Computational modeling of metabolism and the inference of lethal genetic interactions

Proefschrift

ter verkrijging van de graad van doctor
aan de Radboud Universiteit Nijmegen,
op gezag van de rector magnificus prof. dr. J.H.J.M. van Krieken,
volgens besluit van het college van decanen
in het openbaar te verdedigen op maandag 31 oktober 2016
om 14:30 uur precies

door

Wouter Leonard Megchelenbrink

geboren op 9 september 1983
te Enschede
Promotoren:
Prof. dr. Elena Marchiori
Prof. dr. Martijn A. Huynen

Copromotor:
Dr. Richard A. Notebaart

Manuscriptcommissie:
Prof. dr. Tom Heskes (voorzitter)
Prof. dr. Yves Moreau, Katholieke Universiteit Leuven, België
Prof. dr. Bas Teusink, Vrije Universiteit Amsterdam

The research reported in this thesis has been carried out under the auspices of the institute for Computing and Information Science (iCIS) and the Radboud Institute for Molecular Life Sciences (RIMLS) of the Radboud University Nijmegen.

This research was supported by The Netherlands Organisation for Scientific Research (NWO) and The Netherlands Organisation for Health Research and Development (ZonMW) with a grant to the Centre for Systems Biology and Bioenergetics (CSBB) [grant nr CSBR09/013V].

Copyright © 2016 Wout Megchelenbrink

ISBN: 978-94-6233-397-0
Printed by: Gildeprint - Enschede
## Contents

1 Introduction
   1.1 The purpose of (metabolic) modeling ........................................ 3
      1.1.1 Modeling in systems biology ........................................... 3
   1.2 Computational models in biology ............................................. 4
   1.3 Biologically relevant flux paths in COBRA networks .................... 7
      1.3.1 Applications of metabolic modeling ................................. 8
   1.4 Genetic interactions within and beyond metabolism ...................... 11
   1.5 Outline of the thesis ....................................................... 12

2 Sampling the solution space of genome-scale metabolic networks 17
   2.1 Introduction ............................................................... 18
   2.2 Methods ....................................................................... 19
      2.2.1 Uniform random sampling ............................................. 19
      2.2.2 optGpSampler: an improved sampling tool for metabolic networks ............................................................... 23
   2.3 Experiments ................................................................... 23
      2.3.1 Datasets ................................................................. 23
      2.3.2 Evaluation .............................................................. 24
   2.4 Results ...................................................................... 25
      2.4.1 Efficiency .............................................................. 26
      2.4.2 Quality ................................................................. 26
   2.5 Discussion .................................................................. 29

3 Estimating metabolic fluxes using maximum network flexibility 33
   3.1 Introduction ................................................................. 34
   3.2 Materials and method ....................................................... 36
      3.2.1 Computing the total flux range (TFR) distribution ............ 36
      3.2.2 Computing the flux distribution with maximum metabolic flexibility (MMF) ......................................................... 38
## Contents

3.2.3 Using the TFR distribution to select flux measurements . . 39
3.2.4 Metabolic models and flux data ................................. 39
3.3 Results ............................................................... 40
3.3.1 Reducing the total flux range ................................. 40
3.3.2 Quantitative flux estimation ................................. 42
3.3.3 Selecting measurements to optimally reduce the solution space ......................................................... 44
3.4 Discussion and conclusion ........................................ 45

4 Synthetic dosage lethality in the human metabolic network 49
4.1 Introduction ............................................................ 50
4.2 Results ............................................................... 51
4.2.1 Overview of the IDLE algorithm ............................. 51
4.2.2 The metabolic SDL network ................................. 52
4.2.3 SDL is predictive of in vitro shRNA essentiality screens 54
4.2.4 Cancer cells select against SDL ............................. 55
4.2.5 SDL correlates with smaller BC tumor size ................ 56
4.2.6 SDL correlates with increased cancer survival time .... 57
4.2.7 SDLs predicted by IDLE are not expected to be specific for BC ......................................................... 58
4.2.8 Cumulative effect of SDLs in a tumor correlates with better survival ......................................................... 58
4.3 Discussion ............................................................ 60
4.4 Materials and methods ............................................. 62

5 Predicting genetic interactions from cancer genome evolution 63
5.1 Introduction ............................................................ 64
5.2 Materials and methods ............................................. 65
5.2.1 Data sources ....................................................... 65
5.2.2 Extracting the pattern for SL pairs from genomic variations 66
5.2.3 Under-sampling .................................................... 70
5.2.4 Constructing the ensemble-based prediction model .... 70
5.2.5 Constructing the genome-wide human SL interaction map 70
5.3 Results ............................................................... 71
5.3.1 SL interactions are reflected in cancer genome evolution 71
5.3.2 An ensemble-based model for predicting SL interactions 75
5.4 Discussion ............................................................ 76

6 Weighted co-expression predicts new genes for molecular systems 81
6.1 Introduction ............................................................ 81
6.2 The WeGET analysis pipeline ......................... 83
6.3 WeGET validation and comparison to other co-expression databases 84
6.4 The WeGET database and web access .................. 85
6.5 Evaluation of WeGET results for a query gene set .......... 87
6.6 Using WeGET to predict genes involved in neuropathic pain .... 88

7 Discussion ............................................ 93
  7.1 Constraint-based modeling of metabolic networks ........ 93
  7.1.1 Sampling the solution space ..................... 93
  7.1.2 Reducing the feasible flux space ................... 95
  7.1.3 Future directions ................................ 97
  7.2 Inferring gene pairs causing a synthetic (dosage) lethality ... 99
  7.2.1 Synthetic lethality .............................. 100
  7.2.2 Future directions ............................... 100
  7.3 Automated dataset weighting and feature extraction from high-throughput data ................................ 102
  7.3.1 Future directions ................................ 102
  7.4 Combining model-driven and data-driven approaches ......... 103

Bibliography ........................................... 105

Summary ................................................ 125

Samenvatting .......................................... 131

Dankwoord / Acknowledgments .......................... 137

Curriculum Vitae ....................................... 143

List of publications .................................... 145
Metabolism is one of the key cellular processes that enable life. The metabolic machinery involves thousands of chemical reactions, enzymes, genes and other cellular compounds. Seemingly tiny -inborn- errors in this complex machinery are known to give rise to severe diseases. The complexity of metabolism and its inseparable relationship with life and disease make it a highly interesting research area. Moreover, it is a broad area. Topics range from more fundamental studies of metabolic evolution in model organisms on one side, to applied studies that ultimately focus on curing a particular metabolic disease in humans on the other side.

Metabolism consists of a cascade of chemical reactions, known as pathways, in which metabolites are converted into intermediate compounds. In the end, this process provides the building blocks and the energy for a cell to function, grow and reproduce. Metabolic reactions rarely occur spontaneously, but require enzymes to catalyze the reactions. These enzymes are proteins or protein complexes which are encoded by our genes. Metabolites produced by one reaction are typically consumed by other reactions. Therefore, metabolism can be viewed as a network of metabolites that are connected by reactions, and these reactions in turn are connected to enzymes and genes. A considerable part of this thesis focuses on computational methods applied to these network models of metabolism. Figure 1.1b shows a simplified example of a metabolic network of 7 metabolites connected by 10 reactions.

Some pathways, such as glycolysis and the citric acid cycle have been discovered and unraveled starting halfway the 19th century. They are now classic textbook biochemistry pathways [154]. However, the development of whole-genome sequencing techniques in the mid-1990s and the ongoing developments in high-throughput genetic data acquisition and processing greatly accelerated the dis-
Figure 1.1: A toy metabolic network and basic computational analysis. a) The metabolic reaction $B + C \rightarrow 3E$ (labeled $V_4$) is catalyzed by the enzyme $EZ$. This enzyme requires two proteins, encoded by the genes $G_1$ and $G_2$. b) Often, a metabolite is produced by one reaction and consumed by another, which determines a metabolic reaction network. Nutrients are taken up by the cell ($v_1$, $v_2$, and $v_5$) for cellular maintenance and growth. Metabolites are also secreted to the environment ($v_5$). c) This network can be described by a matrix ($S$) in which the number of metabolites produced and consumed by each reaction is depicted (the stoichiometric coefficients). For example reaction $V_4$ requires 1 molecule of $B$ and 1 of $C$ to produce 3 molecules of $D$. Assuming a steady-state and bounding the reaction fluxes constrains the feasible flux paths. d) Still, many flux paths satisfy these constraints. Together, they determine the steady-state solution space. Flux paths that optimize a biological objective (e.g. maximize growth or ATP production) can be found by linear programming (green dots). Only specific flux paths lead to this optimum and therefore this approach narrows the range of feasible flux paths. Another possibility is sampling the solution space (red dots) to obtain the ‘most likely’ flux paths.
covery of novel metabolic genes and enzymes. This enabled the genome-scale reconstruction of metabolic networks for many organisms, ranging from the bacterium *Escherichia coli* [58] to humans [50, 218]. Nowadays, detailed metabolic networks [196] and protocols for their reconstruction based on scientific literature and experimental high-throughput data exist [59, 217, 90]. Importantly, these network reconstructions provide a biochemically, genetically and genomically (BIGG) sound model [176, 161]. Moreover, they provide a solid computational framework that combines network optimization and (omics) data integration to generate and validate new biological hypotheses.

### 1.1 The purpose of (metabolic) modeling

Why go through the effort of building models? To better understand the purpose of models, their advantages and possible applications, I will briefly introduce the discipline called ‘systems biology’, to which metabolic modeling belongs.

#### 1.1.1 Modeling in systems biology

Without providing a formal definition, systems biology pursues to understand biological systems by building models of it. A model aims to mimic the system under study and allows us to make predictions or formulate hypotheses that can be verified with experiments. Ideally, correct model simulation and predictions indicate a good understanding of (that part of) the system. In contrast, false predictions indicate wrong assumptions, a lack of knowledge, modeling errors or other mistakes. These new insights are used to adapt the model, run new simulations and predictions until we understand how the system works. Importantly, good models not only capture the biological data observed, but also correctly predict new relations that could be inferred from the model, but have not explicitly been used for its construction. In addition, the model can be used to zoom in to the system in order to obtain a deeper level of understanding. Another option is to use the model to make predictions in similar organisms or systems, or connect the model of the studied biological system to those of related biological (sub)systems.

**What is a biological system?**

What makes something a system? Well known examples are a computer system, a car engine or a television. Without providing a formal definition, three things are required for a functioning system: an input, an output and something that processes the input and produces the output. In the systems above, an input...
can be a key stroke, button push on the remote control or stepping on the gas pedal. The output is a letter on the screen, a switch of television channel or an accelerating car. For most purposes, we don’t care about how the system works; to us it is just a *black box*. However, this black box processes the input in a coordinated fashion; many smaller parts ‘cooperate’ in time and space to process a task and finally provide the output we observe.

Biological systems work in the same basic fashion [61, 185]. For instance, the growth medium or temperature can be changed or a gene can be knocked-out (removed). Changing the cell’s environment mainly affects the cellular input, whereas a gene knock-out affects the functioning of the system itself. In both cases, the cell responds by regulating other genes and pathways, building other enzymes and routing metabolic flux through other pathways. This may be observed as a different cellular phenotype, an increased growth rate or the secretion of different metabolites. Understanding why this cellular response happens and in particular how it is realized by building models that mimic this behavior is the aim of systems biology in general, and understanding metabolism is the purpose of metabolic modeling in particular.

### 1.2 Computational models in biology

There are many modeling formalisms for biological processes [136, 61]. Which formalism to adopt mainly depends on the process that is modeled, the purpose of the analysis, the available data and the scope. In this thesis I mainly used COnstrained Based Reconstructions and Analysis (COBRA) networks to model metabolism at the genome-scale. COBRA models impose constraints on the reaction network and are widely adopted to model metabolism at this large scale. To put this modeling approach in context, I will briefly discuss some other popular modeling formalisms.

**Boolean networks**

Boolean networks [112] model active and inactive ‘states’ and logic transitions between those states. They are often used to model how transcription factors regulate other genes (which themselves may be transcription factors) in gene regulatory networks. In this way, they can model a pattern of gene activation and repression. They do not require a lot of quantitative knowledge, but therefore are also largely restricted to modeling qualitative (i.e. ‘active’ vs ‘inactive’) states.
1.2. Computational models in biology

Kinetic models

In contrast, kinetic models (using differential equations) have successfully been used to model metabolic processes on a smaller scale, but in greater detail [119, 115]. However, detailed metabolic concentrations and kinetic (enzyme) parameters are also required for these models. This information is often organism- and condition specific and typically only known for a limited set of reactions. However, rapid advances in high-throughput omics techniques allow researchers to obtain the required parameters at an increasingly larger scale, which recently resulted in a complete kinetic model of the human red blood cell [21]. Despite these advances, kinetic models are currently mainly used to study the dynamics in smaller systems. In contrast, constraint-based models study the genome-scale capabilities of the cell, but in a steady-state. Not surprisingly, there are various modeling formalisms in between the ‘large-scale and general’ boolean models and ‘specific and small-scale’ kinetic models.

Petri-nets

Petri-nets consist of places, transitions and tokens. The assignment of tokens (e.g. metabolites, proteins or genes) to places determine the state (called the marking) of the network. Places are connected to transitions (interactions or reactions) and together they determine the system’s behavior. Stochastic- and colored petri nets are expansions that allow the assignment of probabilities to (time-delayed) transitions and different values (metabolite concentration, gene activation level) to the system. Petri nets have successfully been used to model biological systems such as metabolic- [33, 117] and signaling networks [83, 192].

Bayesian Networks

Bayesian networks (BNs) are probabilistic graphical models in which nodes are variables and arcs denote directed connections between variables [120]. BNs model how nodes (e.g. metabolites, proteins or genes) affect connected nodes using a probabilistic formalism. A BN is a popular and well-established modeling formalism to reason under uncertainty. Algorithms to infer networks from experimental data have contributed to their popularity in computational biology. An often mentioned downside of the ‘classical BNs’ is that they cannot model feedback loops, which are common in biological systems such as signaling-, transcription- and metabolic pathways.
Chapter 1. Introduction

Constraint-based models

Metabolic models (in particular their stoichiometry) are among the most well-defined models in computational biology. Most metabolic reactions, the enzymes catalyzing these reactions and the enzyme coding genes are known from literature and high-throughput genomic experiments for a large number of organisms. Although much of this information is known, it is challenging to infer which reactions are active under varying environmental conditions, such as nutrient availability, stress or pathologies. Moreover, for higher order organisms the metabolic activity is highly dependent on the cell- and tissue type under study. The COBRA approach aims to deal with this uncertainty by using a computationally efficient, flexible and easy to interpret constraint-based formalism. In its most general form, the COBRA model imposes two types of constraints (eq. 1.1). First, it is assumed that the total rate at which a metabolite is produced equals the rate at which it is consumed. Thus, this mass-balance constraint assumes a cell to be in a pseudo steady-state, which means there is no net production or consumption of metabolites within the cell. However, nutrients can be taken up from the cell’s environment and metabolites can be secreted when they are mass-, chemically- and redox balanced. The second constraint limits the flux capacity and directionality of the reactions in the network. Briefly, this constraint does not allow thermodynamically infeasible flux directions and limits the reaction rate by assuming an upper bound on the enzymatic capacity.

\[
\frac{d\bar{x}}{dt} S \bar{v} = 0
\]

\[
v_{lj}^{lb} \leq v_{lj} \leq v_{lj}^{ub}, \forall j \in \{1, 2, \ldots, n\}
\]

The number of metabolic molecules that are consumed and produced in each reaction is called the reaction stoichiometry. The topology of the complete metabolic network can therefore be defined as the so-called stoichiometric matrix \(S\), consisting of \(m\) metabolites (rows) and \(n\) reactions (columns). Metabolic flux can be viewed as a flow or flux path of metabolites through the cell. Typically, there are (infinitely) many of such paths that satisfy the mass-balance and flux capacity constraints. More formally, the matrix \(S\) is system of \(m\) linear equations and \(n\) unknowns. Since there are many more reactions than metabolites \((n > m)\), this system is underdetermined and there is no unique (flux) solution, but rather a convex space of feasible solutions. These concepts are illustrated in figure 1.1.
1.3 Biologically relevant flux paths in COBRA networks

A steady-state flux path or flux distribution is a \( n \)-dimensional point in the space of feasible flux distributions. For many organisms, this flux space has hundreds of dimensions and metabolic modeling pursues to find paths that are likely to be used by the cell. Therefore, we -often implicitly- pursue to narrow the hypervolume of this space, which boils down to either adding new constraints, or ‘tightening’ existing ones [20, 184]. There is a great deal of computational methods, often combined with experimental data for exactly this purpose. Most of these methods can be characterized in three broad categories, with overlaps between them.

1. Direct flux measurement

In theory, a reaction’s flux capacity constrains the allowable flux to its maximum enzymatic turn-over rate \( v_{max} \). In practice, this maximum is often unknown and an arbitrary large value is chosen for many reactions. However, if the reaction rate is actually measured, the range between the lower- and upper bound for this reactions rate can be drastically reduced. Unfortunately, direct measurement of metabolic reaction rates (flux) is currently only feasible for a limited set of reactions. Most importantly, uptake and secretion rates of nutrients from the environment are routinely measured. This is especially helpful for modeling unicellular organisms grown on a single carbon source (e.g. glycerol or glucose) in the laboratory that excrete a limited set of metabolites (e.g. lactate, ethanol or acetate). Since the network is assumed to be in steady-state, measuring the major in- and output fluxes of the cell often greatly reduces the feasible flux paths. Measurement of intracellular fluxes is more involved and is at this moment only possible for a small part of the metabolic network.

2. Optimization of a biological objective

A large part of the early success of COBRA models stems from growth rate predictions for unicellular organisms. The bacterium \textit{E. coli} for example shows an exponential growth rate given sufficient nutrients and faster growing bacteria have evolutionary benefits over slower growing variants. Therefore, it is not unreasonable to assume that this bacterium tries to maximize its growth rate given the environmental conditions. Indeed, it has been shown in the laboratory that \textit{E. coli} cultured for many generations evolves and adapt its metabolism towards a maximum growth yield in the culture medium [96, 64]. This maximum growth yield can be predicted from the network model using linear programming. This method, termed Flux Balance Analysis (FBA) [163] is one of the old-
Chapter 1. Introduction

The most successful examples of constraint-based modeling of metabolism. There are many reasonable biological objectives one can think of [200], which have been successfully tested by applying optimization techniques to metabolic networks in recent literature.

3. Integration with genome-scale ‘omics’ data

Given the successful applications of metabolic networks in unicellular organisms it is not surprising that people tried to adapt them to more complex, multicellular organisms, such as mice and humans. One of the major obstacles for applying COBRA models and methods to multicellular organisms is that they have no clear global biological objective function that can easily be optimized. This objective is highly cell or tissue specific, and even then it is difficult to determine what such an objective would be and how the objectives of different cells or tissues are unified in one organism. One notable exception is the modeling of cancer metabolism, where the maximum growth assumption of a cancer cell has shown important applications [221, 204, 63]. The advent of high-throughput techniques such as gene expression micro-arrays, RNA sequencing and mass-spectrometry made a more data-driven approach also feasible for other cell- and tissue types. In the last decade, a series of papers have shown successful integration of gene-, protein- and metabolite data with genome-scale models [135]. Most are driven by the same basic assumption: genes that are not expressed do not encode proteins (and thus enzymes) and therefore the reactions catalyzed by these enzymes cannot be active. In contrast, the proteins and enzymes can be present for expressed genes. Given that a cell dedicates energy to express the genes, it is likely that it requires the enzymes to be active and therefore most of these metabolic reactions should be active. Global optimization techniques that maximize flux through ‘active’ reactions and minimize flux through ‘inactive’ reactions aim to fit the metabolic state of the cell to the observed data. A similar rationale can be applied to the metabolites [199]. Although the relation between gene-expression and flux is not as straightforward as described here, these methods have been successfully applied to various cell- and tissue types in different organisms.

1.3.1 Applications of metabolic modeling

How can computational models help to address questions about metabolism? There are many types of computational models used in biology [136] (see section 1.2) and many biological processes and systems that can be better understood by modeling them [61]. In this part, I will constrain myself to applications of constraint-based models of metabolism; the main modeling formalism used
in this thesis.

**Bio-engineering** COBRA tools have been developed that can pinpoint genes that should be over-expressed or knocked out in order to increase the yield of valuable byproducts [155, 116, 213]. These techniques have been applied for instance in *E. coli* to increase the yield of succinate [181], lycopene [5], polyactic acid, lactic acid and the antibiotic actinorhodi [65]. In yeast, metabolic network models have been used to optimize the production of vanillin [25] and biofuels such as ethanol [82].

**Cancer and other human diseases** Metabolic models provided systematic insights into the phenomenon known as the *Warburg effect* [72, 225, 225, 204], which is frequently observed in cancer cells. Otto Warburg [229] showed that cancer cells often favor the use of glycolysis and fermentation pathways rather than the oxidative pathways that are used in normal cells, even when sufficient oxygen is present. This seems strange, because oxidation provides a much higher energy yield than fermentation pathways. One hypothesis for this phenomenon is that although the yield is lower, the total production rate can be higher given a nutrient surplus [221]. Another explanation might be that cancer cells try to avoid oxidation, because it is related to programmed cell death [221]. Gene expression data has been mapped to human metabolic models, to elucidate the pathways that are active in certain cancer cells [103, 101]. By combining these cancer specific models with optimization algorithms, potential drug targets could be found that significantly lower the fitness of cancer cells [63]. Metabolic models have been used to predict ‘anti-metabolites’; anti-growth factors that target metabolites required for cancer cell survival and proliferation [76, 2]. Many more examples exist, where (metabolic) modeling is used to obtain a better understanding of the complexes processes in cancer, or to guide anticancer drug design [63, 204, 238, 240]. Beyond cancer, metabolic models have also been used to gain insights in liver disease [141], obesity and diabetes [222] to name just a few.

**Other** Another application of metabolic networks is to address evolutionary questions [96, 239, 166, 167]. Pál et al. simulated adaptive evolution of endosymbionts from a common ancestor, by first constraining the uptake medium to match that of the natural environment of the endosymbiont. Then, evolution was simulated by computing the impact of randomly selected gene losses on the organism’s fitness. Genes that did not significantly contribute to the organisms fitness were iteratively removed in this random fashion, until a minimal set of essential genes was obtained. Indeed, this predicted minimal set of metabolic
genes had high overlap with the actual set of metabolic genes of these organisms. This shows that genomic evolution can be predicted by simulation on a metabolic network, using only knowledge of a species ancestor and its natural environment. Furthermore, it was shown that the minimal gene set not only depends on the simulated environment, but to a large extent on chance (i.e. the order and random choice of alternative gene deletions). This explains why closely related endosymbionts may possess substantially different minimal metabolic gene sets.

Another study [96] showed that *E. coli* does not grow at the optimal growth rate predicted by FBA, on every carbon source. However, after growing *E. coli* for 700 generations, its metabolism was adapted to this carbon source, and the in vivo growth rate closely matched the optimum predicted by FBA.

Two recent studies [158, 80] showed that the metabolic network of *E. coli* contains a large number of so called underground reactions or pathways. These metabolic reactions are not active when the organism grows in its natural environment, but can be active when they provide a marked fitness advantage. This fitness advantage was realized by introducing gene knock-outs [80] or growing *E. coli* in a large number of different nutrient sources [158], such that the network of underground reactions could be studied. These studies showed that, first, adaptive evolution can be explained by the existence of a network of latent metabolic reactions. Furthermore, knowledge of the network of underground reactions can be utilized in bioengineering, in order to optimize strains for the production of valuable metabolites.

Metabolic models for plants have more recently become available. One of the first metabolic reconstructions in plants was that of the model organism *Arabidopsis thaliana*, but models for crops like sugarcane and maize are also available [174, 149, 162]. Many of the modeling principles and lessons learned from microbial metabolic modeling have successfully been applied to plants and already contributed to the understanding of their metabolism. It is highly likely that such models will have important applications, for instance in understanding plant diseases, increasing yield or engineering of crops that are better suited to grow in dry environments.

One of the important ‘applications’ of metabolic models, is that they allow the systematic integration of high-throughput data, such as gene- and protein expression. Why are certain metabolic genes co-expressed? Which reactions -and genes- are coupled? How are they regulated? Metabolic modeling can assist in answering these questions. Moreover, models of various related cellular systems become increasingly more integrated to answer more complex questions.
1.4 Genetic interactions within and beyond metabolism

Metabolic networks provide an integrated view on the functional dependencies between metabolic enzymes, genes, reactions and metabolites. One particularly interesting feature of such a model is that perturbations to any of these components can be simulated to predict their effect on other components. This idea has been widely used, for instance in metabolic engineering of bacteria. From the network, an optimal set of perturbations (e.g. gene deletions) can be computed in order to maximize the production of valuable byproducts in bacterial metabolism [236, 27]. Another valuable application is the computational discovery of metabolic vulnerabilities in disease networks. For instance, essential genes have successfully been discovered in the malaria causing protozoan Plasmodium falciparum [54], which make them interesting potential drug targets. Metabolic targets that disrupt tumor metabolism have been identified using a similar approach, applied to various cancer cell lines [63].

Metabolic networks have a highly robust architecture and are relatively robust to many single gene deletions. Moreover, important vulnerabilities are not always visible by using only simple flux simulation on the metabolic network. By combining the metabolic network topology with genome evolution, Lu et al. [131] identified a predictive pattern in converging metabolic pathways. It was shown that the genes involved in the protein complexes of two converging metabolic pathways (a ‘fan-in motif’) often have a negative genomic interaction. In particular, genomic evolution showed that in this fan-in motif, one of the two converging reactions was often not active, because the enzyme-coding genes were not expressed. It was shown that this information can be exploited in cancer cells, by identifying target genes in the protein complex of the other converging reaction. Interestingly, this pattern generalized to protein interactions derived from genome evolution beyond the scope of metabolism. The second part of this thesis focuses on such genetic interactions, either derived from a computational model, or inferred from genome-wide copy number and transcription data.

Chapter 4 describes genetic interactions in metabolism that can possibly be leveraged to target tumor cells. These interactions are derived from a computational model of metabolism and therefore genetic interactions unrelated to metabolism cannot be detected. Ideally, one would combine computational models and genome-scale data of these other cellular processes to discover important genetic interactions beyond metabolism. Indeed, computational models for other cellular processes such as transcription regulation, cell signaling or the cell cycle exist. However, they are often highly cell- or tissue specific or span a limited number of genes. Moreover, mapping large-scale genomics data on these mod-
Chapter 1. Introduction

ters is often non-trivial, for example because data needs to be discretized such as the boolean models described above. Other models, such as kinetic models can be detailed, but also highly specific to the process and conditions they model. Therefore, such models often do not generalize well to other conditions, making them less suitable for integration with other data. Therefore, rather than using a computational model, functional interactions are directly inferred from the data in the fifth and sixth chapter of this thesis. A major advantage of this approach is that it is not limited to any particular domain captured by a computational model. The major disadvantage is that interactions are derived from a statistical model, which indicates that a relationship between two genes exists, but in most cases does not provide a systematic framework that explains why such a relationship exists.

In chapter 5, we look at genome-wide genetic interactions in humans, with a special interest in interactions targeting cancer cells. Important concepts for chapter four and five are the Synthetic Dosage Lethality (SDL, Fig 1.2c) and Synthetic Lethality (SL, Fig 1.2b) respectively. Both concepts indicate an interaction between two genes that is lethal or detrimental to the cell. A SL is a combined knock-out of two genes that is lethal to the cell, whereas a single knock-out is not. A SDL is an interaction that requires one gene to be repressed and the other one to be (over)active. SLs and SDLs can be derived from a network model (chapter 4) or inferred from data such as gene-expression data sets (chapter 5).

