FCT Fundação para a Ciência e a Tecnologia
REPÚBLICA PORTUGUESA | CIÊNCIA, TECNOLOGIA E ENSINO SUPERIOR
NORTE2020 PROGRAMA OPERACIONAL REGIONAL DO NORTE
PORTUGAL 2020
UNIÃO EUROPEIA Fundo Social Europeu

Integrative Pathway Analysis Approaches for Cancer Research And Drug Development
José Vítor de Castro Vieira
UMinho | 2021

Universidade do Minho
Escola de Engenharia

José Vítor de Castro Vieira

Integrative Pathway Analysis Approaches for Cancer Research And Drug Development

July 2021

Universidade do Minho
Escola de Engenharia

José Vítor de Castro Vieira

Integrative Pathway Analysis Approaches for Cancer Research and Drug Development

Doctorate Thesis
Biomedical Engineering

Word developed under the supervision of:

Dr. Miguel Rocha
Dr. Julio Saez-Rodriguez

July, 2021

COPYRIGHT AND TERMS OF USE OF THIS WORK BY A THIRD PARTY

Acknowledgements

This document is the culmination of a project that I will always look back with joy. I am glad to say I had the utmost pleasure of working with many talented people during these four years spent in my pursuit for knowledge. But I am also thankful for the support of friends and family that have always given me a reason to keep going forwards. I dedicate this section to each and every person who supported me.

I would first like to thank Fundação para a Ciência e Tecnologia for the PhD studentship I was awarded with (SFRH/BD/118657/2016). This work is only possible due to their support. I would also like to thank the Centre of Biological Engineering at the University of Minho for hosting me and providing an excellent working environment. I also thank the BioQuant research institute at the University of Heidelberg, my secondary host instituition for their support.

A razão pela qual comecei toda esta aventura deve-se ao meu orientador principal, o Doutor Miguel Rocha. A partir de uma breve conversa há cinco anos atrás criamos um plano ambicioso e agradeço todo o apoio e supervisão mas também a liberdade criativa que me proporcionou durante todo o plano.

Agradeço também ao Jorge Ferreira que me acompanhou durante a maior parte deste trabalho. Primeiro, agradeço-te por todos os *commits* e discussões sobre modelos e dados ómicos sem os quais o meu trabalho não seria tão fácil.

A todos os restantes membros do grupo BISBII com quem tive o prazer de trabalhar e socializar, agradeço o ambiente caloroso que foram proporcionando durante estes anos!

I would like to thank Dr. Julio Saez-Rodriguez for his supervision during the final year of my PhD. It was a pleasure working with him and his group in Heidelberg, even if the circumstances did not allow me to stay abroad for too long. I also thank Aurelien Dugourd for his insights which greatly helped shape up a very difficult part of my work. I also thank the rest of the group for their warm welcome and the environment they provide, which I was very fortunate to share.

Também ao Jorge, assim como à Cris e ao Óscar, agradeço imenso a vossa amizade, cervejas, desabafos, jantares, almoços vegetarianos e férias! Sem o vosso apoio e paciência para me aturar, não seria capaz de completar este capítulo da minha vida! Bora ao Luís às 18h30?

Ao Diogo agradeço a nossa longa amizade. Desde sempre pude contar contigo para me apoiares para falar sobre tudo!

Agradeço à HURNA por estarem lá no Discord ao fim de semana para descontrair na run anual da Penedus Mansion ou para disputar a Copa Fiat Uno online! Mesmo em 3 países diferentes estiveram sempre lá para me apoiar e ajudar-me a descontrair sempre que possível!

Finalmente, às pessoas mais importantes da minha vida: os meus pais. Agradeço à minha mãe por acompanhar de perto todo o meu percurso e por ser sempre o meu porto de abrigo e a pessoa em quem posso sempre confiar. Agradeço também ao meu pai, que abdicou de muito para que eu pudesse chegar a este ponto da minha vida. A ambos devo tudo, e por isso dedico este trabalho a eles.

A todos os amigos e família que não menciono aqui, obrigado também pelo vosso apoio, por mais breve que tenha sido.

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the Universidade do Minho.

