Constrained Multi-Objective Optimization for different metabolic models and conditions. 41
3.6 Results of the optimization procedure in different settings. 45
3.7 Simple workflow of the multi-stage optimization process. 46
3.8 The various stage of optimization. At the end of the first two steps a notable strain is selected as the base of the subsequent optimization. 52

4.1 Frequencies of MNs active genes; two different clusters are present, for aerobic and anaerobic conditions. 58
4.2 Fraction frequency of MNs shared active reactions through a pairwise comparison. Considering the networks in the same conditions the shared reactions are 60 to 90%, while comparing networks in different conditions the fraction is 30 to 50%. 59

4.3 Minimal Metabolic Networks as described by basic measures from the complex network theory. 60
4.4 Evaluation of the Latora-Marchiori efficiency of the interactions networks with the same number of genes as the mandatory genes removed. The network with a higher number of mandatory genes removed have a sensibly lower efficiency, on average. 62
4.5 Distributions of the degree and betweenness of the nodes in the interaction network. 63
4.6 Cumulative distribution of the nodes degree in the network, showing different curves for nodes with different frequency. 64
4.7 Power laws in the degree probability plot. 65
4.8 Comparison of prediction for synthetic double deletions in the MNs. 66
4.9 Results comparison for the 4 Fungi species considered. 67
4.10 BLASTp comparison of the mandatory genes in the models of *Fungi*. 68
4.11 Global comparison of all the models used in minimization of the simulated genome. 69
4.12 Comparison using the BLASTp search using the JCVI-sn3.0 genes as a query. 70

A.1 Pathway oriented Sensitivity Analysis for the *S.cerevisiae* model . . 89
A.2 Clustering results on the observed Pareto Front of the optimization of ethanol production and biomass formation in *E. coli*, anaerobic condition. 90
A.3 Pareto optimal ethanol production and feasible solutions as an observed function of the knock-out cost. 91
A.4 Frequencies of MNs active genes with a relaxed constraint on the growth rate reduction; two different clusters are present, for aerobic and anaerobic conditions. 93
A.5 For each condition the MNs active genes number using a Growth rate threshold of 1% or 10% is reported. 95
A.6 Comparison of prediction for synthetic double deletions in the MNs. 96
A.7 Improved predictions on the synthetic double deletions. 98
A.8 Redundancy and Robustness measures of the Minimal Networks. . . 99

A.9 Impact of the external compounds on the Growth Rate predictions for the Minimal Networks. 100
A.10 Validation of the results for the genome minimization with other minimal genome studies. 101

List of Tables

3.1 Maximization of Ethanol and Biomass production for *E. coli*. 37
3.2 Number of essential genes and lethal gene pairs present in the genome-scale metabolic models. 37
3.3 Solutions with low number of gene sets knocked out obtained in the optimization. 47
3.4 Some selected solutions with low number of gene sets regulations obtained by the second stage fine tuning redirector simulation. . . . 48

4.1 Models used in this study with the number of genes simulated in each of them and characteristics of Minimal Metabolic Networks found by the algorithm. 71
4.2 Media composition and Bounds. 84

A.1 Maximization of Ethanol and Biomass production for *S. cerevisiae* using the redirector approach for enzymes regulations with the iMM904 genome-scale model. 88
A.2 Maximization of Ethanol and Biomass in the selected Pareto optimal strains. 92
A.3 The Mandatory genes for all the MNs of S.cerevisiae in the 7 different external conditions. 94
A.4 Quantitative analysis of the genes in the MNs using Functional Category and Compartment annotations 97

List of Algorithms

1 MOME Optimization Algorithm 26
2 Minimal Media Algorithm 74
3 Evolutionary Algorithm 76
4 Genetic Operator 77
5 Sorting Population Function 77
6 Aging Function 78

Chapter 1

Introduction

The complexity of biological systems has always been, despite the latest an fundamental steps that has been taken in the last two decades, conceals a variety of aspect that are still far to be understood properly.

Even the most simple unicellular organisms, whose comprehension would allow a plethora of chances in biotechnology, present puzzling results and an intrinsic difficulty, especially in their genome and how this is related to the actual behaviour of the organism.[2]

In the last decades the interest over these topics increased exponentially, on one hand the industrial applications, such as those involving the production of specific chemicals from microorganisms able to synthesize products for both pharmaceutical interest and strict production processes improvement. On the other hand, a deeper comprehension of the basic mechanisms of life in its simpler forms could potentially have implication in a variety of other fields, from the cancer therapy, to the development of new therapy tackling infectious disease or parasites affecting humans.

Over the year two fields, in particular, have emerged as tools to be used to systematize and reveal some of these mysteries of the nature. The first is the *Systems Biology*, the second is the *Synthetic Biology*.

The number of recent successes in these fields[3] seems indeed to shake off all but little doubt that in the near future the latter will be standard practice in the production of therapeutic drugs [4], renewable bio-materials [5] and bio-

fuels [6, 7]. These advancements in turn led to the first studies in the field of systems metabolic engineering, where production strains are systematically built and improved through a combination of systems biology, synthetic biology, and evolutionary engineering .

Systems Biology is the branch aimed at the comprehension of the organization of the cells and its possible functional description and schematisation; the study of the cell is usually conducted at a large scale, and it is primarily aimed at the construction of logical, mathematical or computational models that are able to properly represent this organization. The constituents of such a study are for examples the enzymes, the genome, the metabolome, the proteome, but more importantly, all the functional interactions between all these "blocks", such as the chemical pathways happening in the cells and their relationship with the genome and, more generally, the relationship between the genotype and the phenotype. The ideal model realized using the systems biology tool would be able to obtain accurate predictions at different scales. One must not see the systems biology, however, as a mere description of the cell insight that can go as far as our knowledge has already arrived. Conversely, systems biology tools might give a larger picture of complexity, leading to an understanding of the phenomena behind more specific observations, and to gap-filling procedures and discoveries that would be impossible otherwise. Clearly, such an approach requires a lot of measurements and data to obtain a reconstruction of the cell's insight. One of the best example of the systems biology advances are the *genome-scale models*, representing the metabolic networks of a cell and that are the main tool that I have used in my research and that I will discuss throughout this thesis. The metabolism of a cell is the set of all the chemical reactions happening inside it. Clearly a thorough model shall include a great number of metabolites and reactions ($\sim 10^3$ or even larger); such a network is extremely complicated to analyse without the schematic approach typical of the systems biology, but I will return to this later on in the work.

Synthetic Biology is a concept related to disparate research directions, since the notion of synthetic itself is not unique. One of those is the "protocell" creation, that is basically aimed at the construction of approximations of fully functioning cells and the contemporary understanding of basic principles regarding the origin of

life itself; another one, that arose in the last years as probably the main meaning, is the engineering of the cells to obtain new strains with new abilities.[8] The engineering can also be regarded as a study aimed at the construction of fully functioning genome to be inserted in cells to control their processes and obtain a specific ability of interest.

In the bigger ensemble of synthetic biology, the *minimal cell* problem finds then its spot, somehow as a compromise between the two approaches described before.

The concept of minimal cell was firstly introduced in the 1930s, when physicist Max Delbrück founded the American Phage Group,that reasoned on a reductionist approach for the understanding of the basic principles of life, that can be can be compared to the one conducted by the physicists for the hydrogen atom, the simplest chemical element in which it is possible to find common mechanisms shared by all the elements[9]. The same could be said for the cells, making the assumption that the simplest cell would have all the necessary and sufficient elements able to define life and that could be generalized to all the other cells of all the other organisms.

At the same time, the minimal cell would be the simplest blueprint of cell and it could be expanded by the subsequent addition of several genes, pathways or biological elements in general, to become a more complex cell. It is not immediate (actually it is quite far to be so) that such a cell exist for real and that can be recreated or synthesized, but the concept itself is extremely interesting and fascinating.

In my work I will explore the possible applications of the simple mathematical models of metabolic networks defined as *Linear Programming* Problems, and the subsequent *Flux Balance Analysis*[10], in its various forms, as a tool for tackling some of the problems posed by Systems and Synthetic Biology.

In the first chapter I will give a general overview of the metabolic networks and of the genome-scale models in which they are defined; this is the common basis that I have been using throughout my work. Analogously, I then present the flux balance analysis and two variants that permit to obtain a set of fluxes for each chemical reaction in the metabolic network, basically giving the actual prediction of the behaviour of the cell in a steady state corresponding to the exponential growth[10]. I will also introduce other general concepts used in my analyses, such

as the robustness of strains and the sensitivity of a network.

A small following section is dedicated to the inclusion of genome in the models, explaining the way in which it relates to the metabolic network, what is its role, and how changing this genome affects the prediction given by the Flux Balance Analysis.

Finally, I introduce the genetic interaction networks, that I will use in the last chapter of the thesis, and that are a powerful tool for the description and the understanding of the biological phenomena, and that I tried to include in the explanation of some of the results on the minimal cell problem.

Not all of the methods applied to obtain the results are described in the first chapter, as I also included some specific topics in the relative sections.

The second chapter is dedicated to a systematic study for the optimization of chemical production in engineered organisms using the genome-scale models for prediction of optimal engineered strains, with a focus first on the production of Ethanol and then on the production of Lactate. The key point in this chapter is the genetic algorithms that has been used to explore the solution space defined by all the possible strains that can be constructed through different manipulations from the wild type models.

By building upon recent achievements of *in silico* driven engineering of bacteria strains [11, 12, 5], extensive *in silico* optimization and analyses are performed for the production of ethanol as output of the metabolic network of the engineered cell. In fact, the attention given to optimization algorithms for the design of microbial strains overproducing metabolites of interest has drastically increased in the last few years [13]. However, the intrinsic complexity of biological systems and organisms, as I stated before, makes of paramount importance the design of mathematical and computational approaches to fully exploit the potential of this discipline [14].

Alongside the classical simulations involving the gene knockouts inside the models and the iterative selection of the most important knockouts for the production of the chemicals selected as the objective function of the optimization, I also used other two different ways of intervention on the metabolic networks. The first is the *redirector*[12], that simulate the over- and under- expressions of the genes through a change in the objective function, i.e. "redirecting" the optimization to a dif-

ferent objective that should represent the changed genetic expression within it. The second is the optimization of the simulated *medium* to further optimize the production of chemicals.

I combined these two standalone approaches with the classical knockouts one, to obtain a multi-stage procedure that tries to minimize the variations in the single approaches to obtain strains with an optimal phenotype.

The next chapter is dedicated to an application of the genome-scale models to the minimal cell problem. The main algorithmic pipeline that I developed for the task is based on an evolutionary algorithm[15] that iteratively reduce the size of the genome by excluding the genes from the model, ensuring that the predicted growth rate doesn't go below a strict threshold; information on the essential genes were also included to increment the likeliness of the results.

At the best of my knowledge, this is the first time that an approach like this is used systematically with the genome-scale models, even though there has been recently a close attempt of developing an automatic procedure based on the whole-cell model of *M.genitalium*.[16]

The procedure, consistently with the issues of the minimal cells problem, returns a variety of "minimal" genomes. I performed an extensive analysis on all these possible configurations, trying to make emerge some common properties and in particular analysing the genes that are more or less involved in the procedures, unveiling a less or more important role of the reactions catalysed by the corresponding enzymes in the metabolic networks, respectively. In particular, a small set of *mandatory* genes, i.e. the genes that are (almost) never excluded from the minimal genomes, emerged as one of the most important properties of these minimal network.

Other analyses on the network were also performed using the complex network theory on the resulting genetic interactions network; namely, I used basic properties of networks, such as the *Latora-Marchiori efficiency*[17], to establish another important property of the mandatory genes as genes affecting the efficiency of the networks (or other measures) more than other genes, a phenomenon that could be used also for predictions of important genes with the exclusive knowledge of the interaction network, as it was also hinted by some previous work on the essential genes.[18]

To summarize, in this work I present applications of the genome-scale models for two different problems, apparently distant one from each other, but also linked; the network minimisation to effect the channelling of metabolites to desired products is in fact a highly desirable product, although far from a full realization. The genetic and evolutionary algorithms developed for the tasks are linked with the models and every point is evaluated through the Flux Balance Analysis, which, despite its simplicity and its limitations, is a powerful method, and it also allows a great number of evaluations in a reasonable amount of time, a feature that I believe to be fundamental in large scale analyses that does not have pre-determined constraints on the solutions.

In general, the advancing of the computational tools and of the knowledge of biological systems and organisms that allows the refinement of the models[19], will increasingly lead the community to an increased utilization of the computational biology tools (not only the genome-scale models) for a more and more wide variety of biological and biochemical problems in a strict cooperation, that I believe being fundamental for further progresses in science.

Chapter 2

Metabolic networks and Genome-Scale Models

In this Chapter I will introduce some basics and common concepts on the genome-scale models and that will be widely used throughout the thesis.

2.1 The Metabolic Networks

The definition of a mathematical model for the description of biological phenomena is a challenging task to be addressed by the biologists, computer scientists and mathematicians. There are many possible approaches to the issue, and they usually present features to be decided as a trade-off between complexity within the mode, the time required for a single computation using it and the extension of the model itself. In the following I will describe a simple approach to this problem that have been extensively used in the last decades and that is the foundation of the studies I present.

Instead of simulating the entirety of the cell it describes only the *metabolism*, which is defined as the entire set of chemical reactions happening inside every living cell; the reactions contribute to the sustaining of all the basic functions of the cells. A linked series of reactions is called a *metabolic pathway*, that is then a sequence of transformations in which a compound is transformed into another, acting as the substrate of the sequent step; other molecules might be used or pro-

duced by the reactions as cofactor or "waste" respectively. All the reactants and products are also known as *metabolites*. All the pathways can be divided into two macro-categories of opposing streams of reactions: the *catabolic* and *anabolic* (or *biosynthetic*) pathways. The first includes the pathways releasing energy or producing small useful molecules as building blocks for other processes; the anabolic pathways instead produce more complex molecules such as amino acids, proteins or nucleic acids from simpler metabolites.

The entirety of the reactions define the *metabolic network*, with the metabolites as nodes connected by hyperedges, the reactions. An alternative formulation is to consider the network as a directed bipartite graph in which both the metabolites and the reactions are nodes, and the directed edges link the reactants to the corresponding reaction and the reaction to the products. The resulting networks have a large-scale structure and interesting properties, shared among different organisms, such as the topological scaling, and common to the *scale-free networks*[20]. Another characteristic property of the network is its small-world behaviour; almost every couple of metabolites is connected by a path composed of a much reduced number of steps (usually less than 7), if compared to the number of total metabolites. This is due to the presence of the so-called *currency* metabolites, i.e. metabolites such as oxygen or hydrogen, that appear in a great number of reactions, but also more complex molecules, such as the ATP, have similar presence in the network. These metabolites create a lot of "bridges" between distant pathways and make extremely difficult the analysis of the pathways from the raw network. It is almost impossible to reconstruct all the biologically significant pathways from the network without a pre-existent knowledge of the biochemical behaviour of the organisms.

When the metabolic network has been defined (with enormous efforts from the biochemists) it requires a mathematical representation as simple as possible to be used and evaluated. The usual tool used for this is the *Stoichiometric matrix* S, where all the stoichiometric coefficients of all the metabolic chemical reactions are included. Every single row of the matrix represents a different metabolite, while every column is a representation of a single chemical reaction. It follows that the generic element $s_{i,j}$ is the stoichiometric coefficient with which the metabolite i appears in the reaction j. Clearly the majority of the elements in a row are equal

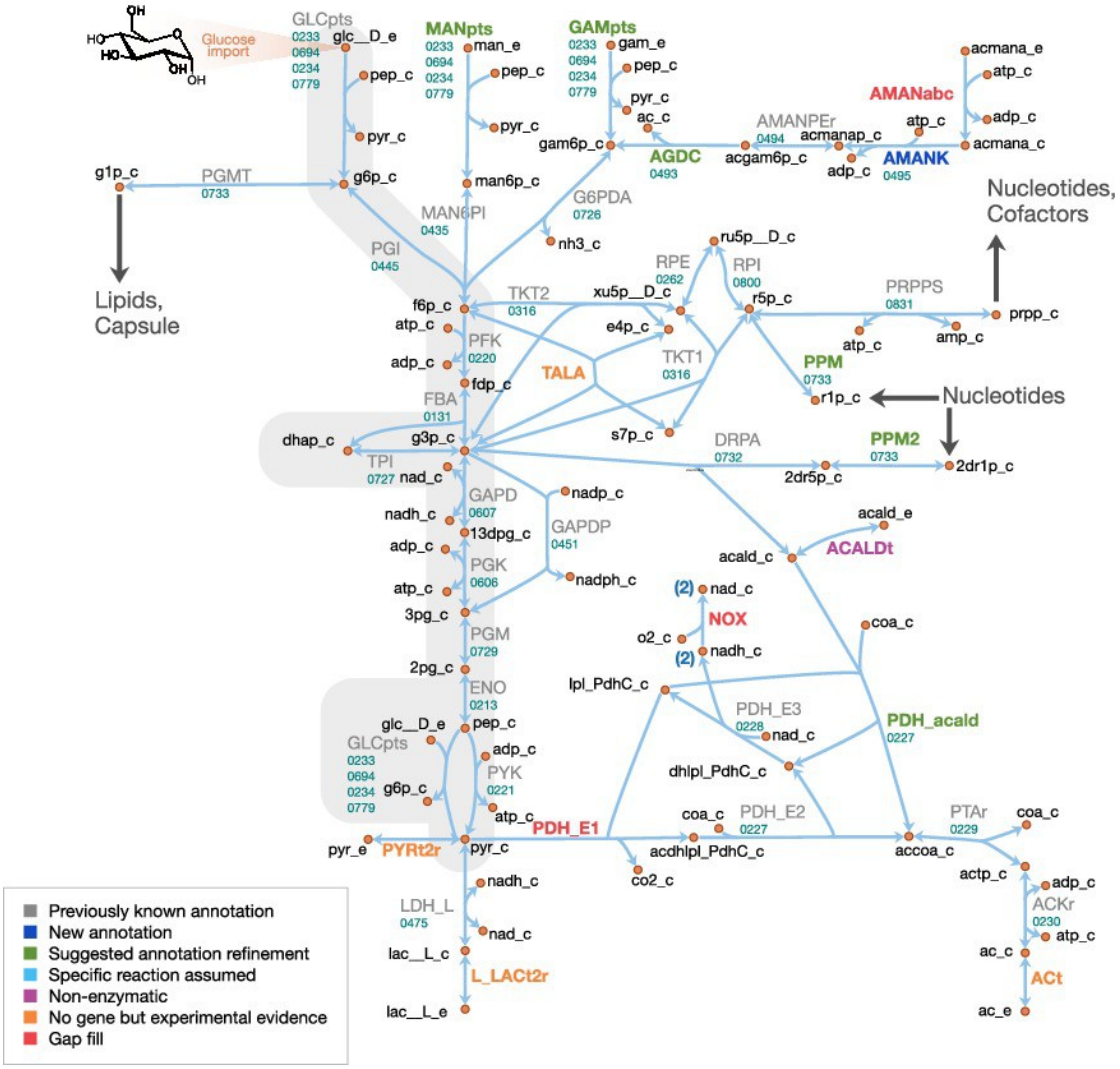


Figure 2.1: Central metabolism in JCVI-syn3A. Figure taken from [1].

to zero, because a reaction usually involves only a few reactants and products, hence S is a *sparse* matrix, i.e. most of its elements are equal to zero. Every coefficient, that in the stoichiometric laws is always positive, can be stored in the matrix with a positive or negative sign if it is a product or a substrate of the reaction, respectively.

Once the matrix S has been defined, it constitutes a set of constraints for a

simple linear problem that can be expressed as

$$\sum_{j=1}^{n} s_{ij} v_j = 0, \quad \forall i \in 1, \ldots, m \qquad \Longleftrightarrow \qquad S\mathbf{v} = 0, \tag{2.1}$$

where m, n are the number of metabolites and reactions, i.e. the dimensions of the matrix S, and $\mathbf{v} = (v_1, \ldots, v_n)^T$ is a vector representing the *fluxes* through every reaction.

Each flux is usually bounded to an interval $v_j \in [lb_j, ub_j]$ defining its admissible values; usually these bounds are such that $lb_j \leq 0, ub_j > 0$ and constitutes two vectors lb, ub. One consequence of the lower bound value is the reversibility of the reaction: if $lb_j < 0$, the reaction flux might have a reverse behaviour, exchanging the role of reactants and products.

2.1.1 Medium definition

Another important consequence of the lower bound value is for the exchange reactions defining the maximum uptake rates of a particular nutrient from the environment. If the null value is not included in the interval, the flux through the reaction is forced to assume a specific value, for example to recreate a specific experimental condition in which there is a known uptake or secretion of one or more molecules.

Another important setting of the model is the representation of the growth medium. In the metabolic network, the exchange of metabolites with the external environment is simulated with the presence of a number of exchange reaction. The exchange reactions, e.g. for the Glucose is:

$$D - glucose_e \leftrightharpoons$$

The sense of reactions is so from the immediate outer space of the cell (the extracellular region included in the model) to the external environment. In order to simulate the uptake of some compounds the flux value has so to be negative. The maximum possible compounds uptake is so regulated by the lower bounds of the correspondent reactions, that have to be properly set. If a lower bound is set to 0, thus the flux will be non-negative, there can only be a secretion of the selected

compound to the environment.

2.1.2 Flux Balance Analysis

Hence, a set of equations and constraints representing the mass balance of each metabolite included in the metabolic network is the obtained. The equations and the inequalities define a solution space, i.e. the set of vectors $\mathbf{v} \in \mathbb{R}^n$ that satisfies all the equations. Every configuration of fluxes is *feasible* in terms of mass conservation, but it does not necessarily mean that is a configuration feasible as a representation of an actual metabolism of a living cell; the null vector satisfies the equations but represent no activity for any reaction. To choose among the set of vectors, an optimization problem is defined:

$$\begin{aligned} \min \quad & \sum_{j=1}^{n} c_j v_j = \mathbf{c}^T \mathbf{v} \\ s.t. \quad & S\mathbf{v} = 0 \\ & lb \leq \mathbf{v} \leq ub \end{aligned} \tag{2.2}$$

where $\mathbf{c}$ is the vector of coefficient of a linear combination defining an objective function for the problem. The objective function must then be set to force the optimal fluxes to lie on similar value to the ones from the real cells. This is a delicate task, usually requiring a precise study on the metabolic network and a deep knowledge of the metabolome and reactome of the cell.

The most common approach to the task is, rather than defining an objective function as a mere linear combination of many fluxes in the cell, to construct an artificial reaction inside the network that simulate the production of biomass, i.e. the growth of the cell, using a variety of internal compounds. The precursors of this reaction are then all required for the growth of the cell, and their production in turn requires many other reactions to be active. A proper and correct definition of the biomass reaction is indeed a fundamental and delicate task to be addressed when constructing a metabolic network.

2.1.3 Flux Variability Analysis

Due to the complexity of the defined solution space, there is potentially an enormous number of solutions, i.e. points of the solutions space, that have the same flux through the biomass formation reaction. All this points are located on a hyperedge of the polytope of the solution space, and the usual FBA does not provide a criterion for the chose among these solutions. To first evaluate the intervals of the fluxes through the single reactions that guarantee the optimal value of the objective function, an approach called *Flux Variability Analysis (FVA)*[21] is used. Assuming that an FBA problem (2.2) has been solved, maximizing the flux v_{bio} related to the biomass reaction and that f^* is this optimal value, the FVA problem is defined as

$$\begin{aligned} \min / \max \quad & v_j \\ s.t. \quad & S\mathbf{v} = 0 \\ & lb_i \leq v_i \leq ub_i, \quad \forall i \\ & v_{bio} = f^* \end{aligned} \tag{2.3}$$

where the last simple constraint ensure that the optimal value of the Growth Rate is reached. The result of the optimization problem are the corresponding minimum or maximum value that the reaction's flux v_j can assume ensuring the optimal value of the biomass reaction, i.e. the variability of the j-th variable on the hyperedge of the solutions space coresponding to the optimal value f^*.

2.1.4 Parsimonious Flux Balance Analysis

Using the *FVA* it is possible to show the great variability of the fluxes in the *FBA* prediction. It is not uncommon for *FBA* to return a distribution of fluxes where some of the values are equals to the upper bounds, for example if there is a dummy cycle in the model that can be activated without affecting the other reactions of the model. To limit this kind of problems in the results I also used a more fine approach called *parsimonious Flux Balance Analysis*[22]. The simple assumption behind this method is that a cell, under exponential growth, tends to take a behaviour that requires the lowest flux through the metabolic network, this because of the limited quantity of enzymes catalysing reactions that are resent in a

cell. The least "expensive" setting is then more likely than the one with unlimited fluxes of reactions happening at the same time.[22] To simulate this, a dual step approach is again used, similarly to the *FVA*. After the problem (2.2) is solved, a *Quadratic Programming* problem is considered, namely

$$\begin{aligned} \min \quad & \sum_i v_i^2 \\ s.t. \quad & S\mathbf{v} = 0 \\ & lb_i \leq v_i \leq ub_i, \quad \forall i \\ & v_{bio} = f^* \end{aligned} \tag{2.4}$$

A method solving this problem returns the fluxes distribution with the lowest sum of squares, i.e. the lowest flux through the network. This distribution is then more reliable when the fluxes values have to be considered, and I used it for all the analysis involving the fluxes of reactions that are not the objectives of the optimization.

2.2 Sensitivity and Robustness of Strains

2.2.1 Sensitivity Analysis

If the knockout string y has been set, the output(s) of a molecular machine depends on the inputs, i.e. nutrient metabolites. The sensitivity analysis allows to analyse which inputs have a strong influence on the output(s). The Morris method to perform this analysis[23], it consists of an experimental plan composed of multiple randomised one-at-a-time (OAT) designs. For each OAT design, an input is changed and the corresponding change in the output is measured. The Morris method can determine if the effect of the input factor x_i on the output v is negligible (v is the vector of the fluxes), linear and additive, nonlinear or involved in interactions with other input factors $\mathrm{x}_{\sim i}$.

The input factor space is *discretised* and the possible input factor values will be restricted to be inside a regular k-dimensional p-level grid, where p is the number of *levels* of the design. The elementary effect of a given value x_i is defined as a

finite difference derivative approximation:

$$eei(x) = \frac{[g(x_1, x_2, ..., x_{i-1}, x_i + \Delta, x_{i+1}, ..., x_k) - g(x)]}{\Delta} \tag{2.5}$$

for any x_i between 0 and $1 - \Delta$, where $x \in \{0, 1/(p-1), 2/(p-1), \ldots, 1\}$, and Δ is a predetermined multiple of $1/(p-1)$. The influence of x_i is then evaluated by computing several elementary effects at randomly selected values of x_i and $x_{\sim i}$. If all samples of the elementary effect of the i^{th} input factor are zero, then x_i doesn't have any effect on the output v, the sample mean and standard deviation will both be zero. If all elementary effects have the same value, then v is a linear function of x_i. The standard deviation of the elementary effects will then of course be zero. For more complex interactions, due to interactions between factors and nonlinearity, Morris states that: high mean indicates a factor with an important overall influence on the output; high standard deviation indicates that either the factor is interacting with other factors or the factor has nonlinear effects on the output. In my work I used the sensitivity analysis to establish the performance of the metabolic models under small changes in the pathways, to highlight the most sensitive ones. See Figure A.1 for an example of the sensitivity analysis of different pathways.

2.2.2 Robustness Analysis

Various *Robustness Analysis* (RA) routines[24] have been discussed in the past. Briefly, RA indexes were used for the *in silico* test and validation of the strains obtained in the optimization phase; in the specific I investigate how notable Pareto optimal (or approximated Pareto-optimal) individuals adapt to "small" variations that inevitably occur either in within the considered bacterium itself, or in the environment surrounding the latter. Fully analyses of the robustness of each notable strain assigning it two different *robustness indexes*: i) *Local Robustness* (LR_i), associated with each reaction of the metabolic network considered; and ii) *Global Robustness* (GR), which evaluates the robustness of a strain under a global point of view. Consider a strain and let v^* be the distribution flux vector associated with it, computed by applying FBA modelling to the strain. Let ϕ be a strain-

dependent function (e.g. production rate of a particular metabolite). Let $\sigma \in [0,1]$ be a perturbation strength and $\epsilon \in [0,1]$ a threshold value. Let $\Delta_i \sim \mathcal{N}(0, \sigma v_i^*)$ be a normally distributed random variable and $V^i = (0, \ldots, \Delta_i, \ldots, 0)$. The i^{th} local robustness index of parameter σ and ϵ for the considered strain are defined as:

$$LR_i^{\sigma,\epsilon} = P\left(\left|\phi\left(v^*\right) - \phi\left(v^* + V^i\right)\right| \leq \epsilon \left|\phi\left(v^*\right)\right|\right)$$

which is the probability that a perturbation of percentage strength σ for the flux of the i^{th} model reaction will produce a small perturbation on the value of ϕ. The definition for the GR index of a strain is a straight forward extension to that of local robustness. Let $V = (\Delta_1, \ldots, \Delta_n)$. The global robustness index of a strain is then defined as:

$$GR^{\sigma,\epsilon} = P\left(\left|\phi\left(v^*\right) - \phi\left(v^* + V\right)\right| \leq \epsilon \left|\phi\left(v^*\right)\right|\right).$$

In the implementation, the LR_i and GR indexes are computed using a Monte-Carlo approach used to estimate the probabilities in the definitions.