A last concept explored in this thesis is that of gene co-regulation. Often, tens or hundreds of proteins are involved in the functioning of a molecular system or biological pathway. For the system to function, these proteins need to be present at the same time. Therefore, their genes are regulated in a similar fashion (co-expression, or co-regulation). Co-regulation can often be observed as genes with a similar gene-expression pattern in micro-array or RNA-sequencing data. If a sufficient number of genes that co-regulate in a molecular system are already identified, novel genes can be predicted from gene-expression data sets. This is the topic of chapter 6.

1.5 Outline of the thesis

Chapters 2-4 of this thesis are about constraint-based modeling of metabolism. Imposing the constraints in eq. 1.1 results in a space of feasible flux distributions, as illustrated in Fig. 1.1d. Chapter 2 and 3 respectively deal with the characterization of this space and the aim to reduce this space to the biologically most relevant flux paths. Chapter 4 exemplifies how computational simulation can be used to detect and prioritize cancer cell vulnerabilities.
1.5. Outline of the thesis

Figure 1.2: The concept of Synthetic (Dosage) Lethality. 

a) A cancer cell takes up nutrients ($v_1$ and $v_2$) and directs metabolic flux to cell growth. 

b) SL: A knock-out of the genes $G_1$ and $G_2$, which encode the enzymes catalyzing reaction $v_7$ and $v_8$ blocks both paths required for cell growth, which kills the cell. Note that a single knock of either gene $G_1$ or $G_2$ is not lethal to the cell, since in that case not all paths to growth are blocked. 

c) SDL: the over-expression of gene $G_2$ routes the metabolic flux away from cell growth (reaction $v_6$). On top of that, a knock-out of $G_1$ blocks the alternative flux path ($v_7 \rightarrow v_9$) to cell growth. Instead, the metabolite $D$ is excreted to the environment.
Chapter 1. Introduction

Chapter 5 and 6 are about genetic interactions (the SL) and co-regulation and are not limited to metabolism. In chapter 5, a method is described that finds SL patterns from a large set of genomic data and combines it in an automated method to select the most promising novel SL candidates. Chapter 6 is about a webtool that crawls a large amount of gene-expression micro-arrays to find genes that belong to any molecular system of interest. A short introduction of every chapter is given in the following paragraphs.

In chapter 2, an improved sampling tool is introduced that samples feasible flux distributions (n-dimensional points) from this space. By taking samples from the so called steady-state solution space one can -to some extent- determine which flux paths are likely and which ones are not. A big challenge is to sample points uniformly distributed over this space in reasonable time. The uniform requirement informally states that all possible flux paths should have an equal chance of being sampled, without biasing towards certain flux paths. As models get larger, sampling time increases and it becomes harder to obtain uniformly distributed samples.

Chapter 3 focuses on shrinking the large solution space, in which it is still unknown which flux paths are utilized by the cell. For many human cells it is unknown what their metabolic objective is, not to mention how to convert it into a mathematical objective. Two basic ideas are used in this chapter. First, it has been observed that organisms living in various environments do not reach the theoretical maximum growth level, but roughly 60-90% of that. One hypothesis is that a full commitment of a cell’s metabolic machinery to one energy source greatly affects their ability to switch to other nutrient environments. In contrast, a slightly smaller commitment largely retains their adaptability [201] making cells much more robust to possible environmental changes. Second, this robustness issue is exploited in an algorithm that excludes flux distributions that severely affect an organisms robustness. Finally, it is used to prioritize experimental measurements that provide maximum information about the flux paths utilized by the cell.

Chapter 4 is an example of how metabolic networks can be applied in cancer research. In this chapter FBA is used to find a combination of a highly active reaction and a near inactive reaction that reduce tumor growth when combined. These so called Synthetic Dosage Lethality (SDL) pairs were found and ordered by their predicted degree of lethality. Comparison of the predicted SDL pairs with gene-expression data from a large cohort of breast- and ovarian cancer patients demonstrate the ability of the SDL algorithm to predict SDL tumor vulnerabilities in silico. Patients with many such SDL pairs exhibit longer cancer survival times and decreased tumor sizes compared to other patients. Unfor-
Fortunately, most patients do not express many of these SDL pairs as cancer cells select against their expression in order to grow and proliferate.

In this study, I conceived and implemented the computational algorithm called IDLE, ranked the SDL pairs and performed the tumor size and (cumulative) survival analysis on the breast- and ovarian cancer patients.

In chapter 5, an ensemble-based classifier that predicts synthetic lethal (SL) interactions from cancer genomic evolution data is described. Here, genome-wide copy number variation and gene-expression data from a large number of cancer- and healthy tissue was used. Five patterns of gene loss, co-loss and compensation events were inferred from this data. These five predictive signals were successfully combined to train a classifier that ranks interactions between pairs of genes by their likelihood of being lethal to the cancer cell.

Identifying which genes are involved in a molecular system, pathway or disease is important to understand its function. A number of genes are often known, but it is unknown whether that information is complete or if other genes also play a role. The topic of chapter 6 is to identify other genes involved in or belonging to such a system, given a set of genes known to be active in that system (called query genes). Novel genes can be identified by observing patterns of gene co-expression in a large number of gene expression micro-arrays. However, a molecular system is usually not active in all possible cells, tissues or conditions and therefore not all micro-arrays are equally relevant for the query system. Micro-arrays in which many of the query genes are expressed seem to be more important and receive more weight in the webtool than those in which they are less expressed. This idea had already been introduced and successfully applied before [10], but an easily accessible webtool that automated this process was not available yet. Apart from enabling users to find novel candidate genes that co-express with their query genes, it also contains a database of genes that strongly co-express with annotated sets of genes. These genes are known from literature to carry out a common molecular function, are involved in the same biological pathway or take part in the same cellular component.

In this study, I implemented a part of the webserver and database and precomputed the weighted gene co-expression for the terms annotated in the Gene Ontology, Reactome and KEGG databases. I also performed the comparative analysis with the other co-expression webtools.

Finally, chapter 7 provides concluding remarks and discusses future perspectives for (metabolic) modeling in biology. Computational models have strongly increased in size and complexity in the last decade and probably will do so in the next ones. The creation and validation of computational models has greatly been facilitated by the introduction of high-throughput experimental procedures in molecular biology. Likewise, these procedures have generated a
wealth of data and dramatically increased the demand for computational models and algorithms that process this data and extract new knowledge to make better predictions and generate novel hypotheses. There is a strong need for next-generation models that integrate data from multiple biological phenomena in order to elucidate their relations and interactions.
Sampling the solution space of genome-scale metabolic networks

Constraint-based models of metabolic networks are typically underdetermined, because they contain more reactions than metabolites. Therefore the solutions to this system do not consist of unique flux rates for each reaction, but rather a space of possible flux rates. By uniformly sampling this space, an estimated probability distribution for each reaction’s flux in the network can be obtained. However, sampling a high dimensional network is time-consuming. Furthermore, the constraints imposed on the network give rise to an irregularly shaped solution space. Therefore more tailored, efficient sampling methods are needed. We propose an efficient sampling algorithm (called optGpSampler), which implements the Artificial Centering Hit-and-Run algorithm in a different manner than the sampling algorithm implemented in the COBRA Toolbox for metabolic network analysis, here called gpSampler. Results of extensive experiments on different genome-scale metabolic networks show that optGpSampler is up to 40 times faster than gpSampler. Application of existing convergence diagnostics on small network reconstructions indicate that optGpSampler converges roughly ten times faster than gpSampler towards similar sampling distributions. For networks of higher dimension (i.e. containing more than 500 reactions), we observed significantly better convergence of optGpSampler and a large deviation between the samples generated by the two algorithms.

This chapter is based on Megchelenbrink W.L., Huynen M.A. and Marchiori E. (2014), OptGpSampler: An improved tool for uniformly sampling the solution-space of genome-scale metabolic networks, published in PLOS ONE. Supplementary material is online available at PLOS ONE.
2.1 Introduction

Modeling metabolic networks helps to unravel the complex machinery of metabolism within the cell. A classic approach is to model the reaction pathways in a dynamic fashion, using detailed kinetic data. For genome-scale models, often involving hundreds or thousands of reactions and metabolites, it is experimentally prohibitive to obtain the kinetic parameters involved. A constraint based approach has successfully been applied to model and address a wide range of biological questions in the absence of detailed kinetic data [111, 180]. By using a steady-state assumption, a first type of constraint dictates that all metabolite concentrations stay constant over time (mass-balance). A second type of constraint limits the flux rate for each reaction (flux-capacity and directionality).

The relation between the $m$ metabolites and $n$ reactions is described in the $m \times n$ stoichiometric matrix $S$. A positive stoichiometric coefficient $S_{i,j}$ means that the metabolite $i$ is produced by reaction $j$ and a negative entry indicates that the metabolite is consumed in that reaction. At steady-state, the mass balance and flux capacity constraints can be formulated as in eq. (2.1) and ineq (2.2) respectively.

\[
\frac{d\vec{x}}{dt} = S\vec{v} = 0 \tag{2.1}
\]

\[
v_{j,\text{min}} \leq v_j \leq v_{j,\text{max}}, \forall j \in \{1, 2, \ldots, n\} \tag{2.2}
\]

where $\vec{x}$ and $\vec{v}$ are vectors of metabolite concentrations and flux rates respectively. Each flux rate $v_j$ is bounded by ineq. (2.2). Although in some cases the bounds are known from experiments, for most reactions this is not the case and arbitrarily large values are used. Since the matrix $S$ is a system of linear equations, the constraints in eq. (2.1) and ineq. (2.2) form a bounded convex space [178], containing all possible values of $\vec{v}$. A major challenge is to characterize the biologically interesting flux distributions among all alternatives. Since the stoichiometric matrix is fixed, the remaining possibility is to add additional constraints or tighten the inequality bounds in ineq. (2.2). This can be done by incorporating measured flux data, which is often laborious, expensive or difficult to obtain, even for a small subset of all reactions.

Many constraint-based methods have been proposed to find flux distributions that are of biological interest. A successful approach is Flux Balance Analysis (FBA) [224], which introduces a biologically relevant objective function and uses linear programming to find a flux distribution that optimizes this objective. FBA proved to be especially useful for cell types with a well-defined objective function, such as maximum growth for unicellular species. Although FBA has
successfully been applied to determine possible phenotypes and byproduct secretion in various experimental settings, it is often unable to determine the underlying (internal) flux states [211]. Furthermore, for many objective functions, a wide range of alternative optima exists [138]. In order to obtain an estimated probability distribution of attainable flux values for each reaction in the network, methods based on uniform sampling are used. A fast algorithm for this task is called hit-and-run (HR) [206]. HR collects samples by iteratively choosing a random direction and a random step size in that direction such that the next point also resides in the solution space. For the irregularly shaped solution spaces of metabolic networks the Artificial Centering Hit-and-Run (ACHR) algorithm [113] is better suited, because it is tailored to sample in the elongated directions of the solution space. Partly based on ACHR, the uniform random sampling procedure known as gpSampler is often used to sample metabolic networks and implemented in the COnstrained Based Reconstruction and Analysis (COBRA) Toolbox [195]. Although these algorithms are often referred to as uniform random samplers, their convergence behavior in the context of genome-scale metabolic networks has not yet been thoroughly investigated. Besides its irregular shape, the solution space of genome-scale metabolic networks are often high-dimensional, containing hundreds to almost a thousand dimensions as in the human metabolic network reconstruction. Therefore in this paper we investigate uniform random sampling in the context of metabolic networks. Our contributions are threefold: (1) we introduce an efficient and effective random sampling algorithm, which combines the advantages of ACHR and gpSampler; (2) we propose a new measure to quantify the deviation between samples obtained from two independent sampling runs; (3) we perform a thorough analysis on five metabolic network models.

### 2.2 Methods

#### 2.2.1 Uniform random sampling

One of the first attempts to sample the flux states in a metabolic network used a rejection sampling technique [230]. In rejection sampling, the space of interest is enclosed by a regular shape, such as a parallelepiped. Samples are drawn from the uniform distribution over the parallelepiped and rejected if they violate the constraints for the enclosed space of interest. The nice feature of rejection sampling is that the samples are uniformly distributed over the enclosed space. However, fitting a regular shape tightly around the space of interest is hard and often impossible. This means that in higher dimensions, the volume of the enclosing shape grows explosively compared to the volume of the shape of
Chapter 2. Sampling the solution space of genome-scale metabolic networks

interest [206]. In this case, a very large fraction of the samples have to be rejected, making rejection sampling an inefficient method for genome-scale models.

The hit-and-run (HR) algorithm [206] mitigates this problem, because it samples directly from the solution space (figure 2.1). Hit and-run starts from a point \( \vec{x}_0 \) in the bounded space. It chooses an arbitrary direction \( \vec{u}_1 \) from the uniform distribution on the boundary \( \partial B \) of the unit sphere in \( \mathbb{R}^n \). The distance from \( \vec{x}_0 \) to the boundary of the solution space in the direction of \( \vec{u}_1 \) determines the maximum distance it can travel. An arbitrary step size \( \lambda_1 \) is selected in the (negative) direction of \( \vec{u}_1 \), such that the sampler does not step out of the constrained space. The next point \( \vec{x}_2 \) is determined by traveling distance \( \lambda_1 \) in the direction \( \vec{u}_1 \). By iterating this process, HR generates a chain of consecutive sample points. The fact that HR uses only the current point to obtain the next sample makes it a Markov Chain Monte Carlo (MCMC) method [77], which has been shown to converge towards the target distribution (uniform in our setting). In general, the constraints in metabolic network models lead to a convex space of irregular shape, which is elongated for reactions whose flux rates are loosely constrained, and is narrow for tightly constrained fluxes. A consequence of this phenomenon is that many sample points are close to the boundary of the solution space. Since HR chooses a direction \( \vec{u}_i \) uniformly from all possible directions, this enforces the sampler to perform small steps, so the next generated point is close to the previous one. In practice this prevents the sampler to fully explore the rest of the solution space. ACHR [113] alleviates the problem of getting trapped in regions close to the boundary because it tries to sample in the elongated directions, thus making larger steps possible. It iteratively generates samples by using an ‘artificial’ center of the space, which is empirically estimated at each iteration using the points sampled so far. ACHR consists of two phases: warm-up and (main) sampling. In the warm-up phase an arbitrary initial point \( \vec{x}_0 \) in the solution space is selected to generate a chain \( \{ \vec{x}_0, \vec{x}_0, \ldots, \vec{x}_W \} \) of \( W \geq n \) points. The requirement \( W \geq n \) ensures that after the warm-up phase, the set of directions spans \( \partial B \) with probability one [113]. Then the main sampling phase starts from \( \vec{x}_W \). By iteratively updating the empirical center and by using a direction from a randomly chosen previous sampled point to this center, ACHR explores elongated directions of the space. Figure 2.2a illustrates the warm-up and the main sampling phase of ACHR.

The sequence of ACHR iterates is not a Markov chain, due to the dependence of directions on prior iterates. Thus the sequence of iterates is not guaranteed to converge to a uniform distribution [113]. ACHR is at the core of gpSampler [195], a popular sampling algorithm for metabolic network analysis. Figure 2b illustrates how gpSampler works. The highly irregular shape of the solution space of metabolic networks makes a uniform direction choice on \( \partial B \) a poor
2.2. Methods

Figure 2.1: Illustration of Hit-and run (HR) sampling. HR starts at the point $\vec{x}_0$ in the solution space $S$. It chooses a random direction $\vec{u}_1$ and determines the maximum distance it can travel forwards or backwards in that direction. A random step size is chosen on the line $\vec{u}_1 \in S$. The next point $\vec{x}_1$ is obtained by traveling $\lambda_1$ in the direction $\vec{u}_1$. By iterating this process $T$ times, samples are obtained that are uniformly distributed in the space, when $T \rightarrow \infty$.

choice. Therefore, in order to generate a number of $T$ samples, gpSampler’s warm-up phase uses linear programming in a two parts procedure. In part (1) $2n$ warm-up points are generated by consecutively minimizing and maximizing the flux rate of each reaction. In part (2) the remaining $T - 2n$ warm-up points are generated by assigning random weights to the fluxes that should be optimized. The optimal solutions (and thus the warm-up points) for a linear program reside on the boundary of the constrained space [233]. Often, this causes the allowed step sizes to become very small, which makes it hard to move away from the boundary. Therefore, gpSampler uses a linear transformation to ‘pull’ the warm-up points more into the interior of the solution space. Then, each moved warm-up point is used in the main sampling phase to generate $T$ separate chains of length $k$. The user provided step count parameter $k$ determines the number of ACHR iterates between the starting point $\vec{x}_W$ and the end point $\vec{x}_{W+k}$ of each chain. Finally, the $T$ end points of the chains are returned as samples.
Chapter 2. Sampling the solution space of genome-scale metabolic networks

Figure 2.2: Conceptual difference between (a) ACHR, (b) gpSampler and (c) optGpSampler. Warm-up points are depicted as gray rectangles, samples that are stored as gray circles. Uncolored circles denote points that are visited by the sampler, but are not stored as a sample. 

a) The original ACHR algorithm starts at a point \( \vec{x}_0 \) and iteratively moves to a next point \( \vec{x}_i = \vec{x}_{i-1} + \lambda_i \vec{u}_i \). One chain is used, with step count \( k=1 \). The chain contains \( W \) warm-up points and \( T \) samples.

b) GpSampler uses the linear programming procedure described in the main text to find \( T \) warm-up points. Then, each of the warm-up points is iteratively moved in the space in the same fashion as the ACHR algorithm in (a), leading to \( T \) sampling chains. Each chain of length \( k \) returns its end point as a sample.

c) OptGpSampler obtains \( 2n \) warm-up points. For each of the \( p \) processors used, a warm-up point is chosen randomly as the initial point \( \vec{x}_{j0} \), with \( j \in \{1, 2, \ldots, p\} \). Starting from the warm-up points, new points are found in the same fashion as for the ACHR algorithm in (a), but now only every \( k^{th} \) point is kept as a sample. Again, the result is \( T \) sample points, but now these have traveled \( k \) up to \( kT/p \) steps from a warm-up point. Compared to gpSampler, it uses less but much longer sampling chains.
2.2.2 optGpSampler: an improved sampling tool for metabolic networks

We propose to combine a part of the warm-up phase of gpSampler and ACHR. The resulting algorithm is called optGpSampler (algorithm 1). First, $2n$ warm-up points are generated using part (1) of gpSampler’s warm-up procedure by successively minimizing and maximizing the flux through each reaction. We do not generate $T$ warm-up points as in gpSampler because running a linear program on a large network is much more time-consuming than random sampling. Moreover, although the weight vector for the linear program is randomly chosen, the constraints in eq. (2.1) and ineq. (2.2) often lead to the same or a similar optimal solution. This could bias the starting points and directions choice of our sampler.

The sampling phase of optGpSampler is similar to that of ACHR (figure 2.2c), but only selects a sample at each $k$ iterates. Furthermore, instead of generating one chain of consecutive sample points, optGpSampler exploits $p$ processors and generates $p$ chains in parallel. In practice, the desired number of sample points $T$ is much larger than the number of available processors $p$. This makes the length of the chains generated by optGpSampler a factor of $T/p$ longer than those generated by gpSampler. We implemented optGpSampler in C++ and used Armadillo [194], a fast linear algebra library. OpenMP [19] was utilized to start a separate chain on $p$ processor cores in parallel. Interfaces for Matlab and Python enable users to easily sample existing models with optGpSampler or integrate our sampler in new methods.

2.3 Experiments

2.3.1 Datasets

We benchmarked gpSampler and optGpSampler on five publicly available reconstructions of genome-scale metabolic networks [196]. All reactions and associated metabolites that could not carry a flux were removed from the models prior to the sampling. The network size remaining after this preprocessing step, i.e. the number of $m$ metabolites and $n$ reactions is given between brackets. In ascending size order, we used *E. coli* central metabolism (68, 87), *C. thermocellum* iSR432 (288, 351), *S. cerevisiae* iND750 (479, 631), *E. coli* iAF1260 (1032, 1532) and *H. sapiens* recon 1 (1587, 2469).
2.3.2 Evaluation

We assessed efficiency and quality metrics of gpSampler and optGpSampler. Efficiency was measured as the algorithms runtime (single core). Quality performance in our context amounts to measure the capability of a sampler to effectively generate points uniformly distributed in the solution space. Since both gpSampler and optGpSampler have no theoretical convergence guarantees, we use two methods: empirical convergence diagnostics and a new method called xy-deviation.

Empirical convergence diagnostics

Empirical convergence diagnostics are used to test whether the sampled distribution converges towards a stationary one. There is not always good agreement between different convergence diagnostic methods [39]. Therefore we use the following three convergence diagnostics tests: Gelman and Rubin [74], Geweke [75] and Heidelberger and Welch (HW) [88]. They are available in the Convergence Diagnosis and Output Analysis (CODA) toolbox for MCMC [172]. The Gelman and Rubin test is multivariate: it returns a so called $R$ value (with $R \geq 1.0$). $R$ values smaller than 1.2 indicate convergence, with values closer to 1.0 indicating better results. The other two tests are univariate. The Geweke test returns a z-value for each reaction, with lower values indicating better convergence. The HW test returns whether a sample distribution for a given reaction converged (value=1) or not (value=0).

xy-deviation

In general, a larger step count provides a sample distribution that is closer to the target distribution, at the expense of a longer runtime. Therefore, we introduce a measure called xy-deviation that quantifies how samples generated by sampler $x$ using a given step count $k_1$ deviate from those generated by sampler $y$ using a much larger step count $k_2$. Specifically, a small deviation indicates that the sample distribution generated by sampler $x$ converged empirically to the target distribution of $y$. Given samplers $x$ and $y$, step counts $k_1$ and $k_2$, with $k_2$ very large ($k_2=5000$ in our experiments), the number $c$ of runs, and the number $T$ of samples, xy-deviation is computed as follows:

1. Perform $c$ runs of sampler $x$ with step count $k_1$, and $c$ runs of sampler $y$ with step count $k_2$, where each run generates $T$ samples.

2. Sort the points in each run, producing $c$ sorted chains of points for sampler $x$ and $y$: $< x_{b,i}^j >$, $< y_{b,i}^j >$, $b \in \{1, 2, \ldots, T\}$, $j \in \{1, 2, \ldots, n\}$, $i \in \{1, 2, \ldots, c\}$.
3. For each reaction \( j \), normalize the flux rates: we divide \( x_{b,i}^j \) (resp. \( y_{b,i}^j \)) by the difference between the upper and lower bounds of \( v_j \):

\[
X_{b,i}^j = \frac{x_{b,i}^j}{v_{j,max} - v_{j,min}}, Y_{b,i}^j = \frac{y_{b,i}^j}{v_{j,max} - v_{j,min}}, \quad i \in \{1, 2, \ldots, c\}, j \in \{1, 2, \ldots, n\}, b \in \{1, 2, \ldots, T\}
\]

4. Compute the mean chain of the \( c \) runs of \( y \):

\[
\bar{Y}_b^j = \frac{1}{c} \sum_{i=1}^{c} Y_{b,i}^j, \quad j \in \{1, 2, \ldots, n\}, b \in \{1, 2, \ldots, T\}
\]

5. Our assumption is that chains generated by sampler \( x \) that converge will have small deviation from the ‘average-chain’ generated by \( y \). For each reaction \( j \), we measure such deviation by computing the mean absolute deviation over the chains generated by \( x \) from \( \bar{Y}_b^j \):

\[
D_j = \frac{1}{cT} \sum_{i=1}^{c} \sum_{b=1}^{T} ||X_{b,i}^j - \bar{Y}_b^j||, \quad j \in \{1, 2, \ldots, n\}
\]

Our choice to use a mean deviation over the standard deviation is motivated by the fact that the mean deviation is more efficient than the standard deviation in the realistic situation where some of the measurements are in error, and more efficient for distributions other than perfect normal [79]. Note that the range of \( D_j \) is [0,1], and can be expressed as a percentage, which makes it convenient to compare deviations across different reactions.

6. Then the xy-deviation over \( c \) chains with respect to \( k_1 \) and \( k_2 \) is defined as the average of the reaction deviations:

\[
\bar{D} = \frac{1}{n} \sum_{j=1}^{n} D^j \quad (2.3)
\]

Small values of xy-deviation indicate convergence of \( x \) to \( y \). In the experiments we analyzed self-deviation (xx-deviation) and cross-deviations (xy-deviation).

### 2.4 Results

We used the gpSampler implementation in the COBRA toolbox. Results of extensive experiments are given in supplementary material. Tables S1–S5 in File S1 provide results for all the runtime experiments performed. Figures S1–S6 in File S1 provide the results of the convergence diagnostic tests and xy-deviation. We visualized how a small or large xy-deviation translates to a similar or (highly) dissimilar sample distribution in figures S7–S11 in File S1.
2.4.1 Efficiency

We sampled all networks four times using T=10, 50 and 100 thousand samples, with step count $k=50$. Both gpSampler and optGpSampler were executed in parallel mode on an AMD desktop computer using $p=4$ processor cores in parallel. Efficiency results (Table 2.1), show that optGpSampler is roughly 6 to 40 times faster than gpSampler.

2.4.2 Quality

To assess the quality of the results, we performed four independent runs with each sampler. Each run collected $T=50,000$ samples and was repeated for six different step counts; $k \in \{50, 250, 500, 1000, 2500, 5000\}$. We ran all convergence diagnostics with the default settings in the CODA package. The convergence tests results were averaged over the four runs. A univariate test for a sampler outputs a vector of length $n$. Significance of the difference between gpSampler and optGpSampler was assessed by applying the Wilcoxon signed-rank test to the corresponding vectors. The results for the Gelman and Rubin test indicated convergence for all experiments, in disagreement with results of the Geweke and HW tests. The obtained chains were also used to compute the $xy$-deviation for the above given step count $k_1$, and $k_2=5000$. All sample chains were compared by the average of the four chains obtained at the highest step count of $k=5000$.

Small networks (less than 500 reactions)

![Figure 2.3: Empirical convergence for C. thermocellum iSR432. a) Convergence according to the Geweke diagnostic. b) Convergence according to the Heidelberger-Welch diagnostic. Convergence for optGpSampler is observed at approximately 500 steps. For gpSampler, both diagnostics only agree on convergence after $k=5000$ steps. Notice the higher HW convergence fraction of the latter at $k=50$ steps and at $k=5000$ steps compared to the steps in between.](image)
Figure 2.4: xy-deviation for *C. thermocellum* iSR432. *xy*-deviation from samples obtained with sampler *x* at step count *k*\(_1\) to sampler *y* using *k*\(_2\)=5000. **a)** Deviation to samples obtained by *y*=gpSampler. **b)** Deviation from samples obtained by *y*=optGpSampler. In both cases optGpSampler converges much faster to sampler *y*.

On the *E. coli* central metabolism network, the Geweke test returned relatively low *z*-values for both gpSampler and optGpSampler (figure S1 in File S1). Results of the HW test indicated that optGpSampler converged rapidly, at *k*=50 step count. Both the Geweke and HW tests showed that gpSampler needs a step count close to 500 to converge.

Results of *xy*-self-deviation demonstrate a large deviation of samples generated by gpSampler with a small step count from those generated with a higher step count. In general, results showed that optGpSampler with a low step count generates samples that are both close to those generated using a higher step count with the same sampler and to those generated by gpSampler with a higher step count. On the *C. thermocellum* iSR432 network the HW test showed significantly higher convergence rates for optGpSampler (figure 2.3).

Both the Geweke and HW tests showed that optGpSampler converged at step counts bigger than 1000. The situation for gpSampler is different: the values of the HW test dropped at step count 250 and then increased significantly when the step count reached 5000. This indicates two distinct points of convergence, one at small step count values and one at large values. This behavior can be explained by the way gpSampler collects its samples. It starts at a warm-up point and uses *k* iterates of the ACHR algorithm to obtain a sample point. It repeats this process, each time starting from a different warm-up point (figure 2.2). These warm-up points turn out to be close to each other, as a consequence of the linear programming procedure used. Thus gpSampler often starts the sampling from the same area of the solutions space. In higher dimensions, a small step count can prevent gpSampler to ‘travel’ far from the warm-up points. In this
Chapter 2. Sampling the solution space of genome-scale metabolic networks

case the small step count causes gpSampler to show a bias towards the regions close to the warm-up points and the sampling chain converges towards these regions. The large xy-deviation shows that the samples collected are different from those sampled at a higher step count (figure 2.4). The results for gpSampler at large step count ($k=5000$ steps) again indicates convergence. In this case the small xy-deviation indicates that an extended region is covered by the samples. Since optGpSampler does not restart at a warm-up point, it is better able to ‘escape’ from the regions near the warm-up points because it effectively uses much longer chains. Therefore, its convergence and xy-deviation results are better.