____________________________, ____

(Place) (Date)

____

(José Vítor de Castro Vieira)

“We are at the very beginning of time for the human race. It is not unreasonable that we grapple with problems. But there are tens of thousands of years in the future. Our responsibility is to do what we can, learn what we can, improve the solutions, and pass them on.” (Richard Feynman)

Resumo

Abordagens Integrativas de Análise de Vias Para a Investigação e Desenvolvimento de Fármacos para Cancro

O cancro é um grupo altamente heterogéneo de doenças que constitui uma das principais causas de morte no mundo moderno. A complexidade envolvida nos mecanismos moleculares que induzem neoplasias suscita a necessidade de desenvolver métodos de análise de dados para os identificar e compreender. O número cada vez maior de dados ómicos e ferramentas computacionais para a sua análise expandiu o conhecimento sobre vários tipos de cancro. Os modelos metabólicos baseados em restrições são particularmente interessantes, apresentando-se como uma estrutura flexível para integração de dados ómicos, com várias aplicações comprovadas no estudo do cancro. No entanto, estes modelos estão restritos à representação do metabolismo, descartando processos de expressão, regulação e sinalização. Além disso, os métodos atuais de integração de dados ómicos carecem de um processo padronizado e unificado para o seu uso. Neste trabalho, serão apresentados dois métodos de contextualização de dados ómicos. Foi desenvolvido um processo para a reconstrução de modelos metabólicos contextualizados, integrando transcriptómica para extrair, de forma genérica, modelos para tecidos humanos. Seguidamente, foi concebida uma nova abordagem de representação de modelos e um método de previsão de fenótipos (ipFBA), estabelecendo uma plataforma capaz de representar vários tipos de redes biológicas no mesmo método. As duas abordagens fazem parte de plataformas de software modulares e abertas para uso e contribuições por parte da comunidade. A validação dos métodos foi realizada usando dados ómicos detalhados para a linha celular de cancro da mama MCF7, revelando o impacto da parametrização nas abordagens acima mencionadas, e estabelecendo uma base sólida para um caso de estudo mais alargado em que os métodos desenvolvidos foram usados para identificar aspectos importantes do metabolismo do cancro consistentes com a literatura. Os modelos contextualizados revelaram melhores previsões de genes essenciais quando comparados com trabalhos anteriores, enquanto que o método ipFBA melhorou significativamente as previsões de atividade de fluxo. Este método também foi usado para caracterizar diferenças entre tecido saudável e cancro renal com uma representação detalhada da interação entre fluxos, genes metabólicos e os seus reguladores.

Palavras-chave: Modelos metabólicos, análise de vias, metabolismo de cancro, integração de dados ómicos

Abstract

Integrative Pathway Analysis Approaches for Cancer Research and Drug Development

Cancer is a highly heterogeneous group of diseases that constitutes one of the leading causes of death in the modern world. The complexity involved in the molecular mechanisms that induce neoplasms elicits the need for data-driven approaches to identify and understand it. The increasingly large number of multi-omics datasets and *in silico* tools for their analysis, contributed positively with new insights on many cancer types. Constraint-based models of metabolism are particularly interesting as a flexible scaffold for omics data integration with several proven applications in cancer. However, the scope of constraint-based models is usually restricted to metabolism, discarding gene expression, regulation and signalling pathways. Furthermore, current methods for omics integration lack a standardised and unified pipeline for their usage. In this work, two methods for the contextualisation of multi-omics data are presented. Firstly, a pipeline for the reconstruction of context-specific metabolic models was developed, integrating transcriptomics data to extract models for human tissues. Using insights from this effort, a novel model representation approach was devised, complemented with a phenotype prediction method (ipFBA), providing a scaffold for multi-omics integration and representing various layers of biological networks in the same formalism. Both methods have been made available through the development of modular software frameworks that are open for usage and contributions from the community. Validation was performed using detailed MCF7 breast cancer cell line multi-omics measurements, revealing the impact of parametrisation, setting a solid basis for a larger case study with the aim of identifying critical aspects of cancer metabolism consistent with those reported in literature. Context-specific models revealed higher predictive accuracy for gene essentiality predictions than previous works, while ipFBA greatly improved flux activity predictions. The latter approach was also used to characterise differences in healthy and renal cancer patients, allowing a detailed visualisation of the interplay between fluxes, metabolic genes and their regulators.

Keywords: Metabolic modelling, pathway analysis, cancer metabolism, omics integration

Contents

List of Figures

List of Tables

Acronyms

AKG	α-ketoglutarate
ANOVA	analysis of variance
ATP	adenosine triphosphate

BN	Boolean network

CARNIVAL	CAusal Reasoning for Network identification using Integer VALue programming
CBM	constraint-based modelling
CCLE	Cancer Cell Line Encyclopedia
CoBAMP	Constraint Based Analysis of Metabolic Pathways
CORDA	Cost Optimization Reaction Dependency Assessment
COSMOS	Causal Oriented Search of Multi-Omic Space
CSMR	context-specific metabolic reconstruction

DHAP	dihydroxyacetone phosphate
DNF	disjunctive normal form

EC	extreme current
EFM	elementary flux mode
EFP	elementary flux pattern
EP	extreme pathway