Since the string y and the inputs have been set, a further analysis permit to obtain the robustness of the molecular machine able to perform a specific task (e.g. maximising acetate and biomass). Indeed, the molecular machine (or strain) can be subject to noise that can be received either from the environment or from the inside. To evaluate the robustness, I consider a basic principle that consists in applying a stochastic noise σ to the system Ψ. This generates a trial sample τ (assuming that the noise is defined by a random distribution) and a set T_τ of trial samples τ (to obtain statistically meaningful). Each element τ of the set T_τ is considered robust to the perturbation for the stochastic noise σ and the given property ϕ if $|\phi(\Psi) - \phi(\tau)| \leq \delta$, where Ψ is the *reference system*, ϕ is a *metric* (or property), τ is a *trial sample* of the set T_τ, and δ is a *threshold*. The robustness of a system Ψ is defined as the number of robust trials of T_τ with respect to the total number of trials $|T|$.

There are two types of robustness analysis: Global Robustness (GR) and Local Robustness (LR). In the first case, the input variables are perturbed simultaneously, so as to evaluate the overall fragility of the molecular machine; while in the second case the perturbation is carried out for one input at a time, thus obtaining

a robustness value for each input [25]. and selecting the minimum as reference value. I also implemented the analysis described in the work of Hafner et al. [26], where the authors implement a procedure that calculates the normalised volume (R) occupied by those parameters such that a system maintains the desired characteristics.

In my work I used the robustness analysis as a tool to evaluate the resulting strains, in particular referring to the minimal metabolic networks (see Figure A.8).

2.3 Genome inclusion

In this paragraph I will discuss the inclusion of the genomic information in the metabolic networks, to construct the genome-scale models to use. After the definition of a metabolic network it is important to have information on the genes related to some of them; the genes encoding enzymes and transporters can be included due to their direct role in the reactions simulated in the metabolism. Other type of genes can not unfortunately be immediately reproduced in a metabolic model.

The genes are included as a set of "rules" for each reaction, every rule is simply a logical proposition composed of variables, representing the genes, as atomic statements. The variable will have a true or false value respectively if the genes is simulated as present or not in the model. The rules can include brackets and the logical connectors *AND*, *OR*, that formalize the case of cofactors or isoenzymes, respectively. A reaction's rule represents the condition that must be satisfied for the reaction to take place; if for example a rule is composed of just one gene A, that reaction will require its presence to happen; clearly there will be reactions with complex rules involving several genes, and reactions with empty rules, i.e. they might happen in any case in the model, this is the case of the spontaneous reactions.

Impact of gene KnockOuts

The main aim of the programs described is to find an optimal solution, defined as a set of genes to be knocked out, which change the prediction of the fluxes distribution of the chemical reactions in the model. The deletion of a gene influences the

model in two different ways. The first is the elimination of one (or more) constraint derived from the matrix S. Through the set of rules map, sometimes referred as the *gene-protein-reaction (GPR)* map, it is possible to point out which reactions are turned off because of the manipulation by evaluating the corresponding logical rules defined for it. For each reaction that has to be excluded from the model, the rows in S corresponding to these reactions, which define the constraints in the model, can be deleted from the matrix at all. The second influence in the model is that there are less constrains from the upper and lower bound, because the ones of the deleted reactions are no more considered in the model.

The global result of such a manipulation at the genome level in the model is a redefinition of the feasible space, and hence the objective functions have to be recalculated in it. Once again, if the biomass function has a value greater than 0, the solution is feasible, otherwise will not be considered.

In Figure 2.2 an example of how the fluxes predicted by the *pFBA* vary after a single knockout in the *S.cerevisiae* metabolic network is reported. In this specific case the Growth rate is reduced, and there are reactions, e.g. the ATP synthase that have a flux reduced by $\sim 20\%$.

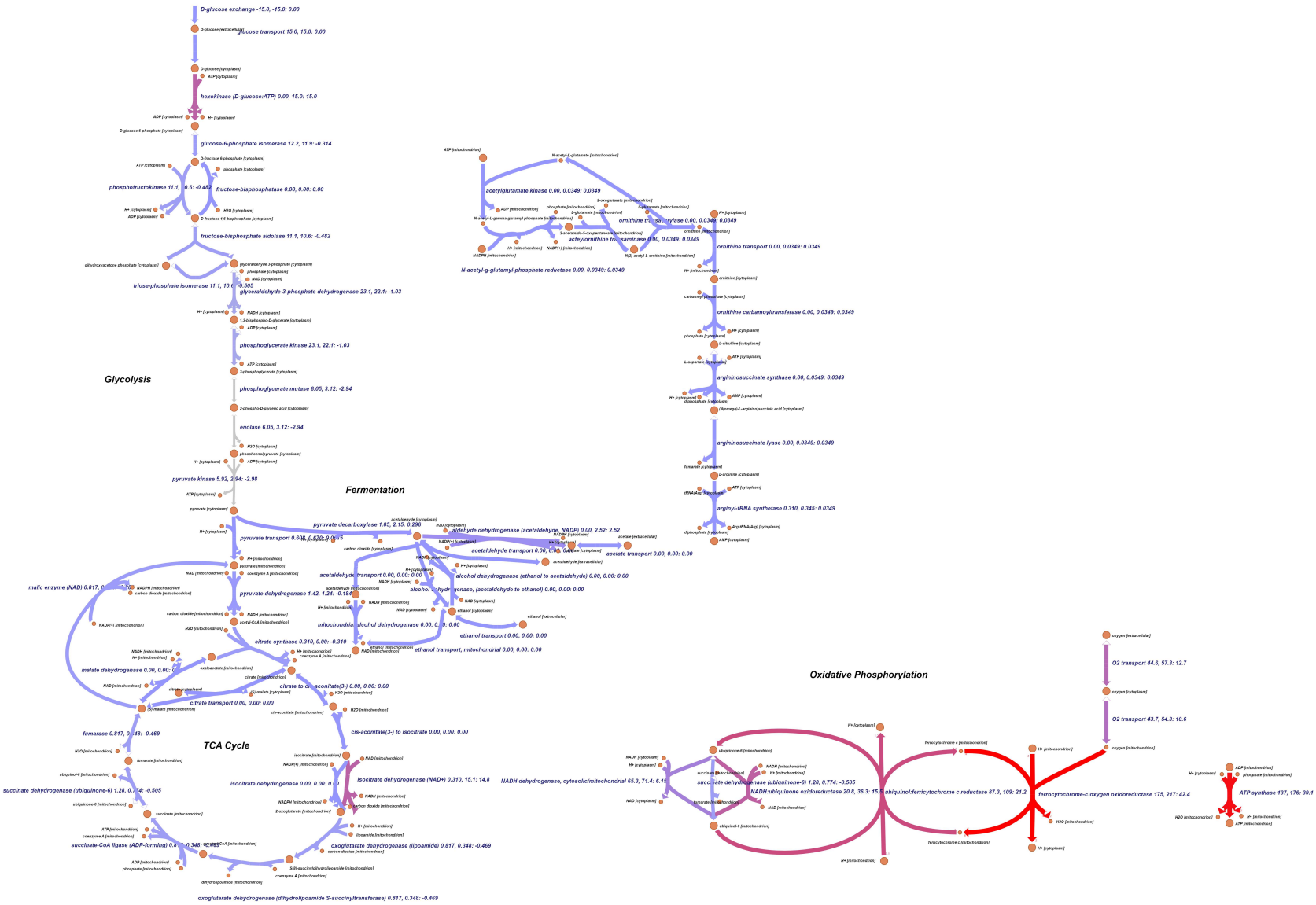


Figure 2.2: Affected reactions in the central metabolism of *S.cerevisiae* after a single gene knockout, as predicted by *pFBA*. The red edges are more strongly affected.

2.4 Redirector

The other kind of simulation used for the predictions of cells' phenotype is the *redirector*[12]. The main idea of this simulations is to iteratively change the objective function used for the optimization in flux balance analysis to represent, in a simple way, the over- and under- expression of some of the genes. While the function is usually set as the classical biomass flux, by changing it, the behaviour of the network change too, redirecting the metabolic fluxes to an other objective function, no more focused only on the biomass reaction. The mathematical formulation of the problem is the following. Let's assume that the vector $y \in \mathbb{R}^g$ express the level of expression of the genes (usually as coefficients in the $[-1, +1]$ interval). If a coefficient is equal to 0 then the gene is normally expressed, if positive, the gene is overexpressed, if negative the gene is underexpressed. When a coefficient is not equal to 0, than all the reactions related to that gene have to be modified. Thus a vector of weights is defined: $w \in \mathbb{R}^n$. The values are so set $w_i = sign(v_i^{WT})y_j$, where j is the index of the gene set related to the ith reaction. The v^{WT} is the vector containing the reaction fluxes of the wild type simulation, and the sign function is needed because the attenuation or the amplification has to be in the right direction, following the flux. If the reaction has not any gene relations, the weight is set to 0. The maximization problem becomes then:

$$\begin{aligned} \max \quad & \gamma v_{biomass} + \sum_{i=1}^{n} w_i v_i \\ s.t. \quad & Sv = 0 \\ & l_j \leq v_j \leq u_j, \quad j \in \{1, \ldots, n\} \end{aligned} \tag{2.6}$$

where γ is the biomass reaction weight, that is interactively set during the genetic algorithm depending of the targets value found. The lower and upper bound of the fluxes are also adjusted during the redirection, in order to ensure that the reaction fluxes don't change sign from the wild type case. So if for example a reaction has a wild type fluxes positive and has to be attenuated, the coefficient will be negative, but the lower bound will be set equal to 0 such that the new prediction will keep the flux non-negative.

I used two different approaches to the redirector framework, the first is called *flat redirector*, and the second *fine tuning redirector*. The difference between each other is the set of values which the coefficients can assume. In the flat redirector, the coefficients y_j can be one of the values $\{-1, 0-1\}$.

The second approach, the *fine tuning redirector*, has a slightly different formulation. The values are $\{-1, -0.9, \ldots, -0.1, 0, 0.1, \ldots, 0.9, 1\}$ are in this case the admitted coefficients. The variables are so more accurate than the flat ones. It is important to notice that this formulation include within itself the flat redirector, although this does not mean that the flat is useless, because the binary variations may allow for example a faster convergence than the fine tuning approach. See Chapter 3 for further comments on the redirector analysis.

2.5 Interactions Network

For further analysis on the genes in the genome scale models, I considered also the network formed of the genetic interactions. The genetic interactions have long been studied for identifying and unveiling the complex relationships, at many levels, that exist between the genes of an organism[27]. These relationships could be of different intensity and nature, depending on which functionality they affect. They are usually evaluated comparing the behaviour of a double mutant to a neutral state; they are then also used as a tool for the classification of genes of unknown functionality. If, for example, a double deletions present a significant slow growth rate if compared to the two single deletions, the two genes may have been involved in compensatory pathways that are able to substitute one the other, but if both are excluded from the organism there is no other pathway able to sustain their role, explaining a resulting fitness significantly non competitive if compared to the single mutations strains. The study of genetic interactions and the subsequent definition of the genetic interaction networks have had a dramatic acceleration in the last year, especially or *S.cerevisiae*, for which a number of systematic studies on all the possible double deletions have been released.[28, 29]

It is also possible to quantify the strength of the interactions through the ϵ-score related to a double mutations. Let f_i be the fitness of the strain obtained

with the deletion of the gene i, the score is defined as:

$$\epsilon_{ij} = f_{ij} - f_i f_j \tag{2.7}$$

If the fitness measure, usually expressing the growth rate of a cell, is normalized in $[0, 1]$ the score has a range of values in $[-1, 1]$. The value -1 express a synthetic double deletions, i.e. a deletion that kills the cell while neither the two single deletions do it.

It is worth to note that the concept of interaction is also used referring to physically events between proteins, and are in fact known also as protein-protein interactions; these are usually established following various biochemical events. When constructing the map of the interactions in an organism these could also be included or form another independent map and I will consider them in the following, alongside the firsts.

Chapter 3

Metabolic Engineering and Pathways Design by Computational Systems Biology

3.1 Introduction

Over the last three decades there has been a dramatic increase in the interest for the bio-production of various chemicals through the metabolic engineering of micro-organism. The microbial fermentation is now widely used for the production of a wide range of chemicals; there are in fact a serious of advantages if compared with other techniques. Ideally, the engineered organisms would be able to produce pure compounds that are difficult to obtain otherwise, and using at the same time economically advantageous feedstocks derived, for example, from industrial wastes. The metabolic engineering is then required to improve and optimize the production chains, both in term of increasing the yield of product for a unit of nutrient, the productivity, and the reliability and stability of the proposed engineered strains.

Engineering the microbes to obtain a determined improvement is indeed a complicated task to be addressed. Many strategies, such as genetic editing or metabolic rewiring, have been tested over the years, with various outcomes of the resulting cell factories [30].

The systems biology tools are of evident help in these tasks, providing and

proposing computational framework for the prediction of the phenotypes of the microbes starting from different cellular genotypes and environmental conditions. Hence, the ultimate goal is to provide an exhaustive computational framework for exact predictions of the actual phenotypes; though, the complexity of such a task is evident and we are still far from a complete unveiling of all the underlying phenomena occurring in the cell factories. The genetic manipulation alone consist of a set of an enormous number of possible changes that are impossible to explore exhaustively.

The industrial interest over this kind of optimization process is however growing exponentially, driven by the many implications that such a working framework would have on a plethora of processes. Chemicals of interest for energy, industry, agriculture, could be produced by cell factories, improving at the same time the environmental sustainability of the production processes addressing the increasing concerns in the field [31].

The genome-scale model, as presented in the previous chapter, have been a dramatically useful tool for the metabolic engineering. Since the first proposed reconstructions for simple organisms, many others have emerged, particularly for more complex model organisms such as *E.coli* and *S.cerevisiae*. The contributions of these models to the field include, but are not limited to, giving a complex metabolic network reconstruction that helped a deeper comprehension of the metabolic pathways and of the metabolism itself, and the chance of studying the network as a whole, with all its related properties [32]. At the same time, the reconstructions offer a simple mathematical tool representing the network, as we have seen.

Clearly there are also a lot of problems and limitations of the metabolic network applications in the systems biology and metabolic engineering. First, the metabolic network itself needs to be properly tuned, including a set of parameters, usually typical of each metabolite and each reaction. Unfortunately the number of parameters requested and the difficulties in obtaining consistently their values make the model fitting an extremely challenging point.

Another critical point is the absence of properly defined objective functions for state that are not the exponential growth state; in the cell factories studies other states of the cellular life could be of interest or have to be taken into account. The

paradigm of the genome scale models still does not allow a straightforward method to simulate the different phases [32].

In the immediate future it is likely that the efforts of the community will be directed in the enrichment and refining of the models, including for example information from transcriptional regulatory network, whose significant part has already been included in the *E.coli*, or through enzymatic constraints as in the latest works regarding the *S.cerevisiae* reconstructions [19].

Even though there are still some improvements needed by the genome-scale models, they still are a powerful tool to be used addressing the metabolic engineering task. In this Chapter I will try to tackle the problem considering improvements that I applied on the *MOME* algorithm that I contributed to develop [33].

I will consider three different approaches to the problem, involving simulated genes deletions for the improvement o the phenotypic prediction over the chemical production, a gene regulations using the redirector approach and finally a medium optimization that is still in the initial stage of its development an that I believe will be of particular interest in the future. Initially I will focus on a case study for the optimization of the production of ethanol in different organisms [33] and then I will switch to the production of Lactic Acid, also introducing an innovative, at the best of my knowledge, approach combining the three that I just mentioned.

3.2 Multi-Objective Optimization and Analysis

In this section I recall some of the basic features of multi-objective optimization and metabolic design through *bioCAD* tools [11] and introduce the MOME algorithm [33] used for the following optimizations.

3.2.1 Multi-Objective Optimization

The concept of Pareto Optimality emerges every time that a problem present two or more competitive functions to be optimized; having objectives in contrast between them make clear the necessity of a trade-off between them, as it is in general not possible to find a solution that optimize at the same time all the objectives. In our case, for example, it is common that a trade-of is needed between the

growth-ratae of a cell and the excretion of a certain chemical, because the latter subtract resources to other pathways leading to the pure growth rate. If some of the metabolic resources are redirected to the production of a chemical, there will be a reduction in the growth and vice-versa. The concept of Pareto Optimality, that is widely used by the algorithm MOME, is then used to classify the points of problems like this where we have trade-off points among different objectives.

Formally, the definition of a partial ordering relationship $\prec$ for each $x, y \in \mathbb{R}^k$ is: $x \prec y \iff x_i \leq y_i \quad i = 1, \ldots, k \quad$ and $\quad \exists\ j$ s.t. $x_j < y_j$, that is, if each component of x is less than or equal to its corresponding component of y, and at least one x component is strictly less than y component. Then, given a generic multi-optimization problem with objective function F an input vector x is said to dominate y with respect to F if $F(x) \prec F(y)$. Finally, the Pareto-front is defined as the set of input vectors x such that there are no input vectors y that dominates x [34]. The goal of the multi-objective optimization algorithm is thus to find (or approximate) the *Pareto front* of the problem.

MOME Algorithm

In silico analysis of FBA models for metabolites overproduction was firstly modelled by directly manipulating the upper and lower bounds on the reaction fluxes [35]. The approach was further improved to account for improved modelling of genetic manipulations, e.g. using of genetic knockouts [36]; or modelling enzymes up/down-regulation [12].

Heuristic optimization techniques have been extensively applied for *in silico* optimization problem associated to synthetic biology in the last two decades. Example specific to the field of metabolic engineering are: Genetic Design through Local Search (GDLS) [37] in which the MILP is iteratively solved in small region of the design space; Enhancing Metabolism with Iterative Linear Optimization (EMILiO) [38], that use a successive linear programming approach in order to solve efficiently a MILP obtained through the Karush-Kuhn-Tucker method. A recent survey of the state-of-the-art is given in [13].

The optimization algorithm MOME [33] that I contributed to develop and that I present and expand here, follow this path of heuristic approximations. In

particular, it is aimed at the discovery and exploration of *Pareto optimal* trade-offs between the production rate, as predicted by a metabolic model of a microorganism, of a chemical or molecule of interest and the modelled organism biological objective, sets of key genetic manipulations are identified, which lead to overproducing strains yet with sensible growth as predicted by the FBA model.

The algorithm lean on the well known algorithm *NSGA-II* [39]; the inspiration is the concept of genetic and evolutionary algorithm [34], in which an initial population, i.e. the initial set of solutions is evolved through a series of genetic or evolutionary operators inspired by the laws regulating the natural selection. Hence the solutions are sequentially adapted to the specific objective functions that evaluate their fitness. In this way, if the algorithm is properly set in the parameters and in the strength of the evolutionary operators, the population evolves to a set of pseudo-optimal solutions that optimize the fitness functions and, in the case of multi-objective optimizations, are generally closer and closer to the theoretical Pareto Front at every new generation obtained.

Algorithm 1 MOME Optimization Algorithm

procedure MOME($pop, maxGen, dup, uKC$)
  $P^{(0)} \leftarrow InitPop(pop)$
  $FBA(P^{(0)})$
  $Rank\_and\_crowding\_distance(P^{(0)})$
  $gen \leftarrow 0$
  while $gen < maxGen$ **do**
    $Pool^{(gen)} \leftarrow Selection(P^{(gen)}, \left[\frac{pop}{2}\right])$
    $Q_{dup}^{(gen)} \leftarrow GenOffspring(Pool^{(gen)}, dup)$
    $Q_{dup}^{(gen)} \leftarrow Force\_to\_feasible(Q^{(gen)}, uKC)$
    $FBA(Q_{dup}^{(gen)})$
    $Rank\_and\_crowding\_distance(Q_{dup}^{(gen)})$
    $(Q^{(gen)}) \leftarrow BestOutOfDup(Q_{dup}^{(gen)}, dup)$
    $P^{(gen+1)} \leftarrow Best(P^{(gen)} \cup Q^{(gen)}, pop)$
    $gen \leftarrow gen + 1$
  return $\left(\bigcup_{gen} P^{(gen)}\right)$

The Algorithm 1 is a pseudo-code description of MOME. As typical for the genetic and evolutionary algorithm, there is a set of parameters to be set and that

influence the behaviour, the process and the results of the algorithm:

(i) *pop*, that is the maximum size of the population at every generation;

(ii) *maxGen*, that is the maximum number of generation for whom the population is evolved, this is indeed the number of iterations of the main loop performed;

(iii) *dup*, that is the strength of the *cloning operator* [40], i.e. the number of offsprings generated at every generation;

(iv) uKC, that is the maximum number of genes KnockOuts allowed at every time, i.e. every point will be required to have a number of deletions less or equal to uKC.

The initial population, $P^{(0)}$, is randomly initialized by the routine *InitPop*, which randomly applies few mutations to the wild type strain, i.e. strains with no deletions, even though in some cases it started directly from a set of wild type points. The FBA is hence applied to each strain in $P^{(0)}$, and, accordingly to the value of the production rates of metabolite of interest, a *rank* and *crowding distance* for each member of the population are computed[39]. The former ensures the *Pareto-orientation* of the procedure, giving priority to the non-dominated solutions i.e. the points theoretically closer to the Pareto Front of the problem. The *crowding distance* is instead a measure estimating the density of the population near each candidate solution. At each generation, there will be more and less isolated solutions in the phenotypic space, hence the candidate solutions lying in a relatively *unexplored* regions of the space (thus having small values of crowding distance) are preferred to those which lie in "crowded" regions. This is due to the necessity of keeping the unexplored regions in the populations so that a broader spectrum of all the solution space is analyzed in the next generation of points. Evaluating this measure is necessary because the algorithm could tend otherwise to keep all similar points from a narrow section of the solution space in a generation avoiding the exploration of the entire space and hence reducing the variability in the results and obtaining a partial approximation of the actual Pareto front of the problem. A generation counter is updated with the number of iterations and used to stop the simulation when reaching the *maxGen* value.

Using the points of the current population $P^{(gen)}$ as a set of "parents", the function *Selection* start a *binary tournament selection* comparing the different points to select the mating pool $Pool^{(gen)}$. The tournaments, comparing the solutions two at a time by their rank and the crowding distance, end when the pool reach a size equal to $[pop/2]$; the pool of parents is then used to obtain new *children* or *offsprings*. The operator $GenOffspring$ generates the Children from the pool of parents performing a series of binary mutations, i.e. adding a simulated gene deletion to the current parent strain. For each parents, *dup* offsprings are generated by the operator, and they constitute the set $Q_{dup}^{(gen)}$. A new selection is then performed in order to keep only the best solutions for each parent, and these points are collected in the population $Q^{(gen)}$. A *dup* number greater than 1 is necessary because the random selection of the new mutation could lead to an infeasible strain; in this way the probability of obtaining a feasible strain is increased. The choice of the *up* parameter has to take into account this necessity and, at the same time, it has to avoid an excessive dispersion of the algorithm that could sensibly increase the computation time of the entire procedure. In fact, the evaluation of all the offsprings generated is the most time-consuming procedure present in MOME; as a trade-off between these necessities, the *dup* parameter was set equal to 10. The evaluation of the $Q_{dup}^{(gen)}$ population first ensure that the number of KnockOuts of the new points is below the threshold uKC; if any of the solution has a number of KnockOuts above it, a random number of genes are then Knocked In to obtain a strain feasible for our constraints. This procedure is fast because it does not require a call to the *FBA* optimization. The next step is the actual evaluation of the fitness of the new points; using this evaluation, the best solution among the *dup* children of each parent is selected by the $BestOutOfDup$ operator and moved to the $Q^{(gen)}$ set. The final step is performed by the function *Best*, that select the best *pop* individuals from the combined set of the current population $P^{(gen)}$ and the new set $Q^{(gen)}$ to obtain the new population $P^{(gen+1)}$ that will be used in the next iteration of the algorithm. When a $maxGen$ number of iterations have been performed, the procedure ends returning all the populations observed at each stage of the algorithm, and that are then combined in the unique set $\bigcup_{gen} P^{(gen)}$. A post-processing procedure is then launched to extract from the set the *observed* Pareto front, i.e. all the non-dominated solutions in the $\bigcup_{gen} P^{(gen)}$. It is im-

portant to stress the difference between the actual Pareto Front, that could only be known having information on all the solutions in the feasible region, and the *observed* Pareto Front, that is the result of the algorithm in which only a small region is covered, even though the algorithm itself is aimed in obtaining the points closer to the actual Pareto Front.

3.2.2 Adaptation to the Redirector Framework

The MOME algorithm could simply adapted to be used for the optimization of the prediction using the redirector framework. The changes needed are, obviously, in the fitness evaluation, in the encoding of the points and in the genetic operator. Whereas for the KnockOut simulations the points are encoded as a binary array in which a true value simply represent a gene deletion, for the *flat redirector* two logical arrays were used, one to storage the positive regulations and the other for the negative ones, combined into a single one. The mutation process is similar to the KO gene set one. The only difference is that there is one binary vector with double length respect of the gene set number. The first half is a representation of the positive variation, while the second half of the negative variations. If the jth gene set has a 1 in the first half and a 0 in the second, then a positive variation occurs. The negative variation occurs when a 0 is present in the first half and a 1 in the second part. In case of equals values for a gene set, two 1 or two 0, nothing happens to that gene set. This formulation allows to maintain the same genetic operator used in the gene set KO simulation, with a single variation among a binary vector.

For the *fine tuning redirector*, instead, the double vector is abandoned, preferring in this case saving the real variables values inside a single vector. These new variables require a new genetic operator which mutates them during the evolutionary process. Every time a solution is considered to be mutated, a random gene set number j^* is chosen and its coefficient y_{j^*} changed. In order to do this, a random

step s in $\{0, \pm 0.1, \pm 0.2, \pm 0.3\}$ is selected, and the the following rules are applied:

$$y_{j^*} = \begin{cases} y_{j^*} + s & if |y_{j^*} + s| \leq 1, |y_{j^*}| < 1 \\ 1 * sign(y_{j^*}) & if |y_{j^*} + s| > 1, |y_{j^*}| < 1 \\ sign(y_{j^*}) * (|y_{j^*}| - |s|) & if |y_{j^*}| = 1 \end{cases} \tag{3.1}$$

We so have a step by step mutation procedure, which gradually redirect the fluxes, looking for an optimal setting.

3.2.3 Medium Optimization

Apart from the redirector approach simulating the over and under expression of the genes, other variables can be used to change the prediction, namely the bounds of the exchange reactions that simulate the uptake of the compounds from the environment to the cell.

To adapt the algorithm to use the external fluxes bounds as variables, that are expressed as real values, a deep change of the genetic operator was needed. In this case, I considered the points of the optimization as the maximum flux through all the exchange reactions present in the model. The wild type values are the exact fluxes as predicted from the wild type strain, and I used this values to construct the initial population of points, defined introducing random variations of some of the fluxes, selected again at random. The variation for a flux v_j is selected defining a quantity

$$var = \left| \frac{ub_j - lb_j}{10 \cdot ub_j} \right|$$

that is then used to perform the actual variation following the rules

$$v_j = \begin{cases} v_j + var \cdot rand & if\ var = 1 \\ v_j + var \cdot (2 \cdot rand - 1) & otherwise \end{cases}$$

where rand is a uniformly distributed random number in the unit interval $[0, 1]$. The initial population is thus created.

In the next iterations, when the algorithm calls the genetic operator, it could performs two different type of variations, each of them occurring in the 50% of the cases; the first generates an offspring point performing a variation on a single

parent point using an operator that is analogous to the one described for the initial population. The second procedure is instead a crossover operator, that starting from two parents p_1, p_2 generate two children c_1, c_2; first a uniformly distributed random number u in $[0, 1]$ is generated, and then a quantity q is defined as

$$q = \begin{cases} (2 \cdot u)^{1/(\mu+1)} & if\ u < 0.5 \\ \left(\frac{1}{2 \cdot (1-u)}\right)^{1/(\mu+1)} & otherwise \end{cases}$$

where μ is a specific mutation parameter that has to be proper set in relation to the numerical properties and ranges in the problem. In this case I set $\mu = 20$. Having defined the quantity q, the two children points are hence generated as

$$\begin{aligned} c_{1j} &= 0.5 \cdot ((1+q) \cdot p_{1j} + (1-q) \cdot p_{2j}) \quad \forall j \\ c_{2j} &= 0.5 \cdot ((1+q) \cdot p_{1j} + (1+q) \cdot p_{2j}) \quad \forall j \end{aligned}$$

At every stage, if the new values goes beyond the lower or upper bound, as initially set in the model, of the specific exchange reaction, the value is set equal to the bound.

Every value of the candidate solutions explored by the algorithm constitute the maximum flux that the corresponding exchange reaction could assume, i.e. the maximum quantity of the specific nutrient that could enter the metabolic network. Clearly, the different values found by the algorithm affect heavily the fitness prediction of each point. The operators are able to efficiently analyze the solution space, that is indeed much more complex than the one in the knockouts or redirector approaches, as here the values are real numbers which have to be selected from a broad interval. I excluded from this optimization the fluxes of Glucose and Oxygen, to constitute a common benchmark to evaluate the newly discovered points. For an example of the results obtained by this procedure see Figure 3.8c.

3.2.4 Robustness Analysis

Using the methodology described in the previous chapter, a robustness analysis on the *E. coli* (model by Orth et al. [41] and model by Feist et al. [42]) was performed.

The y strings used are those obtained by the optimisation of ethanol production and biomass formation (Figures 3.2 and 3.3). The inputs are the upper bound v_j^U and the lower bound v_j^L, $j = 1, \ldots, n$ of the metabolic fluxes. In particular, for each strain, the fluxes corresponding to knocked out gene sets, are maintained equal to zero. In particular, given the string y, only a subset of fluxes are nonzero, so they are used as inputs to perturb the lower and upper bounds of these fluxes.

Clustering

Solutions when represented using, for instance, the production of a metabolite and biomass production tend to form clusters. These highlight the feasible zones of the space of solutions, assuming that an exhaustive search has been performed. Performing clustering on such solutions allows to study the characteristics of the strains that belong to a cluster and potentially identify similarities. There are several clustering techniques, however in this context density-based clustering seems to be the preferable to centroid-based or probabilistic techniques such as k-means and Expectation Maximization, since clusters generally tend to have irregular shapes.