**Large networks ($\geq 500$ reactions)**

The convergence and results for the *S. cerevisiae* iND750 (figure S3 in File S1) and especially the *E. coli* iAF1260 network (figure 2.5) showed an even more surprising result. For these larger models, the convergence results for gpSampler declines when the step count is increased. For optGpSampler, we still observed an increasing convergence performance for the yeast network, although a much larger step count ($k \geq 2500$) was required. The more stable results for the larger *E. coli* iAF1260 network could be an effect of the minimum glucose setting of this network, which significantly reduces the attainable flux states. The xy-deviation results (figure 2.6) indicate that the samplers give completely different sampling results. The self-deviation (figure 2.6) reveals a small variability within the four independent runs for each sampler. This means that both samplers give relatively stable results, and thus that the deviation results observed between the samplers must be due to the difference between the algorithms. Finally, the results for the human network reconstruction (figures S5 and S6 in File S1) indicate that gpSampler converges already at low $k$. Although we believe that the sample distributions indeed converged in this case, it seems unlikely that they represent a uniformly distributed sample. First, the high dimensionality of almost a thousand and the declining results for the Geweke test make this unlikely. Next, the huge deviation with samples obtained by optGpSampler indicate a non-uniform distribution, especially since we saw that optGpSampler performs better on the smaller networks. The convergence results for optGpSampler seem more realistic, with a relatively large z-value and a HW test result that indicates that around 60% of the sampled flux distributions converged.
Table 2.1: Runtimes for the networks analyzed. The number of metabolites and reactions is denoted by \( m \) and \( n \) respectively. The dimensionality of the null space of \( S \), is given by \( N(S) \). Time gp (SD) is the mean runtime (seconds) and standard deviation for sampling \( T=50,000 \) points using gpSampler. Time optGp (SD) denotes the same metrics for optGpSampler. Experiments were performed on a 16 GB RAM AMD Phenom desktop pc.

<table>
<thead>
<tr>
<th>Network</th>
<th>( m )</th>
<th>( n )</th>
<th>( N(S) )</th>
<th>Time gp</th>
<th>Time optGp</th>
</tr>
</thead>
<tbody>
<tr>
<td>E. coli central met</td>
<td>68</td>
<td>87</td>
<td>24</td>
<td>137.16 (± 0.62)</td>
<td>3.6 (± 0.05)</td>
</tr>
<tr>
<td>C. therm. iSR432</td>
<td>288</td>
<td>351</td>
<td>70</td>
<td>258.57 (± 0.54)</td>
<td>10.20 (± 0.05)</td>
</tr>
<tr>
<td>S. cerev. iND750</td>
<td>479</td>
<td>631</td>
<td>180</td>
<td>496.57 (± 3.07)</td>
<td>21.81 (± 0.04)</td>
</tr>
<tr>
<td>E. coli iAF1260</td>
<td>1032</td>
<td>1532</td>
<td>525</td>
<td>1474.01 (± 6.78)</td>
<td>95.78 (± 2.62)</td>
</tr>
<tr>
<td>H. sapiens recon1</td>
<td>1587</td>
<td>2469</td>
<td>932</td>
<td>2910.26 (± 43.57)</td>
<td>349.05 (± 0.48)</td>
</tr>
</tbody>
</table>

Algorithm 1: optGpSampler(\( S \), \( T \), \( k \), \( p \))

**Input:** \( S \): the solution space; \( T \): sample count; \( k \): step count, \( p \): number of processors;

**Output:** \( P \): sequence of \( T \) sampled points;

/* Warm-up phase */
1 Generate \( 2n \) warm-up points as in part (1) of gpSampler’s warm-up phase;
/* Sampling-phase */
2 \( P = <>; \)
3 \( L = \lceil Tk/p \rceil; \)
4 for \( i=1 \) to \( p \) do
5 \( \bar{x}_0 = \) a point randomly chosen from the \( 2n \) warm-up points;
6 for \( j=1 \) to \( L \) do
7 \( \bar{x}_j = \) point generated from \( \bar{x}_{j-1} \) by performing one iterate of the ACHR sampling phase;
8 if \( j \mod k == 0 \) then
9 \( P = P, \bar{x}_j; \)
10 end
11 end
12 end

### 2.5 Discussion

We proposed a new algorithm for uniform sampling of the steady-state solution space of metabolic networks. Our algorithm also implements ACHR, but in a different manner than the state-of-the-art sampling method for metabolic networks (gpSampler). We compared the runtimes with those of gpSampler, and showed its superior efficiency. We investigated empirical convergence using dif-
Figure 2.5: Empirical convergence for *E. coli* iAF1260. a) Convergence according to the Geweke diagnostic. b) Convergence according to the Heidelberger-Welch diagnostic. For both convergence tests, gpSamplers performance deteriorates when the step count is increased up to \(k=2500\). Especially the good scores at low values of \(k\) seem unrealistic and could indicate convergence towards a non-uniform distribution. Results for optGpSampler seem more stable and more reliable.

Figure 2.6: *E. coli* iAF1260 xy-deviation from samples obtained with sampler \(x\) at step count \(k_1\) to sampler \(y\) using \(k_2=5000\). a) Deviation to samples obtained by \(y=gpSampler\). b) Deviation from samples obtained by \(y=optGpSampler\). Self-deviation \((x=y)\) is small for both samplers, but there is a large cross-deviation \((x \neq y)\). For this large network, we do not observe convergence of gpSampler to the samples obtained by optGpSampler or vice versa as in figure 2.4.

Different diagnostics and showed faster convergence of optGpSampler on the two smaller networks studied. Moreover, by using the here introduced xy-deviation measure we compared the sampled distributions. For smaller networks, the samples obtained by optGpSampler using a small step count are close to those obtained with a high step count by both optGpSampler and gpSampler. On three large networks the convergence performance of gpSampler diminishes when the
step count increases. We hypothesized that the approach gpSampler takes by starting each sample chain at a warm-up point, together with the high dimensionality of the solution space restrains its ability to move from the vicinity of these warm-up points. Because our method continues the ACHR procedure from the last collected point it effectively uses much larger step counts. Therefore, optGpSampler is more likely to escape the regions near the warm-up points leading to a better sampling result. For the larger networks, results showed that the convergence observed at lower step counts does not reflect a convergence towards the target distribution since the sample distributions deviate significantly from those generated using a large step count. Therefore, especially larger networks should be sampled with a high step count. To the best of our knowledge there is no method to assess whether samples are truly uniformly random distributed in a convex space of unknown shape. Since the chains generated by ACHR are not Markov chains, asymptotic convergence guarantees also do not hold for both gpSampler and optGpSampler. Therefore convergence results should be interpreted with caution. The accelerated convergence of ACHR towards a uniform distribution was demonstrated by [113] for convex polytopes of known shape. However it remains uncertain to what extent the samples obtained by ACHR for the irregular solution space of metabolic networks are truly uniformly distributed. As expected, our experiments indicate that convergence results deteriorate when the dimensionality of the solution space increases and that for the large genome-scale metabolic networks using a large step count is advisable.

We envisage the provided implementation of optGpSampler will be beneficial to constraint-based metabolic network analysis as it provides an efficient and versatile algorithm for sampling the irregular solution space of metabolic networks.
Genome-scale metabolic networks can be modeled in a constraint-based fashion. Reaction stoichiometry combined with flux capacity constraints determine the space of allowable reaction rates. This space is often large and a central challenge in metabolic modeling is finding the biologically most relevant flux distributions. A widely used method is flux balance analysis (FBA), which optimizes a biologically relevant objective such as growth or ATP production. Although FBA has proven to be highly useful for predicting growth and byproduct secretion, it cannot predict the intracellular fluxes under all environmental conditions. Therefore, alternative strategies have been developed to select flux distributions that are in agreement with experimental “omics” data, or by incorporating experimental flux measurements. The latter, unfortunately can only be applied to a limited set of reactions and is currently not feasible at the genome-scale. On the other hand, it has been observed that micro-organisms favor a suboptimal growth rate, possibly in exchange for a more “flexible” metabolic network. Instead of dedicating the internal network state to an optimal growth rate in one condition, a suboptimal growth rate is used that allows for an easier switch to other nutrient sources. A small decrease in growth rate is exchanged for a relatively large gain in metabolic capability to adapt to changing environmental conditions.

Here, we propose Maximum Metabolic Flexibility (MMF) a computational method that utilizes this observation to find the most probable intracellular flux distributions. By mapping measured flux data from central metabolism to the genome-scale models of Escherichia coli and Saccharomyces cerevisiae we show that i) indeed, most of the...

---

This chapter is based on Megchelenbrink W.L. et al. (2015), Estimating Metabolic Fluxes Using a Maximum Network Flexibility Paradigm, published in PLOS ONE. Supplementary material is online available at PLOS ONE.
measured fluxes agree with a high adaptability of the network, ii) this result can be used to further reduce the space of feasible solutions iii) this reduced space improves the quantitative predictions made by FBA and contains a significantly larger fraction of the measured fluxes compared to the flux space that was reduced by a uniform sampling approach and iv) MMF can be used to select reactions in the network that contribute most to the steady-state flux space. Constraining the selected reactions improves the quantitative predictions of FBA considerably more than adding an equal amount of flux constraints, selected using a more na"ive approach. Our method can be applied to any cell type without requiring prior information.

3.1 Introduction

Advances in obtaining quantitative “omics” data have led to the availability of genome-scale metabolic network reconstructions for many organisms. Successful metabolic modeling examples range from predicting the impact of cell perturbation experiments in micro-organisms [176] and in silico yield optimization of valuable products such as bioethanol [24] to metabolic engineering for drug synthesis [143] and tumor vulnerability studies in cancer cells [101, 2, 160, 3, 63]. At the heart of these models lies the stoichiometric matrix \( S \), containing \( m \) metabolites and \( n \) reactions. Entry \( S_{ij} \) denotes the stoichiometric coefficient of metabolite \( i \) in reaction \( j \). The allowable flux range \( v_j \) for reaction \( j \) is bounded by the mass-balance equations (considered at steady-state) and flux capacity constraints:

\[
\frac{d\vec{x}}{dt} = S\vec{v} = 0
\]

\[
v_{j}^{lb} \leq v_j \leq v_{j}^{ub}, \forall j \in \{1, 2, \ldots, n\}
\]

where \( \vec{x} \) and \( \vec{v} \) are vectors denoting the metabolite concentrations and reaction rates respectively. In metabolic networks the reactions typically outnumber the metabolites, leaving the system of linear equations \( S \) underdetermined [20]. This means that there is no unique solution, but rather a convex space of (infinitely many) feasible flux distributions [163], known as the steady-state solution space. Knowledge of the actual flux distribution the organism utilizes is of great importance for many biological engineering purposes [20, 211], making reduction of the solution space a central problem in metabolic modeling. Since the reaction stoichiometry in eq. (3.2) is fixed, reduction of the solution space can only be achieved by tightening the feasible flux ranges. Methods for reducing the feasible fluxes to those that are biologically most relevant can be divided into three main categories.
3.1. Introduction

i) Computational methods that select flux distributions based on optimization of a biologically sound objective such as biomass or ATP yield. Flux Balance Analysis (FBA) [163, 224] is arguably the most applied technique that has shown to be accurate in predicting maximum growth [52] and byproduct secretion rates [223] for micro-organisms. Often, the flux distribution obtained by FBA is not unique and multiple optima exist. Flux Variability Analysis (FVA) [139] can be viewed as an extension of FBA that instead of finding a unique flux distribution computes the minimum and maximum allowable flux through each reaction while optimizing an objective function. To further reduce the space of alternative optimal solutions, variants of FBA are used. A method that is often applied is parsimonious enzyme usage (pFBA) [127]. This technique selects among the optimal flux distributions the one that minimizes the sum of absolute fluxes, using the rationale that a cell minimizes its enzymatic cost when alternative optimal flux paths exist. Other methods, tailored to minimize the enzymatic cost exist [202, 12, 204, 214] but require organism specific parameters that are not widely available, such as metabolite and enzyme concentrations or the Gibbs free energy change associated with each reaction.

ii) The most reliable method is direct measurement of the unknown intracellular and extracellular fluxes. Extracellular or boundary reactions such as glucose and oxygen consumption, together with growth rates and byproduct secretion rates such as acetate, ethanol and CO₂ are measured on a routinely basis. Unfortunately, measuring intracellular fluxes is currently limited by the available experimental techniques. A successful technique that measures intracellular fluxes is metabolic flux analysis (MFA) [231, 232, 242]. MFA uses isotopic (C¹³) labeling combined with a computational approach to uniquely identify the reaction rates inside central (carbon) metabolism. A drawback of the MFA method is that it is currently mainly limited to central metabolism and can therefore not be applied on the genome-scale.

iii) Computational methods that use other omics sources such as gene- or protein expression [11, 205, 191, 31, 99]. The basic idea behind most of these methods is to maximize the agreement between high (low) expression and active (inactive) pathways. Although the relation between gene-expression and metabolic flux is not straightforward and fluxes may not correlate well with the expression of their enzyme-coding genes [165, 4], this is a method that can be applied on the genome-scale. Many methods have shown to be highly predictive in a qualitative sense (predicting active versus inactive fluxes), but a recent review revealed that most are not well-suited for making quantitative flux predictions [135].

It has been shown that microbes that normally grow in various environments favor a suboptimal growth in order to keep a metabolic state that allows for an
Chapter 3. Estimating metabolic fluxes using maximum network flexibility

easier switch towards other nutrient conditions [60, 201]. Here, we utilize this trade off between robustness and growth to define a novel algorithm that allows for selecting flux ranges from the suboptimal growth space that correspond to maximum metabolic flexibility (MMF) in the network.

We evaluated our method on the genome-scale models of *E. coli* and *S. cerevisiae* and show that measured reaction rates indeed correspond to a high MMF value within the solution space corresponding to suboptimal growth. This information can be used to discard fluxes that would severely affect the organisms’ metabolic robustness and effectively reduces the flux range for each reaction considered. We compared our method with a uniform sampling approach and show that MMF provides a smaller prediction error rate when flux ranges are reduced at a similar cutoff. By applying pFBA after discarding all flux ranges that violate the MMF paradigm by more than 5%, better quantitative flux estimates are obtained. Finally, we demonstrate that our method selects reactions that contribute substantially to the uncertainty in the network. Measurement of the fluxes selected by our method provides a larger reduction and subsequently better prediction than can be expected from an approach that cannot estimate these most likely fluxes.

3.2 Materials and method

3.2.1 Computing the total flux range (TFR) distribution

Our method uses the idea that micro-organisms, which can grow in various environments typically grow at a suboptimal rate on a certain substrate, to facilitate an easier adaptation to other nutrient sources [60, 201]. Our hypothesis is therefore that the routing of metabolic fluxes in these micro-organism is such that they facilitate a substantial growth rate and are furthermore tuned to a maximal metabolic robustness. To clarify this, we introduce a measure called the total flux range (TFR). The TFR is the sum of the feasible flux ranges for a set of *n* reactions: $\sum_{i=1}^{n}(v_{i}^{ub} - v_{i}^{lb})$ and can easily be computed using FVA. The *n* reactions can be composed to cover all metabolic reactions in the network or only a specific subset of interest, such as the TCA cycle or the glycolysis pathway. The method proceeds in two phases. First, the feasible flux range $(v_{i}^{ub} - v_{i}^{lb})$ for each reaction *i* is binned into *b* bins. Denote with $v_{i} = \delta_{j}$ that the lower- and upper flux bound for reaction *i* are constrained to the lower- and upper bound of bin *j*. TFR($v|v_{i} = \delta_{j}$) then denotes the total flux range given the tightened bound on reaction *i*. TFR($v|v_{i} = \delta_{j}$) is computed by adjusting the lower- and upper bound of reaction *i* and propagating this constraint through the other reactions in the network (e.g. using FVA). Finally, this expression is normalized,
such that $\text{TFR}(v|v_i = \delta_j)$ denotes the TFR fraction remaining after constraining reaction $i$.

In the second phase, a unique flux distribution can be chosen with respect to the maximum metabolic flexibility paradigm. A first and straightforward approach is to discard all flux values that cause the TFR to drop below a certain cutoff. This can then be used to further constrain the flux bounds such that the space of possible flux distributions shrinks. This approach is illustrated for a toy network (figure 3.1a). The network consists of three metabolites connected by six reactions. Reaction $v_1$ and $v_2$ produce metabolite $A$, which is converted into biomass either through intermediate metabolite $B$ or $C$. Figure 3.1b illustrates the TFR distribution (purple) for each reaction as a function of the constrained flux through that reaction (here, for simplicity we set the lower- and upper flux bound to the center of the bin). Notice that for instance for the uptake reactions $v_1$ and $v_2$ a higher uptake rate corresponds to a higher TFR. This occurs because reaction $v_3$ and $v_4$ can obtain a wide range of fluxes given enough input through either $v_1$ or $v_2$. Similarly, when reaction $v_3$ obtains the highest rate of 10 mmol/gDW/h, there is only one feasible flux distribution and therefore $\text{TFR}(v|v_3 = [10,10])$ is 0. Notice that when the TFR cutoff is set to 0.5, the maximum flux through reaction $v_3$ is approximately 5.0 mmol/gDW/h and the minimal flux through the growth reaction ($v_6$) should also be 5.0. Using this approach, flux distributions that significantly violate the metabolic robustness principle can be excluded. To obtain a unique flux distribution, for instance pFBA can be applied to the network that satisfies a minimal predefined flexibility or robustness.

**Figure 3.1:** a) A toy network with 3 metabolites, connected by 6 reactions. Flux lower- and upper bounds are denoted between brackets. b) The TFR distribution (relative to the original TFR) for each reaction.
3.2.2 Computing the flux distribution with maximum metabolic flexibility (MMF)

A second option is to choose the flux distribution that maximizes the TFR under the imposed mass-balance and flux capacity constraints directly. This can be useful in the case that a clear biological objective such as growth is not present, which is the case for instance in most mammalian cells. Because the TFR distribution was computed for each reaction separately, selecting for each reaction the flux value that maximizes the TFR yields a non-steady-state flux distribution \( \hat{\rho} \). This can be solved by selecting the steady-state flux distribution that minimizes the overall Euclidean distance to the mode (peak) of each of the binned flux ranges. However, for some reactions the TFR is hardly sensitive to the chosen flux value (a uniform distribution), whereas for others there is a clear mode with maximum TFR (S1 File). The peaked distributions provide a better estimate of the real flux an organism can obtain and we therefore assign more weight to these reactions. The flux distribution that globally maximizes the TFR can be found by solving the following constrained weighted least-squares problem:

\[
\min_v \| Wv - W\hat{\rho}\|_2^2 \\
\text{s.t.} \\
Sv = 0 \\
v_{i}^{lb} \leq v_i \leq v_{i}^{ub}, \forall i \in \{1, 2, \ldots, n\}
\]

Here, \( W \) is a diagonal matrix of weights and \( \hat{\rho} \) is the (unconstrained) flux distribution that maximizes the TFR. The weight of a flux reflects how well it can be predicted with the MMF method. Therefore, distributions with a clear mode should receive more weight than uniform distributions. Shannon’s information entropy is used to determine the weight of each reaction \( i \):

\[
W_i = \log_2 - \sum_{j=1}^{b} \text{TFR}_{i,j} \log_2(\text{TFR}_{i,j})
\]

Note that when the TFR\(_i\) has maximum entropy (uniform distribution), the weight \( i \) is minimal (0). The vector of weights \( \vec{W} \) is normalized such that all entries sum up to 1. Eq. 3.2 can be formulated as a quadratic program that finds the flux distribution such that the network robustness or “flexibility” is maximized under the constraints imposed.
3.2.3 Using the TFR distribution to select flux measurements

MMF can be used to select or prioritize flux measurements. If measured flux data is available, it can be used to reduce the flux capacity constraints (the lower- and upper bounds). Furthermore, propagation (using FVA) of the new bounds typically reduces the bounds of other reactions as well. The amount of reduction caused depends mainly on three factors. First, the reduction of the measured reaction depends on the experimental precision. More importantly, how this reduction affects the feasible ranges of other reactions in the network depends secondly on what the actual flux is that was measured and furthermore on how that reaction is connected to the rest of the network (the network topology). For instance, if a high excretion rate of ethanol was measured, then much of the carbon excreted cannot go to other pathways (under the steady-state assumption). On the other hand, when a low ethanol excretion flux is detected it remains unknown how for instance carbon is excreted in order to satisfy the steady-state constraint. In other words, the organism might excrete other products such as acetate, glycerol and lactate or dedicate much of this flux to growth. Thus, a high excretion rate would typically provide a much larger reduction of the feasible space than a low excretion rate.

The TFR distribution is helpful because it provides a reasonable estimate of the flux through a reaction (supplementary file S1). Furthermore, we can immediately see how much reduction of the TFR can be obtained at each possible measurement outcome. Using the maximum flexibility principle, fluxes are selected where the mode of the TFR distribution is a global minimum. Any measurement outcome will reduce the TFR to at least this value. When the measured flux deviates strongly from the expected value, the TFR reduction will be even stronger.

3.2.4 Metabolic models and flux data

In a recent review about computational methods that integrate quantitative “omics” (gene-expression) data, it was shown that pFBA in general provides better quantitative flux estimates compared to most computational methods that integrate omics data [135]. Although pFBA predicts some fluxes with high accuracy, predicted rates for other reactions can be distant from the rates that have been experimentally determined. We demonstrate the applicability of our method using the same genome-scale metabolic models and experimentally verified fluxes as in [135]. In particular, we used the Escherichia coli iAF1260 [58] and the Saccharomyces cerevisiae iMM904 [151] network reconstructions. The yeast iTO977 network [164] used in [135] was replaced by the iMM904 model, be-
cause the former did not yield a feasible growth rate when FBA was applied. For the \textit{E. coli} model, fluxes have been measured in batch culture by Holm et al. [93] and in a chemostat environment by Ishii et al. [97]. The reaction rates for \textit{S. cerevisiae} have been measured at varying oxygen consumption rates by Rintala and coworkers [188].

### 3.3 Results

Feasible flux distributions form a high-dimensional convex steady-state solution space. The hypervolume of this space indicates how “many” (since the space is continuous, there is actually an infinite number of solutions) alternative solutions exist. Exact computation of this volume is infeasible for genome-scale models [51, 114] and reliable estimation of this volume is extremely hard [22]. Summing the feasible ranges over all reactions is a somewhat crude, but useful approximation.

It is known that organisms often do not grow at the theoretical optimal rate computed by FBA [216], but at a suboptimal rate. This is especially the case when nutrients are available in excess or the available oxygen is limited. The exact deviation between the predicted and actual growth rate depends on the experimental conditions. For the \textit{E. coli} and yeast models we considered, the measured growth rates are between 60% and 80% of the theoretical optimum computed by FBA. Under this suboptimal growth rate, the space of alternative flux distributions is considerably larger than when only distributions satisfying optimal growth are considered (figure 3.2). The hypothesis that metabolism in micro-organisms optimizes for growth allows FBA to effectively reduce the feasible flux ranges to only 10% to 30% compared to those in a minimal glucose medium without any maximization of growth. However, when the maximum biomass output is constrained to the measured suboptimal growth rate, this reduction is considerably less. Between 50% to even 70% of the original flux ranges (without optimizing for growth) satisfy the measured suboptimal growth rate. For organisms growing in a complex medium this reduction is even less and thus a large space of flux distributions that agree with the measured growth rate exist. Thus, there is need to further narrow down this space of suboptimal growth solutions using a secondary objective. We applied MMF to maximize the metabolic flexibility or robustness of the metabolic network.

#### 3.3.1 Reducing the total flux range

By reducing the feasible ranges of the fluxes, better estimates can be made on how flux is actually distributed through the cell. A computational approach
3.3. Results

Figure 3.2: TFR for various genome-scale metabolic reconstructions of E. coli and S. cerevisiae. The TFR was computed while obtaining a minimum growth output of 60, 80 or 100 percent of the maximum value computed by FBA.

suitable for reducing the space of (suboptimal) feasible solutions is uniform random sampling [195, 177, 146]. By random sampling the feasible space, empirical probability density functions (pdf) can be defined for every reaction in the network. These can then be used to tighten the lower- and upper flux bounds of each reaction by discarding flux values with very low probability. The MMF distribution can be used in a similar fashion by excluding reaction rates that cause the TFR to drop below a certain cutoff. This procedure is illustrated for four reactions in the E. coli genome-scale model (figure 3.3a; for the other reactions and networks see supplementary file S1). The sampling distributions (red lines) are narrow, meaning that many feasible flux values are actually never sampled and are therefore unlikely to occur given the network stoichiometry and flux capacity constraints. We also considered the MMF distribution and discarded all flux values that caused the TFR to drop below 0.95 (dashed blue lines). Although flux ranges based on the TFR distributions are wider, in contrast to the sampling procedure, they often include the flux rates measured by C\textsuperscript{13} MFA (orange squares).

In general, it is desirable to reduce the feasible space without excluding the real (i.e. the measured) fluxes. Figure 3.3b-d compares the reduction of the solution space with the fraction of measured fluxes that are outside the reduced space; the mean error. Results are provided for scenario 1, where only the glucose and oxygen uptake rates were constrained (figure 3.3b), when additionally the biomass flux was constrained (scenario 2; figure 3.3c) and all measured exchange reactions were constrained (scenario 3; figure 3.3d). The same trend is observed as in figure 3.3a; while the sampling method obtains a large reduction of the space, it also excludes most of the measured fluxes leading to a high
error rate. At a TFR cutoff of 0.95, the MMF method is more conservative compared to sampling. A smaller reduction of the space is obtained, but most of the measured reaction rates remain in the reduced space. When the cutoff is increased to 0.99, the TFR reduction becomes similar to that achieved by the sampling method (supplementary file S2). Notice that especially in the scenarios for which MMF was designed (finding flux distributions in the suboptimal space, i.e. where the growth rate is known), it achieves a smaller error rate than the sampling approach (figure 3.3c-d). In the remainder of this paper, we favor the smaller error over a large reduction and therefore use the cutoff of 0.95.

3.3.2 Quantitative flux estimation

A reduced solution space still does not provide a unique flux distribution. However, this can be obtained using either MMF to find the flux distribution that
3.3. Results

Figure 3.4: Flux prediction error for scenario 2. By first applying MMF, the predictions made by pFBA (MMF + pFBA) improve. The error improves in particular, when the biomass optimization paradigm may not hold (b and d). Furthermore, global optimization of the flexibility works pretty well, except for yeast growing under high oxygen conditions. In this case, yeast produces much biomass and fluxes estimated by pFBA are more accurate. By first applying ACHR, the feasible flux ranges are pruned to heavily, leading to worse flux estimates.

globally maximizes the TFR or by using pFBA on a model with refined bounds. The predictions made by pFBA improve when the MMF method was used to preprocess the model and narrow the flux bounds (figure 3.4a). In this particular example, E. coli was grown in chemostat culture at a dilution rate of 0.2/h [97] and the model’s glucose and oxygen uptake as well as the biomass production rates were constrained to those experimentally measured (scenario 2). In this case, pFBA underestimates some of the major fluxes in glycolysis (fluxes catalyzed by GAPD and enolase) and overestimates some of the reactions in the TCA cycle (ICT lyase and malate synthase). By first tightening the flux lower- and upper bounds, such that the TFR of each reaction is above 0.95 and then rerunning pFBA, better estimates of the fluxes through these reactions (green squares) are obtained. Similarly, by first applying MMF to find the bounds that retain at least 0.95 of the original TFR, and then applying the global MMF optimization procedure (MMF−2) in equations (3.2) and (3.2), we obtain a flux distribution that is closer to the measured values than those computed using pFBA alone.