ER	extreme ray
FBA	flux balance analysis
GENRE	genome-scale network reconstruction
GIMME	Gene Inactivity Moderated by Metabolism and Expression
gMCS	genetic minimal cut set
GPR	gene-protein-reaction
GRaSP	Gene Regulation and Signalling Pathways
GRN	Gene regulatory network
GSMM	genome-scale metabolic model
GUI	graphical user interface
HGNC	HUGO Gene Nomenclature Committee
HMG-CoA	3-Hydroxy-3-Methylglutaryl-CoA
HPA	Human Protein Atlas
ILP	integer linear programming
iMAT	Integrative Metabolic Analysis Tool
ipFBA	integrated parsimonious flux balance analysis
JSON	JavaScript Object Notation
KEGG	Kyoto Encyclopedia of Genes and Genomes
LP	linear programming
MAPK	mitogen-activated protein kinase
mCADRE	metabolic Context-specificity Assessed by Deterministic Reaction Evaluation
MCC	Matthews' correlation coefficient
MCS	minimal cut set

MGS	minimal generating set
MILP	mixed integer linear programming
MOMA	Minimization Of Metabolic Adjustment
mTOR	mechanistic target of rapamycin kinase
NCBI	National Center for Biotechnology Information
ODE	ordinary differential equation
optlang	Opt
PCA	principal component analysis
PEPCK	phosphoenolpyruvate carboxykinase
pFBA	parsimonious flux balance analysis
PFK	phosphofructokinase
PI3K	class I phosphoinositide-3' kinase
PKN	prior knowledge network
PPP	pentose phosphate pathway
PYK	pyruvate kinase
RAS	reaction activity score
RMF	required metabolic function
ROOM	Regulatory On/Off Minimization
SBML	Systems Biology Markup Language
SIF	Simple Interaction Format
SNP	single nucleotide polymorphism
TAS	transcript activity score
TF	transcription factor
tINIT	Task-driven Integrative Network Inference for Tissues
TNBC	triple negative breast cancer

TPM	transcripts per million
TRN	transcriptional regulatory network
TROPPO	Tissue-specific RecOnstruction and Phenotype Prediction using Omics data

1

Introduction

1.1 Context and motivation

Cancer is a pathological condition that is a currently a major limiting factor in human life expectancy. The alterations in cancer metabolism have been described for decades, particularly through the work of Otto Warburg, that first described the now called Warburg effect, which causes cancer cells to metabolize glucose through glycolysis rather than oxidative phosphorylation.

The key biological adaptations (or hallmarks) that confer cancer cells with the ability to cause and sustain the disease have been described by Hanahan and Weinberg [1], namely:

- Proliferation through activation of signalling pathways that control cell growth, survival and energy metabolism
- Evasion of mechanisms that suppress cell growth or trigger senescence.
- Avoidance of programmed cell death
- Unlimited replication
- Tumour growth sustenance through the promotion of blood vessel formation (angiogenesis)
- Activation of invasion and metastatic processes

The interaction of deregulated signalling and regulatory molecules elicits these adaptations and altered metabolic states, which has been hypothesized as a mechanism to improve cancer cell survival [2]. It is often hard and costly to perform experiments with the purpose of determining the root causes of these changes and find targets to block tumour growth, especially since each type of cancer presents different biochemical and histological findings.

In recent years, the increased availability of molecular biology techniques, along with developments in computational systems biology, has enabled the integration of information from multiple components of human cells into knowledge bases, that can then serve as primers for models with the intent of predicting cellular phenotypes under any experimental condition. These models allow for more rational approaches to analyse and design strategies towards a desired purpose.

Genome-scale metabolic models (GSMM) are a relevant example, with proven results in accurately predicting metabolic states, especially on cancer cells, which is achieved through the creation of constraint-based models that can be used with Flux Balance Analysis (FBA) [3]. Additionally, the simulation and reconstruction of specific models for certain tissues can be achieved with a wide variety of algorithms that integrate omics data and evidence from literature [4–6].

FBA-based analysis requires the specification of a cellular objective, something which is currently ill-defined and controversial for normal human cells. Metabolic pathway analysis methods can provide much clearer answers on the theoretical capabilities of a cell, and have recently been employed in GSMMs, due to the presentation of efficient approaches [7, 8] that can handle the ever increasing scale of these models.

A major drawback of the constraint-based modelling approach is that it typically does not contain the regulatory and signalling layers. As mentioned earlier, this is a very crucial part of cancer metabolism control. There have been some notable attempts to integrate these behaviours, mostly on micro-organisms with models of smaller scale and complexity. Additionally, there is limited integration of pathway analysis methods with these networks, and also limited usage in human studies.