Briefly, DBSCAN distinguishes three different types of points: *core*, *border* and *outliers*. Core points are those points with at least k points within a distance *epsilon*, such points are directly reachable from the core point. a point that is not a core point is a border point if its distance from a core point is less than or equal to ϵ; those points that do not satisfy these conditions are outliers. A cluster is defined by a set of interconnected core points (forming paths of directly reachable core points) and the border points that are connected to them.

If not otherwise specified, the parameters of the algorithm have been set as follows: k=4 and $\epsilon = 0.06$ and the data was normalized before being clustered.

3.3 Ethanol Production

As a first case study of a possible application of the MOME algorithm, I present here an application aimed at the strains design and metabolic engineering for the optimization of the ethanol production [33]. The ethanol production is chosen

for the increasing interest in the biofuels production for ecological reasons and it constitutes a classical objective for strains engineering; in the US there was an increased production of ethanol biofuel from 2015 to 2016 by half a million of gallons, to a total of 15 billions; furthermore, ethanol is the base of more of the 90% of all the biofuels produced [43, 44]. However current ethanol production methods are unable to meet the increasing global demand of bio-ethanol production due to their low yield on feedstock whose primary value is of food and feed [45]. Genetically engineered microorganisms are therefore needed to carry out the production and tackle the increasing demand of product in the market, especially regarding the use of second generation feedstocks of interest [46], such as industrial waste, that could make the production even more economically competitive, having at the same time a huge impact on the environment, reducing the polluting fossil fuels with new biofuels and eliminating the necessity of the biomass waste disposal, that could instead be proficiently used in new cell factories and production chains.

The MOME algorithm was then used to obtain sets of strains overproducing ethanol, if compared to the wild type. The results obtained by the algorithm are analysed when applied to the problem associated to overproduction of ethanol in seven different organisms, as modelled by corresponding genome-scale models of their metabolism. First, focusing on gene KO analysis and discuss extensive comparisons were performed on seven different models. Then comparing the results of using genetic KO with those obtained using enzymes up/down regulation, using the Redirector modelling framework in the specific case of *S. cerevisiae*.

Finally, I developed and run a set of further simulations and analysis, in which informations on the essential genes and others constraints on the growth rate and the external simulated rich media were added, to better simulate a realistic scenario. In the same medium used in the other simulations, the maximum increase in the ethanol production is +195.24% .

3.3.1 Gene KO optimization for Ethanol Production

In this section, I present the results for the optimization of ethanol in *Escherichia coli k12 mg1655* FBA model *iJO1366* [41]. In the following I then compare *E. coli* results with those obtained using other 6 other organisms; that are:

(i) *Staphylococcus aureus subsp. aureus N315 – S. aureus* (model used: *iSB619* [47, 48], 655 metabolites, 743 reactions and 619 genes);

(ii) *Salmonella enterica subsp. enterica serovar Typhimurium str. LT2 – S. enterica* (*STM_v1_0* [47, 49], 1802 metabolites , 2545 reactions and 1271 genes);

(iii) *Yersinia pestis CO92 – Y. pestis* (*iPC815* [47, 50], 1552 metabolites, 1961 reactions, 815 genes);

(iv) *Saccharomyces cerevisiae S288C – S. cerevisiae* (*Yeast 7.6* [51], 2302 reactions, 909 genes);

(v) *Chlamydomonas reinhardtii – C. reinhardtii* (*iRC1080* [47, 52], 1706 metabolites, 2191 reactions, 1086 genes);

(vi) *Yarrowia lipolytica – Y. lipolytica* (*iYL619* [47, 53], 843 metabolites, 1,142 reactions, 619 genes).

Figure 3.1 shows the projection on the codomain space of the feasible region explored by MOME framework and observed Pareto front for the *E. coli* optimization, and 3.1 shows the 10 best trade-offs found (as for values closer to theoretical maximum production). Highest production rate for ethanol found is 19.74 $mmol\, gDW^{-1}\, h^{-1}$ which is a +832.88% improvement with respect to wild-type production. This is obtained by a strain that produce biomass at a rate of f 0.02 h^{-1} (i.e. 98.06% reduction with respect to wild type biomass), and that has a knock-out cost of 21. Specific knock-outs for this strain are: *frmA*, *(fadB or yfcX)*, *fieF*, *uxuB*, *(nuoN and nuoM and nuoL and nuoK and nuoJ and nuoI and nuoH and nuoG and nuoF and nuoE and nuoC and nuoB and nuoA)*, *((pflA and pflB) or (pflA and tdcE) or (pflD and pflC) or ((pflA and pflB) and yfiD))*, *ppk*, *rfaS*, *tpiA*, *avtA*. This solution has indeed an extreme phenotype, in which the predicted growth rate in the exponential phase, that as noted before is the phase considered by the objective function in the genome-scale model close to a null value. Due to this predicted phenotype, this strain is likely to be actually inviable if reproduced, or have a reduced productivity whatsoever.

Figure A.3 summarizes ethanol production as a function of the knock-out cost in strains explored by MOME. Intuitively, strains that are in *knees* of the function represent strains with an optimal trade-off between knock-out cost and ethanol production rate. An interesting strain that this analysis reveal is the strain having a knock-out cost of 6. This produces ethanol at a rate of 18.52 $mmol\,gDW^{-1}\,h^{-1}$ (+775.22%) and has biomass formation of 0.10 h^{-1} (−90.32%). Another single gene knock-out strain is characterized by 16.49 $mmol\,gDW^{-1}\,h^{-1}$ ethanol production, i.e. +679.29% improvement with respect to wild type, and a biomass formation of 0.23 h^{-1} (−77.45%). The genetic target knocked-out in this strain is: *(nuoN and nuoM and nuoL and nuoK and nuoJ and nuoI and nuoH and nuoG and nuoF and nuoE and nuoC and nuoB and nuoA)*. This strain is of a particular interest because the deletions of the *nuo* genes, linked to the NADH-quinone oxidoreductase, and especially of the *nuoB* subunit, has been previously reported to have an effect on the ethanol production [54]. Furthermore, even though the corresponding reactions are linked to all the subunits, the knockouts of few subunits would be sufficient to inhibit the corresponding enzyme [55].

The Pareto front of the values of the objective functions (ethanol production and biomass) of the selected solutions has been clustered using DBSCAN. The results of the clustering are shown in Figure A.2. There are 7 separate and identifiable clusters of solutions. Cluster C3, containing 6294 solutions, provides good ethanol production without penalising biomass. On the contrary, the low biomass production of clusters C1 and C2 would not allow bacteria to survive, while clusters C4-C7 produce modest quantities of ethanol.

Figures 3.2 and 3.3 depicts the Pareto fronts obtained for a set of prokaryote and eukaryote organisms respectively. For ease of comparisons results are normalized by using theoretical upper bounds for both ethanol and biomass production. As a comparison with the *iJO1366* model, the *E. coli iCA1273* [47] is also included. In contrast to the former, no trade off points between the Biomass and the Ethanol production were found by the algorithm, resulting only in points on the two axis. Since the algorithm, set with the same parameters, worked well with all the others models, these results could be caused by the inner features of the model. Among the organisms here explored, *S. cerevisiae* is the one for which the Pareto front computed by MOME is closest to the utopian optimization point

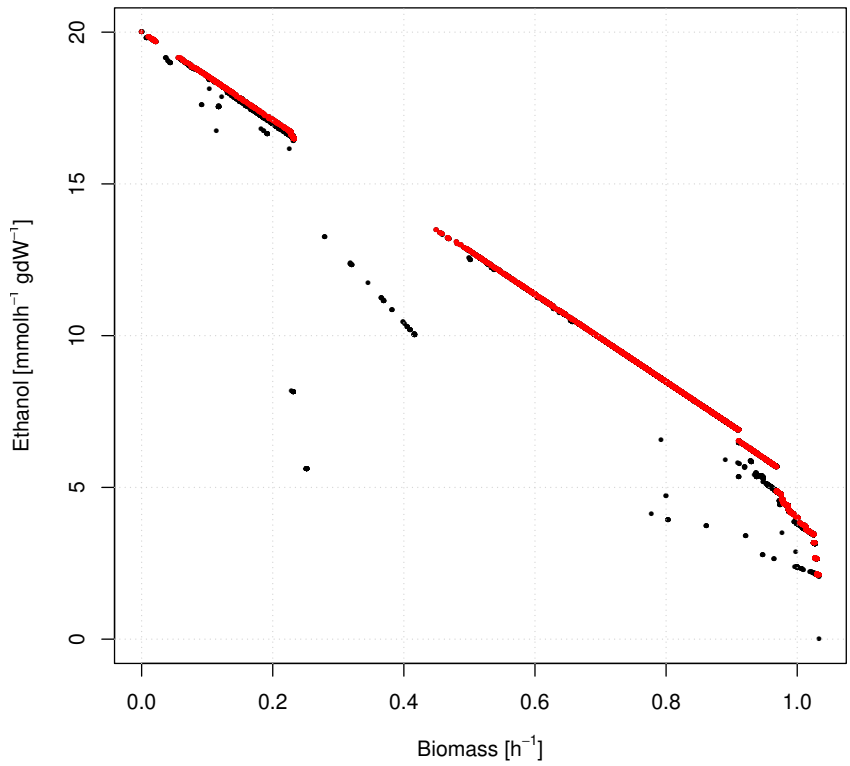


Figure 3.1: Results for optimization of ethanol production and biomass formation in *E. coli*, anaerobic condition, glucose uptake rate 10 $mmol\,gDW^{-1}\,h^{-1}$. Pareto front (in red) and feasible strain (in black).

(that is maximal biomass and maximal ethanol production). The Figure 3.4a plots the whole feasible region explored by the algorithm using the *S.cerevisiae* model.

On the other hand, both *C. reinhardtii* and *S. enterica* do not demonstrate good trade-off between biomass and ethanol; for only small improvements in ethanol production follows consistent decreases in the organism biomass.

3.3.2 Enzyme Regulation in *S. cerevisiae*

Using the MOME algorithm in synergy with the redirector framework, as described before, allow me to obtain different results, typically starting from the wild type strain simulating different regulations on the gene set that affect the reactions in

Strain	Ethanol ($mmol\, gDW^{-1}\, h^{-1}$)	Biomass (h^{-1})	Knock-out cost
Wild Type	2.11603	1.0334	0
S1	16.491892	0.2331	1
S2	18.116 785	0.13082	5
S3	18.798875	0.07981	6
S4	19.72478	0.020068	6
S5	19.724782	0.020068	7
S6	18.949056	0.069831	8
S7	19.741314	0.018862	9
S8	19.724780	0.02068	13
S9	19.741314	0.018862	14
S10	19.724782	0.020068	15

Table 3.1: Maximization of Ethanol and Biomass production for *E. coli*.

the metabolic network, as I described before. Figure 3.4b shows the feasible strains and the Pareto-optimal ones found by MOME for the optimization problem associated to ethanol overproduction in *S. cerevisiae* using enzyme up/down regulation. Notice that a linear relationship between biomass and ethanol production is observed for Pareto-optimal strains, and that feasible strains found by MOME almost uniformly span the region from maximal biomass production ($\approx 0.28\ [h^{-1}]$) to null biomass production, hence discovering a number of different trade-offs *S. cerevisiae* strains. These widespread results over the phenotypic space show that the enzymes regulation approach is in general more flexible than the "binary" KO one.

Organism	Model	Genes	Essential Genes	Synthetic Double Deletions
E. coli	*iJO1366*	1366	113	108
E. coli	*iZ_1308*	1308	105	64
S.cerevisiae	*Yeast 7.6*	909	215	580

Table 3.2: Number of essential genes and lethal gene pairs present in the genome-scale metabolic models.

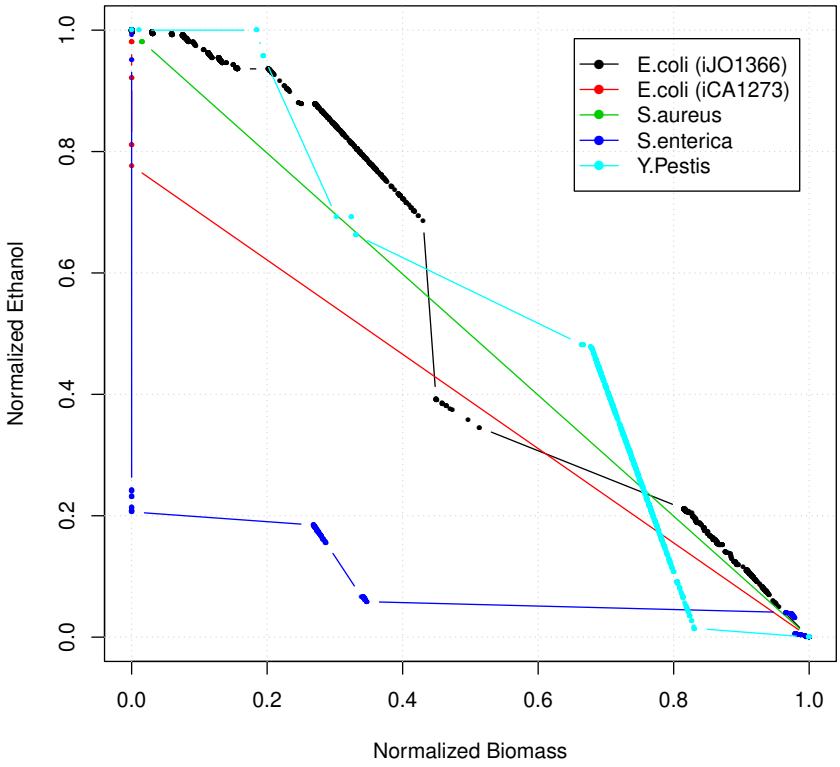


Figure 3.2: Normalized Pareto Fronts of models optimizations for ethanol production and biomass formation in various *Prokaryotic* organisms.

3.3.3 Ethanol production without Essential Genes in *E. coli* and *S.cerevisiae.*

I have described so far the results of the simulations without specific constraints; those results can be considered as an utopian bound of the framework. However, without the introduction of other external constraints simulating the issues of a possible real world application, some of the selected strains could be difficultly applied. To tackle this possible lack of plausibility further simulations were performed, reported in this section for the Ethanol production considering the Essential Genes of the given organisms. I used the *Yeast 7.6* model of *S.cerevisiae* and two models of *E. coli*, the *iJO1366* and the *iZ_1308*[47, 56], the latter modelling the *E. coli O157:H7 strain EDL933*. In contrast to the *iCA1273*, the results

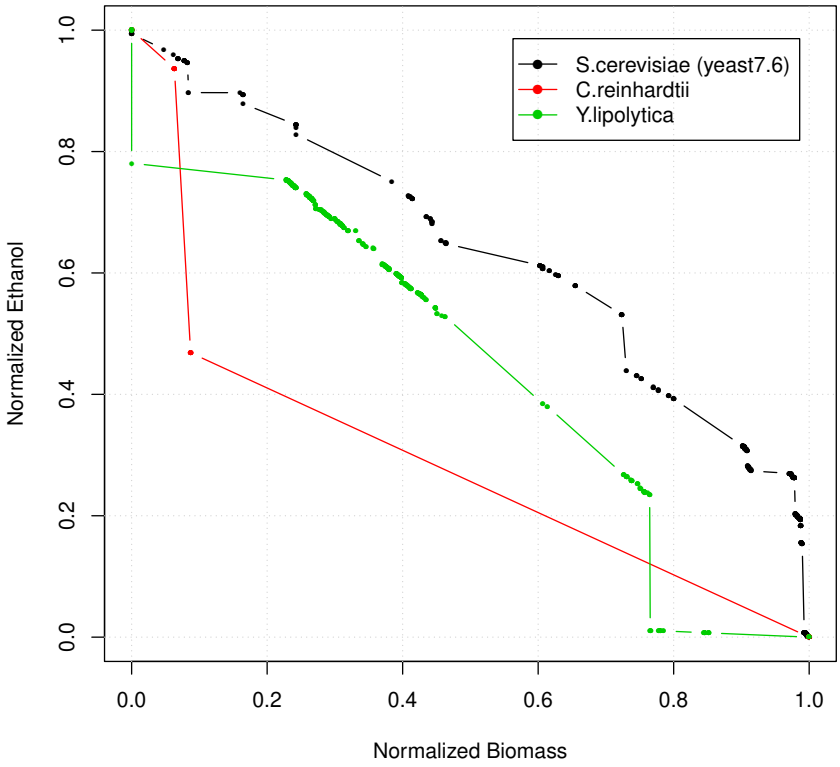
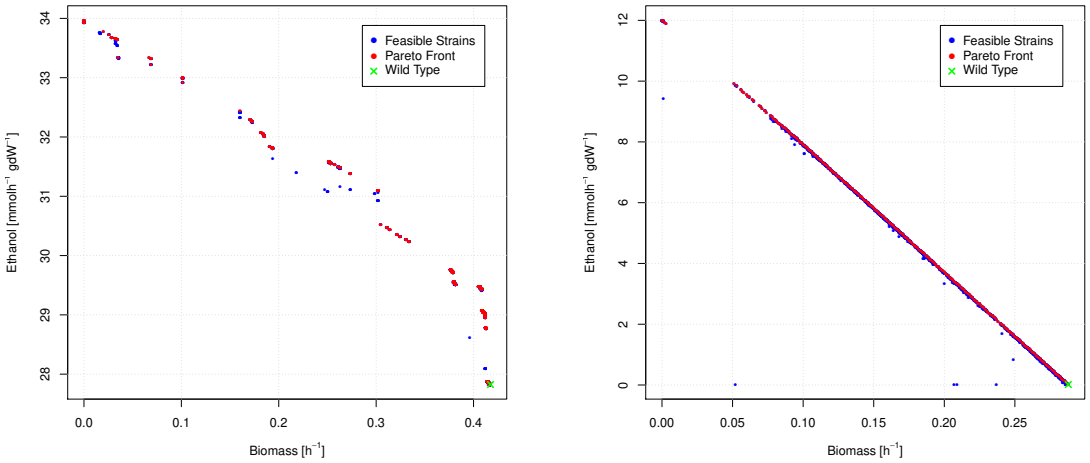


Figure 3.3: Normalized Pareto Fronts of models optimizations for ethanol production and biomass formation in various *Eukaryotic* organisms.

obtained using this new model are quantitatively comparable with the *iJO1366* ones.

Hence the framework was changed to tackle this new task, first introducing some limitations over the genes of the models that can actually be knocked out by the algorithm. Namely, I included information on the *essential genes* and the *lethal gene pairs* of the different organism, taken from external databases. The list of the essential genes of the *E. coli* was taken from the *EcoliWiki* [57], whereas the list of synthetic double deletions for *iJO1366* is taken from [58] and the one for *iZ_1308* from [59]; the genes lists for the *S. cerevisiae* model were taken from [60] (see Table 3.2 for a summary). The essential genes, defined as the genes whose single deletion would result in a non-viable strain of the organism, are thus always



(a) Gene set Knock-Out Multi-Objective Optimization using the *Yeast7.6* model.

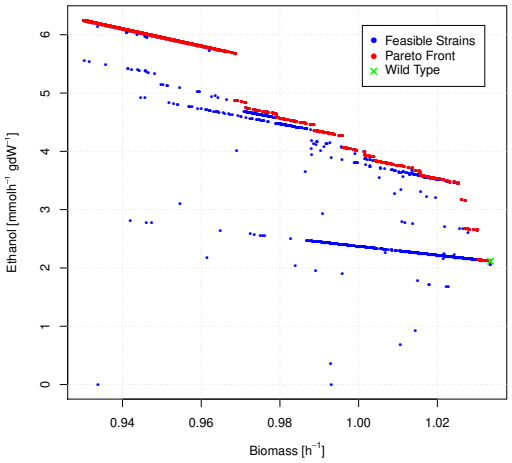
(b) Gene expression Redirector Multi-Objective Optimization using the *iMM904* model.

Figure 3.4: Results for optimization of ethanol production and biomass formation for *S. cerevisiae*.

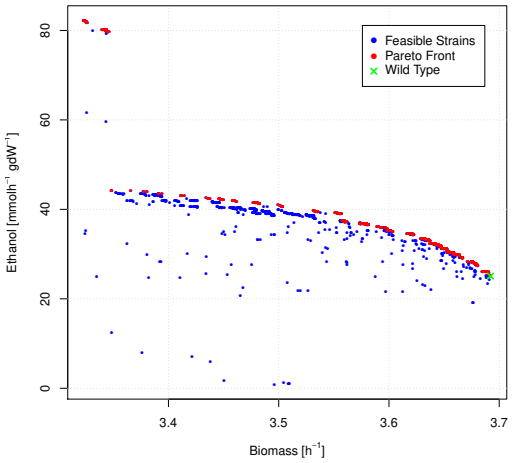
excluded from the framework, i.e. they can not be turned off by MOME. A similar approach is used for the synthetic double deletions, defined as the pairs of genes deletions that, if occurring at the same time, make the strain non-viable, but a single deletion of one of the two does not. While the single knockout of one of the genes of a couple is still allowed (if not essential), the knockouts of both are not; a check of the new possible genes to be knocked out is run in every step of the mutation operator.

Since the databases always refer to single genes, in these simulations I considered the single genes in the models, to obtain a direct comparison between a strain and the lists. However, it is indeed really simple to obtain the gene sets, on which the reactions depend, back again starting from a binary vector representing the single genes of a strain, by just evaluating the boolean expressions of all the gene sets.

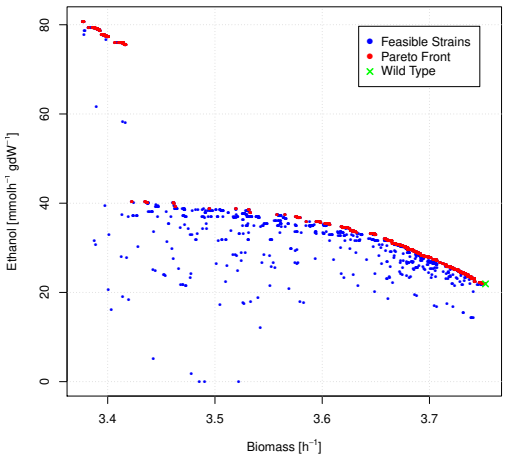
In addition to this, a strict bound on the biomass values is also introduced. Referring to the Wild Type value, all the strain obtained are forced to have a



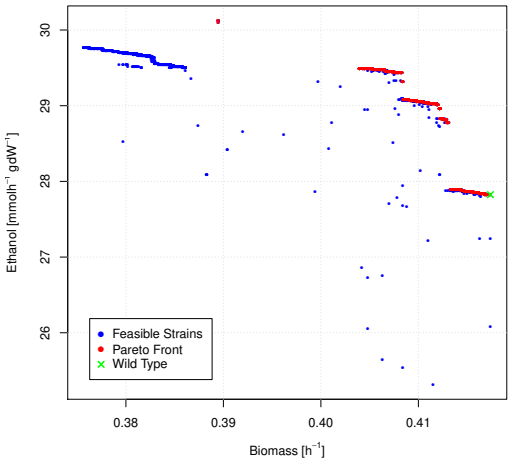
(a) Gene Knock-Out Multi-Objective Optimization using the *iJO1366* model in anaerobic condition.



(b) Gene Knock-Out Multi-Objective Optimization using the *iJO1366* model in LB medium.



(c) Gene Knock-Out Multi-Objective Optimization using the *iZ_1308* model in LB medium.



(d) Gene Knock-Out Multi-Objective Optimization using the *Yeast7.6* model in SD medium.

Figure 3.5: Results of the Gene Knock-Out Constrained Multi-Objective Optimization for different metabolic models and conditions.

biomass reduction not greater than the 10%. So, if a new selected deletion leads the strain to a lower biomass, that gene is restored and the mutation operator

selects a new gene to be deleted; the procedure is repeated until a strain with a new knockout, having a biomass value above the bound, is reached or until a maximum number of attempts (in general 10 trials) has been done. This new constraint also lets the algorithm to more deeply explore a reduced solution space, while forcing the algorithm to discard the strains with a low growth, which can be considered biologically infeasible.

Furthermore, for these new simulations the bounds of the external exchange reactions of the models are properly set in order to simulate the growth in a rich medium, i.e. the well known *LB medium*[59] for the *E. coli* models and the *SD medium*[61] for the *S. cerevisiae* model. A simulation with the same anaerobic medium used in the previous unconstrained tests was also performed.

These new simulations results are shown in Figure 3.5. It is remarkable how by changing the medium setting for the same model (*iJO1366*, ref. to Figures 3.5a, 3.5b) the phenotypic results are different in both the ethanol production and biomass value. Namely in the rich medium there is a much higher value of ethanol production even in the wild type, 25.0757 $mmol\, gDW^{-1}h^{-1}$ against 2.116 $mmol\, gDW^{-1}h^{-1}$ in the anaerobic medium, and a corresponding biomass value of 3.6921 h^{-1} against 1.0334 h^{-1}. This is not surprising, given that the rich medium have a sensibly higher number of nutrients overall that the metabolic network could draw from. Also, the progresses of the algorithm solutions are different, as it can be seen from the trends of the Pareto Fronts found, even using the same parameters. These discrepancies highlight once more the importance of the external environment settings for the *in silico* simulations.

Finally I considered the optimal solutions in the Pareto Front and applied on them a post processing procedure to keep only the necessary genes KOs. Starting from an optimal strain, the procedure iteratively select one gene knocked out in it and restores it obtaining a new strain. If both the biomass value and the ethanol production differences between these strains were less than a tolerance threshold, that is set at 10^{-5}, the gene deletion can be considered superfluous, and so the gene is permanently reintroduced in the strain; otherwise it is kept knocked out. The procedures ends when all the knocked out genes of the strain have been tested.

In the end, there is then a new set of filtered and further optimized solutions, with a low number of deletions (that never involve essential genes or synthetic

double deletions), and with a reasonable value of the biomass function, ensuring that well behaving metabolic pathways are still simulated in the network. Some of these results are shown in Table A.2; the reported strains are selected in this case as the ones with a maximum ethanol production among the strains with the same number of knockouts. Usually the increase in the number of deletions as predicted by the algorithm will result in a potentially higher metabolite production until a maximum number is reached (cf. Figure A.3). There are indeed many other solutions with higher number of deletions, but the overall maximum production found (always labelled as S10 in the tables) can be reached with less than 10 knockouts. It is notable that all the *E. coli* simulations reach a greater maximum ethanol production difference in percentage from the wild type than the *S. cerevisie* simulation. In the anaerobic condition the maximum production rate of ethanol using the *iJO1366* model is 6.2473 $mmol\, gDW^{-1}h^{-1}$, improving the wild type of +195.24%. It is indeed a far lower increase if compared to the ones obtained with the unconstrained algorithm, as expected. Similarly in the LB medium the maximum ethanol production rate is 82.2582 $mmol\, gDW^{-1}h^{-1}$, with a +228.04% improvement , whereas using the *iZ_1308* model the increase is +268.68%, with a maximum production rate equals to 80.7883 $mmol\, gDW^{-1}h^{-1}$. In the *Yeast 7.6*, conversely, the maximum increase is +8.24% and the maximum production rate is 30.1258 $mmol\, gDW^{-1}h^{-1}$. Moreover, while these strains of the *E. coli* models have a biomass reduced of approximately 10%, that is the maximum allowable reduction given the constraints that were used, the strain of *S. cerevisiae* reduces the biomass of 6.69%, highlighting a lack of optimal trade-off points in the region of the solution space closer to the 10% threshold of biomass reduction.

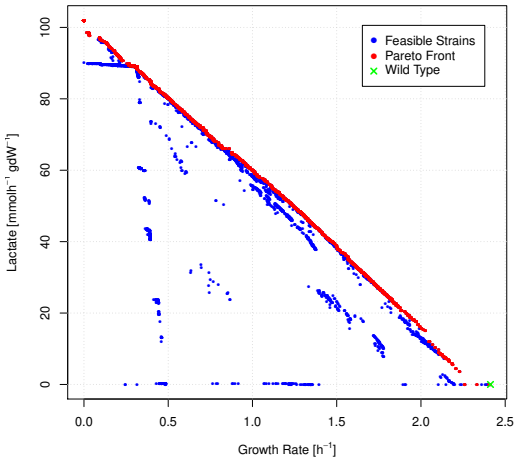
3.4 Lactate Production

After considering the case study of ethanol, I switched my focus on another chemical of particular interest, the *lactic acid* or *lactate*. For this study I focused on the metabolism of *S.cerevisiae*. The interest in the Lactate production is due to the request for substitutes compounds in plastic production, that has rapidly become one of the most prominent issues in the modern industrial research. The possibility to use Lactate to produce of polylactic acid, one of the most common bioplastic, highlights the importance of further research for a sustainable way of production of Lactate. The rapid increase of plastic pollution and the imminent shortage of petroleum resources has drastically accelerated the effort on finding green alternatives to petroleum-derived plastic. In particular, the discovery of a lactate-polymerizing enzyme had facilitated the creation of microbial plastic factories for the synthesis of lactate-based polyesters [62], potentially allowing a sustainable and efficient plastic production process.