Figure 3.4 shows the mean error for the aforementioned models in scenario 2 (see supplementary file S3 for all conditions). Notice that the quantitative fluxes estimated by pFBA improve when MMF was applied in advance to refine the flux ranges. Again, this holds in particular for the chemostat models, where the
maximum growth assumption used by pFBA may not hold due to limited oxygen availability (figure 3.4a and 3.4d). In the absence of oxygen, yeast uses the fermentation pathways to produce biomass and excretes ethanol at a high rate. It is known that the flux predictions made by FBA are incorrect in this case \cite{216} and refining the flux bounds with MMF helps to “guide” pFBA to a more accurate flux distribution. Notice that the MMF–2 method works reasonably well for most models, except for \textit{S. cerevisiae} growing under aerobic (high oxygen) conditions. The pFBA method, which assumes a maximization of the biomass yield is clearly a better approach here. Although MMF–2 does not outperform MMF+pFBA or pFBA alone, it can be used to estimate fluxes for models where an optimization objective such as growth is not applicable.

3.3.3 Selecting measurements to optimally reduce the solution space

We applied the MMF method to select among the measured fluxes those that are expected to maximally reduce the TFR. Selecting fluxes that have a large impact on the TFR may help to reduce valuable experimental time and cost. We iteratively selected one reaction and constrained the model with the measured flux until seven fluxes were selected. Because adding an extra constraint always reduces the TFR, we compared our method with two simple approaches. A rather naive approach is to select reactions randomly from those that have available measurement data. The second approach always selects those reactions that have the largest distance between their upper- and lower bound (called the MaxSpan reactions). Although the flux range of these MaxSpan reactions can be reduced considerably, they may or may not have a large constraining effect on other flux ranges in the network.

As a proof of concept, we looked specifically for reduction of the flux ranges for reactions inside central metabolism. Since flux data for other pathways was not available, little reduction can be expected within these pathways. The flux ranges in the \textit{S. cerevisiae} model are reduced faster when reactions selected by MMF are constrained with measured fluxes compared to the other methods (figure 3.5a; see supplementary file S4 for all models and scenarios). As expected, the larger reduction obtained after measurement of these reactions also results in better prediction of the fluxes by pFBA (figure 3.5b; see supplementary file S5 for all models and scenarios). Notice that although the TFR decreases with each flux measurement, this does not guarantee a better prediction of pFBA. The reason that the error can increase is that the additional constraint causes pFBA to perform a major flux rerouting, which is actually a worse estimate than the flux routing before imposing the extra constraint. This behavior is shown for a subset of the reactions (those in central metabolism) from the genome-scale model
3.4 Discussion and conclusion

A major advantage of constraint-based metabolic modeling is that relatively few constraints are required to predict quantitative traits such as a cell’s maximum growth rate, or uptake and secretion rates of key metabolites. The main drawback is that the underdetermined nature of these networks, combined with a limited set of available constraints makes it hard to compute or estimate many flux distributions accurately. Computational methods such as FBA and pFBA have been successfully applied to find the flux distributions that correspond with a maximization of the growth rate. However, micro-organisms rarely grow at this theoretical maximum rate, but – depending on the available substrates and species – at rates typically between 60% to 90% of this optimum. Impor-

Figure 3.5: TFR reduction and pFBA improvement of MMF compared to random and ‘MaxSpan’ selection on the S. cerevisiae iMM904 model (high oxygen). a) The MMF method selects flux measurements that provide a larger reduction of the flux ranges compared to the MaxSpan or random measurements. b) Fluxes predicted by pFBA obtain the smallest errors when the reactions selected by MMF are measured.

(figure 3.6). Some reactions in the glycolysis pathway have a large flux range and pFBA actually underestimates the flux through these reactions (figure 3.6a). MMF selects the reaction G3P $\rightarrow$ PEP for measurement. Due to the increased flux through glycolysis, more pyruvate enters the mitochondria. Apparently, the minimal sum of fluxes constraint drives a large portion of the flux through a shortcut, which converts citrate into malate. As a consequence, reactions in the right part of the TCA cycle have underestimated rates and the fluxes in the left part are overestimated. A second measurement constrains the flux from succinyl-coa to succinate, which results in a much more accurate prediction.
Figure 3.6: Effect of flux selection by MMF for explained with a network reconstruction of *S. cerevisiae* central metabolism. Red arrows denote overestimated fluxes by pFBA, compared to the measured data. Blue arrows denote underestimated fluxes. Using subsequent measurements in glycolysis (g3p $\rightarrow$ pep) and the TCA cycle (succinyl-coA $\rightarrow$ succinate), the pFBA estimates are much closer to the measured fluxes.

Tantamount, even a small relaxation of the growth rate from 100% to 90% of the maximum computed by FBA, increases the amount of alternative optimal flux distributions and thus the network robustness considerably. The FBA solution to a system that is constrained to have a maximum of say 90% depends mainly on the linear programming algorithm used and has no biological relevance compared to any other flux distribution within this suboptimal space. Using pFBA improves the situation, because the shortest absolute flux path is chosen from the optimal alternatives and thus thermodynamically infeasible cycles are avoided. The MMF method can be viewed as a (computationally expensive) alternative to minimizing the absolute sum of fluxes. Instead, a flux path is chosen that maximizes the “flexibility” or robustness of the network towards varying conditions while maintaining a (sub)optimal growth rate. This robustness is believed to be an evolutionary design principle of the metabolic networks of many organisms and protects the cell against internal defects and varying environmental conditions [100, 122]. The TFR summarizes the allowable flux ranges and thus can be viewed as a measure of metabolic robustness. We rarely observed flux distributions that severely limit the TFR. By using this simple observation we were able to refine the alternative suboptimal solution space found by FVA, by redefining flux ranges such that they satisfy a predefined TFR (here 0.95 times the original
3.4. Discussion and conclusion

TFR). Applying FBA or pFBA to this redefined space prevents that a flux route is selected that severely limits the metabolic robustness of the organism.

We would like to discuss our comparison with the uniform sampling approach. First, uniform sampling of metabolic networks is often done by the ACHR [113] method; a method that -contrary to the original Hit-and-Run method [207]- has no theoretical guarantees to converge towards a uniform distribution. The applicability of the ACHR method for large-scale metabolic networks has recently been challenged [42, 187] and it is questionable to what extent the flux distributions sampled from the steady-state solution space are truly a uniform random sample. On the other hand, binning and propagating the flux values as done in MMF, has some resemblance with belief propagation; a method that has also been applied to metabolic networks [22, 144]. Constraining the flux of one reaction and propagating this constraint through the network is exactly what belief propagation does. Then, the flux distribution with maximum TFR is a proxy to the one with maximum posterior belief. Thus, whether the reduced error rate of MMF compared to sampling is really due to the optimization of ‘metabolic flexibility’ or is mainly due to the limitations of the ACHR method remains unclear.

Our method can also be used on networks that are made tissue- or environment specific with computational methods that integrate omics data with genome-scale networks, such as iMAT [205], GIM3E [199] or EXAMO [191]. Despite the significant reduction that is often obtained by these methods, a large space of solutions remains.

Finally MMF can aid in identifying the key intracellular reactions. Here, to test the performance of the method, flux data measured by MFA was used, since data obtained with other techniques is scarce at the moment. Before MMF would be applicable as a measurement selection tool for MFA techniques, the main obstacle that has to be solved is taking into account the intrinsic coupling that exist between the tracers used and the pathways that are measured. That is, reactions measured with MFA cannot be chosen one at a time and the choice of isotopic label determines which reaction rates will be measured. A possible solution would be to extend the algorithm to a greedy heuristic that tries to select multiple consecutive measurements that best partition the flux space into smaller subspaces. This would be useful to compute a priori which labeling scheme provides most information about the network under study, especially since the error rates can also be computed in advance [197]. Our method can also be applied in association with Kinetic Flux Profiling (KFP) [241]. KFP is a mass spectrometry based approach to measure individual fluxes through metabolic networks by pulsechase feeding of heavy isotope (13C, 15N) labeled nutrients, and does not require specific assumptions about the network.
Taken together, we have presented a novel method that predicts metabolic fluxes by adopting a maximum network flexibility paradigm. Our method can be used to further narrow the solution space in genome-scale metabolic networks and thereby improve existing methods such as pFBA. Our method can predict (intra)cellular fluxes in the absence of known cellular objectives and can be used to find important hubs in the metabolic network that contribute most to the large range of alternative distributions.
A synthetic dosage lethality (SDL) denotes a genetic interaction between two genes, whereby the underexpression of gene A combined with the overexpression of gene B is lethal. SDLs offer a promising way to kill cancer cells by inhibiting the activity of SDL partners of activated oncogenes in tumors, which are often difficult to target directly. As experimental genome-wide SDL screens are still scarce, here we introduce a network-level computational modeling framework that quantitatively predicts human SDLs in metabolism. For each enzyme pair (A, B) we systematically knock out the flux through A combined with a stepwise flux increase through B and search for pairs that reduce cellular growth more than when either enzyme is perturbed individually. The predictive signal of the emerging network of 12,000 SDLs is demonstrated in five different ways. (i) It can be successfully used to predict gene essentiality in shRNA cancer cell line screens. Moving to clinical tumors, we show that (ii) SDLs are significantly underrepresented in tumors. Furthermore, breast cancer tumors with SDLs active (iii) have smaller sizes and (iv) result in increased patient survival, indicating that activation of SDLs increases cancer vulnerability. Finally, (v) patient survival improves when multiple SDLs are present, pointing to a cumulative effect. This study lays the basis for quantitative identification of cancer SDLs in a model-based mechanistic manner. The approach presented can be used to identify SDLs in species and cell types in which ‘omics’ data necessary for data-driven identification are missing.

This chapter is based on Megchelenbrink W.L., Katzir, R. et al. (2015), Synthetic dosage lethality in the human metabolic network is highly predictive of tumor growth and cancer patient survival, published in Proceedings of the Academy of Sciences of the United States of America. Supplementary material is online available at PNAS.
4.1 Introduction

Synthetic lethality (SL) occurs when the combined loss of two nonessential genes renders a lethal phenotype [86]. SLs have been studied by using experimental [37, 212] and computational approaches [210, 46, 203] to address various questions of cell function and evolution. The potential of SLs for cancer therapy has been recognized and accelerated the development of many SL screens [63, 68, 123, 134, 131]. (See refs. [108, 186, 29] for reviews of SLs applied in the context of cancer research.) Less studied are the so-called synthetic dosage lethality (SDL) interactions. An SDL is a genetic interaction between two genes whereby the underexpression of gene A \((A\downarrow)\) together with the overexpression of gene B \((B\uparrow)\) is lethal [121]. The observation that an interaction with an overexpressed gene can be lethal makes it particularly interesting for targeting cancer cells with (over) expressed oncogenes. This is because many oncogenes that drive tumor growth are essential to cell function and thus difficult to target directly. Targeting the oncogene’s SDL partner, which is a nonessential gene in normal cells, may nevertheless kill cancer cells. That SDLs can have important implications for cancer research, for instance to aid in the design of new therapies, has also been recognized [108, 16, 193, 208]. Moreover, it has been shown that the overexpression of specific genes can be detrimental to cancer cell growth [228].

Recently, a data-mining approach was used that identifies SLs and SDLs by analyzing large volumes of cancer genomic data [105]. Here we aim to complement data-driven computational efforts with a biological network model approach to identify SDLs. This has recently become feasible in the realm of metabolism, with the advent of genome-scale metabolic modeling. We introduce a method that uses a constraint-based genome-scale model of metabolism (GSMM) [150, 198, 178, 218, 215] to predict metabolic SDLs. GSMMs have successfully resolved a wide range of research questions in model organisms [178, 57, 160, 176, 153, 158] and have been the basis for many computational studies of cancer [63, 68, 104, 3, 102, 73]. Furthermore, they have contributed to a systematic understanding of the underlying mechanisms leading to lethality and SL [212, 210, 46, 203, 63]. A major advantage of a model-based approach is that it can provide insights into the underlying network mechanisms causing SDLs. Furthermore, the modeling approach presented is general and can be used to identify SDLs in species and cell types in which ‘omics’ data are missing. We introduce a computational approach for identifying dosage lethality effects (IDLE) in metabolism. IDLE predicts enzymatic SDLs from a GSMM with application to cancer. For each enzyme pair \((A, B)\) in the human GSMM, we systematically knock out the enzyme flux through A combined with a stepwise flux increase through enzyme B and quantify the level of growth reduction. Pairs in which the growth is significantly
more reduced than when either enzyme is perturbed individually are ranked as SDLs ($A^\downarrow, B^\uparrow$) with a corresponding value of ‘strength’. We demonstrate the predictive power of our approach in five different ways: First, by analyzing genome-wide experimental shRNA screens, we show that $A^\downarrow$ in predicted SDLs ($A^\downarrow, B^\uparrow$) is indeed more likely essential in an overexpressed enzyme $B^\uparrow$ background than when $B$ is not overexpressed. When $A$ is underexpressed and $B$ is overexpressed in a predicted SDL in a given tumor sample, we denote that SDL as ‘active’, that is, bearing potential functional effects on the tumor growth and the patient’s survival. Second, we show that SDLs are less frequently active across patients with cancer compared with randomly selected enzyme pairs, indicating that tumor cells select against the presence of SDLs to avoid cell death. Third, we illustrate that tumor size in patients with breast cancer (BC) with one or more active SDLs is significantly smaller than in patients expressing randomly selected enzyme pairs. Fourth, we show that the predicted impeding effect of active SDLs on tumor growth correlates with a significantly longer patient survival time. These results become even more pronounced when one includes only highly ranked active SDLs (that show a stronger $A^\downarrow$, $B^\uparrow$ pattern at the transcriptional level), illustrating that our method successfully identifies the clinical impact of SDLs. Finally, we report that observed effects become stronger when more active SDLs are present in a given tumor, pointing to the cumulative effect of active SDLs in clinical tumors.

4.2 Results

4.2.1 Overview of the IDLE algorithm

The IDLE method (figure 4.1, 4.2 and supplementary material section 1) computes the effect on cell growth when an enzyme $B$ increases its activity (referred to here as the reference GSMM) compared with its activity in a KO GSMM in which, additionally, enzyme $A$ is knocked out. The objective of IDLE is to find enzyme pairs ($A, B$) in which this differential growth effect is marked, searching over the space of all possible pairs. For a given pair ($A, B$), we define a reference WT GSMM and compute the maximum growth (biomass, $\mu_{\text{max}}$) with flux balance analysis [224]. Similarly, $\mu_{\text{max}}$ is computed for the KO GSMM, whereby reaction $A$ is knocked out. In both models, the maximum flux through $B$ is computed without any constraint on $\mu$ (i.e., the lower bound is zero; Figure 4.1 a and b show the reference and KO GSMM, respectively). Now, the lower bound of the biomass reaction is increased stepwise (by using $n=10$ steps) toward $\mu_{\text{max}}$ in both the reference (figure 4.1c) and KO (figure 4.1d) model. For each increase, the maximal allowable flux through reaction $B$ is computed again. The increas-
Figure 4.1: Conceptual overview of the IDLE method. a) The maximum flux through enzyme B is computed when there is no biomass pressure (i.e., the flux lower bound is zero). b) This process is repeated for the KO model. c) The biomass pressure is increased in a stepwise fashion and the maximum flux through enzyme B is computed at each step. d) This is repeated for the KO model. e) The maximum relative flux of B ($V_{B,\text{max}}$) is plotted at each biomass step ($\mu_{\text{max}}$) and the angle $\theta$ between the reference and KO vector is computed. f) SDL pairs are ranked based on their growth impact, quantified by their angle $\theta$ (see also 4.2).

4.2.2 The metabolic SDL network

Our method discovered 12,447 SDL interactions (supplementary material; section 2 and supplementary Dataset S1). Reassuringly, the ranked list of SDLs significantly matches the top-ranked metabolic SDLs identified by the data mining
4.2. Results

Create a metabolic model $m$ and a metabolic model $m'$ in which a specific enzyme A is knocked out (i.e. allowable flux = 0). Compute the maximum growth rate $\mu_{\text{max}}$ with FBA.

For $m$ and $m'$ set the minimum growth rate = 0 and the maximum growth rate = $\mu_{\text{max}}$. Now compute the maximum allowable flux through enzyme B.

Increase the minimal growth rate (here using 6 steps) in both models and again compute the maximum allowable flux through B.

The angle $\theta$ measures the vulnerability of the growth rate to an increased flux through enzyme B in the knock-out cell, compared to the normal cell. At the same growth rate ($\mu = 8 \text{ 1/h}$), the maximum allowable flux through enzyme B is lower in the knock-out cell, compared to the normal cell. Therefore, this growth rate can only be reached when the flux through enzyme B is ≤ 20 mmol/g-DW/h. Enzyme pairs $(A, B)$ are considered a "high-impact" SDL pair, when $\theta > 15°$.

Figure 4.2: The IDLE method. IDLE measures the ‘vulnerability’ of the growth rate to a flux increase through enzyme B. This reference model $m$ is compared with a model $m'$ that computes this vulnerability when additionally enzyme A is knocked out. This difference can be quantified as the angle $\theta$ between the vectors in the $m$ and $m'$ models. To accommodate for differences in flux scaling, the computation is done using relative differences.

synthetic lethality identification pipeline (DAISY), an approach for data-driven inference of genetic interactions in cancer that is based on the discovery of underrepresented gene pairs in cancer genomic data [105] (Wilcoxon rank-sum $p < 0.0038$). SDLs are asymmetrical by definition, i.e., $A \downarrow, B \uparrow$ denotes a different interaction than $A \uparrow, B \downarrow$ and each may have a very different magnitude; in the first interaction, enzyme A is the KO partner, whereas in the second interaction it is the overexpressed partner. Surprisingly, we discovered that six enzymes are major ‘master’ hubs, being the KO partners of many other overactivated $B \uparrow$ in the SDL network (supplementary figure S2). These major hubs (TPI, ENO, PGM, PYK, PGK, and GAPD) all reside in the glycolysis pathway. Interestingly, when examining the hub partners we observed that the $B \uparrow$ partners are the same for $\approx 80\%$ of the SDLs. The metabolic pathways that are enriched for these overexpressed partners are shown in supplementary table S3. To better understand the putative mechanisms underlying the workings of these SDLs, we conducted a further model-based analysis. First, we charted SL interactions of the six master hubs, i.e., searched for genetic interacting pairs involving these six hub reactions in which the growth reduction after their combined KO is larger compared with that observed after the single KOs. We were surprised to see that although these...
SDL hub reactions are highly sensitive to a synthetic dosage load (each being essential for $\approx 500$ overexpressed partners), they have only very few SL partners (a list of these reactions and their pathways is shown in supplementary table S4). Examining the SDL partners of the six central glycolytic hubs, we find that they are quite distributed across the metabolic network in ten different pathways that are significantly enriched with the SDL partners (supplementary table S5). When further investigating these SDLs, we discovered that glycogen production is decreased by (on average) 60% when such SDLs are active compared with the WT and KO conditions. Interestingly, it has recently been shown that glycogen metabolism and its initial accumulation is a key pathway induced by hypoxia, and its activity is necessary for optimal glucose utilization in tumors [55].

### 4.2.3 SDL is predictive of in vitro shRNA essentiality screens

We expect that a knockdown of enzyme A ($A^\downarrow$) will be lethal in a $B^\uparrow$ background in the case of a SDL $B(A^\downarrow, B^\uparrow)$. To study this, we exploited gene essentiality at a genome-wide scale in cancer cell lines by using experimental shRNA screens [140] and matched it with gene-expression profiles [8]. In a typical shRNA screen in a given cell line, each gene is individually knocked down by targeting its mRNA (inhibiting and degrading it) by specific shRNAs that bind to it. Then, the effect of each individual gene knockdown on cell growth is measured, from which scores are calculated that indicate gene essentiality (a $p = 0.05$ cutoff was used to consider a gene essential [140]). For each cancer cell line, we divided SDLs into two groups: group 1 consists of SDLs in which at least one of the B enzymes that form a SDL with enzyme A is overexpressed ($B^\uparrow$) and group 2 consists of SDLs in which none of the B enzymes are overexpressed (Materials and methods provides the definition of overexpression and supplementary material section 3 explains mapping genes to reactions). Then, the number of essential and nonessential A enzymes observed experimentally in the shRNA screen was compared between group 1 and group 2 in each cell line (one-tailed Fisher exact test). Using a $p = 0.05$ cutoff, we counted the number of cell lines in which enzymes A from group 1 are more frequently essential compared with these enzymes in group 2. This procedure was also repeated 5,000 times for a set of random enzyme pairs of equal size. As expected, the number of cell lines in which essentiality of A in a $B^\uparrow$ background is enriched (group 1) is significantly higher for SDL than for random pairs (empirical $p \leq 0.002$).
Figure 4.3: Percentage of active enzyme pairs (i.e., A, B) with A underexpressed and B overexpressed. When the angle $\theta$ increases, the fraction of active SDLs approaches zero. SDLs are significantly less frequently active than randomly chosen enzyme pairs. For all cutoffs, the p-values obtain their maximum significance.

4.2.4 Cancer cells select against SDL

Cancer cells are expected to select against the negative effect that SDLs have on (tumor) growth. Thus, when the enzyme pair (A, B) is a SDL, underexpression of enzyme A and overexpression of enzyme B should occur less frequently than for random enzyme pairs. We analyzed a gene expression dataset of 7,362 patients from the TCGA cohort [34] and determined for each gene whether it is underexpressed ($\downarrow$), overexpressed ($\uparrow$), or unchanged compared with expression levels in normal tissue samples [132] (supplementary material section 4). We then computed for all SDLs the number of patients, $F_{\text{SDL}}$, with an active SDL (A$\downarrow$, B$\uparrow$) relative to those patients having only enzyme A underexpressed (A$\downarrow$, B$\uparrow$) or having only enzyme B overexpressed (A, B$\uparrow$; supplementary material section 4). This was repeated for 5,000 randomly constructed enzyme pair sets of equal size ($F_{\text{Random}}$). As expected, $F_{\text{SDL}}$ is significantly smaller than $F_{\text{Random}}$, illustrating that an underexpression of A combined with an overexpression of B when A and B have a SDL relation occurs significantly less frequently than when the enzyme pair have no SDL relation (figure 4.3). In fact, when the angle $\theta$ increases, the fraction of patients that have an active SDL approaches zero, testifying to the
strong negative selection exerted on such SDLs.

**4.2.5 SDL correlates with smaller BC tumor size**

As SDLs negatively affect growth in cancer cell lines, we expect that the tumor size will be smaller for patients with at least one active SDL compared with those who do not. To address this, we used a dataset in which gene expression and matched tumor size data are available for 1,587 patients with BC [40]. We divided the patients in this heterogeneous dataset based on the estrogen receptor (ER) sensitivity of their tumor (key properties of the data set are provided in supplementary material section 5). We analyzed whether the tumor size of patients with an active SDL (\(A^↓, B^↑\)) is significantly smaller compared with patients who have one of the single effects, meaning only an under- (\(A^↓, B\)) or overexpression (\(A, B^↑\)) of enzyme A or B, respectively. To investigate \(A^↓, B^↑\) in relation to \(A^↓, B\), we separated patients into two groups: patients whose tumor overexpresses enzyme B (Materials and methods provides the definition of overexpression) with varying underexpression of enzyme A (\(\sigma\) between 0 and 3 given the underlying gene expression distribution) and patients whose tumor does not overexpress enzyme B with varying underexpression of enzyme A. When comparing \(A^↓, B^↑\) with \(A, B^↑\), we also separated the patients into two groups: patients who have enzyme A underexpressed (Materials and Methods provides the definition of underexpression) with varying overexpression of enzyme B (\(\sigma\) between 0 and 3 given the underlying gene expression distribution) and patients who have enzyme A not underexpressed with varying overexpression of enzyme B. Finally, we created random enzyme pairs (\(n=5,000\)) to serve as control for testing the specific effects of the SDLs. Statistical significance for all comparisons was computed with a signed Wilcoxon rank-sum test, analogous to the signed Kaplan-Meier test as previously defined [105] (see supplementary material section 6). As expected, we observed in ER\(^+\) BC that patients with (at least one active) SDL have significantly smaller tumors compared with patients with only overexpression of enzyme B (\(p < 4 \times 10^{-8}\); figure 4.4). We found for ER\(^-\) disease that the tumor sizes in patients with SDL are also significantly smaller compared with patients with only overexpression of enzyme B (\(p < 5 \times 10^5\)), as well compared with those with only underexpression of enzyme A (\(p < 7 \times 10^{-5}\)). Moreover, smaller tumors are observed for patients with ER\(^-\) and ER\(^+\) disease with active SDLs compared with patients with randomly selected enzyme pairs with the \(A^↓, B^↑\) pattern active (\(p < 2 \times 10^{-3}\)).
4.2. Results

4.2.6 SDL correlates with increased cancer survival time

As SDLs decrease BC tumor size, we hypothesized that their presence also affects patient survival. For the BC data, matched survival times were available such that we could correlate them to the level of SDL activation [40]. We hence performed a survival analysis analogous to the tumor size analysis described earlier. The significance of the results obtained for SDL were compared with the single effects and random pairs by a modified signed Kaplan-Meier test introduced in a previous publication [105] (see supplementary material section).
As expected, we found that patients with ER⁺ BC with at least one active SDL have significantly better survival times compared to patients with only an underexpression of enzyme A (p < 4 × 10⁻³, figure 4.5a and b). Patients with activated highly ranked SDLs show the longest ER⁺ BC survival times, as long as a median of more than 12 years (figure 4.5a). In line with expectation, the survival time of patients with active SDL is significantly better compared with patients with only enzyme B overexpressed (p < 3 × 10⁻⁴; figure 4.5c and d). Moreover, significantly longer survival is also observed for patients with SDLs compared with those with random enzyme pairs with the A↓, B↑ pattern active (p < 1 × 10⁻³). Supplementary material section 7 provides survival analysis of ER⁻ patients. As overexpression of enzyme B is generally not beneficial when enzyme A is not underexpressed, we wondered whether underexpressing enzyme B alone would be beneficial. Supplementary figure S4 indicates that this is not the case. In particular, severe underexpression of enzyme B correlates with increased tumor sizes (supplementary figure S4a and S4c) and decreased survival times (supplementary figure S4b and S4d) in patients with ER⁺ and ER⁻ BC.

4.2.7 SDLs predicted by IDLE are not expected to be specific for BC

To examine their predictive power in another cancer type, we analyzed a large cancer type-specific cohort of 921 patients diagnosed with serous epithelial ovarian cancer (OC) [81] with matched survival times. Indeed, the same observations were made as in the case of patients with ER⁺ BC, i.e., patients with OC with at least one active SDL have significantly better survival times compared with those with the single or random effects (p < 0.09 vs. A↓, B and p < 0.01 vs. all others; supplementary figure S5). These results are even more apparent in the relapse-free survival times (OC-RFS) of these patients (p < 0.02 vs. (A↓, B) and p < 9 × 10⁻⁴ vs. all others; supplementary figure S6).

4.2.8 Cumulative effect of SDLs in a tumor correlates with better survival

As SDL activity in a tumor correlates to survival prognosis, we asked if survival time would increase when patients have more SDLs active. We tested the presence of such a cumulative effect in the two largest cancer subtypes: patients with ER⁺ BC (n = 1,174) and those with serous epithelial OC (n=921). Patients were categorized into three groups, those having one to three, four to eight, or more than eight active SDLs in their expression profiles (over- and underexpression are defined in Materials and methods). The Kaplan-Meier survival curve
Figure 4.5: Median ER\textsuperscript{+} BC survival time (in years). Arrowheads denote the median survival for all patients with ER\textsuperscript{+} BC (7.4 y). The numbers of patients whose disease expresses at least one enzyme pair are denoted inside the figures. Note that the axis of figure a scales differently. a) Patients with at least one active SDL (A\textsuperscript{+}, B\textsuperscript{+}) with constant overexpression of enzyme B. b) Patients whose disease only underexpresses enzyme A (A, B\textsuperscript{-}) of the SDL. c) Patients with at least one active SDL (A\textsuperscript{+}, B\textsuperscript{+}) with constant underexpression of enzyme A. d) Patients whose disease only overexpresses enzyme B of the SDL (A, B\textsuperscript{+}).

(figure 4.6) shows, as expected, better survival for patients with large numbers of active SDLs compared with those with only a few active SDLs. Indeed, a log-rank test [17] revealed significantly improved survival times in both cancer types when the number of active SDLs increases (p < 8 \times 10^{-3} for ER\textsuperscript{+} BC and p < 2 \times 10^{-3} for OC and OC-RFS). The largest cumulative effect in the BC survival is related to SDLs being active with enzyme A as one of the major glycolytic hubs. Interestingly, the observed cumulative effect in OC is already present for patients who have four to eight active (figure 4.6b). The underacti-
vated enzymes A in these SDLs are enriched for pathways that use glutamine through glutamate metabolism, the TCA cycle and mitochondrial transport (p < 0.001, hypergeometric test). It has recently been shown that severe types of OC, such as the epithelial subtype we considered are driven by glutamine metabolism, in contrast to BC tumors that depend on an overactivity of glycolytic enzymes [237].

Figure 4.6: Kaplan-Meier survival curves for patient groups that have one to three, four to eight, or more than eight active SDLs. a) Survival times for the patients with ER+ BC. b) Survival times for patients with serous epithelial OC.

4.3 Discussion

Here we introduced what is, to our knowledge, the first computational method that captures enzymatic SDL effects in metabolic networks. Our method does not only identify SDLs that are strictly lethal to the cell, but also those that have a significant effect on tumor growth or proliferation in clinical settings (i.e., ‘synthetic dosage sick’). We show that our method is able to assign a measure of strength $\theta$ to each SDL, which correlates to its predictive power in an array of different tumor clinical attributes. It is therefore of interest to focus further research toward therapeutic interventions on the basis of ‘high-impact’ pairs, which may have the largest beneficial effect on killing cancer cells. We show that SDLs are less frequently active than expected in cancer cells. This shows that rapidly expanding cancer cells select against interactions that reduce their growth rate. The activation of ‘high-impact’ SDLs is associated with smaller tumor sizes and longer patient survival. The effect strongly depends on the extent to which SDLs are activated, but most SDLs we found do not require a complete enzyme KO to exert a functional predictive signal. Last, we demonstrated a cumulative ef-
fect of SDL presence; the more SDLs active in a tumor sample, the better this is for a patient’s prognosis. This observation may shed light on targeting cancers that rely on glycolysis. Down-regulating glycolytic enzymes that are the major hubs in the SDL network is hence expected to have a large growth-inhibitory effect in tumor cells that overexpress many of the glycolytic SDL partners. As glycolysis is usually less active in normal cells and SDL partners of glycolytic hubs are less frequently overexpressed in normal cells compared with cancer cells in the majority of tissue types (supplementary material section 2.2), targeting these glycolytic SDLs may be of therapeutic interest, especially when a large number of their partners are overexpressed. The present study, being the first of its kind of which we are aware, naturally focuses on harnessing the generic human metabolic model to identify a common core of SDLs that may be shared by many different cancer types. However, the IDLE approach is general and could be extended in the future to identify cancer type-specific SDL interactions more precisely by integrating patient- and tumor-specific omics data such as gene expression or proteomics. The results of our metabolic network modeling do not support the hypothesis that SDLs arise as a result of draining alternative compensatory pathways that compensate for the loss of the KO enzyme. This is because we do not find that the flux in such backup reactions of the major key glycolytic enzyme hubs is reduced following the overexpression of their SDL partners. Intriguingly, we do find that disrupted glycogen metabolism is predicted to be the major mechanism by which hundreds of SDLs of key glycolytic enzymes exert their growth-inhibitory effects. Indeed, it has recently been shown that glycogen metabolism and its initial accumulation is key for optimal glucose utilization in tumors [55]. Thus, SDL relations do not arise via simple proximal interactions, but are likely to be the result of complex stoichiometric network relations that withdraw flux from biomass production through activation of other pathways.

Our results testify to the potential contribution of model-based approaches to identify and uncover the mechanisms behind SDLs. Model-based SDL prediction via IDLE is widely applicable and not limited to cancer. It could be used to identify SDL networks in pathogenic bacteria or fungi, providing new antibiotic therapeutic leads. Other possible applications include metabolic engineering to increase the yield of valuable metabolic byproducts. Specifically, this may be achieved by engineering a SDL effect to inhibit the production of undesired byproducts, or inversely, neutralizing the SDL effect to force an increased flux through desired pathways. Taken together, IDLE is expected to contribute to various research fields ranging from medical sciences to biotechnology.
4.4 Materials and methods

IDLE requires a GSMM with \( m \) metabolites, \( n \) reactions and a well-defined cellular objective function. We used the human metabolic network (recon1) [50], supplemented with a biomass reaction to simulate growth. A rich environment was simulated by allowing a maximum metabolite uptake rate of 5.0 mmol/gram dry weight/h through all boundary reactions. The goal of IDLE is to find SDL enzyme pairs \((A, B)\) that severely interrupt cell growth when the flux through enzyme \( A \) is decreased and the flux through \( B \) is increased (denoted as \( A \downarrow, B \uparrow \)). As illustrated in figure 4.1 and supplementary material section 1, SDL is measured by an angle \( \theta \). Only those enzyme pairs with a significant difference between the reference and KO were analyzed, i.e., all pairs with \( |\theta| \geq 2^\circ \) were selected, resulting in a list of 12,447 putative SDLs. Supplementary material section 1 provides a detailed description of IDLE with an example.

In all analyses, we defined an enzyme/gene to be under- or overexpressed when its expression was below or above 0.5\( \sigma \) to 1.0\( \sigma \) from the mean in the gene expression distribution (see Results for references to gene expression datasets). Detailed procedures of mapping gene expression to enzyme reaction level and calculating the fraction of SDLs in cancer cells \( (F_{sdl}) \) and descriptions of tumor size and patient survival statistics are provided in supplementary material section 1.
Predicting genetic interactions from cancer genome evolution

Synthetic Lethal (SL) genetic interactions play a key role in various types of biological research, ranging from understanding genotype-phenotype relationships to identifying drug-targets against cancer. Despite recent advances in empirically measuring SL interactions in human cells, the human genetic interaction map is far from complete. Here, we present a novel approach to predict this map by exploiting patterns in cancer genome evolution. First, we show that empirically determined SL interactions are reflected in various gene presence, absence and duplication patterns in hundreds of cancer genomes. The most evident pattern that we discovered is that when one member of an SL interaction gene pair is lost, the other gene tends not to be lost, i.e. the absence of co-loss. This observation is in line with expectation, because the loss of an SL interacting pair will be lethal to the cancer cell. SL interactions are also reflected in gene expression profiles, such as an under representation of cases where the genes in an SL pair are both under expressed and an over representation of cases where one gene of an SL pair is under expressed, while the other one is over expressed. We integrated the various previously unknown cancer genome patterns and the gene expression patterns into a computational model to identify SL pairs. This simple, genome-wide model achieves a high prediction power (AUC = 0.75) for known genetic interactions. It allows us to present for the first time a comprehensive genome-wide list of SL interactions with a high estimated prediction precision, covering up to 591,000 gene pairs. This unique list can potentially be used in various application areas ranging from biotechnology to medical genetics.

This chapter is based on Lu X., Megchelenbrink W.L. et al. (2015), Predicting human genetic interactions from cancer genome evolution, published in PLOS ONE. Supplementary material is online available at PLOS ONE.


5.1 Introduction

A synthetic lethal (SL) genetic interaction is defined as a functional relationship between two genes where the loss of either gene is viable but the loss of both is lethal [85]. A comprehensive map of SL interactions sheds light on the relationships between genotype and phenotype [92, 246, 18, 89], potentially advancing the understanding of the mechanisms of complex human disease [56, 152], and even providing therapeutic treatment strategies for human diseases such as cancer [26]. For instance, several studies have shown that inhibiting one gene in an SL pair could be lethal to cancer cells in which the other gene of that pair is mutated [6, 110, 145]. The underlying concept is that, in a cancer cell, a mutation in one (A) of the two genes in an SL pair (A-B), which is not mutated in the normal cell, allows for selectively killing tumor cells by inhibiting B. Despite recent breakthroughs in technologies to identify SL interactions on a genome-wide scale [189, 94, 37, 7], these interactions remain largely unknown in human, underlining the need for predictive computational approaches.

Previous computational approaches have mostly been developed to predict SL interactions in model microorganisms, such as *Saccharomyces cerevisiae* and *Caenorhabditis elegans* [243, 235, 168]. However, genetic interactions are not strongly conserved between species, for instance only 29% of genetic interactions were found to be conserved between the fungi *S.cerevisiae* and *Schizosaccharomyces pombe* [49] and the conservation of SL interactions between microorganisms and human still has to be established. Recently, a study proposed to use cancer genomic data [105] to identify SL interactions by using a ‘compensation’ pattern: one gene (A) is inactive while the other one (B) is highly active, thereby selecting against the situation that both genes become lost and, as such, causing a lethal phenotype. We recently showed another genomic pattern of SL interacting gene pairs: SL interactions are reflected in present-day species genomes and their ancestral genomes in a way that the combined loss of two genes in an SL pair does not frequently occur across evolutionary history [131]. This raises the question whether we can use this ‘co-loss underrepresentation’ pattern to predict SL pairs from human cancer genomes (figure 5.1a). Here, we used copy number variations, i.e. gene loss or gene gain, across hundreds of cancer genomes to ask i) are empirical SL interactions reflected in cancer genome evolution and, if so, ii), which gain and loss patterns correlate most with SL interactions, and iii) can they be captured into a simple computational model to predict SL interactions genome widely?

By exploiting the availability of gene expression data for a large number of cancer samples collected by the Cancer Genome Atlas (TCGA) ¹ and recent em-

---

¹The results shown here are in whole or part based upon data generated by the TCGA Research
5.2. Materials and methods

5.2.1 Data sources

We retrieved the experimentally measured SL pairs and non-SL pairs from two studies [123, 226]. We collected 297 SL pairs and 6,358 non-SL pairs in total. After excluding the pairs of which both genes are located on the same chromosome, we obtained 270 SL pairs and 5,660 non-SL pairs (supplementary material table S1).

The CNV data was retrieved from the cBioPortal for Cancer Genomics [70]. The CNV signals in the database are generated as homozygous deletion, heterozygous deletion, normal copy, duplication and amplification. Using the cgdsr R-package, we obtained the CNV data for 14136 tumor patients from 31 cancer types.

RNAseq data were obtained from the Broad Institute’s Genome Data Analysis Center (GDAC) Firehose 2. The link for downloading the RNAseq data is http://gdac.broadinstitute.org/runs/stddata__2014_03_16/data. For a cancer study, we first downloaded the files named as _RSEM.genes.normalized.data.Level.3, which contains the estimated expression levels for each gene in human genome from RNAseq data by using the RSEM package [128]. In total we collected an expression profile for 7,362 tumor patients with coverage of 26 cancer types. Then, for each gene in a tumor, we computed the z-score and p-value to infer its over- or under-expression relative to expression levels in normal tissue. If at least 25 normal samples from the same tissue type as that of the cancer

Network: http://cancergenome.nih.gov

are available, we used this as the comparison set. Otherwise, all normal tissue samples, regardless of the tissue specificity, were used. The numbers of normal samples for each type of tumor are listed in supplementary table S2. To adjust for multiple hypothesis testing, we used the False Discovery Rate (Benjamini-Hochberg) method to adjust p-values [13, 14] in R. A cutoff of the adjusted p-value, 0.05, was applied to generate the over- or under-expression signal.

5.2.2 Extracting the pattern for SL pairs from genomic variations

The copy number variations can be, -2 = homozygous deletion, -1 = heterozygous deletion, 0 = normal copy, 1 = duplication, and 2 = amplification. For a gene pair (A, B), the co-loss event can be i) homCL: homozygous co-loss (-2, -2), ii) hetCL: heterozygous co-loss (-1, -1) or iii) mixCL: mixed co-loss (-2, -1 or -1, -2). For each co-loss event, we defined a fraction that quantifies the likelihood of the co-loss event. For instance, for the homozygous co-loss event, we defined the fraction for a gene pair A-B as \( f_1 = \frac{n_{\text{homCL}}}{n_t} \), where \( n_{\text{homCL}} \) is the number of patients with the homozygous co-loss of A-B and \( n_t \) is the total number of patients where A-B have a status as (-2, -2), (-2, 0) or (0,-2). We calculated the \( f_1 \) of a gene pair without including samples that have homozygous deletions of more than 2000 genes (tail of the distribution in Figure A in S1 File). We noticed that several tumor samples have a very high number of homozygous deletions (supplementary figure S1A). Such samples can lead to an inflation of the co-loss likelihood regardless of whether they have an SL interaction or not. Similarly, we defined two fractions, \( f_2 \) and \( f_3 \), for heterozygous co-loss event and mixed co-loss events correspondingly (Table 1 and figure 5.1). It should be noted that we did not use an approach in which we, in order to quantify under representation of co-loss events, compared the empirically observed co-loss rate of gene pair A-B with the product of the single loss rates for genes A and B. This approach assumes independence between the loss of randomly chosen genes, which is not what we observe (supplementary figure S1A).

The variations in gene expression can be: -1 = under-expression, 0 = normal, and 1 = over-expression. Here, we defined two fractions, \( f_4 \) and \( f_5 \) (table 5.1 and figure 5.1). \( f_4 \) quantifies the likelihood of both genes in a pair (A, B) are under-expressed. \( f_5 \) is used to quantify how likely gene pair A-B has the expression up-down events, i.e., A is over expressed and B is under expressed or vice versa.

Here, each defined fraction is a signal where SL pairs show difference from non-SL pairs. For \( f_1, f_2, f_3 \) and \( f_4 \), we expected that SL pairs have smaller values for these fractions than non-SL pairs. However, for \( f_5 \) we expected that SL pairs have larger values than non-SL pairs. To test these hypotheses, we com-
5.2. Materials and methods

**Figure 5.1:** Patterns across cancer genomes reflecting selection against gene co-inactivation and the workflow to predict SL interactions.
Figure 5.1: Patterns across cancer genomes reflecting selection against gene co-inactivation and the workflow to predict SL interactions. a) A SL interaction SL1 between gene A and B can show a ‘compensation’ pattern across cancer genomes in which it is more likely that when A is inactive (denoted by -1), B is overactive (denoted by 1) to compensate the inactive A (genomes 110), compared to when A is active (genomes 1130). SL interaction SL2 can show a ‘co-loss underrepresentation’ in which a combined loss of A and B (denoted by -1 and -1, genome 10) across cancer genomes is underrepresented compared to a loss of either one of the two (genomes 29 and genome 1418). Note that SL1 can also be identified via the co-loss underrepresentation pattern, but the SL2 can only be identified via the co-loss underrepresentation pattern. b) The model requires two types of data as input, i) CNVs measured by SNP arrays and ii) gene expression variations measured by RNAseq. In CNVs, the status of a gene can be a homozygous deletion (two dashed lines), a heterozygous deletion (one dashed and one solid line) or normal (two solid lines). For CNVs, we generated three fractions to quantify the likelihood that a gene pair has a homozygous co-loss ($f_1$), a heterozygous co-loss ($f_2$) or a mixed co-loss ($f_3$) event. In gene expression variations, a gene can be under-expressed (one dash line), normal (one solid line) or over-expressed (one bold line). For expression status, we generated two fractions, $f_4$ and $f_5$. $f_4$ is the likelihood that both genes in a gene pair are under-expressed. $f_5$ is the likelihood that a gene pair has an expression up-down event where one is over-expressed while the other one is under-expressed. All these five fractions showed a distribution difference between SL and non-SL pairs. By integrating these five fractions into a prediction model, we can identify SL interactions that can be presented as a network.

pared the fractions in SL pairs with the fractions in non-SL pairs via one-sided Wilcoxon rank tests in R. We carried out four comparisons of homozygous deletion, heterozygous deletion, mixed deletion and co-underexpression to estimate the difference of co-loss tendency between SL and non-SL pairs. In the analysis of up-down compensation, we carried out two comparisons of expression up-down or genomic up-down. Bonferroni correction was used to correct for 4 multiple comparisons in the analysis of co-loss tendency and 2 multiple comparisons in the analysis of up-down compensation (p-values are indicated with Padj).

To validate the robustness of the signals, we compared the fractions in SL pairs to the fractions in random pairs. In each randomization, we first generated 300 random pairs from all human genes for which gene expression and CNV were available and then compared the mean of the fractions in the random pairs with the mean in SL pairs. We expected that the random pairs have a smaller mean of $f_1$, $f_2$, $f_3$ or $f_4$ but a larger mean of $f_5$ than SL pairs. To test the
5.2. Materials and methods

Table 5.1: Five fractions derived from genomic variations for SL interaction identification.

<table>
<thead>
<tr>
<th>Fraction</th>
<th>Calculation</th>
<th>Conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_1$</td>
<td>$n_{\text{homCL}}/n_t$</td>
<td>$A = -2$ and $B = 2$; $n_t = \begin{cases} A = -2 \text{ and } B \in Z \ A \in Z \text{ and } B = -2 \end{cases}$ where $Z = 0$ or $-2$</td>
</tr>
<tr>
<td>$f_2$</td>
<td>$n_{\text{hetCL}}/n_t$</td>
<td>$A = -1$ and $B = -1$; $n_t = \begin{cases} A = -1 \text{ and } B \in Z \ A \in Z \text{ and } B = -1 \end{cases}$ where $Z = 0$ or $-1$</td>
</tr>
<tr>
<td>$f_3$</td>
<td>$n_{\text{mixCL}}/n_t$</td>
<td>$A = -2$ and $B = -1$; $n_t = \begin{cases} A = -2 \text{ and } B \in Z \ A \in Z \text{ and } B = -1 \end{cases}$ where $Z = 0$ or $-2$</td>
</tr>
<tr>
<td>$f_4$</td>
<td>$n_{\text{co-under}}/n_t$</td>
<td>$A = -1$ and $B = -1$; $n_t = \begin{cases} A = -1 \text{ and } B \in Z \ A \in Z \text{ and } B = -1 \end{cases}$ where $Z = 0$ or $-1$</td>
</tr>
<tr>
<td>$f_5$</td>
<td>$n_{\text{comp}}/n_t$</td>
<td>$A = -1$ and $B = 1$; $n_t = \begin{cases} A = -1 \text{ and } B \in Z \ A \in Z \text{ and } B = -1 \end{cases}$ where $Z = 1$, $0$ or $-1$</td>
</tr>
</tbody>
</table>

hypotheses, we counted the randomizations ($n_1$) where the difference of mean between the random pairs and SL pairs is contradictory to the expectation. For each comparison, we conducted 1000 randomizations and calculated the p-value for each hypothesis test as $p = (n_1+1)/1001$. 

69
5.2.3 Under-sampling

The training set is significantly skewed with only 4.6% of the pairs belonging to the positive class (SL pairs) and the rest belonging to the negative class (non-SL pairs). Such a skewed training set can affect the performance of most standard classification algorithms [147]. Thus, we generated a more balanced training set by randomly under-sampling the negative class so that the number of gene pairs in it is equal to that of the positive class. The under-sampling is conducted with ROSE package [133] in R and repeated 100 times. All the classifiers in the study are trained on the balanced set.

5.2.4 Constructing the ensemble-based prediction model

We adopted an ensemble-based model to integrate the aforementioned five signals for predicting whether a gene pair has an SL interaction or not. The balanced training set (described above) was used to train the ensemble-based prediction model that combines multiple classifiers, namely AdaBoost, J48, LogitBoost, RandomForest, Logit, JRip and PART. The combination rule is simply based on the mean function $p(x) = \frac{1}{N} \sum_{i=1}^{N} p_i(x)$ where $x$ is a given gene pair and $p_i(x)$ is the probability that $x$ is predicted to be SL by classifier $i$. The probabilities $p_i(x)$ from all classifiers, except for RandomForest, are obtained from the ‘RWeka’ package [95]. The RandomForest classifier is implemented with the ‘randomForest’ package in R [129].

To quantify the performance of the ensemble-based model, we used a 10-fold cross-validation framework on all empirically measured 270 SL pairs and 5,660 non-SL pairs. In each cross-validation, the ensemble-based model is trained on nine of the randomly constructed 10 fractions and predictions are made for the test samples in the remaining fraction. The performance of the model in each cross-validation is evaluated by a ROC curve, the corresponding AUC score and a precision-recall curve. Repeating this procedure ten times, a mean ROC curve, a mean AUC score and a mean precision-recall curve are calculated as the evaluation for the performance of the ensemble-based prediction model.

5.2.5 Constructing the genome-wide human SL interaction map

To predict SL interactions in human at a genome-wide scale, we first selected 15,620 genes that are measured for both CNV and mRNA variations in cancer cells. As mentioned in the results section, due to the presence of arm-level copy number variations, gene pairs on the same chromosome are more likely to be co-lost regardless of the status of SL interaction. Thus, we applied our model to approximately \( \approx 115 \) million gene pairs that are located on different
chromosomes. To construct a highly accurate SL interaction map, we predicted a list of more than 591,000 SL interactions based on a probability score \( p(x) \) threshold of 0.81, which achieved a precision of 67% at a recall of 10%.

5.3 Results

5.3.1 SL interactions are reflected in cancer genome evolution

We first asked whether empirically observed SL interactions are reflected in gene presence/absence and gene expression in cancer cells. To answer that, we used two types of genome variation from the Cancer Genome Atlas (TCGA) i.e., i) copy number variations (CNVs) and ii) gene expression variations. The TCGA consortium measured 14,136 tumor samples for CNVs and 7362 tumor samples for gene expression variations. To determine whether genes in cancer samples are significantly over- or under-expressed, we determined their expression-levels relative to normal samples of the same tissue type (Methods). We obtained the empirical SL interactions from two recent studies [123, 226] that measured SL interaction in colon tumor cell lines and have the highest genome coverage among all the studies available. In total we collected 270 SL pairs and 5,660 non-SL pairs (supplementary table S1).

We first tested whether SL pairs are less likely to be co-lost in a genome than non-SL pairs. A gene can either be homozygously or heterozygously deleted. We first focused on homozygous losses in which both copies of a gene are lost. We express the likelihood of homozygous co-loss of both genes in a gene pair by the fraction \( f = n_1/n_2 \), where \( n_1 \) is the number of tumor samples with a co-loss of both genes and \( n_2 \) is the number of tumor samples in which at least one gene is lost (see Methods and figure 5.1). Indeed, we found that SL pairs are less likely to be homozygously co-lost than the non-SL pairs (0.00728 vs 0.0104, one-sided Wilcoxon rank test, \( P_{adj.} \leq 0.008 \), figure 5.2a).

We performed several additional analyses to show that this result is valid and robust. First, we showed that the difference in co-loss events is not caused by the difference in single gene loss rates. Indeed the homozygous gene deletion rate of the genes in SL pairs is not different from the deletion rate of the genes in non-SL pairs (0.00402 vs 0.00406, two-sided Wilcoxon rank test, \( p \leq 0.38 \)). Secondly, given the limited genome coverage of the known SL and non-SL pairs available for our analysis, we also compared the likelihood of co-loss events of SL pairs with random pairs from the human genome. We found a significant difference in co-loss between SL pairs and random pairs (0.00728 vs 0.0128, 1000 randomizations, \( P_{adj.} \leq 0.012 \), figure 5.2a). This shows that the difference in the likelihood of co-loss events between the SL pairs and the random gene pairs is
Figure 5.2: SL pairs are reflected in copy number variations. SL pairs are less likely to have a) homozygous co-loss events, b) heterozygous co-loss events and c) mixed co-loss events than non-SL pairs or random pairs. The fractions for these three types of co-loss events are described as $f_1$, $f_2$, $f_3$ in Methods and figure 5.1. Each dot is the fraction for a given pair and the horizontal bar represents the mean of the fractions. P-values for the comparison between SL and non-SL pairs were calculated using one-sided Wilcoxon rank test. P-values for the comparison between SL and random pairs were calculated from 1000 randomizations. P-values were adjusted for multiple comparisons using the Bonferroni correction (see details in Methods).
5.3. Results

a consistent signal across the human genome. The difference between SL pairs and random pairs is larger than the difference between SL pairs and non-SL pairs (figure 5.2a). This is likely due to the fact that the genes included in the experiments tend to be biased towards those that are frequently lost, i.e. the homozygous deletion rate of genes in SL/non-SL pairs is higher than that in random pairs (0.0049 vs 0.0042, one-sided Wilcoxon rank test, $p \leq 0.04$). It should furthermore be noted that we require the gene pairs included in the analysis to be composed of genes on different chromosomes. The reason for this is that the presence of arm-level copy number variations will always cause a high probability of co-loss for the gene pairs on the same chromosome regardless of whether they have an SL interaction or not.

Besides a homozygous co-loss, with both genes homozygously deleted, there is also the possibility of a heterozygous co-loss, where both genes are heterozygously deleted and a mixed co-loss where one gene is homozygously deleted and the other is heterozygously deleted. For the heterozygous co-loss and for the mixed co-loss event we carried out the same analysis as done above for the homozygous co-losses. For both types of co-loss events, we found a significant and robust signal, i.e., the SL pairs are less likely to be co-lost than the non-SL pairs (for heterozygous co-loss 0.1935 vs 0.216, one-sided Wilcoxon rank test, $P_{adj.} \leq 1.08e-08$, figure 5.2b; for mixed co-loss 0.189 vs 0.2008, one-sided Wilcoxon rank test, $P_{adj.} \leq 0.02$, figure 5.2c). As was the case for the homozygous co-losses, both signals are consistent when SL pairs are compared with random gene pairs (for heterozygous co-loss 0.1925 vs 0.218, $P_{adj.} \leq 0.004$, figure 5.2b; for mixed co-loss 0.189 vs 0.210, $P_{adj.} \leq 0.032$, figure 5.2c).

We next examined gene expression levels, where we expected to find a similar signal to the one we found at the level of gene absence/presence, since the under-expression of one gene can also result in the loss of its activity. Indeed, we found that SL pairs are less likely to be both under-expressed than non-SL pairs (0.0443 vs 0.0586, one-sided Wilcoxon rank test, $P_{adj.} \leq 2.39e-10$, Fig 3A). Only pairs composed of genes on different chromosomes are included in the analysis. Again the signal is consistent when SL pairs are compared with random gene pairs (0.0443 vs 0.0570, $P_{adj.} \leq 0.004$, figure 5.3a).

Previous studies [44, 109] have shown another pattern in genes in SL pairs at the transcription level. In this pattern one gene of an SL interacting pair is over-expressed while its partner is under-expressed. Thus, we expected that compared with non-SL pairs, SL pairs would have higher probabilities to have an expression pattern where one gene is over-expressed while the other is under-expressed. We refer to this as expression up-down. The probability of this expression pattern is quantified by the fraction $f = n1/n2$, where $n1$ is the number of tumor samples that have the pattern and the $n2$ is the number of tumor
Figure 5.3: SL pairs are reflected in gene expression variations. a) SL pairs are less likely to be co-underexpressed relative to the control i.e., non-SL or random pairs. The fraction for co-underexpression events is described as $f_4$ in methods and figure 5.1. b) SL pairs are more likely to have expression up-down events where one gene is over-expressed while the other is under-expressed. The fraction for such pattern is described as $f_5$ in Methods and figure 5.1. Each dot is the fraction for a given pair and the horizontal bar represents the mean of the fractions. P-values for the comparison between SL and non-SL pairs were calculated with a one-sided Wilcoxon rank test. P-values for the comparison between SL and random pairs were calculated from 1000 randomizations. P-values were adjusted for multiple comparisons using the Bonferroni correction (for details see Methods).
samples that have an under-expression of at least one of the genes (see Methods and figure 5.1 for details). As expected, we found that SL pairs are more likely to have this expression pattern than non-SL pairs (0.250 vs 0.211, one-sided Wilcoxon rank test, $P_{adj.} \leq 2.10e^{-04}$, figure 5.3b). Again, we validated the consistency of the signal by comparing the likelihood of this expression pattern in the SL pairs against its likelihood in random pairs (0.250 vs 0.146, 1000 randomizations, $P_{adj.} \leq 0.002$, figure 5.3b). We note that the difference between SL pairs and random pairs is higher than that between SL pairs and non-SL pairs. This is possibly due to the fact that the genes included in the experiments were biased towards those that are more likely to be over-expressed when one is mutated, i.e., the over-expression of genes in non-SL pairs is higher than that of random genes (0.0957 vs 0.0789, one-sided Wilcoxon rank test, $p \leq 1.08e^{-06}$).