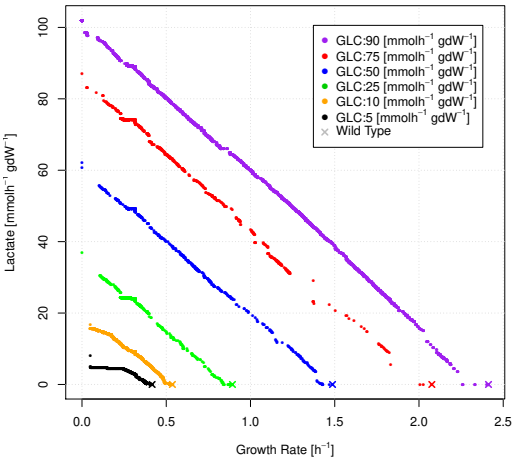
The metabolic engineering and the *in-silico* methods are then again a tool that could play a key role in the optimization and design of strains aimed at obtaining optimal Lactate production levels. I then used the MOME algorithm to optimize the lactate production and significant results were obtained, as summarized in Figure 3.6a. It is extremely evident from this plot how the two objectives are in competition, and how the algorithm at the same time is able to reach an observed Pareto Front close to the theoretical linear relationship between the two objectives. The points with the maximum growth rate exhibit the lowest value in lactate production and vice-versa. In this case I used some extreme conditions on the simulated medium, in particular I set the bounds of the exchange reactions of D-Glucose and Oxygen equal to 90 and 10 $mmol\, gDW^{-1}h^{-1}$ respectively. With these values a very high growth rate is obtained, because the model follow a linear relationship from the nutrients to the growth, without clear limitations happening in the real world conditions. The ratio between the two quantities ensure the contemporary activity of the respiratory and fermentative pathways. An interesting point on the predictions is made in the Figure 3.6b, where are reported the Pareto Front found by the MOME algorithm in different conditions of Glucose in the medium, and the oxygen bound is instead fixed to the same value equal to 10

$mmol\, gDW^{-1}h^{-1}$. The differences between the Pareto Front highlight the impact that the bounds have to the predictions of the model, but more interestingly it is possible to note that the ratio between the curves is not the same as the one occurring between the the glucose values.

As we have seen, however, the knockouts of some of the genes included in the metabolic model is only one of the possible approach to the optimization problem related to the chemicals production, I introduced the redirector framework and the medium optimization, and in the next section I will discuss a proposal for using them all together to improve the predictions.



(a) Results of the optimization for lactate production. The Pareto-Front is highlighted in red.



(b) In 6 different environmental condition, changing only the available glucose and oxygen external bounds, the resulting pareto fronts are sensibly different.

Figure 3.6: Results of the optimization procedure in different settings.

3.4.1 Multi Stage Optimization

Finally, I developed a new type of approach to the problem. The main idea was to combine the gene set KnockOuts simulation with the fine tuning redirector and the medium optimization. Since the gene set KnockOuts are a drastic way to

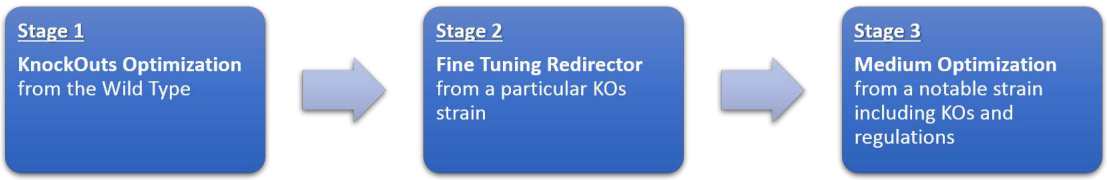


Figure 3.7: Simple workflow of the multi-stage optimization process. In my implementation the first step, starting from the Wild Type, select the KnockOuts improving the phenotypic predictions; one of these deletion strains is then selected as the starting point for the following stage, in which the redirector approach is used. Finally, a notable strain, including both the KOs and the regulations from the previous two stages, is selected and its predictions are further improved using the medium optimization algorithm in the last stage.

change the model, I first selected a feasible solution from the results of the standard simulation performed on the model. Hence this strain is used as the starting point for a second stage optimization, using the fine tuning redirector algorithm for further optimize the predictions on the production. All the regulations evaluated by the algorithm will then be applied on a strain including the deletions as selected from the first stage. In the same manner, at the end of the fine tuning redirector algorithm another notable strain could be selected for further optimization by the medium optimization algorithm.

The order in which the stage could happen is not fixed and could be decided and changed with respect to the specific application to be considered.

In the case of Lactate production, among the solutions found by the KOs algorithm the ones with the lowest number of gene involved in knock outs are reported in Table 3.3. The trade off between the Growth rate flux, the Lactate Production, and the total number of gene sets knocked out led to a solution, highlighted in the table, with a total of 3 gene sets knocked out involving 4 genes. The predicted growth rate is equal to 1.2327 (48.9 % less than the wild type) and the predicted lactate production is equal to 45.9851 $[mmolh^{-1}gdW^{-1}]$. The genes to be deleted are *PYC1* and *PYC2*, coding two isoenzymes of the Pyruvate Carboxylase, *GPD1*, catalyzing the Glycerol-3-Phosphate Dehydrogenase, and *GLT1* Glutamate synthase; all the geens are involved in initial phases of the metabolism, in particular of Pyruvate and Glutamate metabolism. The GPD1 mutant is re-

ported to show a significant increas in ethanol accumulation, that is reasonable because this pathway is close to the one producing Lactate. The double deletions of PYC1 and PYC2 is reported to affect the growth, so the interactions with the other two deletions found by the algorithm should be further analyzed before an actual reproducibility *in vivo*. The *GLT1* and *GPD1* deletions are not reported to affect the growth.

Lactate Production $[mmolh^{-1}gdW^{-1}]$	Growth Rate $[h^{-1}]$	Ethanol Production $[mmolh^{-1}gdW^{-1}]$	KO cost	Gene sets
89.2004	0.28243	69.9313	3	YDR050C YNL241C YPL262W
87.5051	0.3183	81.9188	3	YDL022W YDR050C YJL121C
78.8212	0.33251	78.9899	4	YDR050C YJL121C YMR083W YOL126C
86.0992	0.3254	70.721	4	(YER062C or YIL053W) YDR050C YHR104W
60.6003	0.93758	104.5517	5	(YGL062W or YBR218C) YDL022W YDL171C YPL061W
67.1999	0.78851	99.8315	6	(YGL062W or YBR218C) YDL022W YDL171C YOR184W YPL061W
10.57	1.7723	128.7437	3	(YGL062W or YBR218C) YDL022W
18.4143	1.5807	137.836	4	(YGL062W or YBR218C) YDL022W YLR058C
19.7614	1.5603	136.7941	5	(YGL062W or YBR218C) YDL022W YDR300C YLR058C
45.9851	**1.2327**	**115.5424**	**4**	**(YGL062W or YBR218C)** **YDL022W** **YDL171C**
30.1127	1.684	121.2078	5	YDL022W YDL171C YLR348C YNL241C YOL126C
48.6864	1.1776	113.0071	5	(YGL062W or YBR218C) YDL022W YDL171C YNL241C
49.3044	1.1636	112.4877	6	(YGL062W or YBR218C) YBR166C YDL022W YDL171C YNL241C

Table 3.3: Solutions with low number of gene sets knocked out obtained in the optimization. Glucose uptake rate: 90 $[mmolh^{-1}gdW^{-1}]$ oxygen uptake rate: 10 $[mmolh^{-1}gdW^{-1}]$. The KO cost is the total number of genes to be turned off, i.e. involved in the gene sets deletions.

The solution is not a Pareto optimal one, but considering the really small

number of knockouts is a good trade-off. Starting from this selected strain, a fine tuning redirector simulation was set and launched. So all the feasible points obtained by this new simulation share the absence of the 3 gene sets, while the over- and under- expressions of all the others constitute the variables, as we have seen for the fine tuning redirector. As reported in Figure 3.8b, the Pareto Front of this second-step simulation is really similar and near to the one achieved in the standard gene sets KnockOut one. The difference is that all the strains cannot increase the Growth Rate of the starting strain, so the results are limited only at the upper part of the plot of the phenotypes. In the lowest part of the Pareto Front, it is still a bit far from the original one, following the behaviour of the starting point, but other solutions achieved a good performance instead, comparable to the KnockOuts results.

Lactate Production $[mmolh^{-1}gdW^{-1}]$	Growth Rate $[h^{-1}]$	Regulations	Sign	Gene sets
66.2379	0.82315	4	UnderExpression: OverExpression:	-0.3: (YKL127W or YMR105C) 0.2: (YJL052W or YJR009C or YGR192C) 0.3: YDR272W 0.3: YML004C
66.7163	0.81214	4	UnderExpression: OverExpression:	-0.3: YNL241C 0.2: YDR050C 0.6: YML004C 0.3: YMR062C
68.9013	0.76387	4	UnderExpression: OverExpression:	-0.2: YDR226W 0.2: (YAL038W or YOR347C) 0.3: YDR272W 0.3: YML004C
66.1212	0.8258	4	UnderExpression: OverExpression:	-0.2: (YBR117C or YPR074C) 0.2: (YAL038W or YOR347C) 0.3: YDR272W 0.3: YML004C
66.1212	**0.75482**	**4**	**OverExpression:**	**0.2: (YAL038W or YOR347C)** **0.3: YBR291C** **0.3: YDR272W** **0.3: YML004C**
71.3569	0.69647	5	OverExpression:	0.2: (YAL038W or YOR347C) 0.3: (YNR001C or YPR001W) 0.2: YBR006W 0.3: YDR272W 0.3: YML004C
68.883	0.76432	5	UnderExpression: OverExpression:	-0.3: YER065C -0.1: YLR058C 0.2: (YAL038W or YOR347C) 0.3: YDR272W 0.3: YML004C
71.3364	0.69697	5	UnderExpression: OverExpression:	-0.3: YER065C 0.2: (YAL038W or YOR347C) 0.3: YDR272W 0.3: YML004C 0.2: YMR250W
72.9662	0.67009	5	UnderExpression: OverExpression:	-0.3: YJR148W -0.2: YLR027C 0.2: (YJL052W or YJR009C or YGR192C) 0.3: YDR272W 0.3: YML004C

Table 3.4: Some selected solutions with low number of gene sets regulations obtained by the second stage fine tuning redirector simulation.

Again, among all the results of this step, a new strain is selected, with a good trade-off of Growth Rate, Lactate production and total number of regulations on the gene expressions. The solutions involving the lowest changes are reported in table 3.4. The selected solution, highlighted in the table, has 4 gene sets over-expressed and none under-expressed. The genes involved are *CDC19, PYK2, CTP1, GLO1* and *GLO2*; the latter two are not surprising, as they are directly involved in the pathway producing lactate in *S.cerevisiae*. The first two are instead genes catalyzing the pyruvate kinase, and this is not surprising as well, as the lactate produced is directly derived form the pyruvate. The last gene, *CTP1*, is a tricarboxylic acid transporter localized in the mitochondrial membrane, involved in transport of citric acid through the membrane and so related to the citric acid cycle pathways.

On this solution, a final simulation was run, but this time changing the medium of the strain, namely the maximum flux of the exchange reaction, as we have seen in the *Medium optimization* section. So the behaviour of this strain, with 3 gene sets knocked out, 4 gene sets over-expressed, was simulated in different external mediums. The Pareto-Front obtained after this simulation is very close to the one that results from the redirector phase (see Figure 3.8). I believe that this behaviour highlights once more the different chances of optimizations that could lead to similar phenotypes. Applying either the gene knockouts search, or the redirector, or the medium optimization, it is possible to obtain the same behaviour, at least considering these two quantities of interest at the moment. It is then reasonable to further investigate this kind of hybrid methodology that could, on one hand, minimize the impact of a single approach, e.g. on the genome with the number of deletion from the wild type, and, on the other hand, ensuring at the same time a sub-optimal result. It is important to point out the relative simplicity of this mixed approach, still largely based on the FBA, and that only requires a smart algorithm to explore the different solutions spaces.

3.5 Future Improvements

I believe the multi-stage approach could be further expanded and used for a variety of applications in metabolic engineering and strain design.

Alongside new optimization with the approach that I have just described, it would be possible to change the order in which the stages are performed within the optimization procedure; perhaps applying a circular workflow rather than a linear one, so with the different stages performed more than once in a recursive way could be an even more powerful tool.

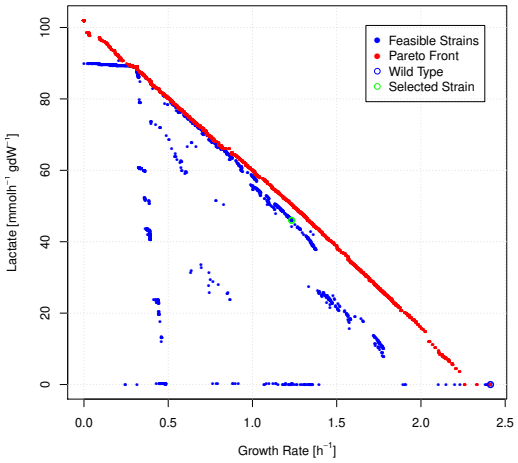
A key point is also the definition of the medium; the chemicals that supply the metabolic network are indeed crucial for the prediction; they both affect the behaviour and the fluxes inside the network in the wild type and the selected genes to be knocked out in the model and that enhance the synthetic objective production during the algorithm that I have described in the previous section. The optimization alone try to yield enhanced values in the production, but on the other hand, it could be of interest to initially optimize the medium and then run the other two stages in order to find strains that could adapt to a decided specific medium setting. The optimization of the medium could then be seen as an optimization of the costs that the nutrients come along with. Again on this topic, it could be also interesting to define a specific medium, for example derived from industrial or biomass waste to be used for the production of various compounds, and then force the two optimization algorithms to find performing strains in this specific conditions. This is, I believe, a key point that need to be addressed over the next years, to improve the sustainability of the production chain and processes, and to follow the concept of circular economy.

Moreover, the same approach I have described since now, could also be used changing the objective function to be maximized. Barely considering, as before, the simple value of the exchange flux relative to the metabolite of interest is not always the best choice of objective and involving other measures to measure the fitness of the strains could be the right choice. Simple quantities to be used for this could be the theoretical *yield* and the *productivity*; the yield is defined as the rate of production of the desired chemical for each unit of glucose (or of another used substrate) that has been fed to the cell; the yield is then expressed as $\left[\frac{g_{prod}}{g_{sub}}\right]$. The productivity is instead defined as the yield times the growth rate of the cell,

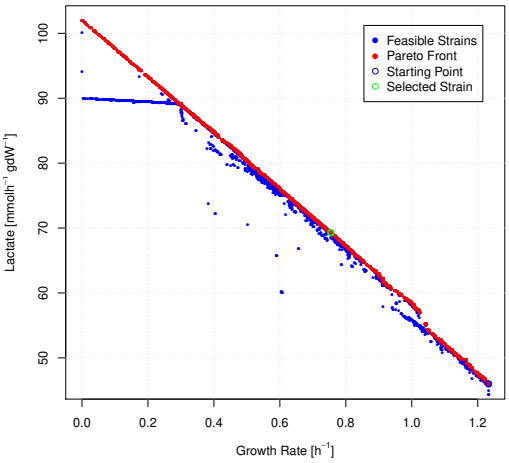
thus having the dimension of $\left[\frac{g_{prod}}{g_{sub}} \cdot time^{-1}\right]$

$$Yield = \frac{v_{obj}}{v_{glc}} \qquad Productivity = \frac{v_{obj}}{v_{glc}} \cdot GR$$

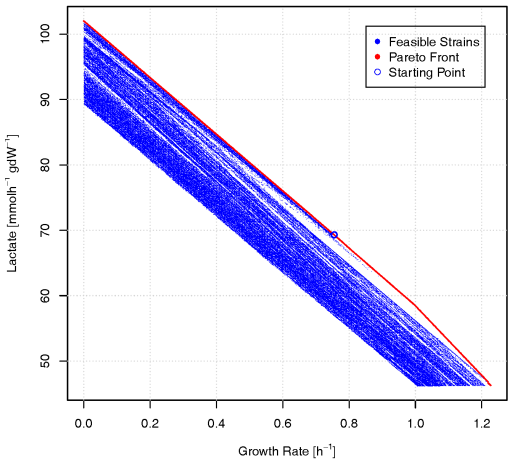
These quantities involves more than one flux at the same time and are of particular interest in the design of engineered strains and can be safely considered as objectives of the optimization framework. In the case of the maximization of productivity, since the growth rate is present as a factor within its definition, it would be redundant or at least less significative to include the growth rate alone as another objective. A second objective that in the future I am planning to consider is to maximize both the minimum and maximum productivity at the same time, using an approach similar to the *Flux variability Analysis* [21], in order to obtain strains that have an increased productivity in all the range of conditions, at least as it is predicted by the *FBA*.



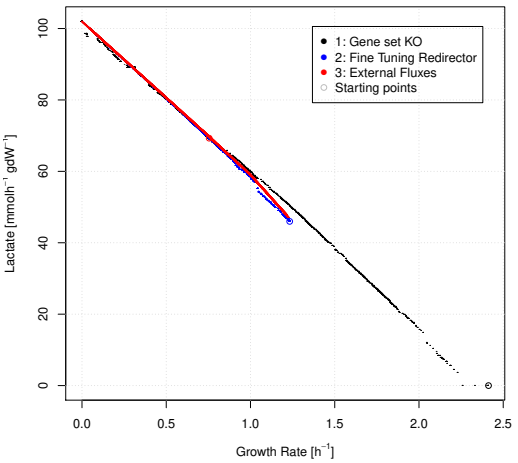
(a) The first stage is the standard knockout optimization selection through the genetic algorithm.



(b) The second stage take start from a notable strain found in the first stage and further optimize the production of Lactate through the fine tuning redirector evolutionary algorithm.



(c) The final stage changes the external bounds regulating the uptake of nutrients from the simulated environment to further optimize the production starting from a notable strain selected from the stage 2.



(d) The three Pareto-Front from the three stages of the optimizations are compared. The behaviour are similar, highlighting the various chances of optimizin the predictions using the genome-scale models.

Figure 3.8: The various stage of optimization. At the end of the first two steps a notable strain is selected as the base of the subsequent optimization. Note that the knockouts selected in the first stage seem to set bounds for production and growth rate, as no points found in the following have a lower or higher values respectively.

Chapter 4

Minimal Metabolism Design by Computational Synthetic Biology

4.1 The minimal genome

In this part of the work I will focus on an application of the genome scale models approach on the minimal cell problem.

The Synthetic Biology is at the threshold of a golden era after the unveiling of the powerful technology of CRISPR-Cas and its broad-spectrum applications, enabling wholesale genome engineering with almost any biotechnological organism[63]. Also the progress in synthetic genome projects such as Yeast 2.0, facilitate pathway engineering and chromosome reconfiguration[64], while the first example of minimal synthetically generated functional cells have been deployed [2, 1].

The minimal cell has been defined as the unicellular organism with the smallest genome that is able to sustain all the basic activities of cellular life, namely the ability to replicate the DNA, transcribe the RNA, translate proteins, perform mitosis and sustain a normal growth. The theoretical minimal cell would include only the *essential* genome that is needed to perform these tasks efficiently[65]. Such a cell, that has also been defined as the hydrogen atom of biology, could highlight a variety of mysteries on the principle of life. All the genes present would have a specific and determined function inside the cell, there would be no

redundancies in the genome, and we would have a complete comprehension of the cellular mechanics linked to specific genes[9]. Clearly, there are still some critical points in these analysis; not all the genes have a known functionality, and some genes have more than one mechanic that can be expressed from them; other genes' behaviour and impact in the cell is affected by the presence and expression of others and so on and so forth.

On one hand, this is a major problem for the design and the realization of a real minimal cell because, despite the latest analysis, a complete knowledge of the genes is still missing, even for the simplest organism, has still not been reached at this time[66]. On the other hand, a minimal cell could theoretically be used as a tool for new analyses and on the genes by adding them to the genome and observe the changes of the cell's behaviour in the same environmental condition. In a similar approach manner, a minimal cell could be used as the first building block to design and create more complex cells able to do specific task with a more precise methodology, given that it would be easier to avoid waste of energy and nutrients and focus on the pathways of interest.

The search for the theoretical minimal genome of a living cell has been intensively studied in the last decades and has then arisen as one of the most important issue to achieve a deeper understanding of the principles of life[65, 9]. A minimal cell would be capable of all the basics functions[67], and would comprehend only an essential genome, in which every single gene is needed for the survival[68]. The identification of essential genome has several important application[69] and has been increasingly addressed by new researches, from the theoretical and comparative genomics approaches[70, 71, 72, 73, 74], to the latest real minimal cells[75, 2]. Although the great steps taken in the field, there are still many genes, as I said, that have not a clear function; also, there are potentially many possible configurations of a minimal genome[65, 71].

One could imagine that the minimal genome is formed only by recognized essential genes of the organism, hence, for the organism with (almost) complete information on the gene deletions effect, the construction would be relatively straightforward. Indeed, this is not the case occurring in nature; the set of only all the essential genes does not lead to a viable strain of the organism. The main phenomenon behind this is the redundancy that some organisms have in their genome.

As an example of that, there are genes inside the cell that can provide the same essential function even though they are not paralogs. In such a case, the single deletion of one of those genes maintain the cell viable, but if both of them are not present, there are no longer genes able to supply that essential function hence the cell is no longer viable. There is still a lack of proper definition for what is really essential for the survival of the cells[65], as the definitions could be contingent or open to different interpretations, aside a still imperfect knowledge on the functionality that I have already mentioned.

Now, it is necessary to strip down the metabolic network of key cell factories such that they provide a basic workbench on which novel pathways may be elaborated while minimising diversion of metabolites away from the desired product.

4.1.1 The proposed computational approach

I developed an automatic procedure to tackle the minimal cell problem in a systematic way; the computational pipeline can define a minimal metabolic network for any sequenced organism and tested it using a *Saccharomyces cerevisiae* genome-scale model, demonstrating significant quantitative and qualitative differences between the minimal networks required for fermentative and respiratory growth. Extending the study to 20 other species, from bacteria to humans, comparing the resulting networks with the JCVI-syn3.0 genome[2, 1]. The *in-silico* pipeline sequentially removes genes encoding enzymes and transporters. At the end of the procedure, described in detail in the Methods section, the resulting minimal set of genes still ensures the fundamental metabolic functionalities expressed by the growth rate value predicted by Flux Balance Analysis[76]. I have used this pipeline to define a large number of unique minimal metabolic networks for each genome-scale metabolic model of various eukaryotic and prokaryotic organisms but have focused on Saccharomyces cerevisiae in different nutrient environments. The mandatory genes present in most networks and their role in the corresponding genetic interactions network are then investigated using simple metrics from the complex network theory. The results demonstrate the complexity of the minimization problem, and I analysed the networks to discover similarities and recurrent redundancies among the minimal metabolic networks and to highlight some of

the homologous genes that are retained in the networks of a majority of different species.

4.2 Results for *Saccharomyces cerevisiae*

the framework takes start from a wild-type stoichiometric model, including all the genes that have been included encoding the enzymes and transporters catalysing all the reactions happening in the model. Thus the genome of the starting (wild-type) strain contains all these genes and the algorithm seeks to minimize the number of active reactions (i.e. maximize the number of genes removed from the initial inventory), while keeping the biomass formation value predicted by Flux Balance Analysis above a strict threshold based on the wild-type value. The algorithm then selects a "minimal" network (MN), as a mutant strain in which all the reactions that are still active are "essential", i.e. no more enzyme-encoding genes can be deleted without violating the biomass formation threshold. The maximum admissible reduction in the biomass formation value was set to either 1% or 10% of the initial value. To further improve the reliability of the results, another constraint was added, namely that any gene in the model that was known, from experimental data, to be essential[60], could not be deleted by the algorithm. At first I focused on the metabolic network of Saccharomyces cerevisiae s288c, using the last available version of the consensus yeast genome-scale stoichiometric model at time, yeast 8.3.1[77] (see the Methods section and Table 4.1). Environmental conditions, and particularly the chemical composition of the growth medium, affect not only the metabolic behaviour of the cell, but also determine whether a given gene or reaction is essential for growth[78, 79] and even the definition of the minimal genome itself[80]. Hence the predictions on the minimal network and on the genome minimizations do also change in different external conditions, which are defined in the genome-scale models as a set of exchange reactions whose lower bounds regulate the maximum possible uptake rates of nutrient compounds, as I have introduced in the previous chapters. I have considered seven different growth media for S.cerevisiae from literature or defined (entirely or partially) by an ad-hoc algorithm (see Methods). For two of the media, simulations were performed under anaerobic conditions. In the following, if not specified, I always consider the

richest condition (SD in aerobic and "Glucose" in anaerobic), in agreement with previous studies[2, 1].

The optimization algorithm is based on an Evolutionary Algorithm[81], evolved from a greedy hill-climbing approach[15], to iteratively select the reactions/genes to be deleted in order to find the MNs. Over a maximum number of generations, the algorithm selects new knockouts for a given set of strains that constitutes the current population of points, promoting the solutions with a greater Hamming distance from the others. with a metric evaluation. If a solution is not improved, i.e. no new knockouts are selected, it is increasingly probable that it will be discarded and substituted by an "ancestor" point obtained by backtracking the corresponding leaf of the search tree. (See Methods for an exhaustive description of the algorithm). When the algorithm reaches the last generation ends, a post-processing procedure is launched to expunge the solutions that show no improvements and extract the genuine, and distinct, MNs. This procedure also performs a series of further analyses, including the evaluation of flux distributions using parsimonious Flux Balance Analysis (see Methods). The number of unique MNs found is a little more than 1000 in all the conditions (see Table 4.1).

Figure 4.1 shows the basic properties of the MNs in the 7 different conditions and with a maximum of 1% reduction in the rate of biomass formation. From this Figure, it is immediate clear how that the most significant difference between all the MN solutions is that between the MNs found under the two anaerobic conditions and those arrived at for aerobiosis. The minimal solutions in the anaerobic states are sensibly smaller in terms of the number of genes involved. This difference is maintained when the more relaxed constraint of a 10% reduction in the rate of biomass formation is employed, although (as might be expected) the distributions are shifted to a slightly smaller number of genes/reactions in the MNs.

Figure A.5 present pairwise comparisons of the results with the two different thresholds and the seven different growth environments.

I have also used basic measures from the complex network theory[82] to describe the MNs (Figures 4.3a-4.3b-4.3c-4.3d) considered as a bipartite graph of reactions and metabolites; there are two different behaviours in the weights distributions for aerobic and anaerobic (a pattern that is kept from the Wild Type networks), while the degree distributions are all similar. The mean and standard deviation of

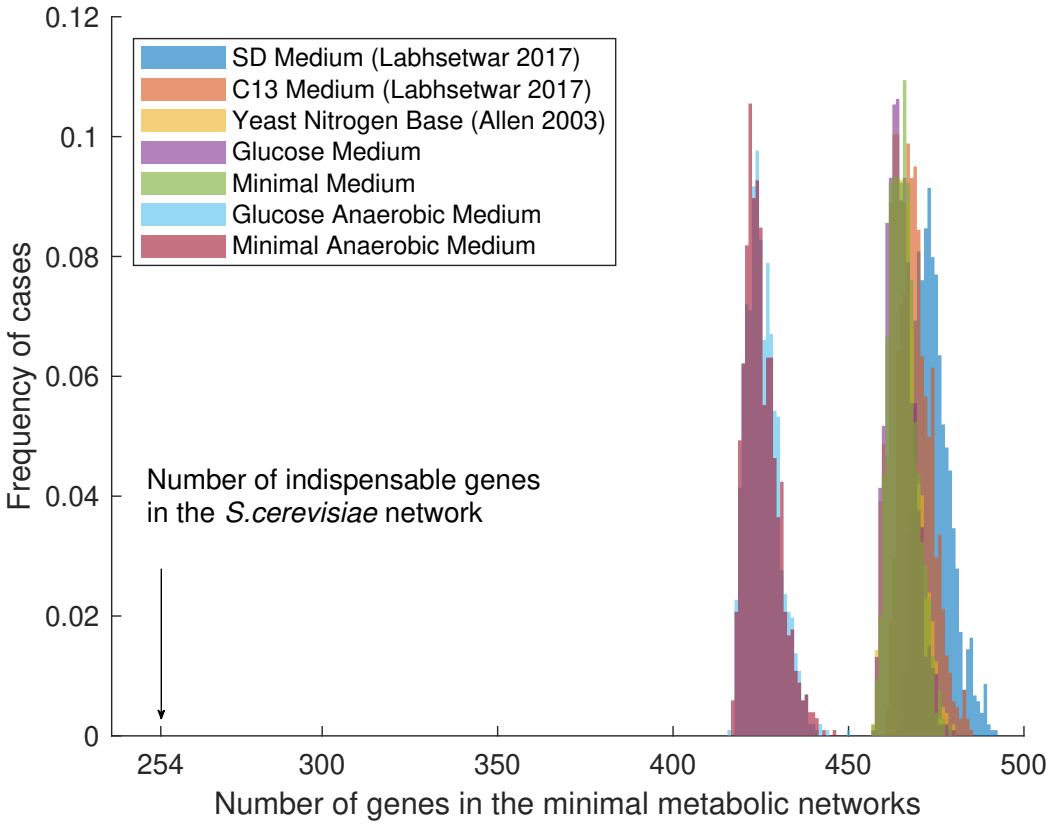


Figure 4.1: Frequencies of MNs active genes; two different clusters are present, for aerobic and anaerobic conditions.

the weights show how the aerobic networks in different conditions are divided in two sub-clusters.