We also analyzed a genomic pattern at the gene presence/absence level by calculating the probability for each gene pair to have a CNV pattern where one gene is duplicated or amplified while the other one is homozygously or heterozygously deleted, referred to as genomic up-down in the remainder of the text. We found that SL pairs indeed have a higher probability to have the genomic up-down combination at the DNA level than non-SL pairs (0.300 vs 0.274, one-sided Wilcoxon rank test, $P_{adj.} \leq 1.65e^{-07}$), but this is not significant when we compared the SL pairs to random gene pairs.

In total, we found five patterns in the CNVs and gene expression variations in cancer cells, all of which showed that synthetic lethal interactions are reflected in cancer genome evolution. These five patterns fall into two categories: i) genes in SL pairs are more likely to be over-expressed when their interaction partner is under-expressed and ii) genes in SL pairs are less likely to be co-lost either at the DNA level or at the gene expression level.

### 5.3.2 An ensemble-based model for predicting SL interactions

We next asked whether these five genomic patterns are strong enough to reliably predict SL pairs in human on a genome-wide scale. To do that we developed an ensemble-based model that integrates the five patterns. It should be noted that we did not include the genomic up-down pattern found in CNVs since SL pairs are not significantly different from random pairs. An ensemble-based model is a classifier that combines the prediction results from multiple classifiers, such as decision trees and logistic regression. It is known that such an ensemble-based model can improve performance relative to a single classification procedure [190], especially for complex problems such as SL prediction involving noisy inputs [48].

We used the empirically measured 270 SL pairs and 5,660 non-SL pairs as
described in the previous analysis. To construct the prediction model, we first needed to handle the imbalance of sample size between the negative class, i.e. non-SL pairs, and the positive class, i.e. SL pairs. The skewed distribution of the classes can affect the performance of prediction models [147]. To solve this issue, we randomly under-sampled the negative class (non-SL pairs, 95.4% of the training set) to produce a set of negative samples of the same size as the positive class (SL pairs, 4.6% of the training set). This balanced combination of two sets is used to train an ensemble-based model for SL prediction. Note that the under-sampling is only applied to the training set. In total we selected seven different single classifiers as the base for the ensemble model: AdaBoost [67], J48 [179], LogitBoost [69], RandomForest [23], Logit [124], JRip [36] and PART [66], which are either robust against noisy data or over-fitting. After being trained with the balanced set, each single classifier generates a probability that a gene pair has an SL interaction. Then we integrated all seven probabilities from these single classifiers by calculating the mean of the seven probabilities and used that as the final predicted probability.

To assess the performance of the ensemble-based prediction model, we used a 10-fold cross-validation on all the empirically measured 270 SL pairs and 5,660 non-SL pairs. The plot of sensitivity (i.e., true positive rate) versus false positive rate of the ensemble-based model shows that our model achieves an area under ROC curve (AUC) of 0.75 (standard error = 0.016, figure 5.4b). It should be noted that this high AUC is only achieved when combining all patterns (Fig 4A). We also found that the ensemble-based model achieved the highest AUC compared to all seven single classifiers (figure 5.4b). In order to predict a genome-wide SL interaction map, we estimated the average precision and recall values from the 10-fold cross-validation (figure 5.4c). We then applied the model to all gene pairs on the genome. Among ≈115 million pairs for which gene expression and CNV data were available, we predicted more than 591,000 SL interactions based on a probability score threshold of 0.81 (figure 5.4c), which corresponds to an estimated precision of 67% based on our training set, i.e., 14-fold higher than expected from chance (supplementary dataset S1). Note that the model achieves a similar precision (60% at p = 0.81) when using an independent set of experimentally measured SLs (supplementary figure S1C).

5.4 Discussion

In this study we present a novel computational model that identifies SL interactions from cancer genomic data on a genome-wide scale. To develop such a model, we first systematically explored how SL interactions are reflected in cancer genomes and their gene expression levels. We found that compared
Figure 5.4: Receiver operating characteristic (ROC) curves. 

\(a\) The ensemble-based prediction model based on all five combined patterns has an area under curve (AUC) of 0.75 (blue line), which is estimated by 10-fold cross validation. Ensemble-based prediction models based on the non-combined individual patterns, i.e., co-loss in CNVs, co-underexpression and expression up-down, are shown in red, green and purple respectively and have lower AUCs. Standard error bars are added to each ROC. 

\(b\) The ensemble-based prediction model (the blue ROC curve) has a better performance than all the seven single. 

\(c\) The precision and recall curve is estimated from 10-fold cross validation. Standard error bars are added. The curve is colored according to the cutoff of probability. The color panel of the probability is plotted at the right side. The cutoffs of probability scores (\(p(x)\)), 0.81, are printed at the corresponding curve positions. The grey line represents the prediction precision by chance alone.
with non-SL pairs, genes in SL pairs are significantly less likely to be co-lost in a cancer genome, both at the level of gene expression and at the level of copy number variation. Moreover, SL pairs are more likely to have an expression up-down pattern where one gene is over-expressed while the other one is under-expressed, which is consistent with another recent study [105]. Based on these results, we constructed an ensemble-based model to predict SL interactions via integrating these unique patterns in cancer genome variations, achieving a high prediction performance (AUC = 0.75). Our work presents a direct way to predict SL interactions from cancer genomic data, in contrast to most existing computational models, which identify SL interactions either specific for the model organisms yeast and C. elegans [243, 235, 168], or predict SL pairs in human in an indirect way by mapping SL interactions from yeast to human via orthology [45]. A strategy that uses human genomes by exploring the ‘compensation’ pattern also requires, as an additional criterion, that the genes are generally co-expressed [105]. As SLs have the characteristic that only one of the two genes is strictly needed, co-expression is not crucial. As such, co-expression as an additional criterion limits the coverage of SL interactions encoded in the genome, which is reflected in the total number of predicted SL pairs by Jerby-Arnon L. et al. (2,816 with accuracy of 0.779) [105]. In contrast, our approach, which does not rely on co-expression, predicts many more SL interactions with a comparable accuracy (591,000 with an estimated accuracy of 0.75).

Future work should focus on the following issues to improve the performance of the model. First, given the genomic and micro-environment heterogeneity among different types of tumors [28, 106], the empirically detected SL interactions included in our analysis might be only specific to colon cancer in which the experiments were carried out. As genetic interactions were found to be growth condition specific [84], it might be that two genes are co-lost in certain tumors simply because the functions of these SL pairs are not essential for that particular cancer type. Such discordance of tissue types might have dampened the effect size we discovered. To improve this, one can focus on gene-expression and CNVs that are taken from the same tumor type as the empirical SLs. A model can then be constructed to predict tumor type specific SLs, which is valuable to overcome the challenges posed by inter-tumor heterogeneity in cancer treatments. Secondly, our model only considered gain or loss of gene function caused by CNVs and variations in gene expression. There are other mutations that can result in gain or loss of gene functions, such as mutations of miRNA [130, 169] and epigenetic mutations [98, 91]. When knowledge becomes available on how these other types of genomic variations affect gene function and genetic interactions, these mutations could also be taken into account. Thirdly, our model achieves a good prediction performance by a 10-fold
cross validation. However, we note that the model is trained on a relatively small number of available SL and nonSL pairs, which constrains a precise estimation of the model performance for genome-wide prediction. The performance can be better estimated in the future when more empirically measured SLs become available. Finally, it still remains to be seen to what extent these predicted SL interactions from cancer genomes are relevant to understand other diseases. For diseases where CNV or gene expression data are available, one can prioritize disease-associated SL interactions from our prediction list by selecting pairs that are co-lost in the disease.

Taken together, we systematically investigated and showed that SL interactions are reflected in genome evolution of cancer in various forms. Based on the unique patterns discovered in cancer genomics, we proposed a simple approach to identify SL, which strongly improves existing frameworks. We generated a unique SL interaction network in human at the genome-scale covering up to 591,000 pairs with a high estimated precision. In the light of medical genetics, this list is highly valuable in the search for anti-cancer drug targets and in understanding human diseases.
Chapte

6

Weighted co-expression predicts new genes for molecular systems

We have developed the Weighted Gene Expression Tool and database (WeGET) for the prediction of new genes of a molecular system by correlated gene expression. WeGET utilizes a compendium of 465 human and 560 murine gene expression datasets that have been collected from multiple tissues under a wide range of experimental conditions. It exploits this abundance of expression data by assigning a high weight to datasets in which the known genes of a molecular system are harmoniously up- and downregulated. WeGET ranks new candidate genes by calculating their weighted co-expression with that system. A weighted rank is calculated for human genes and their mouse orthologs. Then, an integrated gene rank and p-value is computed using a rank-order statistic. We applied our method to predict novel genes that have a high degree of co-expression with Gene Ontology terms and pathways from KEGG and Reactome. For each query set we provide a list of predicted novel genes, computed weights for transcription datasets used and cell and tissue types that contributed to the final predictions. The performance for each query set is assessed by 10-fold cross-validation. Finally, users can use the WeGET to predict novel genes that co-express with a custom query set.

6.1 Introduction

Ever since the publication of the first gene expression arrays, the correlated expression of genes involved in a related molecular process has been used to predict functional relations between gene pairs [35]. Large amounts of microarray data...
Chapter 6. Weighted co-expression predicts new genes for molecular systems

and RNA-seq transcript expression, measured under a plethora of conditions enable mining for concordantly expressed genes. Indeed, this concept has been successfully employed in databases such as COEXPRESSdb, GeneFriends, GeneMANIA and STARNET 2 [107, 159, 41, 245]. Nevertheless, relative to other types of genomics data, co-expression has lower sensitivity and selectivity [157]. To improve the quality of the predictions, various strategies have been applied, like exploiting the conservation of co-expression between species [227], combining many gene expression datasets [125, 244] or biclustering datasets to identify groups of genes that co-express within a subset of the experiments (see [137] for a review). Expression screening, an extension of biclustering methods [10], weights gene expression datasets based on the co-expression of genes within a molecular system and uses those weights to predict new genes involved in that system. It has been successfully applied to predict new mitochondrial proteins essential for the organelle [10] and to discover new players in heme biosynthesis [156]. The principle behind this method is appealing: it systematically exploits the available gene expression data and, via its weighting scheme, implicitly solves the question facing many researchers: which gene expression data to use to predict new genes for a pathway? Nevertheless, it is computationally costly, as the weighting has to be recalculated for each pathway separately and additional cross validation requires multiple runs per pathway. We have therefore developed and implemented a fast expression screening algorithm that includes a dataset weighting and allows for the rapid computation of genes that co-regulate with a query gene set. Our algorithm was employed to compile a weighted co-expression database for all Gene Ontology (GO) terms and human pathways annotated in the KEGG and Reactome databases. Furthermore, we provide information regarding the original experimental setup of the highly weighted datasets. In particular, WeGet reports the cell and tissue types in which the query genes are consistently up- and downregulated with each other. Finally, the robustness of the predicted results is assessed by 10-fold cross-validation and reported as the receiver operating characteristic (ROC) curve.

We compared WeGET with five popular web tools and databases that predict novel genes based on their co-expression with specified query gene sets, using two query gene sets published by Baughman et al. [10] and show that indeed, weighting the datasets results in improved precision, in particular at low recall rates (the top 100 genes). The complete WeGET database, together with a custom query submission system, is available through the WeGET website.
6.2 The WeGET analysis pipeline

WeGET uses a compendium of 465 human and 560 murine gene expression datasets ranging from 6 to 192 samples per dataset. In total, approximately 30,000 samples from multiple mammalian platforms were collected from the Gene Expression Omnibus (GEO) [9].

Figure 6.1: The WeGET computational pipeline used to create the database. a) Determining the dataset weight $W_{\text{dataset}}$. The transcriptome measurements are converted into a correlation matrix. The average correlation with the query set ($S_{\text{gene}}$) is used for gene ranking and the dataset weight calculation $W_{\text{dataset}}$. b) Data integration across datasets, platforms and species. Gene scores $S_{\text{gene}}$ from all datasets are combined taking into account the precomputed weights. Subsequently different transcriptome platforms and species data are integrated to arrive at the final ranking. The process is repeated after excluding each query gene to construct a receiver operating characteristic (ROC) curve that visualizes predictive power of the method for a specific query set of genes.

The WeGET computational pipeline starts with selecting the normalized expression values for all probes associated with the query genes. For genes with multiple probes, the probe with the highest average Pearson correlation coefficient with all other query probes is selected. Subsequently, the pipeline calcu-
lates the average Pearson correlation between each gene and the set of query genes in every dataset (figure 6.1). Then, all probes are ranked based on their average correlation with the query probes and mapped back to their associated gene. Each gene \( i \) obtains a score \( s_i \) depending on the fraction of the query set that has been ranked above that gene. These calculations are then repeated four times for the same query set and gene expression dataset, where the expression values have been randomly permuted between the genes in every measurement. This step estimates the number of genes that are expected to highly correlate with the query set in a random model. To calculate the dataset weight, an N100 value is calculated that is the fraction of query genes found among the top 100 genes with highest average Pearson correlation. The ratio between the N100 from the original dataset and the average N100 value for the randomized datasets constitutes the weight of the experiments. A species score is the weighted average of all its datasets. The final ranked gene list is obtained by integrating the ranked human and mouse list (mouse genes that are unambiguous human one-to-one orthologs). This is performed using the ‘RobustRankAggreg’ R-package [118] that computes the final gene rankings using a rank-order statistic [1, 209].

Thus for each set of expression data the pipeline measures whether genes in a given pathway co-express better than expected and uses that to assign weights to that expression dataset. These weights are subsequently used in determining the (weighted) co-expression of all genes with that pathway. The source of the variation in the weights between the datasets can be technical, e.g. variation in the probes that have been used, or biological, e.g. variation in the tissues in which gene expression has been measured. The important assumption behind the method is that new genes for a pathway are significantly co-expressed with the majority of the genes of a pathway that they belong to, rather than only with some of its members. This, in turn, depends on the pathway definition. To aid in finding the genes from a pathway that co-express with each other, the results include a visualization of co-expression between query genes displayed as a network. This allows the user to select a subset of co-expressed genes from that pathway to repeat the procedure.

### 6.3 WeGET validation and comparison to other co-expression databases

To assess and compare the predictive power of different co-expression methods (supplementary table S1), we used two query gene sets [10]: 19 query genes in the cholesterol biosynthesis pathway and 76 genes involved in oxidative phos-
phorylation (OXPHOS). We manually performed leave-one-out or 10-fold cross validation by multiple submissions (see supplementary methods for details). We took into account the top 100 ranked genes as a likely use case scenario. Figure 6.2 shows the WeGET results for the cholesterol biosynthesis pathway and OXPHOS system compared to other online tools employing the co-expression analysis. Baughman et al. [10] carried out one-time computations for cholesterol and OXPHOS datasets. WeGet webserver achieves identical (cholesterol) or marginally better performance (OXPHOS, 86.4% sensitivity at 99.8% specificity, compared to 85% and 99.4%, respectively, supplementary figure S1).

![Figure 6.2: ROC performance curves for online co-expression tools (see supplementary table S1). Performance measured by multiple cross-validation runs is indicated by the area under the curve (AUC) for the top 100 genes corresponding to a typical use case scenario. a) Results for 19 genes in the cholesterol pathway using leave-one-out cross-validation. b) Results for 10-fold cross-validation in the oxidative phosphorylation (OXPHOS) query set.](image)

### 6.4 The WeGET database and web access

Figure 6.3 depicts the architecture of the WeGET database. Human pathways and their associated genes from GO and KEGG are stored in a central database. The WeGET parallel algorithm that calculates each dataset on a separate thread precomputes the co-expressed genes and dataset weights for all pathways using the transcriptome compendium. The results are presented to the user using the WeGET webtool (implemented in Python Flask) and can additionally be downloaded.

On the WeGET website, pathways are shown in a data grid (figure 6.4), which can be sorted and searched. Detailed information (figure 6.5), such as the best scoring genes, dataset weights, cell and tissue types in which the genes highly co-express (see also supplementary figures S2 and S3) and cross-
Figure 6.3: The WeGET system architecture. Results for predefined pathways (GO, KEGG and Reactome) are precomputed and exposed through the WeGET webtool. Custom defined gene sets can be analyzed by submitting the gene ids or gene symbols to the webserver.

validation results are shown when a row entry is selected. A pathway can be accessed directly as http://weget.cmbi.umcn.nl/pathwaydb/identifier where “pathwaydb” denotes the pathway database (one of: GO, KEGG or Reactome) and identifier the category identification (e.g. http://weget.cmbi.umcn.nl/GO/GO: 0000398). User queries (different than the predefined sets) can be entered using the “Custom pathway” tab, specifying genes as Entrez ID or HUGO gene symbol. The query is then scheduled for analysis. After the analysis, the user receives an email with results, including the cross-validation and a network that displays the co-regulation of the query genes within the datasets (see below) in a spreadsheet. The website provides an opportunity to learn more about the experimental conditions in which the concordant expression of the query molecular system has been observed. The tab “Dataset Weights”, accessible for each precomputed query set lists GEO datasets with a concordant expression patterns of the query system, indicating congruent coexpression of the gene components of the molecular system. The dataset identifier is directly hyperlinked with the GEO entry description (both online and in Excel output file) such that users can read details of the experiment that lead to harmonious expression of the query set.
6.5 Evaluation of WeGET results for a query gene set

The robustness of the results is tested by $k$-fold cross-validation and graphically displayed with a ROC curve. The curve illustrates the performance of the WeGET method, by plotting the true positive rate (successfully cross-validated query genes) versus all human genes (figure 6.2). The curve is plotted for every molecular system stored in the database (GO, KEGG and Reactome pathways) separately. The area under the curve (AUC) is a measure of the prediction quality and robustness for that pathway. The average AUC computed for all pathways is around 0.7. Well-studied and clearly defined cellular components such as mitochondria and biological processes such as cilium movement and assembly have a higher AUC (average 0.83 and 0.84, respectively) reflecting their concomitant expression patterns. For pathways with less than 50 genes we use leave-one-out cross-validation, for larger pathways 10-fold cross-validation is carried out. Finally, the cohesion of the query gene set is displayed as a network using a node-force algorithm (figure 6.1). Query genes that consistently co-express perform a large attractive force and therefore cluster together. In contrast, genes that show little evidence of co-regulation exhibit a smaller force and do not cluster with the other query genes. Using this visualization, the user can resubmit the query gene set to omit genes that do not show evidence of co-regulation.
6.6 Using WeGET to predict genes involved in neuropathic pain

Previous studies indicate that mutations in genes coding for voltage-gated sodium channels and related processes may impair the nociceptive pathway and influence response to pain stimuli [87]. From the literature we collected genes implicated in neuropathic pain (Table 1) and used the WeGET database to predict novel candidate genes for this pathway. Table 2 shows genes co-expressing with the neuropathic pain molecular system as calculated by WeGET. Next to sodium channels and its regulators (PRMT8, UNC80 rank 41 and 74, respectively) also genes of the voltage-gated potassium system are strongly represented among top coexpressing genes (MAP1A, PPP2R2C, KCNH3, KCNQ2 rank 6, 7, 66 and 87, respectively) consistent with their involvement in nociceptive processing [219], their expression in dorsal root ganglion neurons analogous to voltage-gated sodium channels [173] and with recently discovered genetic variants that modulate neuropathic pain [38]. The PIEZO2 gene, a nociceptive component mechanically activated in nerve endings [43] ranks 74th among all genes. Additional poorly characterized genes such as SERP2, TMEM130 and CCDC155 (ranks 5, 9 and 20, respectively) are also present among genes co-expressing with the system and constitute novel candidate genes for nociceptive pathway. Figure 6.6a shows the higher performance of WeGET integration of all datasets (cross-validated AUC=0.82) compared to integration of all datasets.
with equal weights (average coexpression across all experiments, AUC=0.71) and a high weight individual dataset GDS1634 of dorsal root ganglia neurons (AUC=0.68). Weights assigned to GEO datasets reveal a high contribution of transcriptome measurements related to neurons: a murine nodose and dorsal root ganglia study (GDS1634, weight 3.0) (22), gene expression in human neurofibrillary tangles (GDS2795, weight 2.5) and DNA methylation effect on neural stem cells (GDS538, weight 3.0). The peripheral roles of DPYSL2 (trafficking subset, table 6.1), MSN and NEDD4L proteins (peripheral subclass) are visualized in the query gene network (figure 6.6b). Currently we screen patients with a familial form of neuropathic pain for genetic variants that may impact the function of the candidate genes.

<table>
<thead>
<tr>
<th>No.</th>
<th>Gene Symbol</th>
<th>Sub-system</th>
<th>No.</th>
<th>Gene Symbol</th>
<th>Sub-system</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ACTG1</td>
<td>CORE</td>
<td>11</td>
<td>DPYSL2</td>
<td>TAG</td>
</tr>
<tr>
<td>2</td>
<td>ANK3</td>
<td>CORE</td>
<td>12</td>
<td>KCNK3</td>
<td>TAG</td>
</tr>
<tr>
<td>3</td>
<td>SCN10A</td>
<td>CORE</td>
<td>13</td>
<td>NRCAM</td>
<td>TAG</td>
</tr>
<tr>
<td>4</td>
<td>SCN11A</td>
<td>CORE</td>
<td>14</td>
<td>ANXA2</td>
<td>TAG</td>
</tr>
<tr>
<td>5</td>
<td>SCN1B</td>
<td>CORE</td>
<td>15</td>
<td>PRKACA</td>
<td>TAG</td>
</tr>
<tr>
<td>6</td>
<td>SCN2B</td>
<td>CORE</td>
<td>16</td>
<td>PRKCB</td>
<td>TAG</td>
</tr>
<tr>
<td>7</td>
<td>SCN3A</td>
<td>CORE</td>
<td>17</td>
<td>SYN2</td>
<td>TAG</td>
</tr>
<tr>
<td>8</td>
<td>SCN3B</td>
<td>CORE</td>
<td>18</td>
<td>TNR</td>
<td>TAG</td>
</tr>
<tr>
<td>9</td>
<td>SCN8A</td>
<td>CORE</td>
<td>19</td>
<td>MSN</td>
<td>PI</td>
</tr>
<tr>
<td>10</td>
<td>SPTBN4</td>
<td>CORE</td>
<td>20</td>
<td>NEDD4L</td>
<td>PI</td>
</tr>
</tbody>
</table>

Table 6.1: Genes implicated in neuropathic pain collected from the literature. Core molecular sub-system associated with voltage-gated sodium channels (CORE), trafficking-associated genes (TAG) and the peripheral involvement (PI) classes are indicated [175, 53, 220].
Figure 6.6: Visualization of performance of the WeGET results for neuropathic pain genes. **a)** ROC curves for the neuropathic pain query set (table 6.1). The $x$-axis represents the fraction of human genes, the $y$-axis the fraction of the neuropathic pain molecular system. Shown are ROC curves for final results (blue), the cross-validation (CV) of integrated datasets (green), the average co-expression across all datasets (integration with equal contribution of each dataset) with CV (red) and results of co-expression within a single high-weight dataset (GDS1634, a nodose and dorsal root ganglia comparison (cyan)). **b)** Network visualization of the co-expression allows identification of genes less co-expressed with the core of the query gene set.
6.6. Using WeGET to predict genes involved in neuropathic pain

<table>
<thead>
<tr>
<th>Rank</th>
<th>P-value</th>
<th>Entrez ID</th>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>0.00000</td>
<td>6854</td>
<td>SYN2</td>
<td>synapsin II</td>
</tr>
<tr>
<td>02</td>
<td>0.00005</td>
<td>6327</td>
<td>SCN2B</td>
<td>sodium channel, voltage-gated, type II, ...</td>
</tr>
<tr>
<td>03</td>
<td>0.00010</td>
<td>23705</td>
<td>CADM1</td>
<td>cell adhesion molecule 1</td>
</tr>
<tr>
<td>04</td>
<td>0.00018</td>
<td>60</td>
<td>ACTB</td>
<td>actin, beta</td>
</tr>
<tr>
<td>05</td>
<td>0.00018</td>
<td>730102</td>
<td>LOC730102</td>
<td>quinone oxidoreductase-like protein 2, ...</td>
</tr>
<tr>
<td>06</td>
<td>0.00020</td>
<td>7532</td>
<td>YWHAG</td>
<td>tyrosine 3-monooxygenase</td>
</tr>
<tr>
<td>07</td>
<td>0.00025</td>
<td>283209</td>
<td>PGM2L1</td>
<td>phosphoglucotase-2-like 1</td>
</tr>
<tr>
<td>08</td>
<td>0.00030</td>
<td>64207</td>
<td>IRF2BP1</td>
<td>interferon regulatory factor 2 binding, ...</td>
</tr>
<tr>
<td>09</td>
<td>0.00037</td>
<td>71</td>
<td>ACTG1</td>
<td>actin gamma 1</td>
</tr>
<tr>
<td>10</td>
<td>0.00040</td>
<td>288</td>
<td>ANK3</td>
<td>ankyrin 3, node of Ranvier (ankyrin G)</td>
</tr>
<tr>
<td>12</td>
<td>0.00049</td>
<td>8455</td>
<td>ATRN</td>
<td>attractin</td>
</tr>
<tr>
<td>12</td>
<td>0.00050</td>
<td>5579</td>
<td>PRKCB</td>
<td>protein kinase C, beta</td>
</tr>
<tr>
<td>13</td>
<td>0.00054</td>
<td>3785</td>
<td>KCNQ2</td>
<td>voltage gated KQT-like subfamily Q</td>
</tr>
<tr>
<td>14</td>
<td>0.00054</td>
<td>3745</td>
<td>KCNB1</td>
<td>voltage gated Shab related subfamily B</td>
</tr>
<tr>
<td>15</td>
<td>0.00055</td>
<td>6191</td>
<td>RPS4X</td>
<td>ribosomal protein S4, X-linked</td>
</tr>
<tr>
<td>16</td>
<td>0.00064</td>
<td>84502</td>
<td>JPH4</td>
<td>juncophilin 4</td>
</tr>
<tr>
<td>17</td>
<td>0.00064</td>
<td>6907</td>
<td>TBL1X</td>
<td>transducin (beta)-like 1X-linked</td>
</tr>
<tr>
<td>18</td>
<td>0.00066</td>
<td>2917</td>
<td>GRM7</td>
<td>glutamatergic receptor, metabotropic 7</td>
</tr>
<tr>
<td>19</td>
<td>0.00076</td>
<td>221336</td>
<td>BEND6</td>
<td>BEN domain containing 6</td>
</tr>
<tr>
<td>20</td>
<td>0.00081</td>
<td>26508</td>
<td>HEYL</td>
<td>hes-related family bHLHrop transcription factor</td>
</tr>
<tr>
<td>21</td>
<td>0.00083</td>
<td>6235</td>
<td>RPS29</td>
<td>ribosomal protein S29</td>
</tr>
<tr>
<td>22</td>
<td>0.00088</td>
<td>2597</td>
<td>GAPDH</td>
<td>glyceraldehyde-3-phosphate dehydrogenase</td>
</tr>
<tr>
<td>23</td>
<td>0.00097</td>
<td>84677</td>
<td>DSCR8</td>
<td>Down syndrome critical region 8</td>
</tr>
<tr>
<td>24</td>
<td>0.00101</td>
<td>8825</td>
<td>LIN7A</td>
<td>lin-7 homolog A (C. elegans)</td>
</tr>
<tr>
<td>25</td>
<td>0.00101</td>
<td>6134</td>
<td>RPL10</td>
<td>ribosomal protein L10</td>
</tr>
<tr>
<td>26</td>
<td>0.00108</td>
<td>4155</td>
<td>MBP</td>
<td>myelin basic protein</td>
</tr>
<tr>
<td>27</td>
<td>0.00111</td>
<td>11280</td>
<td>SCN11A</td>
<td>sodium channel, voltage gated ...</td>
</tr>
<tr>
<td>28</td>
<td>0.00114</td>
<td>3778</td>
<td>KCNA1</td>
<td>calcium activated large conductance subfamily</td>
</tr>
<tr>
<td>29</td>
<td>0.00115</td>
<td>399947</td>
<td>C11orf87</td>
<td>chromosome 11 open reading frame 87</td>
</tr>
<tr>
<td>30</td>
<td>0.00115</td>
<td>6171</td>
<td>RPL41</td>
<td>ribosomal protein L41</td>
</tr>
<tr>
<td>31</td>
<td>0.00117</td>
<td>63895</td>
<td>PIEZO2</td>
<td>piezo-type mechanosensitive ion channel ...</td>
</tr>
<tr>
<td>32</td>
<td>0.00122</td>
<td>222865</td>
<td>TMEM130</td>
<td>transmembrane protein 130</td>
</tr>
<tr>
<td>33</td>
<td>0.00125</td>
<td>2828</td>
<td>GPR4</td>
<td>G protein-coupled receptor 4</td>
</tr>
<tr>
<td>34</td>
<td>0.00126</td>
<td>23201</td>
<td>FAM168A</td>
<td>family with sequence similarity 168 ...</td>
</tr>
<tr>
<td>35</td>
<td>0.00129</td>
<td>57181</td>
<td>SLC39A10</td>
<td>solute carrier family 39 (zinc transp.) ...</td>
</tr>
<tr>
<td>36</td>
<td>0.00129</td>
<td>1816</td>
<td>DRD5</td>
<td>dopamine receptor D5</td>
</tr>
<tr>
<td>37</td>
<td>0.00132</td>
<td>9378</td>
<td>NRXN1</td>
<td>neurexin 1</td>
</tr>
<tr>
<td>38</td>
<td>0.00135</td>
<td>4130</td>
<td>MAP1A</td>
<td>microtubule associated protein 1A</td>
</tr>
<tr>
<td>39</td>
<td>0.00138</td>
<td>7314</td>
<td>UBB</td>
<td>ubiquitin B</td>
</tr>
<tr>
<td>40</td>
<td>0.00148</td>
<td>302</td>
<td>ANXA2</td>
<td>annexin A2</td>
</tr>
</tbody>
</table>

Table 6.2: Results from custom molecular system as received by the user. Top 40 genes prioritized for their involvement in neuropathic pain are shown. Genes that were part of the query set are shown in bold font face.
In this thesis, three main themes were considered. First, *constraint-based modeling of metabolic networks* was discussed in chapter two and three, with focus on the space of feasible flux paths. In chapter 4, a new algorithm was introduced to infer SDL interactions from a genome-scale metabolic network. This chapter bridges metabolic modeling and another important topic of this thesis; *inference of synthetic genetic interactions pairs*. Whereas the SDL in chapter 4 denotes a lethal interaction between an over- and underexpressed (or lost) gene, the fifth chapter considers an interaction involving two lost genes. In this case, we do not use a metabolic model, but an ensemble based classifier that considers synthetic lethal (SL) interactions spanning the whole human genome. In both chapters, the possibility to exploit S(D)Ls to target cancer cells was explored. Finally, in chapter 6 a new webtool was introduced to identify novel genes for a molecular function or pathway given a set of query genes representative for that system. There, we used gene co-expression computed in a large set of micro-arrays; an idea that has been exploited in many research papers before. The more interesting idea in this chapter is how to *automatically select relevant data sets and features from the vast amount of (public) high-throughput data* given a small set of representative members of that cellular system. For each of these three themes, the main results of this thesis and future perspectives are discussed.

### 7.1 Constraint-based modeling of metabolic networks

#### 7.1.1 Sampling the solution space

In constraint-based modeling of metabolism, two main types of constraints are imposed. First, a steady-state assumption dictates that there is no built-up
of metabolic mass in the network. Second, the direction and magnitude of metabolic flux are constrained by limitations on flux capacity and directionality. All flux distributions that satisfy these constraints form the convex space of steady-state flux distributions. Sampling flux distributions (i.e. \( n \)-dimensional points) from this space in an unbiased manner (uniformly distributed over the whole space) is difficult for two main reasons.

First, metabolic networks are large and heavily underdetermined. For example, the latest reconstruction of human metabolism [218] has \( m = 5,063 \) metabolites, \( n = 7,440 \) reactions and a nullspace with \( k = 2,774 \) dimensions. Therefore, this system of linear equations has 2,774 degrees of freedom and thus a solution space with equal dimensions. In order to provide a good estimate of the feasible flux distributions, a large number of samples need to be drawn from this huge space in an unbiased manner. The second difficulty is caused by the large variability in constrained flux ranges. By consensus, unknown reaction rates are typically constrained to \( \pm 1000 \) mmol/gDW/h. However, the range between the lower- and upper bound of measured reactions is typically a small fraction of that. After propagating all constraints in the network, one often faces the situation where the ratio between the largest and smallest flux range is in the order of \( 10^5 \) or worse. As a consequence, the ratio between the longest and smallest axis of the solution space has a similar magnitude. For this reason, ordinary Hit-and-Run [15], which takes a direction uniform on the boundary of the \( k \)-dimensional sphere is inapplicable, even for small scale networks. The smallest axis of the space limits the allowable step size so much that the larger axes are in practice never fully explored, which biases the sampler to a subspace of the feasible solution space. Resorting to Artificial Centering Hit-and-Run (ACHR) [113] greatly alleviates this problem, but does not seem to be a satisfactory solution for sampling the ever increasing genome-scale metabolic networks. In the next paragraphs, I will discuss some alternatives and recent developments that may be better suited to the largest metabolic networks.

Taken together, although the new sampling tool described in chapter 2 significantly improved sampling time and ‘uniformity’, it is still not an ideal solution for deriving the ‘most likely’ flux distributions in the largest metabolic network reconstructions. Fortunately, alternative methods and new developments exist, which I will briefly discuss now.

**Alternatives for ACHR sampling**

It is clear that efficient and uniform sampling from a metabolic network is a challenging task. Hit-and-run is an example of a Markov Chain Monte Carlo (MCMC) sampler and therefore possesses attractive convergence properties, but
does not mix well in the odd-shaped solution space of metabolic networks. Artificial Centering Hit-and-Run (ACHR) [113] mixes much better, but is not a MCMC method and therefore loses these theoretical convergence guarantees.

**MCMC sampling with non-uniform direction choice** Recently, a new method was introduced that seems to combine the best of both worlds in a two phase procedure. First, the shape of the solution space is estimated and this information is used to bias the sampling direction [42] towards the elongated axes during the second phase. Because the sampling direction does not depend on previous iterates, the resulting algorithm is still an instance of a MCMC sampler. Results show that it indeed converges nicely, but does that mean that the samples are indeed uniformly distributed? Showing that samples indeed have a uniform distribution is straightforward for basic and well-known geometric shapes, but not trivial for the unknown shapes of metabolic networks. In the end, if that uniform distribution was known, we would not need these sampling techniques. How to deal with this problem?

**Belief- and expectation propagation** An alternative technique for sampling from the solution-space is (loopy) belief propagation (BP) [171, 170], which has indeed been applied to metabolic networks [22]. In a nutshell, BP starts with choosing flux values for the uptake fluxes. These values are ‘propagated’ to reactions sharing the metabolite that was taken up and for these reactions the possible flux values are computed. By using this local propagation step throughout the network, an empirical flux distribution for each of the reactions can be obtained. In particular, when the network has no cycles, this propagation algorithm is highly efficient and produces exact results. In the case of metabolic networks, where cycles often exists, the propagation steps can be repeated until the flux values converge. In most cases, the algorithm still produces reliable results in a reasonable amount of time. The fact that computations are local not only makes BP very efficient, but also makes it much easier to deal with the heterogeneous axes of the solution space. Expectation propagation (EP) [148] is a similar technique, often applied in Bayesian inference. Remarkably, BP and EP do not seem to be widely adopted by the metabolic community, possibly due to a lack of available toolboxes tailored to metabolic modeling.

**7.1.2 Reducing the feasible flux space**

Discovering the biologically relevant flux distributions from the wide range of possibilities is a central problem in constraint-based metabolic modeling. Where Flux Balance Analysis (FBA) [163] and many other optimization techniques are
rooted in maximizing the growth rate or yield in unicellular organisms, they cannot easily be applied to complex, multicellular organisms. A notable exception is the modeling of cancer metabolism, where tumor growth is an objective with strong clinical relevance that can easily be optimized in network simulations.

In the context of mitochondrial disease, we wanted to know how metabolic fluxes are routed in healthy and diseased subjects. In particular, metabolic modeling was used to prioritize flux measurements; a challenging task for a couple of reasons. First, most metabolic fluxes cannot be measured with current experimental techniques. Exchange fluxes (uptake from, or secretion to the growth medium) and reaction rates for which the metabolites can be isotopically labeled are the main exceptions. The latter is usually applied by using C\textsuperscript{13} carbon labeling, [231, 242] which enables the quantification of reaction rates in central carbon metabolism. Combined, the methods are able to uncover the fluxes in central metabolism and those exchanging metabolites through the cell boundary. Although that is an important part of the metabolic network, only a couple dozen fluxes are revealed in the ideal case. This severely limits the amount of data available for building and validating models. Second, by tracing the isotopic labeling of carbon metabolites in central metabolism, a whole pathway of fluxes is measured. In our experiments we have neglected this fact, because proper experimental design of isotopic labeling is a complex research field by itself. Rather then focusing on one experimental design, we aimed to develop a general method, focusing on reduction of the steady-state flux space. Third, there is no clear quantitative relation between flux and gene- or protein expression [4]. Reactions whose enzyme coding genes are not expressed usually do not carry flux, but the reaction rate is often unknown when the genes are expressed. Thus, gene expression can mainly be used to exclude the measurement of fluxes for non-expressed enzymes. Fourth, the size of the solution space after flux measurement not only depends on which reaction’s rate is measured, but to a high degree also on the actual outcome of the measurement. In many cases, a flux that is relatively high constrains the other fluxes more than a low flux. The exact impact depends on many things; network topology and measurement precision amongst others. This last fact we tried to exploit by estimating the most likely reaction rates.

It has been shown that bacteria that are believed to optimize for growth yield, do this at a suboptimal rate [60, 201]. It is hypothesized that the reason for this behavior is a robustness principle. Dedicating the complete metabolic machinery to optimal growth in one nutrient environment severely affects the capability to quickly adapt to other environments. A slight relaxation of the maximum growth rate on the other hand, greatly improves the adaptation capability and therefore the metabolic robustness of the organism [201]. This robustness prin-
principle is a uniform design principle of metabolic networks [100, 183]. However, robustness on the topology level of the network can still be superseded at the metabolic flux level. This seems unlikely and therefore the basic idea behind our algorithm is to discard flux distributions that severely affect the robustness of metabolic systems. We applied this principle in chapter 3 to predict intracellular flux rates, reduce the steady-state solution space and prioritize measurements.

7.1.3 Future directions

One of the main limitations in metabolic modeling is that metabolic flux measurement techniques are currently not available at the genome scale. To circumvent this problem, researchers resort to indirect evidence of metabolic activity. Traditionally, this was mainly gene expression data, but advances in proteomics and metabolomics allow the measurement and integration of increasingly larger amounts of proteins and metabolites. With ongoing developments in many of the high-throughput omics techniques, there is a rapid growth in the number and types of available molecular datasets. A natural consequence of these developments, is the desire to unravel how data from various cellular processes fit together. Therefore, there is demand for computational methods that can integrate data from various experiment types to obtain a systematic understanding of the cellular machinery. This trend is already visible in metabolic modeling, where next generation models include gene regulatory processes [31] and data on protein structure [32] and concentrations [126, 161]. It is likely that in the future other processes such as signaling, cell cycle specific modifications or epigenetic dynamics can be included. Eventually, it is likely that metabolic modeling gradually shifts to whole cell modeling.

An integrated cellular model, with data from different, but related molecular processes could be captured in a layered constraint-based setting (figure 7.1). The benefit of such a model is that lower layers (e.g. epigenetic marks or proteomics data) can propagate constraints up to higher level processes and vice versa. An interesting use case for such an integrated modeling approach would be capturing the early differentiation dynamics in pluripotent mouse embryonic stem cells (mESC). It has been observed that cultured pluripotent mESC, apart from strong genetic and epigenetic dynamics also show significant metabolic alterations [142]. How this is established is largely unknown, which makes them an ideal candidate for such an integrative modeling approach.

Dealing with increasingly larger networks One might expect that with the introduction of the second genome-scale reconstruction of human metabolism [218], metabolic networks will not get larger or more complex. This seems to be
Figure 7.1: Simplified integrated model of regulation and metabolism. a) The metabolic layer is connected by the enzymes that are build from proteins translated from RNA. b) The translated proteins determine which metabolic states are feasible. In turn, they provide information about the transcription layer below. c) Transcription is determined by active promoters and enhancers. The epigenetic machinery determines which genes are transcribed and which are not. Any layer provides feasible states for the layer above. In addition, observed layers provide information about the regulation patterns that must be present in the lower layers.

Integration of experimental data On the one hand, many methods exist that incorporate experimental data into genome-scale constraints. On the other hand,
sampling methods and also propagation methods aim to obtain an unbiased flux sample. It seems natural to also integrate ‘omics’ data into sampling techniques, to obtain samples that are more likely given the experimental data. Using for instance gene- or protein expression to bias the direction of the sampler such that it preferentially explores higher flux regions for highly expressed genes or proteins seems perfectly feasible. Similar incorporation of prior data can be used with BP or EP techniques. This is an easy way to perform (semi) probabilistic inference in constraint-based metabolic models.

7.2 Inferring gene pairs causing a synthetic (dosage) lethality

Metabolic networks have previously been used to compute synthetic lethal (SL) interactions [240, 63] in various model organisms [203, 210] and cancer. Recently, the detrimental effects of certain dosage reactions on cancer cells have also been discovered [228] using metabolic modeling. However, a method to compute synthetic dosage lethality (SDL) from a genome-scale metabolic network had not been published before. The fact that one of the genes within a SDL gene pair is overexpressed makes them particularly attractive for possible applications in cancer research.

Our method scores SDL pairs based on their predicted impact on cell growth. This score proved to be predictive, in particular for measured tumor sizes and patient survival in breast- and ovarian cancer patients. Moreover, we showed that SDLs have a cumulative effect; expressing more SDL pairs is in general beneficial for a patient’s tumor growth and survival prognosis. By studying the SDL interaction network, we discovered that the vast majority of SDL pairs have a downregulated enzyme in the glycolysis pathway. Indeed, many cancer cells depend heavily on glycolysis [72, 71, 55] and we observed that the largest effect of the SDL is due to the knock-out enzyme in this pathway. Nevertheless, a significantly larger effect was found when additional ‘dosage’ enzymes were over-expressed. Such dosage reactions are generally not simple proximal enzymes, but can be in a distinct part of the metabolic network in which metabolites are drained from biomass. The SDL effect can therefore be seen as a competition between reactions in the supply of and demand for metabolites. Lowering the glycolytic activity limits the supply of metabolites that sustain tumor growth. The effect is increased when other reactions drain these metabolites before they can be used for growth. In essence, our method computes which combination of a lowered flux (limited supply) and increased flux (competitive demand) has the largest effect on the availability of metabolites required for tumor growth. By combining clinical data with tumor gene-expression profiles, we were able to verify our model predictions.
Our method is not limited to cancer, but can in principle also be used to identify SDL networks in pathogenic bacteria or fungi, providing new antibiotic therapeutic leads. Other possible applications include metabolic engineering to increase the yield of valuable metabolic byproducts. This can for instance be achieved by engineering SDLs in order to inhibit the production of undesired byproducts, or inversely, neutralizing the SDL effect to force an increased flux through desired pathways.

7.2.1 Synthetic lethality

A novel computational approach that identifies SL interactions from cancer genomic data on a genome-wide scale was presented in chapter 5. First, cancer genomes were systematically explored to identify predictive patterns of SL interactions. Compared with non-SL gene pairs, SL pairs were observed to be significantly less often co-lost in a cancer genome, both in gene expression and in copy number variation analysis. Furthermore, SL pairs are more likely to express a compensation pattern (called up-down SL pairs), where one gene in the SL pair is over-expressed to compensate for the under expression of the other gene. In total, five such predictive signals in cancer genome variations and expression data were detected and integrated in an ensemble-based model. This model achieved high predictive performance (AUROC 0.75) on validation sets of known SL pairs.

7.2.2 Future directions

The SL and SDL interactions discovered are not specific for any tumor type. In the case of SDLs, a general model of human metabolism was used. Furthermore, in the metabolic models, predictions were done on the metabolic flux level, whereas validations were performed mainly on the RNA expression level. Although we were able to reveal general patterns by applying our method to large numbers of patients and genomic data, it is far from clear how such predictions can be utilized for individual patients who often require personalized cancer therapy. This also holds for the SL pairs derived in chapter 5. An interesting approach would be to integrate omics data with the metabolic network, in order to create models that are specific to each cancer (sub)type. Then, differential analysis between the models and comparison with healthy tissue can reveal vulnerabilities that are either specific to, or shared amongst different cancer types. Ideally, these vulnerabilities can be ranked and top candidates would be validated in vitro.
Clearly, the central role of glycolysis in the metabolism of both cancerous and healthy tissue poses a difficulty in drug design. Since glycolysis is also vital for healthy cells, therapeutic strategies should aim to target specifically the glycolytic pathways in cancer cells. Interestingly, compounds that suppress key glycolytic enzymes only in cancer cells have already been successfully identified [30, 62], showing that the SL and SDL mechanism can indeed be applied in practice. Whether stimulation of dosage reactions can be achieved in practice and how that would affect healthy cells is at this moment unclear.

Another important point is that cancer is not limited to metabolism and most of our observations are probably caused by (epi)genetic defects upstream of the altered metabolic pathways. A growing tumor cell obviously requires metabolites and metabolic flux, but these events need to be regulated. Once more, given the abundance of (epi)genetic data, an integrated approach as depicted in figure 7.1 would be beneficial to get a better understanding of SL and SDL interactions and the mechanisms explaining their cause and effect.

For the SL interactions, we used publicly available copy number variation and gene-expression data from colon cancer patients. In this case, the predicted interactions were not limited to metabolism but genome-wide. A disadvantage of this approach is that it is harder to gain systematic insights, since no mechanistic model was used to derive the SLs. Furthermore, SL interactions were derived from colon cancer tissue and in unclear to what extent predicted interactions hold for other cancer types. Finally, both the SDL and SL study identified a large number of lethal interactions. Although that provides ample choice for further study, one would ideally narrow this set of interactions to a small set of high confidence pairs, to be validated in an experimental setup.

An interesting possibility would be to study the glycolytic shift observed in many cancer types (known as the Warburg effect) using an integrated systems biology approach, such as illustrated in figure 7.1. Evidence collected from different layers could then be used to better understand this effect, in order to design more effective experimental perturbations.

In summary, SL and SDL approaches, whether derived from a mechanistic model or inferred from data provides important opportunities to selectively target cancer cells or pathogens. In general, the cellular complexity of these diseases cannot be fully understood with a relatively coarse-grained metabolic model. Tailoring the model to a particular cancer (sub)type is important to increase its predictive capabilities. Furthermore, a complex disease such as cancer cannot be completely understood by only looking at metabolism. Therefore, it is important that other processes such as gene regulation are also captured, ideally in an integrated cellular model.
7.3 Automated dataset weighting and feature extraction from high-throughput data

In chapter 6, a webtool was introduced that searches novel genes co-regulating with a set of query genes known to be involved in a certain molecular system. The WeGET webtool ranks genes based on their weighted co-expression with the query genes within a large compendium of micro-arrays. Mining micro-array datasets using co-expression; a so called ‘guilt by association’ strategy has been applied in numerous papers [234, 47, 78]. What makes our application interesting is that it automatically weights the most relevant datasets based on the congruent expression of the query genes. If the expression of query genes is jointly activated or repressed in a certain dataset, that dataset might be more relevant than one without a clear co-expression pattern between the query genes. In chapter 6, we only used micro-array data, but the same principle can be applied to other ‘omics’ data types, ideally combining multiple data types of interest.

7.3.1 Future directions

New high-throughput sequencing and mass-spectrometry techniques provide massive amounts of novel data. This data can in principle be used to gain better understanding of any query system. More importantly, since the amount of data provided by these techniques is so large, obtaining the relevant genes, proteins or other features from an experiment often depends on arbitrary cutoffs, such as fold changes or read counts. A system that automatically crawls a large number of similar data sets and selects the most interesting ones based on some relevance criterion is an attractive way to rapidly obtain a set of key players in the system with higher confidence.

As long as a good measure of relevancy can be defined, such a strategy can be applied to any type of data to find the key genes, proteins, metabolites or epigenetic marks relevant to a query set. Moreover, this data can be integrated such that search results are not limited to co-expressing genes. For instance, when one is interested in gene regulation, the search result could also provide relevant proteins (transcription factors), chromatin interactions, epigenetic alterations and DNA binding motifs. Third, this data can provide an integrated view of the query system. In the ideal case, a systems biology approach shows not only which components are associated with the query system, but also provides systematic insights into their relation.

Taken together, the automated selection of relevant data and extraction of the most important features opens important possibilities in the analysis of high-
throughput ‘omics’ data. On the one hand, this data can be used to obtain a better and preferably systematic picture of the query system. On the other hand, such a tool could prove to be advantageous in rapidly obtaining key insights in high-throughput experiments.

7.4 Combining model-driven and data-driven approaches

In this thesis, two approaches for analyzing biological systems have been used. First, existing models can provide novel systematic insights and ‘mapping’ new data on established models is often used to better understand the mechanics of a biological system. Second, if such a model does not exist, relationships can also be derived directly from the data. Surely, both approaches are not mutually exclusive. Models can be created, refined, extended or integrated using new data. Furthermore, where currently the availability of data mostly drives the construction of mechanistic models, the reverse is also possible. Mechanistic models can provide interesting leads for follow-up experiments that could not have been easily discovered by ‘simple’ analysis of the data. Ideally, experimental and computational approaches are combined in a systems biology project, where model-driven discovery is combined with wetlab experiments and validation in an iterative fashion. This trend is already visible and an increasing number of labs appreciate the power and benefits of this combined approach. By exchanging the models and the data through public repositories, we will ultimately advance to an integral cellular model, with significant implications, for example in our combat against complex diseases.
Bibliography


[38] Michael Costigan et al. “Multiple chronic pain states are associated with a common amino acid–changing allele in KCNS1”. In: *Brain* (2010), awq195.


[64] Stephen S Fong, Andrew R Joyce, and Bernhard Palsson. “Parallel adaptive evolution cultures of Escherichia coli lead to convergent growth phenotypes with different gene expression states”. In: Genome research 15.10 (2005), pp. 1365–1372.


[230] Sharon J Wiback et al. “Monte Carlo sampling can be used to determine the size and shape of the steady-state flux space”. In: *Journal of theoretical biology* 228.4 (2004), pp. 437–447.


Metabolism is the process in which a cell or organism takes up nutrients from its environment and converts them to molecules that facilitate the cellular mechanisms required to sustain life. This conversion process is performed by a large number of metabolic reactions. The metabolites produced by one reaction are often the inputs for others. Most metabolic reactions do not occur spontaneously, but require proteins (or protein complexes) called enzymes, which catalyze the reaction. The ‘recipe’ for making these proteins is stored in the genes on our DNA. In particular, a gene is a long sequence of only four possible nucleotides (abbreviated A, C, T or G) and the specific order of these nucleotides determines the protein it encodes. Before the protein is made, the gene needs to be transcribed to RNA (a sort of ‘temporary copy’ of the recipe). The RNA molecule is finally translated into amino-acids that make up the protein. When the cell is actively creating many of these RNA copies, we call the gene expressed and it often indicates that the cell is producing much of this protein. Thus, by regulating the rate of gene transcription and translation, a cell controls which proteins and enzymes are present. Using this mechanism for hundreds or even thousands of genes, a cell regulates (switches on or off) the metabolic pathways that are available.

For most of the metabolic reactions, the metabolites consumed and produces are known. This also holds for most of the enzymes catalyzing the reactions and gene(s) required to produce these enzymes. Therefore, metabolism can be modeled as a network, with connections between reactions sharing metabolites. However, also the connections between reactions, enzymes and genes can be stored in such a network. This network representation of metabolism is convenient, because it allows us to use computational methods and mathematical tools such as graph theory and optimization techniques to analyze their behavior. For example, we can determine which pathways are ‘blocked’, because the enzyme coding genes for many of the reactions in that pathway are not expressed.

Reaction rates or metabolic flux (the reactions ‘speed’) and metabolite con-
centrations depend on many things, can change rapidly over time and may differ between organisms. Measurement of all parameters required to create a detailed model of how metabolic flux and metabolite concentrations change over time is currently infeasible for a metabolic network that spans all known metabolic reactions. A widely used alternative is COnstraint-Based Reconstruction and Analysis (COBRA) of metabolism. The idea of COBRA is to explicitly incorporate physical, biochemical and genetic limitations of the cell in order to model which metabolic states can be reached. Metabolic states of interest are for example a cell's growth rate, the production rate of certain metabolites such as (bio)ethanol or which nutrients must be available in order for the cell to survive.

COBRA assumes two basic types of constraints. First, a steady-state constraint dictates that any metabolite that is produced by a reaction inside the cell must immediately be consumed by another reaction. Though this is strictly not the case, the time scale on which metabolic events occur is typically much smaller than for the processes regulating them (activation of the genes and building of the enzymes). Therefore this is a useful simplification. A second type of constraint limits a reaction's capacity (its maximum flux) and the directionality. A flux path that satisfies these constraints is called feasible. Together, all feasible flux paths form a space (think of it as a box) and the volume of this space reflects the amount of feasible flux paths in the space. The volume also reflects our uncertainty about the metabolic pathways used by the cell, since a large volume means that many alternative flux pathway exist. It is possible and often desirable to add more constraints to further narrow the space of feasible metabolic states, or more informally, to shrink the box. Such constraints can be the result of metabolic flux measurements, integration of protein- or RNA expression or for example by only considering the flux paths that maximize the tumor growth rate in cancer cells.

In constraint-based modeling of metabolism, the space of feasible flux paths is of central interest. Its volume reflects the capabilities of the cell under the constraints imposed. In practice, there are never enough constraints to pinpoint the exact flux path utilized by the cell.

The first part of this thesis consists of three chapters dealing with metabolic modeling and this space of feasible flux paths. An introductory chapter briefly explains how models are created and used in molecular biology, with a focus on metabolic modeling in particular. In chapter 2, a method is described that samples valid metabolic states from the space of feasible steady-state distributions. Though the volume of the flux space was mentioned above, this word is actually reserved for spaces with three dimensions. The space of feasible flux paths actually consists of hundreds of dimensions, even for a relatively simple organism such as bakers yeast. Sampling efficiently from this large space
requires optimized computational methods. An important application of sampling is to find which flux paths are often used compared to others. Though useful, sampling from large metabolic spaces can also lead to undesired bias and therefore sampling results have to be considered with care. In chapter 3, an optimization method is introduced which finds the flux paths that severely reduce the metabolic robustness of the cell. Robustness is important for a cell in order to adapt to changing environmental conditions. We show that this method can also be used to prioritize experimental measurements of metabolic fluxes in order to reduce the flux space. In chapter 4, metabolic modeling is combined with a new algorithm to detect so-called synthetic dosage lethali ties (SDLs), with potential to target tumor cells. Briefly, the method finds specific combinations of one inactive (or at least reduced in activity) and one overactive reaction that reduce the production of metabolites required for a tumor to grow. One can think of this as a supply and demand problem. The pinched reaction limits the supply of metabolites needed by the tumor and indeed by itself already reduces its growth. The overactive reaction competes with the tumor for its scarce construction material and therefore amplifies this effect. Cancer patients with many of these SDLs indeed have reduced tumor sizes and longer survival times compared to patients who do not have these SDLs. Such discoveries can be beneficial to test the potential of new anti-cancer drugs in an early stage using computer simulation or explore new possibilities to target tumor cells.