A particular analysis is needed for the results in the richest condition, the SD medium; as highlighted from the Figure 4.1, the MNs include on average a larger number of genes than the results in the other conditions. This appears counter-intuitive since the SD medium provides a large number of complex biochemicals, e.g. amino acids and nucleic acid bases, thus relieving the need for the minimised strains to retain the metabolic pathways involved in their biosynthesis. This is due to the high initial biomass yield of the wild-type strain imposing a much higher quantitative threshold for the permitted reduction in the rate of biomass formation than was the case with the other (less complex) media; this high minimal value

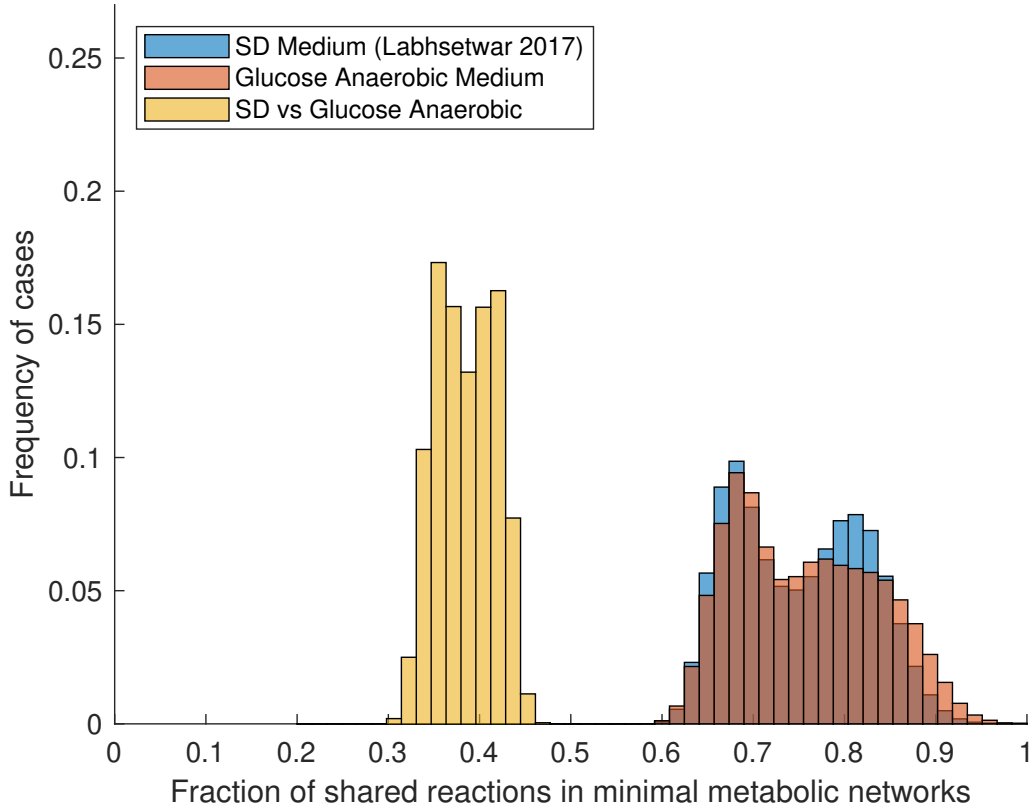
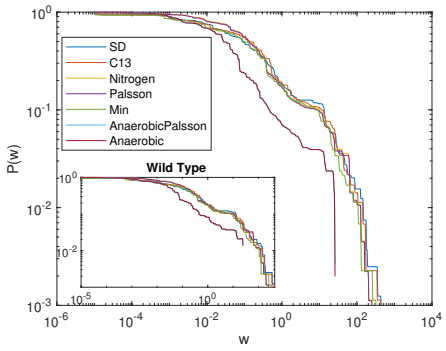


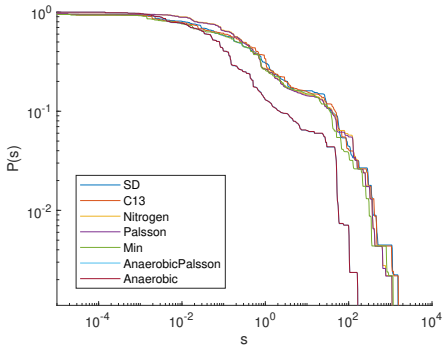
Figure 4.2: Fraction frequency of MNs shared active reactions through a pairwise comparison. Considering the networks in the same conditions the shared reactions are 60 to 90%, while comparing networks in different conditions the fraction is 30 to 50%.

thus requires these anabolic genes/reactions to be present in all the minimised networks. As a confirmation of this, I compared the results with the relaxed biomass constraint, to find out that the difference in the SD medium is wider if compared to the other conditions. In addition to this, comparing the genes present in the MNs in the different conditions, there is a stronger presence of transporters-related genes in the MNs in SD medium. These genes, which are a large fraction of the model genes, influence the final size of the MNs; their presence could be explained by the necessity of import and diffusion mechanics for the larger number of external complex metabolites to the cell compartments.

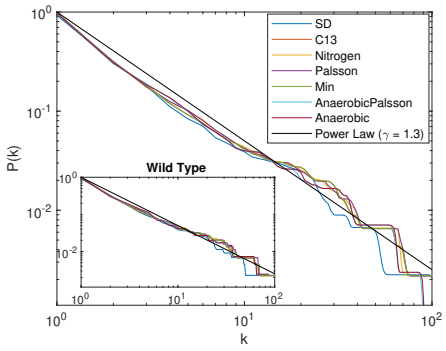
I then performed an extended analysis on the genes of the MNs and the same



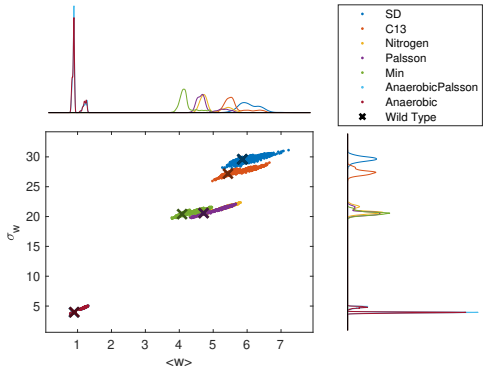
(a) Average weight distribution in MNs; aerobic and anaerobic networks show different distributions



(b) Average node strength distribution in MNs; aerobic and anaerobic networks show different distributions



(c) Average degree distribution in MNs, similar for all the conditions MNs.



(d) Plot and frequency distribution of the mean and standard deviation of weights in the MNs, divided in three operating regions.

Figure 4.3: Minimal Metabolic Networks as described by basic measures from the complex network theory.

analysis can be performed on sub-sets of the conditions, e.g. the five aerobic conditions or the two anaerobic condition, or all seven. A set of *mandatory genes*, i.e. the genes that are always present, or that are present in most of the MNs (more than 95% of them), emerged from this analysis. In Table 4.1 I have reported the number of mandatory genes for each external condition and the shared ones in all the conditions or in aerobic/anaerobic environments only, while the gene list in

Table A.3. I considered two measures of redundancy and robustness applied on the pathways of the model, defined as the fraction of the pathway-related genes that have been deleted and the capacity of a MN to bear small variations in the pathway fluxes (see Figure A.8 and Methods). In the two conditions there are differences in some of the functional categories of the genes and of the pathway related to them. The pathways in presence of oxygen are in general significantly more robust, while the redundancy values are more similar between conditions, with the notable exceptions of the TCA cycle and a few amino acids-related pathways.

4.2.1 Considering the network of genetic interactions

To further evaluate the importance of the mandatory genes I also considered their role in the networks of genetic interactions. It has already been studied how the network topology and the essentiality of a gene are related[18]. I tried then to see if some topological measures or analyses in the genetic interactions network show a significant relation with the importance degree emerging from the results I presented.

The average behaviour shows for example that removing the mandatory genes affect significantly more the network considering the Latora-Marchiori efficiency[17] (Figure 4.4). Also, other simple measures of importance, e.g. degree and betweenness, show an increase in the cumulative distributions related to the frequency in the MNs (Figure 4.5), again with different behaviours in absence of oxygen in the media. It is important to highlight that, even though the average value for the genes that are (almost) never present in the MNs is low, there is a great variability in the set of such genes, because most of the genes are scarcely present. This confirms that simple topological analysis is still not sufficient alone for an exact classification of every gene but it could be useful to point out some candidate genes. Specifically, the mandatory genes showing a significantly higher value in the degree and betweenness are ARO1, ARO2, FUM1, OPI3, GAP1.The GAP1 presence, though, is actually essential only in the SD medium, which include external amino acids. In all the other conditions it is forced to be present by the constraints on the synthetic lethal deletions included in the simulations, but it is not related to essential reactions in the MNs. Thus, the constraints used can in-

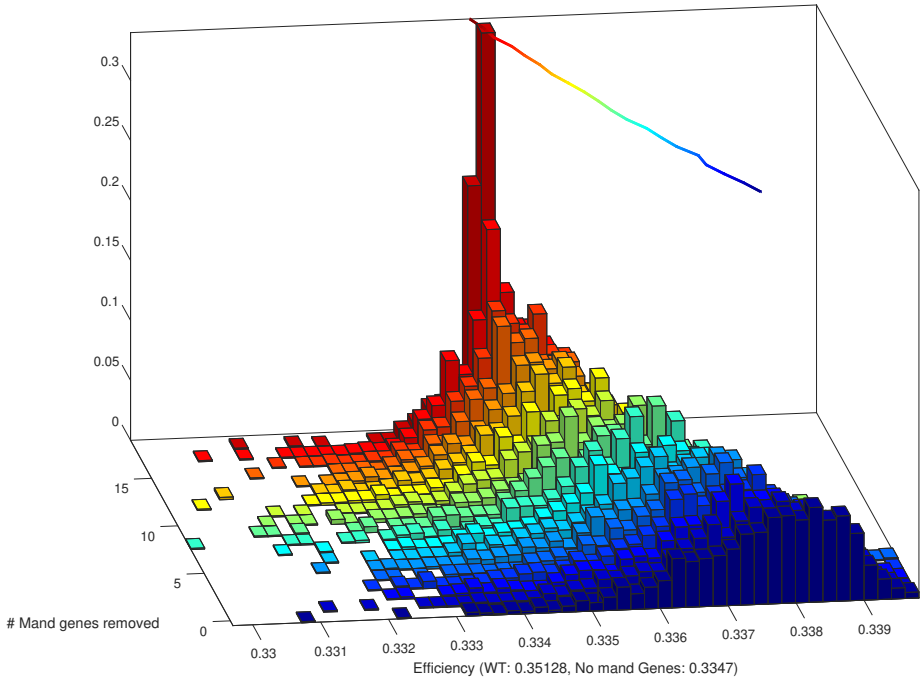


Figure 4.4: Evaluation of the Latora-Marchiori efficiency of the interactions networks with the same number of genes as the mandatory genes removed. The network with a higher number of mandatory genes removed have a sensibly lower efficiency, on average.

fluence the results preserving important genes and enhancing their feasibility, but sometimes introducing unnecessary changes due to the different environmental conditions affecting the essentiality of some of the genes.

Moreover, a few essential chemical reactions can be equally activated by more than one gene[1]. If these genes are not involved in any other reaction the frequencies usually lie on values close to the uniform random distribution, e.g. two genes that can both activate an essential reaction will be present with a frequency close to the 50% each; this is the case of the ARO3 and ARO4 genes. This phenomenon also explains why more genes clusters around certain frequency values, specifically around the 50% and 66%.

A comparison of the cumulative distributions of degrees for genes with similar

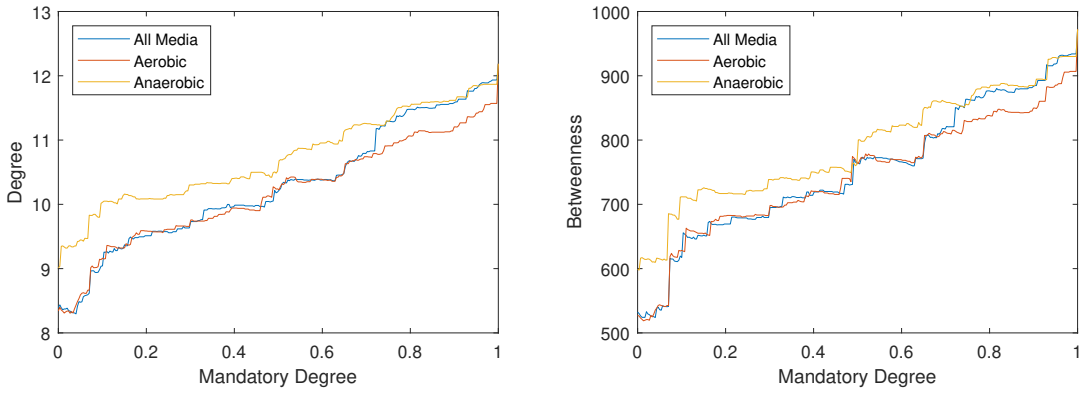


Figure 4.5: Distributions of the degree and betweenness of the nodes in the interaction network; for every frequency value x (mandatory degree) the corresponding point in the plot is the average value for all the points with a frequency less or equal to x.

frequency values (Figure 4.6) confirms the relationship between the two quantities; it is also possible to infer the power laws (Figure 4.7) for the extreme values of frequency sets, i.e. the mandatory and essential genes, and the genes always discarded. Even though the curves are close from each other, the two confidence intervals of the gamma values don't overlap.

4.2.2 Synthetic Double Deletions Prediction

Another benchmark on the predictions I considered is the synthetic double deletions predictions of the MNs compared to the wild type network using data from two different databases[29, 28] (Figures 4.8 and A.6 respectively; see also Figures A.7a and A.7b for boxplots on the same data). A clear improvement in the true positive predictions is obtained, but at the expense of an increase in the false positive as well. Hence the differences are due to a general fragility of the MNs, that are more susceptible to the simulated gene knockouts. This is also confirmed by the single and double deletions of the mandatory genes in the MNs if compared to the wild-type case. The deletions in the MNs reduce on average more heavily the theoretical capacity of the network to produce the biomass precursors as there are less (and more important) pathways in it to reach the compounds of interest.

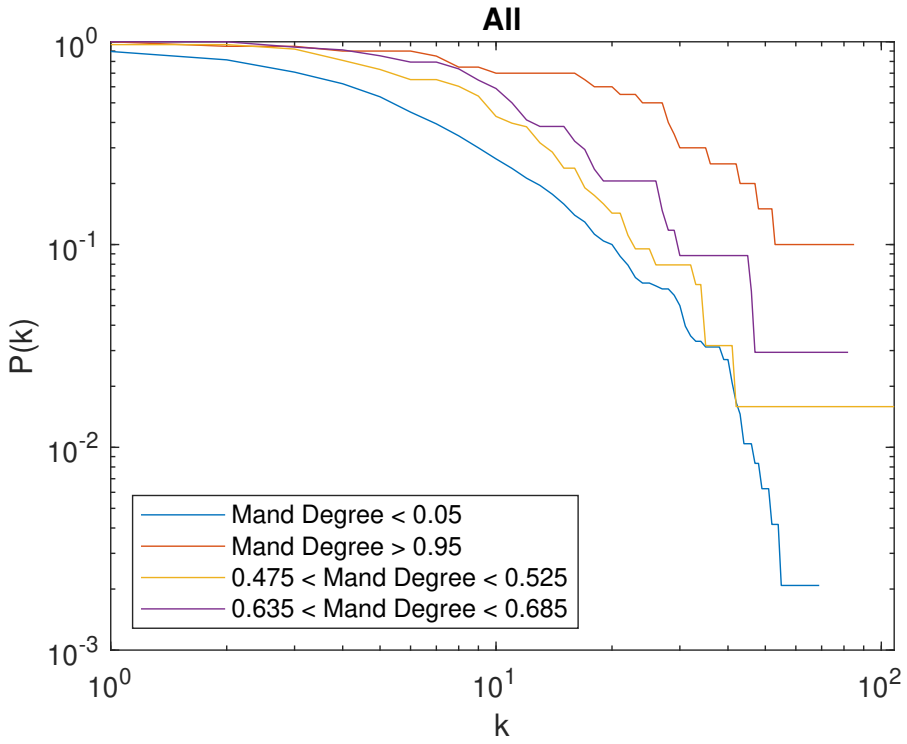


Figure 4.6: Cumulative distribution of the nodes degree in the network, showing different curves for nodes with different frequency.

I analysed then the genes grouped using their multiple GO slim annotations of the genes. In Table A.4 I included the comparative analysis of the WT with the average MNs in the SD and Anaerobic medium. Among the most numerous categories, the most affected are the hydrolase and the transmembrane transporters, with an average reduction of the 77.10% and 82.83% respectively while the least affected are the transferase groups and the amino acid metabolic processes. So, despite the reduction in the transporters is lower in SD, as noted before, it is still significant.

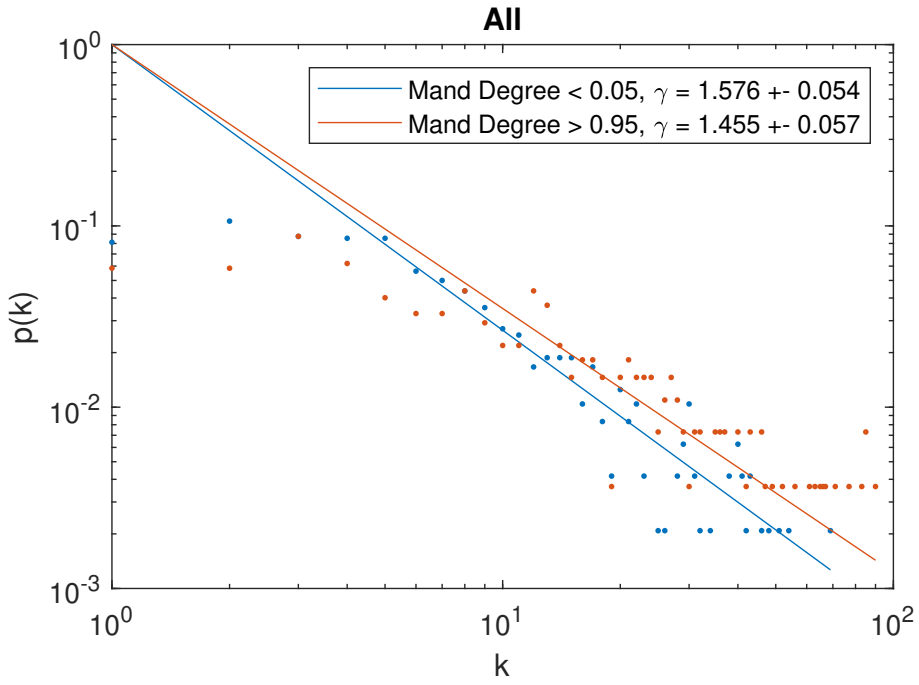


Figure 4.7: Power laws in the degree probability plot.

4.3 Extending to other *Saccharomycetales*

After considering an extensive analysis on the *S.cerevisiae*, the next step was to use the same pipeline on a series of other metabolic models, starting with the metabolic network of three other species in the *Saccharomycetales* class, *Schizosaccharomyces pombe*, *Komagataella phaffii* (syn. *P.pastoris*) and *Yarrowia lipolytica*. Here it is important to remember that the evolutionary distance between *S.cerevisiae* and *Y.lipolytica* is as great as that between *H.sapiens* and *Ciona*[83]. Figure 4.9a shows the comparison of the resulting distributions. The sizes of the models (see Table 4.1) are various, and this, together with the evolutionary distances, are probably the main reason behind the disparity in the results, even considering the same simulated rich medium and, for *K.phaffii* and *S.pombe*, also the same inventory of essential genes.

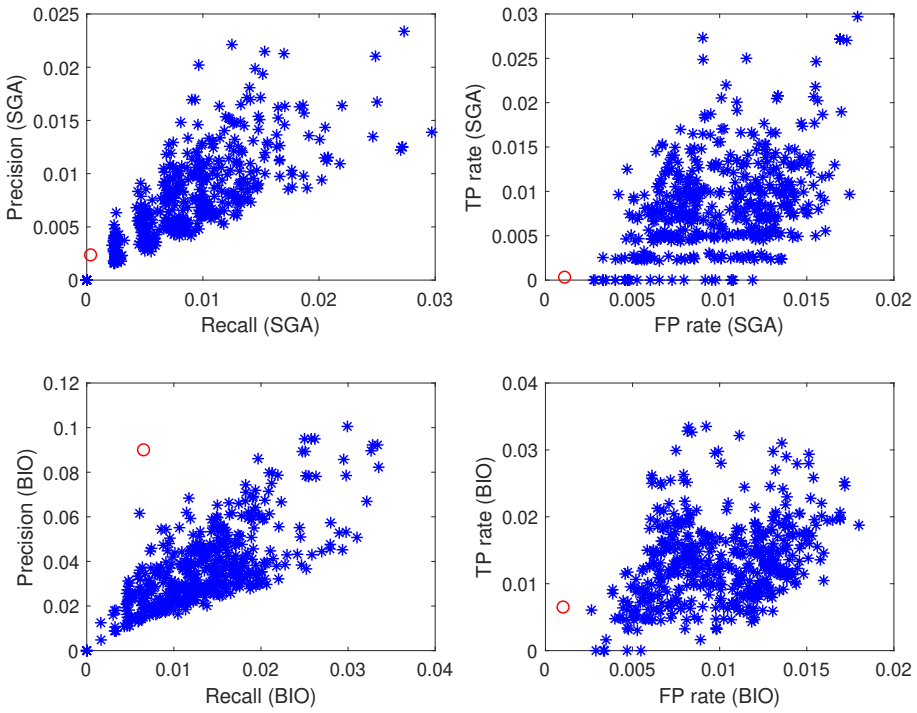
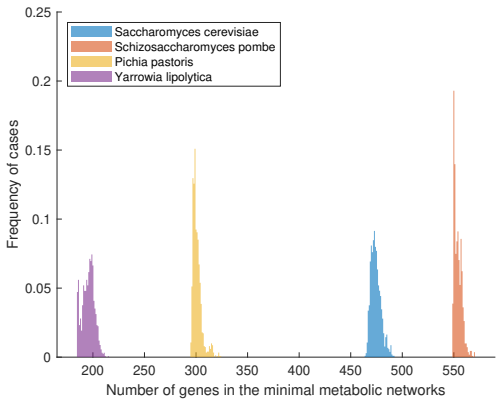
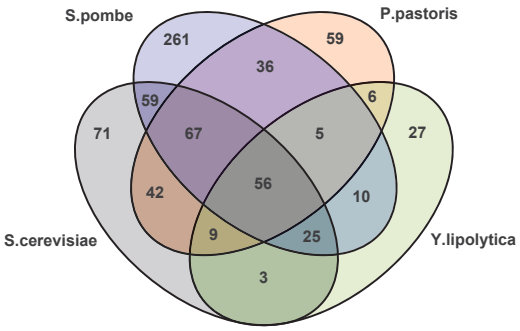


Figure 4.8: Comparison of prediction for synthetic double deletions in the MNs; the red circles and the blue horizontal line in the boxplots represent the WT values, the other data refer to the MNs.

A BLASTp search (10-5 as expected value) using the mandatory genes in the S.cerevisiae results as query and dividing the results by compartment annotation is reported in Figure 4.10; the search of the mandatory genes is performed versus all the models' genes for the other organisms. Figure 4.9b shows instead the mandatory and essential genes shared by the different models; there are several genes that are uniquely present in only one of the models. However, 56 of them have orthologs in more than 85% of the models' networks results; most of these genes are related to various compounds biosynthesis, and in particular 13 of them are ERGosterol biosynthesis genes. There are 4 of these genes that are not marked as essential (YNK1, CPA2, PFK2, OPI3); notably, YNK1 and PFK2 are the manda-



(a) The frequencies of MNs active genes for the 4 different models cover different intervals, due to the size of the models.



(b) Quantification of similar mandatory genes shared by different models using a BLASTp comparison of the mandatory genes.

Figure 4.9: Results comparison for the 4 Fungi species considered.

tory genes with the greatest impact in the production of the biomass precursors in the yeast model.

4.4 Further extension to broad-spectrum models

Finally, I considered a variety of genome-scale models, from M.genitalium to H.sapiens and applied the same pipeline on them. In this case I represented the results comparison using also a heatmap and a hierarchical cluster analysis (Figure 4.11) to highlight the similar results; I used the KEGG Orthology information to compare the genes from the various organisms, including the JCVI-syn3.0 metabolic genes[1] as a benchmark. The clustering shows a reasonable division of the species, meaning that the mandatory genes keep a specificity, even though it does not follow the exact taxonomy. results. The eukaryotic and prokaryotic organisms have in fact more similarities within the two groups even considering the mandatory and essential genes only.

Figure 4.12 shows instead the quantitative results of the BLASTp search using the JCVI-syn3.012 genes as a query on all the mandatory genes. More than half of the JCVI-syn3.0 genes have no hit in the other organisms, but this is expected,

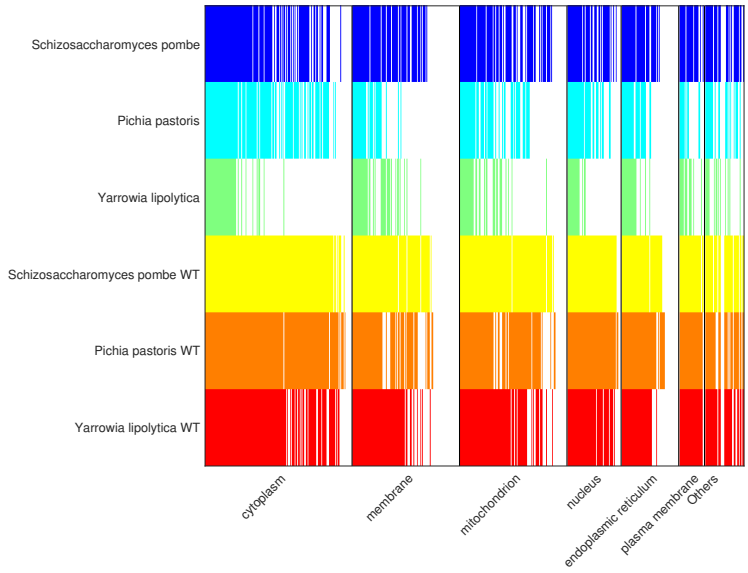


Figure 4.10: BLASTp comparison of the mandatory genes; every column is a S.cerevisiae gene, and a hit in the model of the row is coloured, while a white space means no hit in that model. The genes are divided using the GO slim compartments annotation.

because the genome scale models simulate only the cellular metabolism. The JCVI-syn3.0 has in fact only 82 genes in the metabolic functional group[2] and only 155 genes were included in its genome-scale models[1].

4.4.1 Comparative analysis

Comparing this minimal genome with the results obtained by the algorithm, most of the kept mandatory genes belongs to the central carbon metabolism group, clearly required for the survival across the cells in its different possible configurations. The genes in the other groups (especially of unknown functionality) show a lower or null frequency in the results. Quantitatively, for example 9 genes (56 considering also the essential genes) have a similarity with the mandatory genes of S.cerevisiae as found by the BLASTp search, with a slightly higher correspondence in the membrane transport group if compared to other eukaryotic

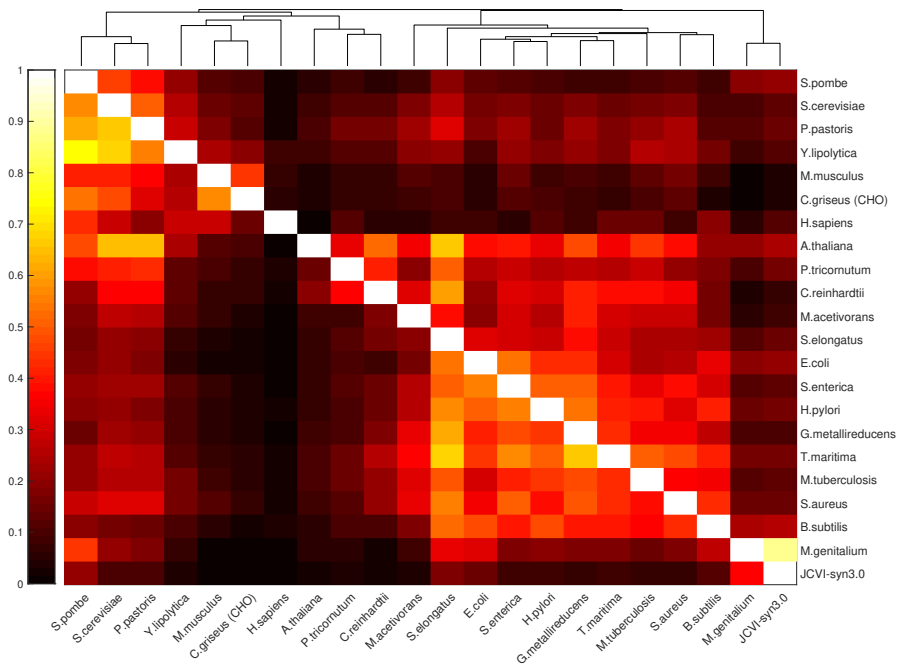


Figure 4.11: Global comparison among all the models used in this study, considering the mandatory genes for each model. The genes are considered using their KEGG Orthology annotation, so the different models can be compared, and the results are reported in a heatmap showing the more and less similar models in terms of the MNs found. Each element i,j represents the fraction of mandatory genes found in the model i that are also present in the mandatory genes of model j. The entire genome of the JCVI-syn3.0 is also included

organisms; 3 genes out of 9 encode ATP synthase subunits, while the other 6 (IPP1, SHM1, LPD1, RPE1, PFK2, ALD6) encode enzymes with various functions in the metabolism. The most significant alignment is, unsurprisingly, with the M.genitalium results with 147 mandatory genes found in the syn3.0; only 89 of these genes, however, have a similarity with some of the 155 genes in the syn3.0 genome-scale model. Almost all of the 58 genes with no correspondence in the model are annotated with translations functionality whereas the corresponding genes in the M.genitalium model are directly related to the simulated growth

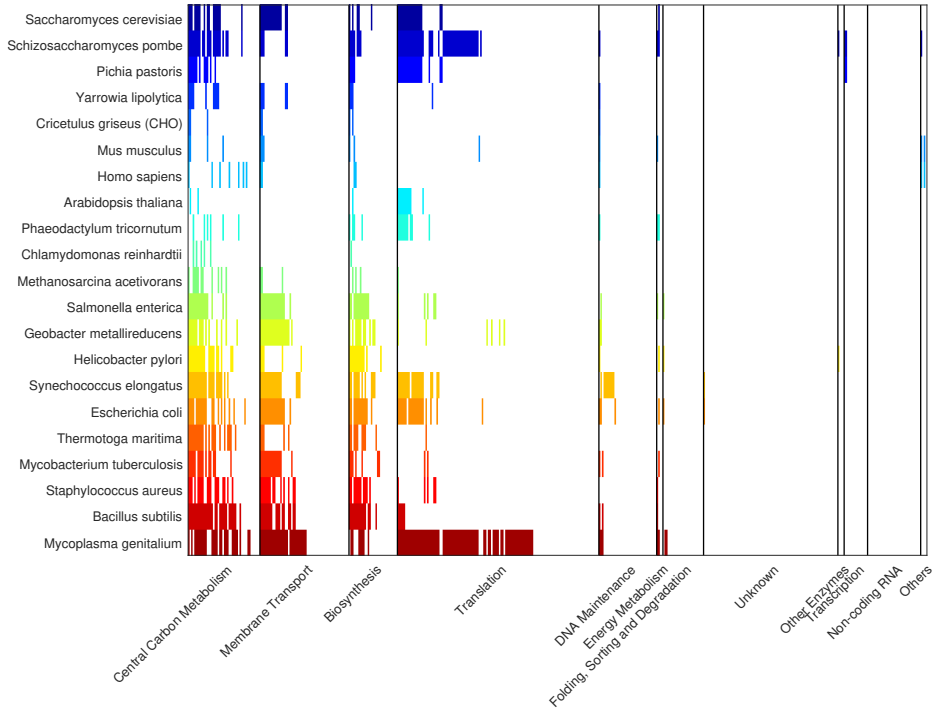


Figure 4.12: A comparison using the BLASTp search, like the one of Figure 4.10, using the JCVI-sn3.0 genes as a query vs all the mandatory genes for all the organisms. All the genes, one per column, are divided by secondary functional classes.

reaction or to various dipeptide transport reactions. For all the other models, most of the sequences are present in the metabolic model as expected. Following the classification in Essential, Quasi-Essential and Non-Essential of the syn3.0 genes[1], I compared the number of sequence alignments across the organisms. The Essential genes are more present, on average, in other organisms' mandatory genes than the 13 Non-Essential genes, showing predictions in accordance with the experimental data on the "minimal" cell (see Figure A.10b). Besides the comparison with the JCVI-syn3.0, I finally considered four other available minimal genomes studies[74, 15, 84, 85] for E.coli and B.subtilis, for a comparison using the corresponding model results from the pipeline (see Figure A.10) and obtained a significant fraction of shared genes between the results and these studies.

Organisms	Model	Genes	Ess	Thr	Medium	#MNs	max KOs	min KOs	mean KOs	Mandatory Genes		
Saccharomyces cerevisiae	yeast8.3.1	1133	254	1%	SD	1039	669	641	658.63	59+20	50 + 21	11 + 28
					Minimal Aerobic	1051	676	654	667.41	57+16		
					C13	1042	672	648	663.45	58+18		
					Nitrogen Base	1046	676	653	667.42	57+19		
					Glucose Aerobic	1063	676	653	668.00	57+16		
					Glucose Anaerobic	1014	717	683	707.36	20+16	20 + 19	
					Minimal Anaerobic	1014	716	687	707.72	20+21		
				10 %	SD	1047	682	640	667.59	39+21	39 + 22	9+ 21
					Minimal Aerobic	1031	683	653	672.99	40+23		
					C13	1070	681	648	669.12	40+24		
					Nitrogen Base	1038	685	645	672.42	40+21		
					Glucose Aerobic	1067	684	655	673.05	40+26		
					Glucose Anaerobic	1045	722	691	712.18	13+19	13 + 18	
					Minimal Anaerobic	1035	722	688	712.10	13+18		
Schizosaccharomyces pombe	iSchpo_972h	1065	360	1%	SD	1109	516	495	511.15	113+5		
Pichia pastoris	Kp1.0	720	230	1%	SD	1504	425	396	419.02	17+3		
Yarrowia lipolytica	iYL619_PCP	619	N/A	1%	SD	1250	434	404	423.48	125+7		
Cricetulus griseus (CHO)	iCHOv1	1766	N/A	1%	Default	400	1614	1567	1595.87	90+1		
Mus musculus	iMM1415	1375	N/A	1%	Default	539	1181	1107	1164.95	113+9		
Homo Sapiens	Recon3D	2248	N/A	1%	Default	284	2131	2078	2113.90	34+2		
Arabidopsis thaliana	AraGEM	1404	N/A	1%	Default	537	1282	1262	1274.83	43+2		
Phaeodactylum tricornutum	iLB1027_lipid	1027	N/A	1%	Default	12	656	615	644.75	241+5		
Chlamydomonas reinhardtii	iRC1080	1086	N/A	1%	Default	711	839	795	824.64	142+7		
Methanosarcina acetivorans	iMB745	745	N/A	1%	Default	1003	459	441	454.15	227+2		
Salmonella enterica	STM_v1_0	1271	N/A	1%	Default	545	951	926	943.74	250+5		
Geobacter metallireducens	iAF987	987	N/A	1%	Default	723	604	584	598.63	284+4		
Helicobacter pylori	iIT341	339	N/A	1%	Default	199	101	97	98.91	229+0		
Synechococcus elongatus	iJB785	785	N/A	1%	Default	794	270	239	264.46	463+7		
Escherichia coli	iJO1366	1367	113	1%	LB	600	1037	1002	1024.61	128+14		
Thermotoga maritima	iLJ478	482	N/A	1%	Default	1562	269	259	266.91	181+2		
Mycobacterium tuberculosis	iNJ661	661	N/A	1%	Default	1110	392	370	387.80	218+3		
Staphylococcus aureus	iSB619	619	N/A	1%	Default	1198	375	364	372.21	194+2		
Bacillus subtilis	iYO844	844	97	1%	LB	1017	605	583	598.3972	81+9		
Mycoplasma genitalium	iPS189	190	N/A	1%	Default	4	28	27	27.5	159+0		

Table 4.1: Models used in this study with the number of genes simulated in each of them and characteristics of Minimal Metabolic Networks found by the algorithm. A Default medium entry in the Medium column refers to a model that has not been changed from its original setting. The last column value is the number of mandatory genes divided in the number of genes always present in the MNs and the number of genes present in at least the 85% of the MNs but not in all of them. For S.cerevisiae the last two columns show the shared mandatory genes in the aerobic or anaerobic results, or in both types.

4.5 Methods

4.5.1 Models

The base of this study are the 21 genome-scale models, corresponding to 21 different organisms, as reported in the first two columns of Table 4.1. Given the deep differences among the considered prokaryotic and eukaryotic organisms themselves and their metabolism, the features of the models vary sensibly in terms of genes, metabolites and reactions.

During the media definition procedure described below for the S.cerevisiae, I used 3 other model to confirm the methods (*yeast 7.6[51], constrained yeast 7.6[61], yeast 7.6 with Fe Metabolism[86], iMM904[87]*).

KnockOut simulation

In addition to the rections, the models also include information on the genes of the organism. An array with all the genes is present in the structure, and many of the reactions are related to a logical rule involving the genes. The logical connectors AND, OR, simulate the isoenzymes and protein complexes. The presence or absence of the genes in the model is represented by a logical array, whit a True value corresponding to a present genes and vice versa. Every different solutions considered in the analysis can then be represented by a logical array. The wild type is so the all True array and every single mutation (see 4.5.3) is the switch of one of the logical value of the array.

4.5.2 Minimal Media

The exchange reactions of a model simulate the external environment of the cell and have a big impact on the behaviour of the metabolic models, both in term of the mere biomass function optimal value and the predicted fluxes of the chemical reactions.

Most of the models were used with the original setting of exchange reactions bounds, while the media for *E.coli* and *B.subtilis* was changed with a simulated *Luria-Bertani* medium [59] to include rich conditions. For *S.cerevisiae* 7 different

simulated media conditions were used (see Table 4.2); while two are entirely taken from literature, other three of them have a known composition but automatically generated minimal bounds; finally two media were entirely automatically generated by a simple algorithm developed for the task.

All the media uses Glucose as a Carbon Source; Water and Oxygen (when present) uptake bounds are left unconstrained in all the media. The SD and $^{1}3C$ media settings are used as defined in [61], and the Glucose uptake rate bound is set to 15 $mmol/gDW/h$ in all the media for a direct comparison with the rich conditions of the SD medium. The media composition for the Nitrogen[88], Glucose[89] and Glucose Anaerobic[89] are taken from literature (we used a name Glucose to name the most common media used in these analysis, but all the media are Glucose-based). The bounds are then set using a minimization procedure that returns a the minimum values ensuring the same growth rate of the unconstrained setting (see 4.5.2).

Alongside the media taken from literature, I developed a simple procedure, inspired by the one used for the genes in [15], that could define automatically new minimal media, given a minimum value of the biomass function to be guaranteed. The idea is to set all the exchange reactions bounds as unconstrained (-10^3) at the beginning, and then sequentially set one of them, chosen at random, to zero, i.e. removing that compound from the simulated medium. The change is kept if the predicted biomass is still above the minimum, otherwise it is restored, and the procedure is repeated. If after a number tol of attempts an exchange reaction to be removed is not selected, a function implementing an exhaustive procedure is called. All the residual removals are tested in it, and if a feasible one is found is returned to the main procedure, otherwise and the procedure ends and returns an empty array, since no further exchange reactions bounds can be set to zero. The glucose, oxygen and water exchange reactions bounds are fixed and not considered.

The algorithm was set up to evaluate the Growth Rate of all the models, and to validate the media composition for all of them in every step. I mapped the exchange reactions of the models and used only the common ones.

The minimum biomasses used by the procedure are the Default one in the case of aerobic condition, and the Glucose Anaerobic Medium one in the anaerobic condition. While in the aerobic case no further settings were required, in the

Algorithm 2 Minimal Media Algorithm

```
procedure MINIMAL MEDIA(minGrowth, models, tol)
    Fluxes = getExchangeReaction(models)
    i = 0
    while i < tol do
        BackUp = Fluxes
        Fluxes = remove(rand(Fluxes))
        if FBA(models, Fluxes) >= minGrowth then
            i = 0
        else
            Fluxes = BackUp
            i = i + 1
            if i == tol then
                Fluxes = exhProc(Fluxes, model, minBiomass)
                if ¬(isempty(Fluxes)) then
                    i = 0
    return Fluxes
```

anaerobic conditions the differences between the models made necessary an addition of some specific metabolites that are not in common. In the yeast 7.6 based models I added the 14-demethyllanosterol uptake reactions, that is not present in the iMM904 model , but it is required for anaerobic growth in the earlier. Vice versa in the iMM904 the linoleate exchange reaction is needed. In the yeast 7.6+Fe the presence of at least one amino acid is also requested, e.g. the L-methionine. All the other fatty acids required for the anaerobic growth [90][87][19] are present in all the models. In all the yeast 7.6 based models another change had to be made, namely the *Heme-a* was removed from the definition of the growth pseudo-reaction, as reported in [19]. I then selected two media over the results and named them *"Minimal"*, one in aerobic and one in anaerobic conditions. The media involve respectively 4 and 9 compounds excluding the Glucose, Oxygen and Water.

The minimal aerobic medium is very similar to the Glucose Aerobic, with only two less compounds present, i.e. potassium and sodium, and keeping ammonium, iron, phosphate and sulphate. In all the run of the algorithm (128) this was the only medium found using the Biomass value of the Default conditions.

On the converse, in the anaerobic conditions, in 128 trials, 89 different media were found by the procedure, mainly because all the fluxes are left unconstrained,

so in some of them the model is allowed to use a large amount of metabolites from the environment to keep the biomass value, e.g. other carbon sources, or amino acids, to compensate the progressive deletion of other exchange reactions, leading to unrealistic uptake rates of the remaining compounds. Hence the medium was manually selected over all the results.

Bounds definition

With the exceptions of the SD Medium and ^{13}C Medium, which were used in [61] with defined bounds for some of the exchange reactions, all the other media are simulated leaving the reactions bounds to an unconstrained level (-10^3). In order to narrow the possible uptake fluxes, I used a simple models-oriented approach to redefine the bounds of the exchange reactions.

Using an approach analogous to the Flux Variability Analysis I determined the minimum flux of the reactions that ensure the Growth Rate value obtained by the FBA, using all the models. Then, for every single reaction, I took the minimum over the models results of the FVA (corresponding to the maximum uptake of that metabolites, given that the exchange reactions have negative fluxes if representing a flux from the environment to the cell). The rounded down value, considering the first two decimal digits, is the bound of that reaction.

See Tables 4.2 for all the media definitions and the reactions bounds for the *S.cerevisiae* experiments.

4.5.3 Algorithm

In this section I describe the evolutionary algorithm used for obtaining the minimal metabolic network. The procedure is the same for all the organisms and the media. In the Algorithm 3 the main procedure is described. The idea is an evolutionary algorithm which iteratively improve the initial population for maximizing the number of KO. Every point of a population represents a candidate solution, i.e. a set of genes that are knocked out. They are represented by a logical array, with every value corresponding to a gene in the model. If the value is equal to 0 (false), the gene is still present in the model, otherwise it is knocked out.

Algorithm 3 Evolutionary Algorithm

procedure EA(*pop*, *gen*, *model*, *minBiomass*)
  $P = initPop(pop)$
  for $i = 1 : gen$ **do**
    $P_t = geneticOperator(P_{i-1}, model, minBiomass)$
    $P_t = sortPop(P_t \cup P_{i-1})$
    $P_t = P_t(1 : pop)$
    $P_i = aging(P_t, P_{i-1})$
    $saveResult(P_i)$
  $R = loadResults()$
  $minSol = findMinimalSolutions(R)$
  return *minSol*

The initial population points are all wild type strains, i.e. all-zeros arrays. During the procedure, the genetic operator function is the first function to be called. It selects a new possible knockout over all the remaining active genes in the strain and it evaluates the new Growth Rate. If the Growth Rate satisfies the constraint the change is kept and the new point will enter in the new population, otherwise the searching is repeated, till a feasible knockout is reached or a maximum number of trials (10) are performed.

The constraint over the Growth Rate is such that the new strains can not have a predicted value smaller than the 99% of the Wild Type one.

The genetic operator function constructs an offspring for every parent point in the population, and all these new points constitute the offspring population. The union of this and the parents one is then sorted using the *sortPop* function. The idea of the sorting is that the points that were improved with a new knockout should be discarded. In order to do this the Manhattan Distance of all the couples of points is evaluated. The Manhattan distance between two solutions p and q in this case can be defined as the lowest number of changes (from 1 o 0 or vice versa in the logical array of the solution) that are necessary to obtain the solutions q starting from the solutions p. If two points $p, q \in P$ have a Manhattan Distance equal to 1, and p has a higher number of KO than q has, i.e. $nKO(p) = nKO(q)+1$), we say that p *dominates* q. This definition let us to consider the non-dominated points as the ones to be kept. I assign them a number 1 corresponding to

the first *front*; the procedure is then repeated ignoring the points already labelled, finding the points of the second front and so on.

The points with the lower front value are then preferred over the others, but also a criterion of selection among the same front is needed. This criterion is based instead on the Manhattan Distance. For every point I assign a value, i.e. the mean of the Manhattan distances between it and its closest 10 neighbours (always in terms of the Manhattan distance). In the same front the points that have the greater value so defined are selected. This is done in order to maintain a diversification in the population.

Algorithm 4 Genetic Operator

```
function geneticOperator(P, model, minBiomass)
    for all p ∈ P do
        ntrials = 0
        isFeasible = 0
        while ntrials < 10 and ¬isFeasible do
            p_t = selectNewRandKO(p)
            if (1 + FBA(p, model)/minBiomass) < 0.01 then
                p = p_t
                isFeasible = 1
            else
                ntrials = ntrials + 1
    return P
```

Algorithm 5 Sorting Population Function

```
function sortPop(P)
    KOs = getKOs(P)
    i = 1
    while ∃p ∈ P : ∄(p.front) do
        for all p ∈ P do
            D_p = ManhattanDistance(p, P)
            if ∄(q ∈ P : D_p(q) == 1 and KOs(q) == KOs(p) + 1 then
                p.front = i
            p.dist = sort(D_p, npoints)
        i = i + 1
    return P
```

All the points have also a feature inside the population, their ages. The age can be defined as the numbers of last generation in which the points has been present. If a point was not improved during the last generations of the algorithm, it is a candidate to be a minimal solution. If this is the case, there are no more knockouts that can be selected for the strain that satisfies the Growth Rate constraint; there is no point then in keeping this point in the population. The age is then used as the variable of a sigmoid function representing the probability of a solution to be discarded from the population (see Algorithm 6).

If the point has to be replaced, the *Backtracking* function is called. This function simply select a random set of genes knocked out in the strain and turn them on, going up in the search tree (not necessarily in a node that has already been visited). If the new strain satisfies the Growth Rate constraint, then it will be the replacement in the population.

Algorithm 6 Aging Function

```
function AGING(P, Q, minBiomass)
    for all p ∈ P do
        if ∃q ∈ Q : q == p then
            p.age = q.age + 1
        else
            p.age = 0
        if 1/(1 + exp(10 − p.age/10)) < rand() then
            p = Backtracking(p, minBiomass)
    return P
```

At the end of the algorithm a post processing procedure is launched to identify the minimal solutions found by the algorithm. Following the definition of non-domination, given for the sorting of the population, the solutions that are not dominated are selected. All of them are then tested using an exhaustive procedure, similar to the one used in the definition of the new media. All the single KO of the genes still active in the string are tested, to ensure that there are not any other genes that can be turned off satisfying the biomass constraint. If this is the case, the solutions can then be defined as *minimal* and be included in the results returned by the procedure.

Parameters and complexity

The usual parameters used in the experiments are a population size equal to 100 and a maximum number of generations equal to 5,000. The theoretical maximum number of candidates solutions explored during the procedure is then equal to 500,000 (potentially to be multiplied by 10 if considering the maximum number of trials in the genetic operator). Of course the actual number of points explored is lower than this. In the *S.cerevisiae* experiments using the *yeast 8.3.1* model, the mean of the numbers is close to 350,000. Among all of these the number of Minimal Networks found by the evolutionary algorithm is about 1,000. An important remark is that the dimension of the raw solutions space, in the case of the *yeast 8.3.1* model, is equal to $2^8 79 \simeq 10^2 64$ which is incredibly complex and infeasible for an exhaustive search. The choice of the parameters is a trade-off that allows an acceptable number of solutions to be evaluated and deepness of the search tree, and a relatively fastness of the algorithm.

4.5.4 Frequency and Redundancy Analysis

Once the algorithm returns the MNs found by the algorithm, a set of analysis could be performed. First I considered the frequency of knockouts for every genes. Three sets of genes are defines: i) genes that are always knocked out in every MN; ii) genes that are knocked out in some of the MNs but are present in others; iii) genes that are always active in the solutions. (Another category would be the essential genes, which are not considered by the algorithm). It is clear how it is also possible to consider this knockouts frequency analysis as a measurement of the redundancy of some of the genes; in particular I consider more or less redundant the genes that have a higher or lower knockouts frequency in the MNs, respectively.

The analysis was also extended, for *S.cerevisiae*, to include all the MNs in aerobic or anaerobic conditions, or both. In general, the genes that are (almost) always present in the solutions are the more interesting because of the important predicted role they have in the metabolic networks. Moreover, if they are not marked as essential (as in the yeast case), that could be a new information on the importance of the gene in the metabolism and/or in the cell, or either a false positive due to the approximations that are included in the models.

4.5.5 BLASTp

Following the frequency approach of the previous section, another analysis I performed was a cross comparison between the different organisms results. I considered a *BLASTp* search between the results; first I used the KEGG or other equivalent databases for retrieving the amino acids sequences of the genes in the different organism. The BLASTp search was then used among the genes that are active in most of the MNs (choosing a threshold of the 85% of the solutions to exclude possible outliers in the MNs set).

The search are conducted using a set of genes, e.g. the S.cerevisiae genes, as the query and all the other organism to be included in the comparison as the database. The similarity cut-off is set to $10^{-}5$.

4.5.6 Pathway-oriented Robustness Analysis

The pathway robustness is a measurement of the robustness of a metabolic network considering the partition of the reactions in the model. I applied this analysis on the results in the SD medium for yeast 8.3.1 model. In the model each reaction belongs to a specific pathway. The total number of pathways in the model is equal to 91 (even though many of them have only few reactions). The robustness of a pathway is evaluated considering the effect on the Growth rate predicted value by changing the fluxes of the reactions of the pathways. Let P be a pathway of a model, $r_i \in P, i \in 1, \ldots, n$ the reactions belonging to the pathway P, σ the maximum variation of a reaction flux, δ the maximum allowable variation of the Growth Rate value, N the number of trials and $v_i, i \in 1, \ldots, n$ the original reactions fluxes and f the original Growth Rate value as predicted by the *pFBA* for the considered strain. For each trial and for each reaction r_i, a value $v_i^* = v_i(1 - \sigma \cdot \mu)$ is considered, where μ is a random number drawn from the standard normal distribution. A set of new constraints is then added to the *pFBA* problem, imposing that $lb_i = ub_i = v_i^*$ for each reaction in the pathway P. Hence the new problem is evaluated using the *pFBA*. If the resulting value f^* of the Growth Rate is such that $|f^* - f| <= \delta \cdot f$ than the trial is positive. The robustness index for

the pathway P is hence defined as

$$\frac{\#\ of\ positive\ trials}{N}$$

The parameters used in the study are $\sigma = \delta = 1/100, N = 1000$.

4.5.7 Complex Network Analysis

The metabolic network given by the model and the set of genes present in it, can be visualized as a graph and can be used as a complex network. There are several ways to do the task[]; I focused on the bipartite directed graph type. Every metabolite and every reaction of the model becomes a node of the graph, while the edges follow the stoichiometry of the reactions, e.g. if the metabolites m_1, m_2 are a reagent and a product of the reaction r respectively, two directed edges $(m_1, r), (r, m_2)$ are added to the graph. The metabolites nodes are then not directly connected by an edge in the graph, and the same for the reactions nodes, hence the bipartite graph. It is possible to use the predicted reactions fluxes as the weights of the edges of the graph, having care of multiplying the flux by the metabolite stoichiometric coefficient in the reaction. Considering the weights, some of the reactions will not be active (i.e. they have a flux equal to 0) and so the correspondent nodes and edges are not considered in the network, which is then simplified. Once the weighted graph is defined in this way it can analyzed by using some basic measures and quantities defined in the complex network theory to analyze the behaviour of the network[82]. Specifically, I considered the Cumulative distributions of edge weights and node degrees.

4.5.8 Code implementation and machine setting

The main algorithm and the analysis codes were developed on MATLAB software by The MathWorks, Inc.. The main procedure was run on *MATLAB r2017a* executed on a HPC cluster node with 2 Intel Xeon Skylake 6142 processors, 2.6GHz 16-core and 192GB RAM, O/S Scientific Linux 7. The analysis were performed on *MATLAB r2018b* executed on a laptop with a Intel i7-6770HQ processor, 2.6GHz 4-core and 16GB RAM, O/S Windows 10 by Microsoft. The Linear and Quadratic

Programming Problems were solved using the MATLAB interface of the Gurobi v. 7.5.2 software.

4.6 Future Improvements

Computational approach to the minimal cell could be further optimized and refined as a tool.

We have seen how results and analyses are based on the metabolic models and particularly on the definitions of the external conditions and the relative simulated biomass reactions within, that is indeed the key point for the simulations using any kind of genome-scale metabolic model [91]. It is reasonable to assume that changing the biomass function definition by removing or adding compounds the results could be sensibly different. The MNs I found are then a minimization of the specific metabolisms and their related genes, aimed at keeping the cells activity as defined for the wild type organisms rather than a minimization of the overall number of genes for a universal biomass definition. It could be of interest in the future to address this comparative task, also from the computational point of view, to excerpt possible optimal pathways from other cells to be included in a synthetic minimal cell; this would require, though, the definition of a global "life" function. Such a function could be inferred by the studies on the minimal organisms as the JCVI-syn3.0, since we have already seen in the comparisons that there are metabolic structures in common, notably the central carbon metabolism, in cells far from each other from many points of view.

Another major improvement to the task could be the implementation of a bottom-up approach rather than a top-down one as the one I have used in my work. This is a more challenging point that would need a precise formulation in order to obtain reasonable results emerging, for example, from just a sequence of nucleic acid simulating the first genes that have emerged since the beginning of the history of life on earth. At the moment, it is not reasonable to imagine that such an approach could have effective formulations as a computational problem, since it would require a deep comprehension of any basic aspect of life and there would be a huge number of variables that need to be taken into account. It is though a utopian and exciting objective, such as many others, which the efforts of

researchers could take the human knowledge closer to in the next years.

Exchange Reaction	Bounds						
	SD	C13	Nitrogen	Glucose	Minimal	Gl. Anaer.	Anaer. min.
(R)-pantothenate exchange	-0.78	-0.78					
4-aminobenzoate exchange	-0.78	-0.78	-0.01				
adenine exchange	-0.78						
ammonium exchange	-1000	-1000	-7.49	-7.57	-7.57	-1.66	
biotin exchange	-0.78	-0.78	-0.01				
citrate(3-) exchange	-0.78						
D-glucose exchange	-15	-20	-15	-15	-15	-15	-15
folic acid exchange	-0.78						
glycine exchange	-0.78						-1.66
iron(2+) exchange	-1000	-1000	-0.01	-0.01	-0.01		
L-alanine exchange	-0.1						
L-arginine exchange	-0.31						
L-asparagine exchange	-0.36						
L-aspartate exchange	-0.72						
L-cysteine exchange	-0.78						
L-glutamate exchange	-0.6						
L-glutamine exchange	-0.23						
L-isoleucine exchange	-0.78						
L-leucine exchange	-0.78						
L-lysine exchange	-0.78						
L-methionine exchange	-0.78						
L-phenylalanine exchange	-0.78						
L-proline exchange	-0.78						
L-serine exchange	-0.47						
L-threonine exchange	-0.78						
L-tryptophan exchange	-0.78						
L-tyrosine exchange	-0.13						
L-valine exchange	-0.78						
myo-inositol exchange	-0.78	-0.78	-0.01				
nicotinate exchange	-0.78	-0.78	-0.01				
oxygen exchange	-1000	-1000	-1000	-1000	-1000		
phosphate exchange	-1000	-1000	-1.07	-1.08	-1.08	-0.24	-0.24
potassium exchange	-1000	-1000	-0.01	-0.01			
pyridoxine exchange	-0.78	-0.78					
riboflavin exchange	-0.78		-0.01				
sodium exchange	-1000	-1000	-0.01	-0.01			
sulphate exchange	-1000	-1000	-0.11	-0.11	-0.11	-0.03	-0.03
thiamine(1+) exchange	-0.78	-0.78	-0.01				
uracil exchange	-0.78						
water exchange	-1000	-1000	-1000	-1000	-1000	-1000	-1000
ergosterol exchange						-0.01	
lanosterol exchange						-0.01	-0.01
palmitoleate exchange						-0.01	
stearate exchange						-0.01	
zymosterol exchange						-0.01	-0.01
14-demethyllanosterol exchange						-0.01	-0.01
ergosta-5,7,22,24(28)-tetraen-...						-0.01	-0.01
oleate exchange						-0.01	-0.01

Table 4.2: Media composition and Bounds.

Chapter 5

Conclusions

Throughout my work, I have been trying to apply and improve algorithms and methods for various application using the simple genome-scale models. The production of chemicals, which has already been explored during the last years, also using this kind of models, it is still an interesting task to tackle, and the new developments and refinements of the models could be of fundamental importance in the future for a more and more precise design of strains for various possible applications.

In my work, I analysed the genome-scale models by using a multi-objective optimization, the redirector, and the medium optimization approaches for the maximization of the ethanol production; the designed approach takes into account for finer analysis of the metabolic network models and processing of Pareto optimal strains. All the analyses, and the new constraints introduced for the optimization, such as the essential genes and medium used, taken from in *in vivo* experimentations, was done in an effort to improve the understanding of the relationship between genotype and phenotype in this particular application scenario.

The complexity of the design problem, and the possible different optimizations used require a specific setting of the initial conditions, and the further development of efficient methods. Here I also proposed a combined methodology, using three different steps of optimization for enhancing the results and minimizing the disruptive interventions on the genome.

Clearly, all these results, and the models in general, are still not sufficiently

precise to obtain a results to be considered safely reproducible *in-vivo*. I believe that further researches and improvements of the mathematical models could make these predictions more and more precise, keeping the simplicity of the formulation, as we have started to see in the last few years.

The second problem that I tried to address in my work is the minimal cell problem. In a novel approach, I developed a custom algorithm for simulating extreme genome reduction in the models, obtaining a set of theoretical minimal metabolic networks. It is important to point out that the results and analyses are based on the metabolic models and particularly on the definitions of the external conditions and the relative simulated biomass reactions within. It is reasonable to assume that changing the biomass function definition by removing or adding compounds the results could be sensibly different. The resulting MNs are then a minimization of the specific metabolisms and their related genes, aimed at keeping the cells activity as defined for the wild type organisms rather than a minimization of the overall number of genes for a universal biomass definition. It could be of interest in the future to address this comparative task, also from the computational point of view, to excerpt possible optimal pathways from other cells to be included in a synthetic minimal cell; this would require, though, the definition of a global "life" function. Such a function could be inferred by the studies on the minimal organisms as the JCVI-syn3.0, since we have already seen in the comparisons that there are metabolic structures in common, notably the central carbon metabolism, in cells far from each other from many points of view. The external conditions are also extremely important in the simulations of these results. As an example of that, I considered the impact of the single compounds of a rich medium (SD) on the yeast MNs (see Figure A.9). While the wild type model is able to survive on a minimal medium of just 6 compounds, some of the MNs of a rich medium have "adapted" and require the presence of amino acids such as the L-leucine, L-lysine or L-methionine for their survival, having apparently lost the ability to synthetize them. This is a two-way problem, because changing the environment changes the theoretical minimal cells that can survive in it and a given minimal cell is likely to survive only in a defined medium capable of sustaining its life. My results confirm this assumption and give a clue of the complexity of this subsequent problem. The results I have presented have implication on the cutting-edge minimal genome

studies. With a relatively simple approach I considered several organisms and I was able to obtain different minimal solutions, i.e. different strain that are predicted to grow at a comparable rate to the WT, but have half of the active genes, or less. This could be interesting in the further studies on the evaluation of genes function in the minimal cells or as a hint for less or more important elements in the definition of a reduced minimal genomes from more complex organisms.