The second part of the thesis studies genetic interactions causing cell death and the automated discovery of gene sets involved in specific cellular processes. One such genetic interaction is the synthetic lethality (SL), which is similar to the SDL described above. In this case, the combined inactivation of two specific genes is lethal to the cell, whereas the inactivation of only one of the genes is not. SLs also have particularly interesting applications in cancer cells where genetic mutations occur frequently. When a cancer cell has lost one of the genes (say gene A) for which a SL partner gene is known (gene B), this provides a good opportunity to silence gene B. Besides its lethal effect on the cancer cell, healthy cells will survive the deletion of gene B, since they do not have the mutation that inactivated gene A. Because human DNA contains thousands of genes, it is practically impossible to screen for all possible combinations of genes in an experimental setup. But, given that large amounts of genetic data obtained from healthy and cancerous tissue are available, we can screen for promising candidates using computer models and simulation. Chapter 5 deals with the computational screening for SL gene pairs. In this chapter we tried to answer two important questions. First, how do you recognize patterns of promising SL candidates in all this genomic data? Moreover, how can we integrate these patterns to come up with a ranked list of the most promising SL candidates?
Importantly, the SL genes found are still candidates. To really say that they interact, such SL combinations have to be tested experimentally. However, we can show that SL interactions that have been tested experimentally are frequently found near the top of our list and thus that such a method can indeed be very valuable in selecting candidates that are more promising compared to others.

Chapter 6 deals with a similar topic for finding genes that are involved in certain cellular functions. One can for instance think of genes involved in a disease, but the method is equally valid for genes playing important roles in normal cellular functions. For many functions and diseases, a number of the genes involved is known, but this information is often incomplete. Can we use our knowledge of the known genes to find novel genes that may also play a role in the cellular function or disease? Again, we can try to exploit the huge amounts of public genetic data that is available. One data source that is often used is gene expression data. Genes that play an important role in a certain cellular process, disease or tissue are often co-expressed. This means that the cell activates and deactivates these genes at the same time, such that the enzymes required for the cellular process or disease are available at the same time. Importantly, cellular processes taking part in certain brain cells may not take place in muscle cells. Similarly, a set of genes that are important in breast cancer, may not be active at all in lung cancer. Therefore, it is critical that the data source used matches the system of interest as close as possible. In this chapter, we describe a tool that finds the most important data sets for the molecular system specified and weights them accordingly. Then, the more important data sets are used to find novel genes that may also be important for the system we study. Such a tool provides a very easy and cheap way to screen for novel genes that can then be tested experimentally. It also provides an easy means to check the potential of novel candidates found by another (computational) method.

Our cells (and those of any other complex species) contain large amounts of genes, proteins and many other cellular compounds that interact in a complex temporal and spatial manner. Modern methods can measure thousands of genes or proteins in a single experiment. Often, such experiments are performed to compare different cell types or to follow a cellular process over time. One can imagine the amounts of data that are obtained in such an experiment and the combinatorial explosion of all the possible genes and proteins that may or may not be important in this process. Computational analysis is needed to discover which genes, proteins or reactions are important and how they interact to fulfill the cellular task. For the latter, it is important to build computer models that mimic their behavior, to unravel how all the small parts work together and try to predict the consequences of perturbations. Such predictions can be tested to refine our models. New experimental techniques are introduced every year,
enabling us to look at new aspects of the complex functioning of the cell in greater detail. There is a great number of interesting opportunities to study this and I will discuss some of them in the concluding chapter. Using computers to analyze and model this complex behavior has greatly advanced the field of molecular biology and our understanding of disease and will continue to do so in the future.
Samenvatting

Metabolisme is het proces waarin een cel of organisme voedingstoffen opneemt uit zijn omgeving en omzet in moleculen die de cellulaire processen mogelijk maken. Dit afbreken en omzetten van moleculen wordt uitgevoerd door een groot aantal metabole reacties. De moleculen (metabolieten) die worden geproduceerd door de ene reactie worden vaak weer afgebroken en omgezet in andere moleculen door de volgende reactie. De meeste metabole reacties vinden echter niet of nauwelijks spontaan plaats, maar vereisen enzymen (eitwitten of eitwit-complexen) die de reactie versnellen. Het ‘recept’ om eitwitten te maken ligt opgeslagen in de genen op ons DNA. Ons DNA bestaat voor een groot deel uit vier soorten zogenaamde nucleotiden (afgekort A, C, T en G). Een gen is een langgerekte stuk DNA van duizenden nucleotiden, die door hun specifieke volgorde coderen voor een specifiek eiwit. Voordat dit eiwit gemaakt kan worden moet het DNA ‘afgelezen’ worden. Dit aflezen gebeurt door het gen om te zetten naar RNA; een soort tijdelijke kopie van het gen. Hierna wordt het RNA vertaald naar een keten van aminozuren die het eitwit vormen. Op het moment dat de cel veel van deze RNA kopiëren van het gen maakt noemen we dat gen actief. Dit is vaak een indicatie dat relatief veel van het eiwit wordt gemaakt. De cel kan deze activiteit van aflezen en kopiëren naar RNA (transcriptie) en omzetten naar eiwit (translatie) reguleren en hiermee dus bepalen welke enzymen op een bepaald moment aanwezig zijn. Door gelijktijdige regulatie van honderden of zelfs duizenden genen bepaalt een cel welke metabole reacties en paden beschikbaar zijn.

Veel van de reacties en metabolieten in onze cellen en die van veel andere organismen zijn bekend. De enzymen die deze reacties catalyseren, evenals de genen die betrokken zijn bij de productie van deze enzymen zijn ook bekend. We kunnen metabolisme daarom modelleren als een groot netwerk van metabolieten die de input of output zijn van verscheidene metabole reacties. Van deze reacties modelleren we welke enzymen de reactie catalyseren en tenslotte welke genen verantwoordelijk zijn voor het maken van deze enzymen. Zo’n netwerk
Samenvatting

model is handig voor de visualisatie van metabolisme, maar vooral omdat er uitgebreide wiskundige technieken en methoden beschikbaar zijn om netwerken te analyseren. Een voorbeeld is om te kijken welke metabole paden niet beschikbaar zijn, omdat de onderliggende enzymen niet actief zijn. Een andere veelgebruikte toepassing is het berekenen van de maximale celgroei gegeven de beschikbare voedingstoffen.

Reactiesnelheden (ook wel metabole flux genoemd) en de concentraties van metabolieten en eiwitten worden bepaald door vele factoren, kunnen zeer snel veranderen en verschillen vaak tussen organismen. Het meten van al deze factoren is op dit moment vrijwel onmogelijk wanneer alle reacties en moleculen die betrokken zijn bij het metabolisme worden beschouwd. Een veelgebruikt alternatief is de metabole capaciteiten van een cel te bepalen door haar fysieke, biochemische en genetische beperkingen in acht te nemen. Voorbeelden hiervan zijn de maximale groeisnelheid van een bacterie wanneer er wel of juist geen zuurstof aanwezig is, of de reacties die actief moeten zijn om zoveel mogelijk glucose om te zetten in bioethanol.

Twee soorten beperkingen worden altijd toegepast. Allereerst wordt aangenomen dat ieder metaboliet geproduceerd in de cel direct weer wordt omgezet in een andere metaboliet. In de cel vindt dus geen opslag van metabolieten plaats. Hoewel deze aannamen een versimpeling van de werkelijkheid zijn kan die in het algemeen worden toegepast omdat de tijdspanne waarop metabole reacties plaatsvinden vele malen kleiner is dan die van de processen die het metabolisme reguleren. De tweede beperking stelt een grens aan de snelheid en de richting (de reactie zet metaboliet A om in B en niet andersom) waarmee die kan plaatsvinden. Een aaneenschakeling van reacties (met hun reactiesnelheid) dat aan deze twee voorwaarden voldoet wordt een geldig(e) fluxpad of fluxdistributie genoemd. Alle geldige fluxpaden samen vormen een ruimte (een soort doos) en het volume van deze ruimte is een maat voor de metabole capaciteiten van de cel. Dit volume is echter ook een maat voor onze metabole kennis van de cel. De metabole capaciteiten van de cel zijn in werkelijkheid vaak kleiner dan in ons model, alleen weten we niet welke reacties wel of niet plaatsvinden en met welke snelheid. Wellicht de grootste uitdaging van deze manier van modelleren is om de capaciteiten van de cel in ons netwerk zo precies mogelijk te definiëren, zonder daarbij de daadwerkelijke capaciteiten te verliezen. We bereiken dit over het algemeen door voorwaarden of beperkingen in de vorm van extra aannames of metingen toe te voegen. Een voorbeeld van zo’n aannamen is dat metabole reacties actief zijn wanneer hun enzymen actief. Die enzymen zijn op hun beurt weer actief als hun coderende genen actief zijn. Het is mogelijk de activiteit van vele genen gelijktijdig te meten en met de genoemde aannames kan men dan de onderliggende activiteit van het metabole netwerk afleiden.
Samenvatting

Het eerste deel van dit proefschrift bestaat uit vier hoofdstukken waarin het metabole netwerk en haar capaciteiten centraal staan. Dit deel begint met een introductie over de constructie en analyse van modellen in de moleculaire biologie, met speciale aandacht voor metabole netwerken. In hoofdstuk 2 wordt een methode beschreven waarin samples worden genomen uit de ruimte van geldige fluxpaden. Hierboven wordt gesproken over het volume van deze ruimte, maar dat woord verwijst eigenlijk naar objecten met drie dimensies. De ruimte van geldige fluxpaden heeft vaak honderden dimensies, zelfs voor een relatief eenvoudig orgaisme zoals bakkersgist. Efficiënt een groot aantal samples nemen uit deze gehele ruimte vereist geoptimaliseerde computerprogramma’s. Een belangrijke toepassing van sampelen is om te bepalen welke fluxpaden waarschijnlijker zijn dan anderen. Hoofdstuk 3 behandelt een methode die fluxpaden vindt die de robuustheid van het metabole netwerk ernstig verminderen. Robuustheid is zeer belangrijk voor een cel om snel in te kunnen spelen op veranderende omstandigheden. Vaak kunnen de fluxpaden die de robuustheid van de cel aantasten buiten beschouwing worden gelaten, omdat deze in werkelijkheid vrijwel nooit gebruikt worden. Ook kan de methode gebruikt worden om te bepalen welke experimenten de meeste informatie verschaffen over de genoemde ruimte van fluxpaden, opdat deze zoveel mogelijk verkleind kan worden.

Hoofdstuk 4 gaat over een nieuwe methode om zogenaamde celdood door synthethische overdosering (CSO) te bepalen aan de hand van het metabole netwerk. Deze CSO is een combinatie van een sterk verlaagde reactiesnelheid van reactie A en een verhoogde snelheid van reactie B die dodelijk is voor een cel. CSO’s vinden een interessante toepassing in de bestrijding van kankercellen, die vaak gekenmerkt worden door sterk verhoogde activiteit in bepaalde metabole reacties. Men kan deze CSO beschouwen als een vraag- en aanbod probleem. De levering van bouwstoffen aan de tumor wordt verminderd door reactie A te blokkeren. Reactie B gebruikt ook de bouwstoffen die A levert en concurreert dus met de tumor. Door de levering van moleculen door A te beperken worden deze schaars en een verhoogde activiteit van reactie B zorgt (nog sterker) dat vrijwel geen moleculen aan de tumor geleverd worden, waardoor deze cellen sterven. Kankerpatiënten met veel van deze CSO bleken zoals verwacht kleinere tumoren en een langere overlevingstijd te hebben. Methoden zoals deze kunnen gebruikt worden om snel en goedkoop kansrijke combinaties van enzymen of genen op te sporen. Ook kan men met behulp van deze methodes ook voor andere combinaties van genen of enzymen snel bepalen wat hun effect is op de berekende tumorgroei.

Het tweede deel van dit proefschrift bestudeert genetische interacties die tot celdood leiden en combinaties van genen die juist belangrijk zijn voor specifieke
processen in de cel. Een interessante genetische interactie is de zogenaamde synthetische lethaliteit (SL). In dit geval leidt de gezamenlijke blokkering van twee genen of enzymen tot celdood, maar het blokkeren van slechts een van hen niet. Ook voor SL’s geldt dat er mogelijk interessante toepassingen bestaan in kankeronderzoek. Omdat veel kancercellen mutaties ondergaan is het mogelijk dat bepaalde genen in de kancercel verloren zijn gegaan (bijvoorbeeld gen A). Dit gen kan redundant zijn als er een gen B is met een gelijke functie. Het uitschakelen van gen B kan nu de kancercel doden, omdat deze niet terug kan vallen op gen A. Een normale cel kan dit wel en dit principe maakt SL’s buitengewoon interessant om tumorcellen heel specifiek te bestrijden. Ons DNA bevat duizenden genen en het is vrijwel onmogelijk om voor alle combinaties experimenteel vast te stellen of ze een SL vormen. Gelukkig zijn er grote hoeveelheden moleculaire data verzameld van tumor- en gezonde cellen. Dit maakt het mogelijk om computerprogramma’s te ontwikkelen die kansrijke combinaties van genen selecteren. Dit screenen van grote hoeveelheden data om SL combinaties te vinden is het onderwerp van hoofdstuk 5. We proberen hier twee belangrijke vragen te beantwoorden. Allereerst, welke patronen in de data kenmerken deze SL’s? Ten tweede, hoe kunnen we deze patronen combineren in een computerprogramma dat geheel automatisch een lijst geeft van de meest kansrijke combinaties? Het programma op zichzelf kan geen uitsluit-sel geven over de gevonden combinaties; deze moeten nog getest worden in het laboratorium. Het voordeel van een goed programma is dat de ‘kansrijke’ SL’s inderdaad veel vaker daadwerkelijk een SL blijken te zijn dan wanneer zo’n programma niet wordt gebruikt. Hierdoor kan dus veel gerichter onderzoek worden gedaan, waardoor tijd en kosten bespaard kunnen worden.

Hoofdstuk 6 gaat over het vinden van genen die betrokken zijn bij een bepaalde celfunctie. Men kan hierbij denken aan genen die geassocieerd worden met een bepaalde ziekte, maar ook aan genen die belangrijk zijn voor reguliere celfuncties. Voor veel functies en ziektes is een deel van deze genen bekend, maar voor lang niet allemaal, of het is niet bekend of deze set compleet is. Kunnen we de genen die reeds bekend zijn gebruiken om andere genen, die mogelijk ook betrokken zijn bij de celfunctie of ziekte op te sporen? Wederom kunnen we de grote hoeveelheid genetische data gebruiken. In het bijzonder kijken we hier naar genexpressie; informatie over de mate waarin genen actief zijn. Genen die betrokken zijn bij een gezamenlijke functie komen vaak gelijk-tijdig tot expressie (co-expressie). Deze processen zijn echter celtype specifiek. Processen die zich in hersencellen afspelen, vinden vaak niet plaats in hartcellen en genen die betrokken zijn bij borstkanker zijn wellicht totaal niet betrokken bij longkanker. Het is daarom belangrijk dat de gen-expressie data die men gebruikt om co-expressie te bepalen zo goed mogelijk matchen met het celtype.
en het proces waarin men geïnteresseerd is. Het is vrijwel onmogelijk om dit handmatig te matchen voor honderden datasets en duizenden mogelijke processen waarin men geïnteresseerd kan zijn. In dit hoofdstuk wordt een methode beschreven die automatisch goed matchende datasets zwaarder weegt en op die manier vaak meer relevante resultaten oplevert.

Onze cellen (en die van ieder ander complex organisme) bevatten grote hoeveelheden genen, eiwitten en heel veel andere componenten die op een complexe manier interacteren in de tijd en ruimte. Met moderne technieken kunnen duizenden genen, eiwitten of andere componenten gelijktijdig gemeten worden in een enkel experiment. Zulke experimenten worden vaak herhaald voor verschillende celtypen of op verschillende tijdstippen om de ontwikkeling van de cel te volgen. Men kan zich de enorme hoeveelheid data voorstellen die hierbij gegenereerd wordt. Bioinformatica en het modelleren van biologische systemen zijn belangrijk om een beter begrip te krijgen van de cel en haar functie. Jaarlijks worden er nieuwe en verbeterde technieken geïntroduceerd die ons een stapje verder brengen in het begrip van de complexiteit van de cel. Computermodellen die deze complexiteit kunnen nabootsen en de consequenties van veranderingen of experimentele ingrepen correct voorspellen zijn zeer belangrijk voor een beter begrip van gezonde en zieke cellen. Dit brengt nieuwe uitdagingen en mogelijkheden met zich mee, die kort in het concluderende hoofdstuk aan bod komen. Bioinformatica en systeembiologie hebben een belangrijke bijdrage geleverd aan de moleculaire biologie en zullen dit blijven doen in de toekomst.
Iemand zei eens tegen me dat een promotietraject een marathon is, en dus geen sprint. Iedere sporter kent het gevoel dat ie door enthousiasme en onervarenheid te snel uit de startblokken is gevlogen en zichzelf te vroeg in de wedstrijd tegenkomt. Gelukkig hoefde ik deze wedstrijd niet alleen te lopen. De steun van collega’s, begeleiders, familie en vrienden zorgde ervoor dat ik snel weer op stoom was en de rest van het traject zonder noemenswaardige problemen heb kunnen afleggen. Nu de finish in zicht is, wordt het dus de hoogste tijd een aantal mensen te bedanken.

Allereerst wil ik mijn promotoren Elena en Martijn bedanken. Elena, bedankt voor zowel je inhoudelijke als persoonlijke advies en begeleiding gedurende mijn promotieonderzoek. Toen ik mijn masterscriptie bij jou deed kwam ik voor het eerst in aanraking met de boeiende vakgebieden machine learning en bioinformatica. Vanaf die tijd, tot aan het einde van mijn promotie kon ik altijd bij je terecht voor advies en vragen, of om nieuwe ingevingen te bespreken. De sessies waarin we samen je whiteboard volschreven met schetsen, uitwerkingen van ideeën en resultaten in de wildste kleurencombinaties waren leuk en leerzaam. Je advies over machine learning en algoritmes was zeer nuttig en daar heb ik erg veel van geleerd. Ook het assisteren bij je colleges, met als hoogtepunt de twee colleges die ik zelf heb gegeven, vond ik erg leuk.

Martijn, bedankt dat ik deel heb mogen uitmaken van je bioinformatica groep (onderdeel van het CMBI) op het NCLMS. Ik vond het leuk en leerzaam om samen met Richard en Sergio mee te werken aan het CSBB project, waarin we onze metabole netwerken hebben ingezet om ernstige mitochondriële aanlopen benaderen. In het begin was het even afgasten en vond ik de groep bijeenkomsten lastig, omdat ik minder achtergrond in de biologie had dan de meeste promovendi op het CMBI. Later is dat goed gekomen en ben ik ook steeds meer gaan samenwerken met anderen. Eerst met Richard en Xiaowen op de genetische interacties in kankercellen en daarna met Radek en Pavel aan de weighted co-expression applicatie.
Richard, bedankt voor onze interessante tijd, met name gedurende de tweede helft van mijn promotie. We hebben samen gelachen (ik denk aan de Skype sessies met Eytan, of de nullematrix ...), maar ook interessante discussies over metabolic modeling gehad, met als hoogtepunt onze publicatie in PNAS en een stukje in de Telegraaf. Ik bewonder je creativiteit en de doelgerichte manier waarop je onderzoek doet. Waar ik nog weleens de neiging heb om teveel te willen beschrijven of de analyses tot in perfectie uit te voeren, heb jij ons altijd op de ‘main road’ gehouden en de rode draad bewaakt.

CMBI en CSBB

Graag wil ik mijn roommates op het CMBI bedanken. Robin en John, bedankt dat ik mijn nutteloze ideeën en theorieën met jullie mocht delen. We hebben veel en vaak gelachen, en ik kom nog altijd graag even ‘buurten’ in jullie kamer. John ook bedankt dat je altijd bereid was mijn knuffels in ontvangst te nemen en dat je mijn biologische vragen zo treffend kon beantwoorden :-) Radek and Pavel, thanks for being nice roommates and for collaborating with me on the weighted co-expression webtool.

Naast mijn roommates wil ik nog een aantal bioinformatici persoonlijk bedanken. Rob, ik ken geen mens (of konijn) die meer wortels, appels, paprika’s of welke groenten en vruchten dan ook meeneemt naar zijn werk, om ze daar uitgebreid in perfecte reepjes van 8 bij 2 cm te snijden. We hebben met ons vaste ‘lunchwandeling clubje’ altijd met, maar ook om je gelachen. Bij dat groepje horen natuurlijk ook Wouter, Jon en Robin. Wouter, ik vond het uiterst pijnlijk om met jou samen een rondje te racefietsen over de Nijmeegse stuwwallen. Maar nu dit boekje bijna af is, ga ik meer trainen en zal ik je uiteindelijk pakken! Jon, unfortunately, we never collaborated on any project. However, our (first) sons Sem and Sepp are born a couple of days from each other. This shared feeling of being a proud young father, as well as struggling with our programming code due to elongated eye bags served as key ingredients for a rather special relationship.

Sergio, thanks for collaborating on the metabolic models and your mentorship during the early phase of my PhD. I’m sure we’ll keep meeting each other on conferences and hopefully we can collaborate in the future. Xiaowen, thanks for working with me on the synthetic (dosage) lethal interactions and your kind offerings of chocolate and other candy.

Tenslotte bedankt aan alle anderen met wie ik niet direct samen heb gewerkt, maar die mijn periode op het CMBI toch hebben veraangenaamd.

I also wish to thank everyone involved in the Centre for Systems Biology and Bioenergetics (CSBB) for the pleasant and inspiring time during my PhD project.
Though we only met one hour a week, these were interesting meetings.

**Machine Learning Group**

Naast Elena, wil ik alle andere mensen in de machine learning groep bedanken. **Tom**, bedankt dat ik een paar jaar mocht meedraaien in je groep, dat je me de planning van de wekelijkse colloquia toevertrouwde en dat je voorzitter van mijn manuscriptcommissie wilde zijn.  

**Twan**, bedankt voor je advies bij sommige methodes en je snelle hulp wanneer de C++ compiler en ik het samen niet eens konden worden. Ik vond het ook erg leuk wanneer we als Starcraft teammates ten strijde trokken tegen al het virtuele kwaad. Ik vond het een eer om als paranimf bij je promotie te zijn, en ben blij dat de rollen binnenkort omgedraaid zijn.

Furthermore, I’d like to thank Ali, Christiaan, Daniel, Dimitris, Elena S, Evgeni, Fabio, Frank, Jonce, Joris, Kasper, Maaike, Max, Maya, Mohsen, Pavol, Perry, Rhido, Saiden, Saskia, Suzan, Tjeerd and Tom C. for their support and suggestions during my pleasant stay.

**Moleculaire Biologie**


**Yaser**, as my partner in the lab, I want to thank you, in particular for your patience during the first couple of months of my postdoc period. At that time, I sometimes had to spend some hours on finishing my thesis, while also working on our embryonic stem cell project. Hopefully, the ‘oliebollies’ at New Year’s Eve made up for it a little bit. I’m looking forward to soon submitting our first manuscript and hope many will follow. Thanks so far to all other people in this department. Though thanking everyone sounds a bit strange, since I do not
Dankwoord / Acknowledgments

intend to leave soon. Instead, I’m looking forward to continue working with many of you and hope it will lead to interesting new biological discoveries.

Familie en Vrienden


Wieger, Hans, Jeffrey, Rick, Erwin, Niels en Jeroen, bedankt dat jullie al jaren deel uit maken van mijn hechte vriendengroep. Ik kijk met veel plezier terug op de vakanties en uiteraard die we als pubers en jonge mannen hebben meegemaakt. Maar ook tegenwoordig doen we nog regelmatig leuke dingen samen, soms alleen met de mannen, andere keren met partners en kinderen.

Wieger, Thomas, Karin, Eveline en Annet, bedankt voor alle speciale en heerlijke avondjes in ons eetclubje. De levensgevaarlijke zeilavonturen onder kapitein Thomas, het gezamenlijk nuttigen van biervellen’s bij Samson of café Jos, en natuurlijk de onvergetelijke klaverjasavonden, dat is precies wat een hardwerkende promovendus van tijd tot tijd nodig heeft. Wieger, ik ken weinig mensen die zo fanaticen zijn met kaarten als jij. Het is heerlijk om met jou samen te winnen van Annet/Eveline en Jolien. Minstens zo leuk is het om jouw (slecht) ingehouden woede te zien als we verliezen, omdat we -geheel tegen alle statistieken in- een boompje lang slechte kaarten krijgen . . . Gelukkig gebeurde het dan ook niet vaak ;-)

Danny, bedankt voor alle mooie momenten tijdens, maar vooral ook na onze studie. De festivals en Rammstein optredens die we samen (met Samantha) bezocht hebben waren fantastisch, maar ook de onvergetelijke Starcraft marathons moeten hier worden genoemd! Ik vind het prachtig dat je een van mijn paranormen wil zijn. Ik zie ons al staan, breedgrijzend in onze pinguinpakjes!

Tenslotte wil ik de twee belangrijkste personen in mijn leven bedanken. Sepp, bedankt dat je alweer ruim anderhalf jaar het mooiste, liefste en grap-
pigste kereltje bent dat ik ken. Ik vind het fantastisch om te zien hoe je elke dag weer dingen leert en ontdekt, en ik geniet van de boevenstreken die je uithaalt. Je bent bijna altijd vrolijk en pakt iedereen in met die brede grijns, of ontwapenende lach van je. Jolien, als ik alle gebieden waarop ik je dank verschuldigd ben zou moeten beschrijven, dan zou dit -toch al niet korte- dankwoord de helft van mijn proefschrift gaan beslaan. Je hebt altijd voor me klaar gestaan; (te) veel huishoudelijke taken op je genomen, naast je co-schappen en later in je drukke bestaan als huisarts. Je hebt vele weekenden alleen voor Sepp gezorgd en geaccepteerd hoe ik vele avonden aan mijn onderzoek werkte. Daarnaast ben je ook nog een prachtige, lieve en intelligente vrouw en ik knijp elke dag mijn handjes dicht als ik naast je wakker word. Ik weet dat je het moeilijk vond dat ik vaak tijd aan mijn onderzoek besteedde, wanneer we ook leuke dingen hadden kunnen doen. Daarom wordt het nu tijd om meer samen te genieten van de vele andere leuke dingen in het leven.

Ik heb erg genoten van mijn tijd als promovendus en ga nog steeds iedere dag met erg veel plezier aan de slag als bioinformaticus. Ik ben blij en trots dat dit boekje af is, maar realiseer me dat het onmogelijk was geweest om de finish te halen zonder de steun en liefde van alle mensen die het traject geheel of gedeeltelijk hebben meegelopen. Ik wil iedereen daarvoor nogmaals hartelijk bedanken!

Sinds maart 2015 werkt hij als bioinformaticus in de moleculaire biologie groep onder leiding van Henk Stunnenberg. Hier gebruikt hij methoden uit de machine learning en netwerktheorie om (epi)genetische data te analyseren en integreren. Het ultieme doel hierbij is de systematische werking van biologische systemen in zowel gezonde als ziekte cellen te begrijpen en het effect van eventuele interventies te voorspellen.
List of publications