In conclusion, I presented these computational application to biological tasks. It is extremely interesting to me to have the chance of working on such complex and realistic problem using only a computer, it is something that have always fascinated me. I tried then to obtain and describe significant results that could be useful for a further understanding of these phenomena, and to assess a starting point for future studies.

Appendix

Metabolic Engineering and Pathways Design by Computational Systems Biology

Strain	Ethanol ($mmol\,gDW^{-1}h^{-1}$)	Biomass (h^{-1})	Variations (neg, pos)
Wild Type	0.015	0.287	0
S1	0.685	0.272	10 (9,1)
S2	1.558	0.251	12 (9,3)
S3	2.876	0.22	10 (7,3)
S4	4.73	0.176	13 (10,3)
S5	5.48	0.158	12 (9,3)
S6	6.73	0.128	12 (10,2)
S7	7.126	0.119	11 (7,4)
S8	8.035	0.097	12 (9,3)
S9	9.493	0.061	11 (8,3)
S10	11.923	0.002	17 (14,3)

Table A.1: Maximization of Ethanol and Biomass production for *S. cerevisiae* using the redirector approach for enzymes regulations with the iMM904 genome-scale model.

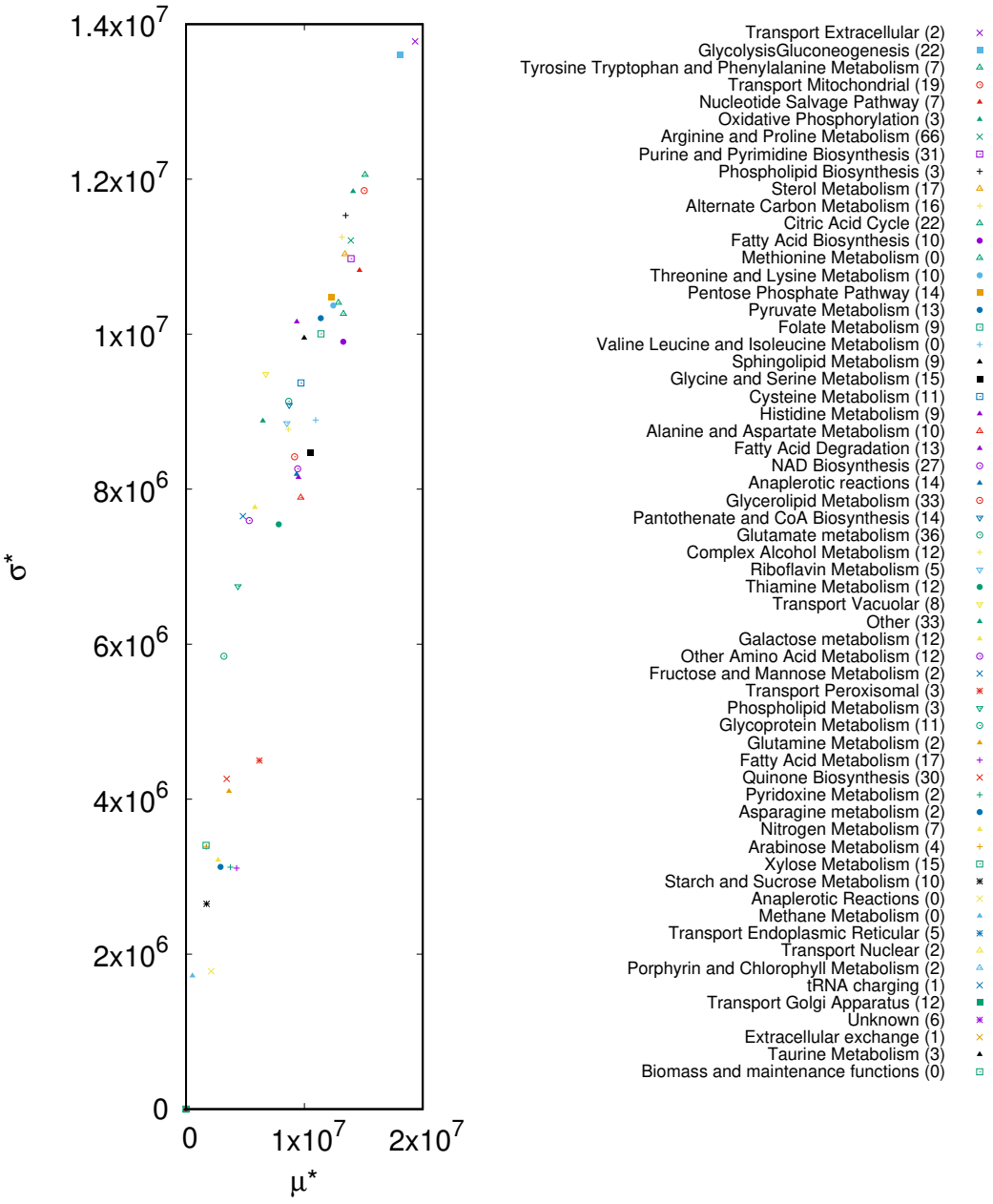


Figure A.1: Pathway oriented Sensitivity Analysis for the *S.cerevisiae* model

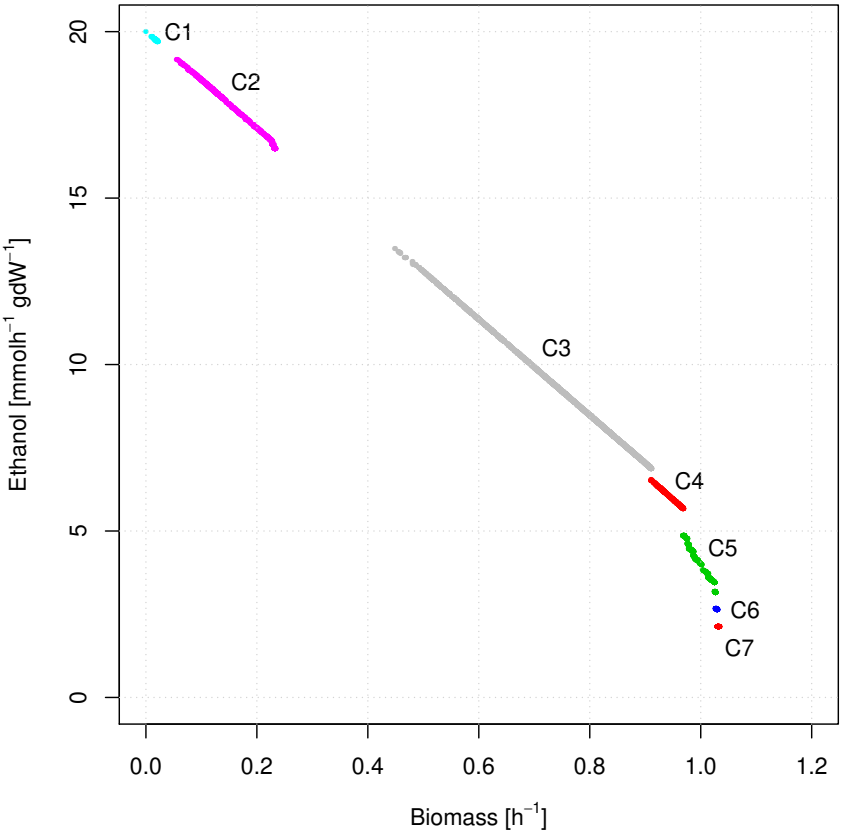


Figure A.2: Clustering results on the observed Pareto Front of the optimization of ethanol production and biomass formation in *E. coli*, anaerobic condition, glucose uptake rate 10 $mmol\, gDW^{-1}\, h^{-1}$.

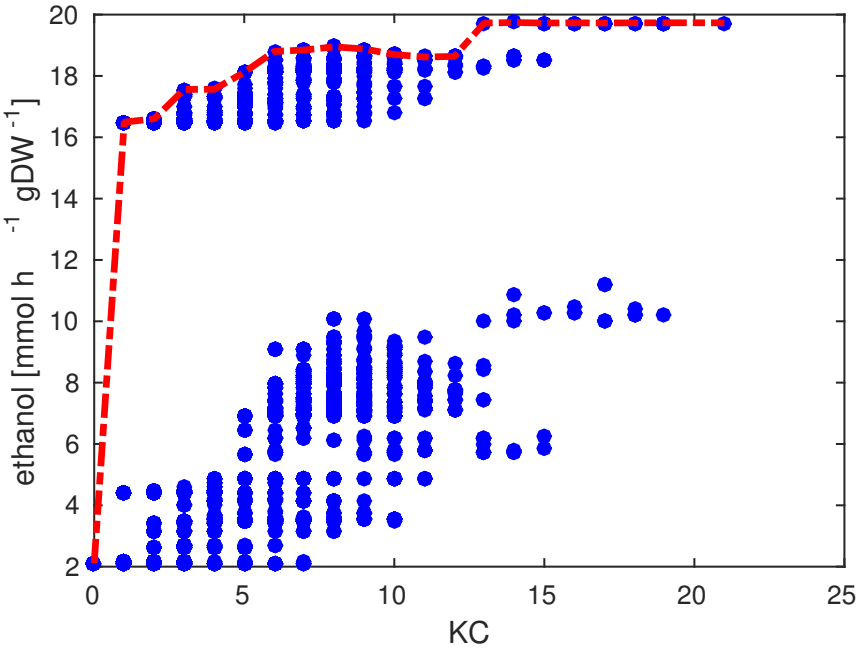


Figure A.3: Pareto optimal ethanol production (in red in the figure) and feasible solutions (in blue in the figure) as an observed function of the knock-out cost.

Strain	Ethanol ($mmol\,gDW^{-1}h^{-1}$)	Biomass (h^{-1})	Knock-out
E. coli – iJO1366 in Anaerobic condition			
Wild Type	2.116	1.0334	0
S1	2.1346	1.0311	1
S2	4.8362	0.97127	2
S3	3.4428	1.0253	2
S4	5.6776	0.9687	3
S5	4.8737	0.96886	3
S6	6.2177	0.93217	4
S7	6.2457	0.93027	5
S8	6.2453	0.9303	6
S9	6.2464	0.93023	7
S10	6.2473	0.93016	9
E. coli – iJO1366 with LB Medium			
Wild Type	25.0757	3.6921	0
S1	31.7284	3.6476	1
S2	79.6763	3.3459	2
S3	31.7387	3.6472	2
S4	81.8296	3.3263	3
S5	80.0633	3.3438	3
S6	80.0521	3.3438	3
S7	82.2129	3.3242	4
S8	82.2404	3.3239	5
S9	82.2569	3.3235	6
S10	82.2582	3.3235	8
E. coli – iZ_1308 with LB Medium			
Wild Type	21.913	3.7522	0
S1	24.0692	3.7345	1
S2	80.7771	3.3776	2
S3	75.6019	3.4169	2
S4	80.7838	3.3773	3
S5	80.7799	3.3776	3
S6	78.8747	3.3921	3
S7	80.7865	3.3773	4
S8	80.7879	3.3773	5
S9	80.7882	3.3771	6
S10	80.7883	3.3771	8
S.cerevisiae – Yeast7.6 with SD medium			
Wild Type	27.8249	0.4174	0
S1	29.4342	0.4083	1
S2	29.0183	0.41195	1
S3	29.4699	0.40653	2
S4	29.4488	0.40742	2
S5	29.4715	0.4065	3
S6	29.4829	0.40531	4
S7	29.4879	0.40478	5
S8	29.4911	0.40446	6
S9	29.4939	0.40402	7
S10	30.1258	0.38946	8

Table A.2: Maximization of Ethanol and Biomass in the selected Pareto optimal strains.

Minimal Metabolism Design by Computational Synthetic Biology

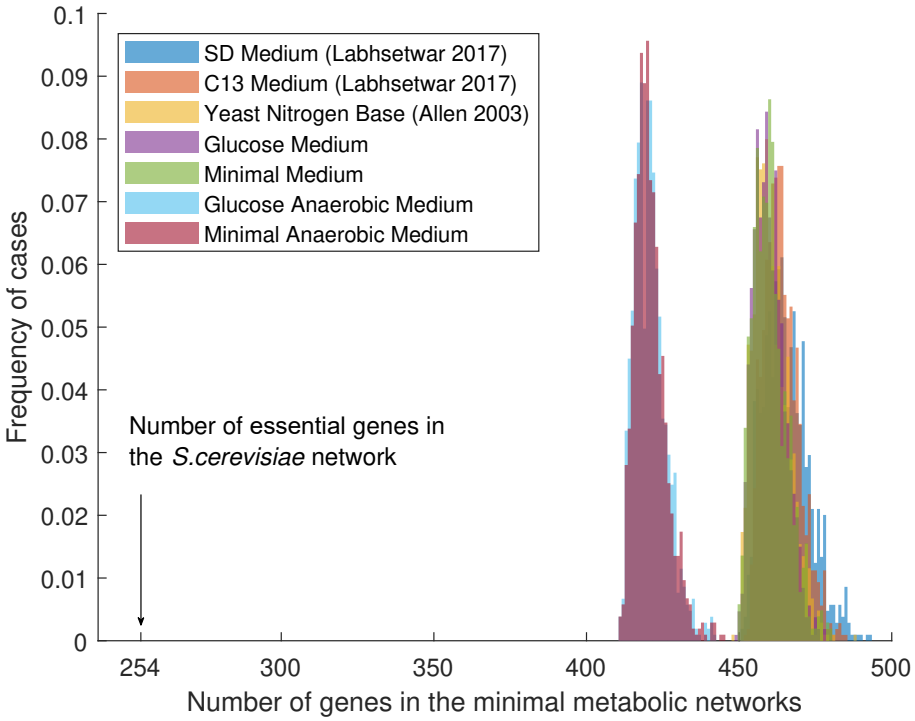


Figure A.4: Frequencies of MNs active genes with a relaxed constraint on the growth rate reduction; two different clusters are present, for aerobic and anaerobic conditions.

Systematic Names	Standard Names	Name Description	Frequency (1% Biomass Threshold)	Frequency (10% Biomass Threshold)
YDR127W	ARO1	AROmatic amino acid requiring	1	1
YGL148W	ARO2	AROmatic amino acid requiring	1	1
YJR073C	OPI3	OverProducer of Inositol	1	1
YJR109C	CPA2	Carbamyl Phosphate synthetase A	1	1
YKL067W	YNK1	Yeast Nucleoside diphosphate Kinase	1	1
YKR067W	GPT2	Glycerol-3-Phosphate acylTransferase	1	1
YMR205C	PFK2	PhosphoFructoKinase	1	0.802
YOL140W	ARG8	ARGinine requiring	1	1
YOR130C	ORT1	ORnithine Transporter	1	1
YOR303W	CPA1	Carbamyl Phosphate synthetase A	1	1
YPR160W	GPH1	Glycogen PHosphorylase	1	0.975
YPL262W	FUM1	FUMarase	0.998	0.945
YOL064C	MET22	METhionine requiring	0.986	0.367
YLR058C	SHM2	Serine HydroxyMethyltransferase	0.981	0.916
YKL029C	MAE1	MAlic Enzyme	0.978	0.725
YDR019C	GCV1	GlyCine cleaVage	0.968	0.906
YJL121C	RPE1	Ribulose 5-Phosphate Epimerase	0.960	0.957
YNL192W	CHS1	CHitin Synthase	0.957	0.970
YKR039W	GAP1	General Amino acid Permease	0.956	0.968
YBL099W	ATP1	ATP synthase	0.951	0.951
...	...	...	...	

Table A.3: The Mandatory genes for all the MNs of S.cerevisiae in the 7 different external conditions.

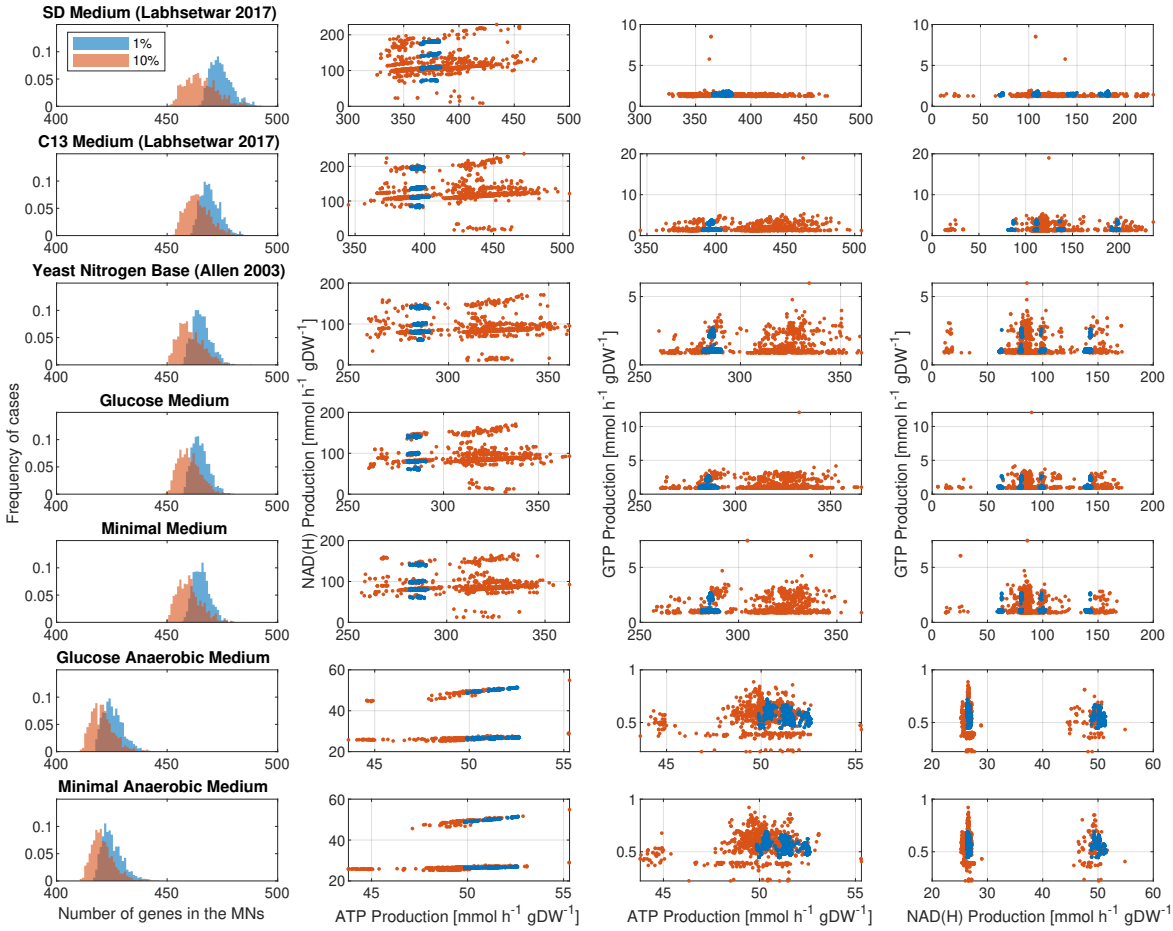


Figure A.5: For each condition the MNs active genes number using a Growth rate threshold of 1% or 10% is reported. On average less genes are required for the less strict bound. Sum of reactions fluxes producing one of three energy molecules (ATP, NADH, GTP), highlighting different regions in aerobic and anaerobic conditions and for the two thresholds.

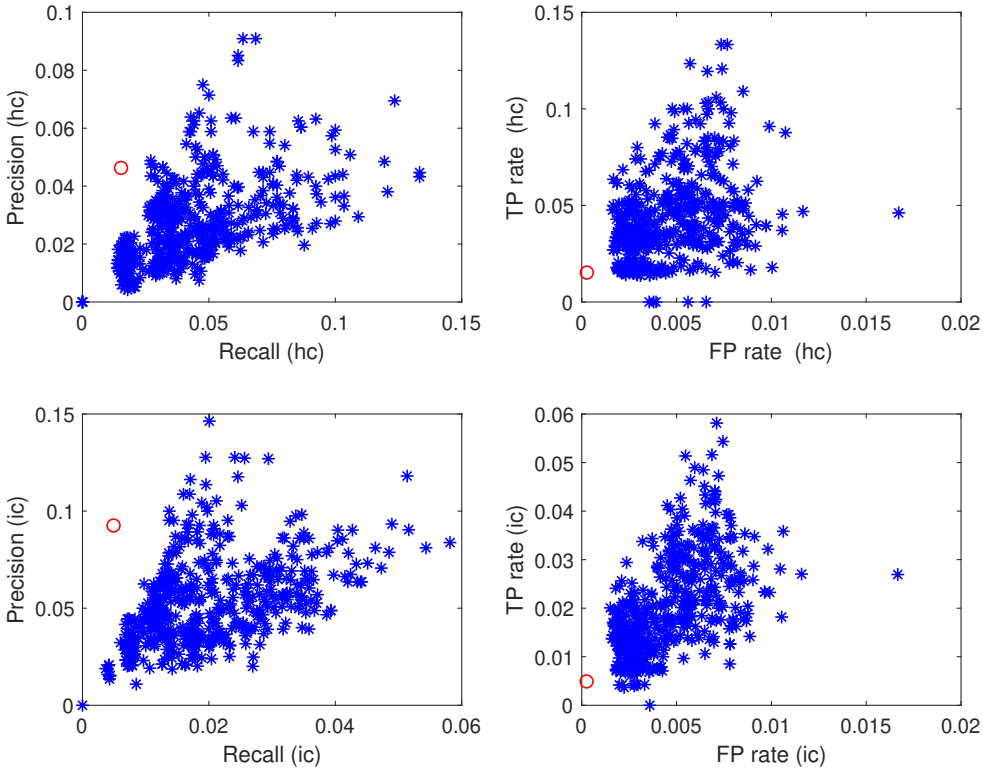
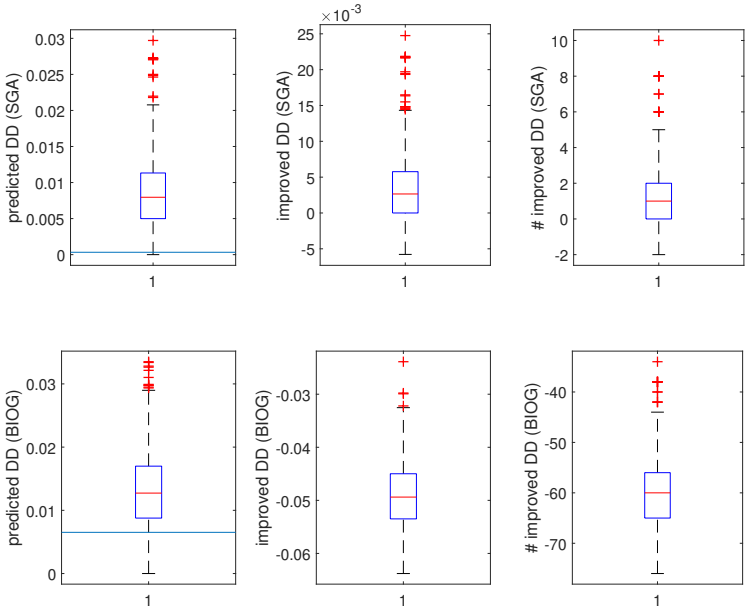


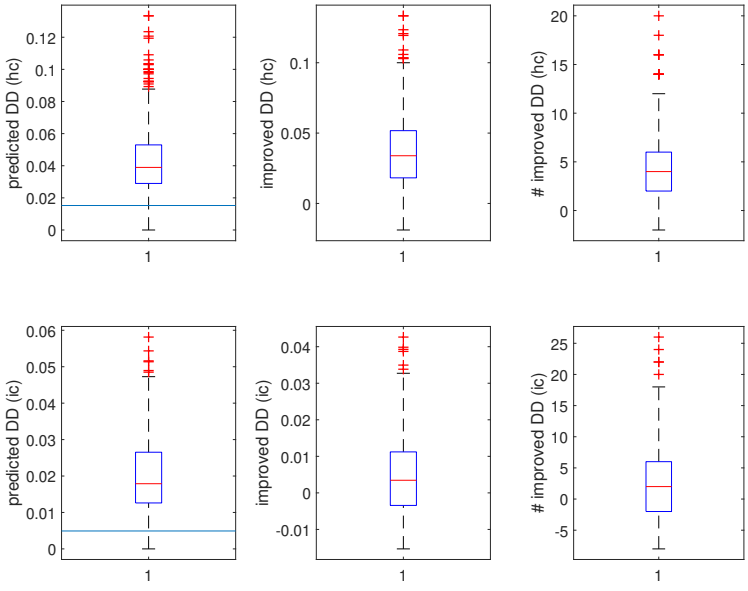
Figure A.6: Comparison of prediction for synthetic double deletions in the MNs; the red circles and the blue horizontal line in the boxplots represent the WT values, the other data refer to the MNs.

Functional categories	# Genes in the Wild Type	Relative Frequency	Mean of Rel. Frequencies (SD)	Mean of KO Perc. (SD)	Mean of Rel. Frequencies (An)	Mean of KO Perc. (An)	Diff. in KO
transferase activity	320	28.24	34.52	48.82	38.13	49.33	-0.51
oxidoreductase activity	210	18.53	18.91	57.29	17.76	64.04	-6.75
hydrolase activity	192	16.95	9.27	77.10	9.75	78.41	-1.31
transmembrane transport	146	12.89	6.41	79.18	6.38	81.43	-2.25
lipid metabolic process	116	10.24	11.18	54.30	12.64	53.67	0.63
ligase activity	83	7.33	9.57	45.29	10.73	45.00	0.29
lyase activity	83	7.33	9.51	45.62	9.88	49.36	-3.74
kinase activity	76	6.71	7.96	50.28	8.90	50.21	0.08
transmembrane transporter activity	69	6.09	2.50	82.83	2.17	86.61	-3.78
ion transport	66	5.83	6.01	56.82	3.56	77.05	-20.23
transferase activity, transferring glycosyl groups	64	5.65	7.72	42.81	8.59	42.93	-0.12
carbohydrate metabolic process	59	5.21	3.30	73.47	4.11	70.36	3.10
unassigned	48	4.24	6.38	36.98	4.48	60.30	-23.32
biological_process	38	3.35	1.15	85.63	1.21	86.44	-0.81
protein glycosylation	36	3.18	3.38	55.50	3.74	55.83	-0.33
amino acid transport	34	3.00	1.07	85.01	1.28	83.97	1.04
isomerase activity	29	2.56	3.79	38.00	4.23	37.93	0.07
tRNA aminoacylation for protein translation	29	2.56	3.49	42.98	3.88	43.04	-0.06
molecular_function	28	2.47	1.02	82.78	0.87	86.80	-4.01
hydrolase activity, acting on glycosyl bonds	27	2.38	0.35	93.85	0.39	93.87	-0.02
carbohydrate transport	26	2.29	0.63	88.46	0.71	88.46	0.00
mRNA binding	25	2.21	2.57	51.32	2.88	51.01	0.31
ATPase activity	23	2.03	2.32	52.17	1.22	77.48	-25.31
cellular amino acid metabolic process	23	2.03	2.61	46.10	2.65	51.07	-4.97
RNA binding	21	1.85	2.02	54.45	2.12	57.02	-2.57
methyltransferase activity	20	1.77	1.25	70.25	1.38	70.56	-0.32
nucleotidyltransferase activity	18	1.59	2.11	44.43	2.35	44.44	-0.02
phosphatase activity	11	0.97	0.00	99.79	0.01	99.62	0.17
Others	144	12.71	10.07	66.52	10.01	69.11	-2.59
Compartments							
cytoplasm	470	41.48	49.95	49.58	55.10	50.14	-0.56
membrane	449	39.63	36.81	61.11	32.34	69.37	-8.26
mitochondrion	376	33.19	36.20	54.32	30.50	65.50	-11.18
nucleus	198	17.48	19.92	52.28	21.79	53.19	-0.91
endoplasmic reticulum	181	15.98	18.75	50.87	20.55	51.73	-0.85
plasma membrane	153	13.50	10.94	66.07	11.43	68.24	-2.17
unassigned	86	7.59	5.12	71.76	5.34	73.58	-1.82
vacuole	61	5.38	2.39	81.42	2.52	82.44	-1.01
Golgi apparatus	40	3.53	2.47	70.72	2.76	70.62	0.09
peroxisome	32	2.82	1.27	81.10	1.53	79.72	1.38
extracellular region	21	1.85	0.45	89.88	0.24	95.04	-5.16
ribosome	9	0.79	0.77	59.41	0.76	64.30	-4.89
cytoskeleton	3	0.26	0.25	59.70	0.29	58.91	0.80

Table A.4: Quantitative analysis of the genes in the MNs using Functional Category and Compartment annotations. In the first two columns are the number of genes that can be reconducted to the specific category and the percentage in the WT genome. The next two columns are relative to the SD Medium mean values in the MNs, with the updated mean percentage and the percentage of genes that were turned off. The next two columns refer to the rich Anaerobic medium. In the last column the difference in the KO values is reported (1% Growth rate threshold).



(a) Data from [29] and *BIOGRID*.



(b) Low and high confidence data from [28].

Figure A.7: Summary of improved double deletions using 3 different databases, in the boxplots of the first column is the fraction of double deletions correctly predicted, with the blue horizontal line representing the predictions of the Wild Type network. The second column is the fraction of improved double deletions as predicted by the minimal networks ($0 = WT$ value) and the last column gives the exact numbers of improvements (Note that the double deletions that are no longer correct in the MNs are counted as negative).

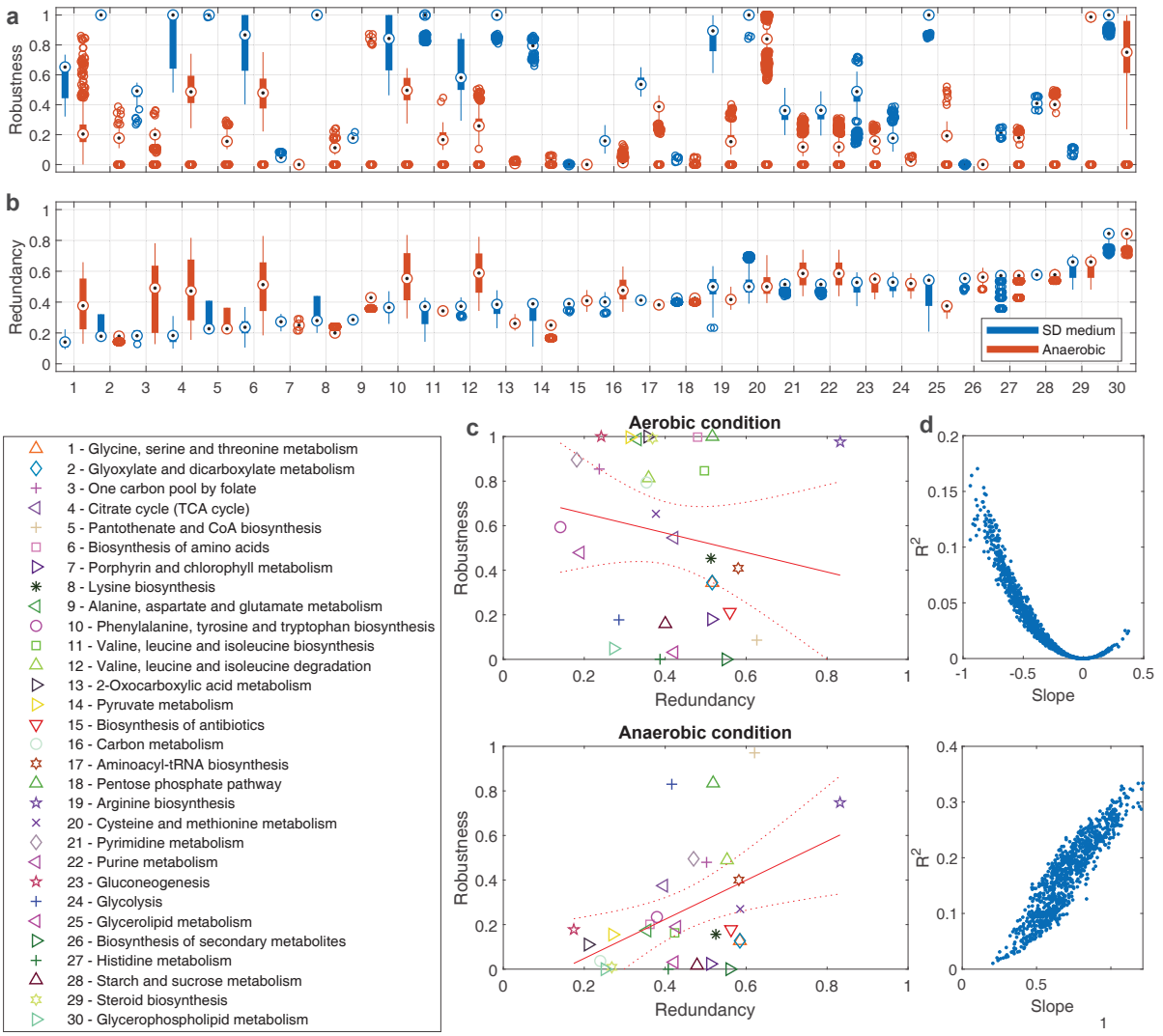


Figure A.8: Redundancy and Robustness measures for the MNs. For each pathway of the model the correspondent boxplots (a, b) of the measures along all the MNs in two different conditions, the richest media in aerobic and anaerobic are reported; the median values are shown in a 2D plot (c) with a least square fit line. It seems that there is no correlation in the aerobic case, while there is a weak dependency in the anaerobic, as it is also shown in the R2 Pearson coefficient plot for all the MNs (d).

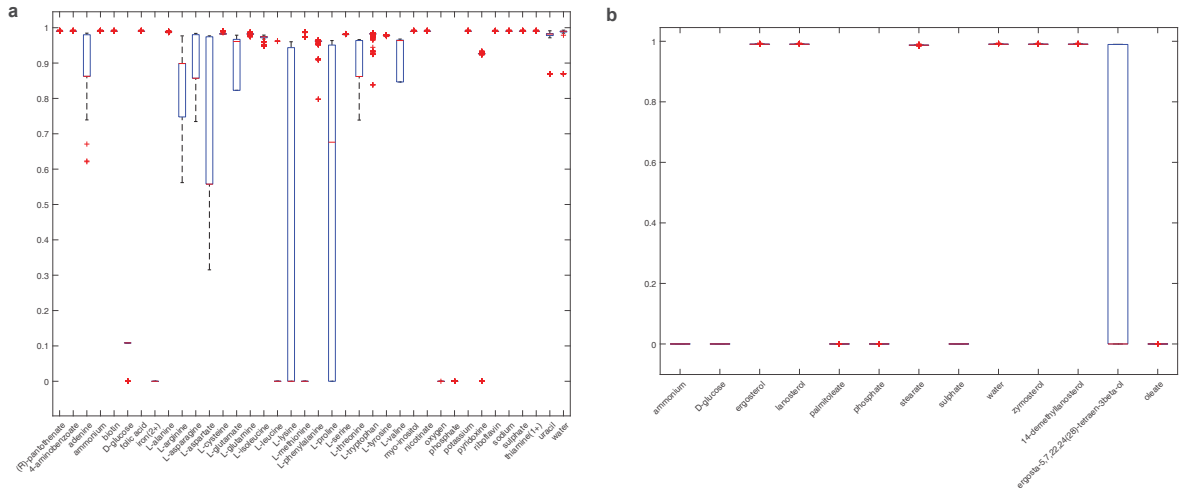


Figure A.9: Impact of the external compounds of SD (a) and anaerobic rich (b) media on the growth rate prediction of the MNs of S.cerevisiae. The y-axis represents the growth rate value expressed as the fraction of the initial (wild-type) value. There are changes in the impact of the removal of a single compound in the MNs compared to the wild-type, especially in the aerobic case; for example, the amino acids L-leucine, L-lysine and L-methionine become necessary for most or all of the aerobic minimal configurations, despite they are not required in the wild-type, expressing an "adaptation" of the networks on the simulated external conditions.

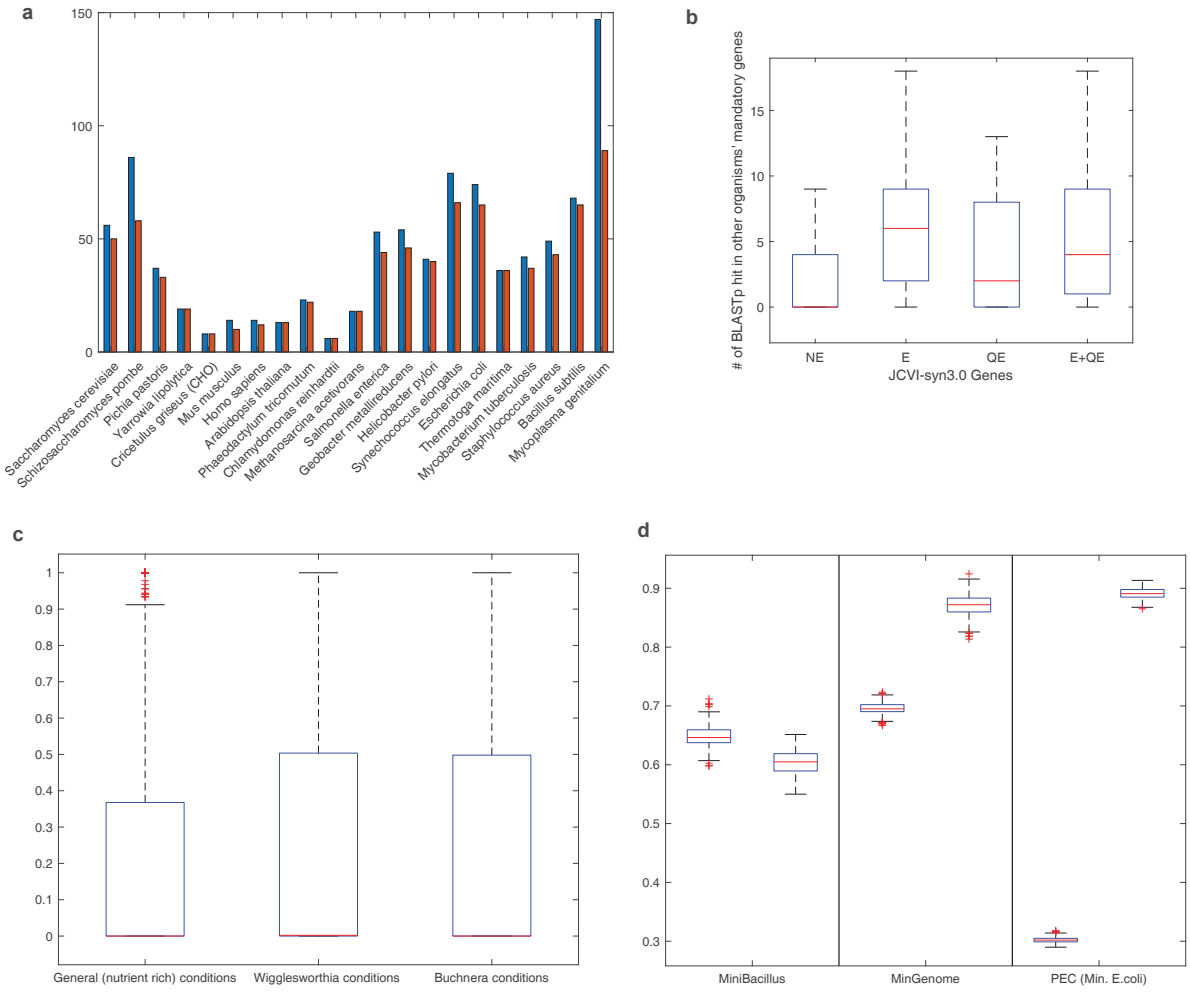


Figure A.10: Validation of the results for the genome minimization with other minimal genome studies. The first Figure (a) represent the number of genes from the JCVI-syn3.0 entire genome (blue bars) and its genome-scale metabolic model (red bars) that are also found in the mandatory and essential genes of other organisms. In (b) I reported the number of set of mandatory genes across the various organisms where the JCVI-syn3.0 genes can be found; the genes marked as essential for the syn have a sensibly higher frequency in other organisms than the non-essential genes. In (c) I considered the results from Pál et al, 2006 paper, in which there are frequencies of presence of genes in minimal in silico E.coli strains. Each boxplot represents the differences of the results with theirs, in one condition, for each gene that is present in both the studies. All the third quartiles are below the 50% of difference, and notably the lower differences are in the Rich conditions, which are the closer to the Luria-Bertani simulated media considered for the E.coli simulations. In (d) there is the comparison with single minimal genomes, with two boxplots for each of them; the first column boxplot in each comparison refers to the fraction of genes of the ref. genome that are predicted to be in the MNs. The second column refers to the fraction of genes in the MNs genomes that are also present in the ref. genome.

Bibliography

[1] M. Breuer, T. M. Earnest, C. Merryman, K. S. Wise, L. Sun, et al. Essential metabolism for a minimal cell. *eLife*, 8:e36842, Jan. 2019.

[2] C. A. Hutchison, R.-Y. Chuang, V. N. Noskov, N. Assad-Garcia, T. J. Deerinck, et al. Design and synthesis of a minimal bacterial genome. *Science*, 351(6280):aad6253, Mar. 2016.

[3] G. M. Church and E. Regis. *Regenesis: how synthetic biology will reinvent nature and ourselves*. Basic Books, 2014.

[4] G. M. Church, M. B. Elowitz, C. D. Smolke, C. A. Voigt, and R. Weiss. Realizing the potential of synthetic biology. *Nature Reviews Molecular Cell Biology*, 15(4):289, 2014.

[5] H. Yim, R. Haselbeck, W. Niu, C. Pujol-Baxley, A. Burgard, et al. Metabolic engineering of escherichia coli for direct production of 1, 4-butanediol. *Nature chemical biology*, 7(7):445, 2011.

[6] S. K. Lee, H. Chou, T. S. Ham, T. S. Lee, and J. D. Keasling. Metabolic engineering of microorganisms for biofuels production: from bugs to synthetic biology to fuels. *Current opinion in biotechnology*, 19(6):556–563, 2008.

[7] C. Bro, B. Regenberg, J. Förster, and J. Nielsen. In silico aided metabolic engineering of saccharomyces cerevisiae for improved bioethanol production. *Metabolic engineering*, 8(2):102–111, 2006.

[8] J. Nielsen and J. D. Keasling. Engineering cellular metabolism. *Cell*, 164(6):1185–1197, 2016.

[9] J. I. Glass, C. Merryman, K. S. Wise, C. A. Hutchison, and H. O. Smith. Minimal Cells—Real and Imagined. *Cold Spring Harbor Perspectives in Biology*, page a023861, Mar. 2017.

[10] J. D. Orth, I. Thiele, and B. Ø. Palsson. What is flux balance analysis? *Nature Biotechnology*, 28:245, Mar. 2010.

[11] A. Patanè, A. Santoro, J. Costanza, G. Carapezza, and G. Nicosia. Pareto optimal design for synthetic biology. *IEEE transactions on biomedical circuits and systems*, 9(4):555–571, 2015.

[12] G. Rockwell, N. J. Guido, and G. M. Church. Redirector: designing cell factories by reconstructing the metabolic objective. *PLoS computational biology*, 9(1):e1002882, 2013.

[13] M. R. Long, W. K. Ong, and J. L. Reed. Computational methods in metabolic engineering for strain design. *Current opinion in biotechnology*, 34:135–141, 2015.

[14] E. Andrianantoandro, S. Basu, D. K. Karig, and R. Weiss. Synthetic biology: new engineering rules for an emerging discipline. *Molecular systems biology*, 2(1):2006–0028, 2006.

[15] C. Pál, B. Papp, M. J. Lercher, P. Csermely, S. G. Oliver, et al. Chance and necessity in the evolution of minimal metabolic networks. *Nature*, 440:667, Mar. 2006.

[16] J. Rees, O. Chalkley, S. Landon, O. Purcell, L. Marucci, et al. Designing minimal genomes using whole-cell models. *bioRxiv*, page 344564, 2019.

[17] V. Latora and M. Marchiori. Efficient Behavior of Small-World Networks. *Physical Review Letters*, 87(19):198701, Oct. 2001.

[18] H. Jeong, S. P. Mason, A.-L. Barabási, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, May 2001.

[19] B. J. Sánchez, C. Zhang, A. Nilsson, P. Lahtvee, E. J. Kerkhoven, et al. Improving the phenotype predictions of a yeast genome-scale metabolic model by

incorporating enzymatic constraints. *Molecular Systems Biology*, 13(8):935, Aug. 2017.

[20] H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási. The large-scale organization of metabolic networks. *Nature*, 407(6804):651, 2000.

[21] S. Gudmundsson and I. Thiele. Computationally efficient flux variability analysis. *BMC Bioinformatics*, 11(1):489, Sept. 2010.

[22] N. E. Lewis, K. K. Hixson, T. M. Conrad, J. A. Lerman, P. Charusanti, et al. Omic data from evolved E. coli are consistent with computed optimal growth from genome-scale models. *Molecular Systems Biology*, 6(1):390, Jan. 2010.

[23] M. D. Morris. Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161–174, 1991.

[24] J. Costanza, G. Carapezza, C. Angione, P. Lió, and G. Nicosia. Robust design of microbial strains. *Bioinformatics*, 28(23):3097–3104, Dec. 2012.

[25] C. Angione, G. Carapezza, J. Costanza, P. Lió, and G. Nicosia. Pareto optimality in organelle energy metabolism analysis. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 10(4):1032–1044, 2013.

[26] M. Hafner, H. Koeppl, M. Hasler, and A. Wagner. 'glocal'robustness analysis and model discrimination for circadian oscillators. *PLoS computational biology*, 5(10):e1000534, 2009.

[27] R. Mani, R. P. S. Onge, J. L. Hartman, G. Giaever, and F. P. Roth. Defining genetic interaction. *Proceedings of the National Academy of Sciences*, 105(9):3461–3466, 2008.

[28] B. Szappanos, K. Kovács, B. Szamecz, F. Honti, M. Costanzo, et al. An integrated approach to characterize genetic interaction networks in yeast metabolism. *Nature Genetics*, 43:656, May 2011.

[29] M. Costanzo, B. VanderSluis, E. N. Koch, A. Baryshnikova, C. Pons, et al. A global genetic interaction network maps a wiring diagram of cellular function. *Science*, 353(6306):aaf1420, Sept. 2016.

[30] Y. Liu and J. Nielsen. Recent trends in metabolic engineering of microbial chemical factories. *Current opinion in biotechnology*, 60:188–197, 2019.

[31] Z. A. King, C. J. Lloyd, A. M. Feist, and B. Ø. Palsson. Next-generation genome-scale models for metabolic engineering. *Current opinion in biotechnology*, 35:23–29, 2015.

[32] C. L. Barrett, T. Y. Kim, H. U. Kim, B. Ø. Palsson, and S. Y. Lee. Systems biology as a foundation for genome-scale synthetic biology. *Current opinion in biotechnology*, 17(5):488–492, 2006.

[33] A. Patané, G. Jansen, P. Conca, G. Carapezza, J. Costanza, et al. Multi-objective optimization of genome-scale metabolic models: the case of ethanol production. *Annals of Operations Research*, 276(1-2):211–227, 2019.

[34] K. Deb. *Multi-objective optimization using evolutionary algorithms*, volume 16. John Wiley & Sons, 2001.

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

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

[37] D. S. Lun, G. Rockwell, N. J. Guido, M. Baym, J. A. Kelner, et al. Large-scale identification of genetic design strategies using local search. *molecular systems biology*, 5(1), 2009.

[38] L. Yang, W. R. Cluett, and R. Mahadevan. Emilio: a fast algorithm for genome-scale strain design. *Metabolic engineering*, 13(3):272–281, 2011.

[39] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[40] A. Ciccazzo, P. Conca, G. Nicosia, and G. Stracquadanio. An advanced clonal selection algorithm with ad-hoc network-based hypermutation operators for synthesis of topology and sizing of analog electrical circuits. In *International Conference on Artificial Immune Systems*, pages 60–70. Springer, 2008.

[41] J. D. Orth, T. M. Conrad, J. Na, J. A. Lerman, H. Nam, et al. A comprehensive genome-scale reconstruction of escherichia coli metabolism—2011. *Molecular systems biology*, 7(1), 2011.

[42] A. M. Feist, C. S. Henry, J. L. Reed, M. Krummenacker, A. R. Joyce, 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(1), 2007.

[43] A. E. Farrell, R. J. Plevin, B. T. Turner, A. D. Jones, M. O'hare, et al. Ethanol can contribute to energy and environmental goals. *Science*, 311(5760):506–508, 2006.

[44] M. Balat and H. Balat. Recent trends in global production and utilization of bio-ethanol fuel. *Applied energy*, 86(11):2273–2282, 2009.

[45] A. Gupta and J. P. Verma. Sustainable bio-ethanol production from agro-residues: a review. *Renewable and sustainable energy reviews*, 41:550–567, 2015.

[46] K. Robak and M. Balcerek. Review of second generation bioethanol production from residual biomass. *Food technology and biotechnology*, 56(2):174, 2018.

[47] Z. A. King, J. Lu, A. Dräger, P. Miller, S. Federowicz, et al. BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Research*, 44(D1):D515–D522, Jan. 2016.

[48] S. A. Becker and B. Ø. Palsson. Genome-scale reconstruction of the metabolic network in staphylococcus aureus n315: an initial draft to the two-dimensional annotation. *BMC microbiology*, 5(1):8, 2005.

[49] I. Thiele, D. R. Hyduke, B. Steeb, G. Fankam, D. K. Allen, et al. A community effort towards a knowledge-base and mathematical model of the human pathogen salmonella typhimurium lt2. *BMC systems biology*, 5(1):8, 2011.

[50] P. Charusanti, S. Chauhan, K. McAteer, J. A. Lerman, D. R. Hyduke, et al. An experimentally-supported genome-scale metabolic network reconstruction for yersinia pestis co92. *BMC systems biology*, 5(1):163, 2011.

[51] H. W. Aung, S. A. Henry, and L. P. Walker. Revising the Representation of Fatty Acid, Glycerolipid, and Glycerophospholipid Metabolism in the Consensus Model of Yeast Metabolism. *Industrial Biotechnology*, 9(4):215–228, Aug. 2013.

[52] R. L. Chang, L. Ghamsari, A. Manichaikul, E. F. Hom, S. Balaji, et al. Metabolic network reconstruction of chlamydomonas offers insight into light-driven algal metabolism. *Molecular systems biology*, 7(1), 2011.

[53] P. Pan and Q. Hua. Reconstruction and In Silico Analysis of Metabolic Network for an Oleaginous Yeast, Yarrowia lipolytica. *PLOS ONE*, 7(12):e51535, Dec. 2012.

[54] S. Steinsiek, S. Stagge, and K. Bettenbrock. Analysis of escherichia coli mutants with a linear respiratory chain. *PloS one*, 9(1):e87307, 2014.

[55] H. Erhardt, S. Steimle, V. Muders, T. Pohl, J. Walter, et al. Disruption of individual nuo-genes leads to the formation of partially assembled nadh: ubiquinone oxidoreductase (complex i) in escherichia coli. *Biochimica et Biophysica Acta (BBA)-Bioenergetics*, 1817(6):863–871, 2012.

[56] J. M. Monk, P. Charusanti, R. K. Aziz, J. A. Lerman, N. Premyodhin, et al. Genome-scale metabolic reconstructions of multiple escherichia coli strains highlight strain-specific adaptations to nutritional environments. *Proceedings of the National Academy of Sciences*, 110(50):20338–20343, 2013.

[57] Essential genes - EcoliWiki. "`http://ecoliwiki.net/colipedia/index.php/Essential_genes`". Accessed on 2018-02-15.

[58] P. F. Suthers, A. Zomorrodi, and C. D. Maranas. Genome-scale gene/reaction essentiality and synthetic lethality analysis. *Molecular Systems Biology*, 5(1):301, Jan. 2009.

[59] R. K. Aziz, J. M. Monk, R. M. Lewis, S. In Loh, A. Mishra, et al. Systems biology-guided identification of synthetic lethal gene pairs and its potential use to discover antibiotic combinations. *Scientific Reports*, 5:16025, Nov. 2015.

[60] B. D. Heavner and N. D. Price. Comparative Analysis of Yeast Metabolic Network Models Highlights Progress, Opportunities for Metabolic Reconstruction. *PLOS Computational Biology*, 11(11):e1004530, Nov. 2015.

[61] P. Labhsetwar, M. C. R. Melo, J. A. Cole, and Z. Luthey-Schulten. Population FBA predicts metabolic phenotypes in yeast. *PLOS Computational Biology*, 13(9):e1005728, Sept. 2017.

[62] M. Sauer, D. Porro, D. Mattanovich, and P. Branduardi. 16 years research on lactic acid production with yeast–ready for the market? *Biotechnology and Genetic Engineering Reviews*, 27(1):229–256, 2010.

[63] P. D. Hsu, E. S. Lander, and F. Zhang. Development and Applications of CRISPR-Cas9 for Genome Engineering. *Cell*, 157(6):1262–1278, June 2014.

[64] S. M. Richardson, L. A. Mitchell, G. Stracquadanio, K. Yang, J. S. Dymond, et al. Design of a synthetic yeast genome. *Science*, 355(6329):1040, Mar. 2017.

[65] M. Juhas, L. Eberl, and J. I. Glass. Essence of life: essential genes of minimal genomes. *Trends in Cell Biology*, 21(10):562–568, Oct. 2011.

[66] M. C. Jewett and A. C. Forster. Update on designing and building minimal cells. *Current opinion in biotechnology*, 21(5):697–703, 2010.

[67] R. Gil, F. J. Silva, J. Peretó, and A. Moya. Determination of the Core of a Minimal Bacterial Gene Set. *Microbiology and Molecular Biology Reviews*, 68(3):518, Sept. 2004.

[68] M. Lluch-Senar, J. Delgado, W. Chen, V. Lloréns-Rico, F. J. O'Reilly, et al. Defining a minimal cell: essentiality of small ORFs and ncRNAs in a genome-reduced bacterium. *Molecular Systems Biology*, 11(1):780, Jan. 2015.

[69] M. Juhas, D. R. Reuß, B. Zhu, and F. M. Commichau. Bacillus subtilis and Escherichia coli essential genes and minimal cell factories after one decade of genome engineering. *Microbiology*, 160(11):2341–2351, 2014.

[70] A. R. Mushegian and E. V. Koonin. A minimal gene set for cellular life derived by comparison of complete bacterial genomes. *Proceedings of the National Academy of Sciences*, 93(19):10268, Sept. 1996.

[71] A. P. Burgard, S. Vaidyaraman, and C. D. Maranas. Minimal Reaction Sets for Escherichia coli Metabolism under Different Growth Requirements and Uptake Environments. *Biotechnology Progress*, 17(5):791–797, Jan. 2001.

[72] P. L. Luisi. Toward the engineering of minimal living cells. *The Anatomical Record*, 268(3):208–214, Nov. 2002.

[73] A. C. Forster and G. M. Church. Towards synthesis of a minimal cell. *Molecular Systems Biology*, 2(1):45, Jan. 2006.

[74] L. Wang and C. D. Maranas. MinGenome: An In Silico Top-Down Approach for the Synthesis of Minimized Genomes. *ACS Synthetic Biology*, 7(2):462–473, Feb. 2018.

[75] D. G. Gibson, J. I. Glass, C. Lartigue, V. N. Noskov, R.-Y. Chuang, et al. Creation of a Bacterial Cell Controlled by a Chemically Synthesized Genome. *Science*, 329(5987):52, July 2010.

[76] B. Ø. Palsson. *Systems biology*. Cambridge university press, 2015.

[77] H. Lu, F. Li, B. J. Sánchez, Z. Zhu, G. Li, et al. A consensus s. cerevisiae metabolic model yeast8 and its ecosystem for comprehensively probing cellular metabolism. *Nature communications*, 10(1):1–13, 2019.

[78] E. C. A. Goodall, A. Robinson, I. G. Johnston, S. Jabbari, K. A. Turner, et al. The Essential Genome of Escherichia coli K-12. *mBio*, 9(1):e02096–17, Mar. 2018.

[79] A. Barve, J. F. M. Rodrigues, and A. Wagner. Superessential reactions in metabolic networks. *Proceedings of the National Academy of Sciences*, 109(18):6810, May 2012.

[80] E. V. Koonin. Comparative genomics, minimal gene-sets and the last universal common ancestor. *Nature Reviews Microbiology*, 1:127, Nov. 2003.

[81] K. A. De Jong. *Evolutionary computation: a unified approach*. MIT press, 2006.

[82] V. Latora, V. Nicosia, and G. Russo. *Complex Networks: Principles, Methods and Applications*. Cambridge University Press, Cambridge, 2017.

[83] B. Dujon, D. Sherman, G. Fischer, P. Durrens, S. Casaregola, et al. Genome evolution in yeasts. *Nature*, 430:35, July 2004.

[84] M. Hashimoto, T. Ichimura, H. Mizoguchi, K. Tanaka, K. Fujimitsu, et al. Cell size and nucleoid organization of engineered Escherichia coli cells with a reduced genome. *Molecular Microbiology*, 55(1):137–149, 2005.

[85] D. R. Reuß, F. M. Commichau, J. Gundlach, B. Zhu, and J. Stülke. The Blueprint of a Minimal Cell: MiniBacillus. *Microbiology and Molecular Biology Reviews*, 80(4):955, Dec. 2016.

[86] D. Dikicioglu and S. G. Oliver. Extension of the yeast metabolic model to include iron metabolism and its use to estimate global levels of iron-recruiting enzyme abundance from cofactor requirements. *Biotechnology and Bioengineering*, 0(0), Dec. 2018.

[87] M. L. Mo, B. Ø. Palsson, and M. J. Herrgård. Connecting extracellular metabolomic measurements to intracellular flux states in yeast. *BMC Systems Biology*, 3(1):37, Mar. 2009.

[88] J. Allen, H. M. Davey, D. Broadhurst, J. K. Heald, J. J. Rowland, et al. High-throughput classification of yeast mutants for functional genomics using metabolic footprinting. *Nature Biotechnology*, 21:692, May 2003.

[89] Yeast FAQs | Systems Biology Research Group. "`http://systemsbiology.ucsd.edu/InSilicoOrganisms/Yeast/YeastFAQs`". Accessed on 2018-04-01.

[90] B. D. Heavner, K. Smallbone, N. D. Price, and L. P. Walker. Version 6 of the consensus yeast metabolic network refines biochemical coverage and improves model performance. *Database*, 2013, 2013.

[91] D. Dikicioglu, B. Kırdar, and S. G. Oliver. Biomass composition: the "elephant in the room" of metabolic modelling. *Metabolomics*, 11(6):1690–1701, 2015.