In this work, the development of an integrated metabolic pathway analysis framework for the discovery of drug targets for cancer will be addressed. The work will include the creation of context-specific models of cancer cells, the development of metabolic pathway analysis tools that can provide important answers on human metabolism and enhance the model creation process. A second phase will focus on the adaptation of the developed methods to integrate regulatory and signalling layers. Ultimately, these methods will be leveraged to obtain insights on the metabolism of cancer cells, providing a phenotype prediction platform with possible applications in personalised medicine.

1.2 Research objectives

The end goal of this work is to use metabolic pathway analysis methods in genome-scale metabolic models of human tissues integrated with regulatory and signalling networks to enhance the analysis of human metabolism, with emphasis on gaining relevant biological insights on cancer cells. This encompasses the development of a generic framework for the improvement of existing *in silico* context-specific model creation approaches with a standardised workflow for omics data integration as well as novel algorithms to represent signalling and gene regulatory interactions in constraint-based models. The software developed throughout this work is to be made available to the scientific community with significant ease of access and use, to provide a basis from which more *in silico* studies of human metabolism can be performed. The ultimate research goal with this effort is to develop a set of tools capable of generating richer biological insights into the metabolic reprogramming that occurs in cancer cells, and demonstrate the applicability of such tools in personalised medicine.

The main objectives of this work are:

- To develop a set of software tools implementing constraint-based methods and pathway analysis algorithms in genome-scale metabolic models;
- To develop a framework for omics data representation, processing and usage with context-specific model reconstruction algorithms as well as model validation;
- To assemble a scalable pipeline for context-specific model reconstruction and validation from transcriptomics data;
- To extend constraint-based models with a new model representation capable of directly integrating gene regulatory and signalling networks;
- To develop a phenotype simulation method to successfully predict active fluxes and metabolic regulation;
- To apply the developed software and extract relevant knowledge on the metabolic heterogeneity and patterns identified in cancer cells using omics data.

1.3 Thesis outline

This document is structured in 6 chapters.

The current chapter provides a brief overview of the context surrounding this work and motivation behind the approaches and results presented in the following chapters. Furthermore, the research objectives and outline of this thesis are also briefly discussed.

Chapter 2 provides a solid background for concepts and subjects discussed throughout the work. Firstly, an overview of genome-scale metabolic models is presented with a heavy focus on medical applications in humans. Concepts, mathematical formalisms and tools associated with the constraint-based modelling framework are also discussed, as well as omics integration methods based on them. Finally, gene regulatory and signalling networks are also presented, as well as methods that allow their integration with metabolism.

In Chapter 3, the structure and implementation of software packages developed within the scope of this thesis are described in detail. These software packages provide support for constraint-based modelling tools for context-specific model reconstruction, pathway analysis and integrative modelling.

In Chapter 4, a pipeline for the reconstruction and validation of multiple cancer cell line models is presented. The implemented software resources are leveraged to devise a set of routines capable of reconstructing cancer models to predict phenotypes that are validated using gene essentiality and fluxomics data. The heterogeneity of cancer metabolism is explored using these models, first by identifying key metabolic phenotypes associated with breast cancer subtypes and attempting to predict a cell line's primary disease with predicted fluxes.

Chapter 5 details a novel approach to extend constraint-based models with gene expression and regulatory interactions as well as a new simulation method that leverages this representation, enabling integration of multi-omics datasets to predict metabolic fluxes. The approach is validated using a large-scale data set with pan and breast cancer case studies as well as a smaller cohort of renal cancer patients, while demonstrating its improved ability of the method to predict phenotypes.

In Chapter 6, a brief summary of the work detailed in this thesis is presented, along with a critical analysis of the outcomes, implications of this work in the community as well as ideas for improvement as part of future work.

2
Background

Systems biology is a very recent field that introduced a paradigm shift in the biological sciences by dealing with biological systems as a whole, rather than only peering at the sum of its individual parts. The field's advances concerning the reconstruction of genome-scale models of metabolism allows *in silico* phenotype prediction, analysis and design, while including multiple layers of omics information. This chapter focuses on constraint-based models, a widely used approach that allows the integration of whole-genome sequencing data in genome-scale reconstructions, yielding models that include the entire metabolic machinery of a given cell. The underlying principles of this approach will be discussed, as well as analysis methods and methodologies to integrate omics data with simulations.

2.1 Genome-scale metabolic models

2.1.1 Systems biology

Systems biology is a relatively new field that attempts to look at a biological system by combining comprehensive information regarding its entirety. Until the 20th century, biology was dominated by a reductionist approach with increasing amounts of collected data about individual components and processes [9, 10]. As molecular biology progressed towards high-throughput techniques for nucleotide sequencing and expression measurements, emphasis shifted towards an integrative approach, using the large amounts of generated information to build models that could help understand, analyse and eventually predict the phenomena that occur in living organisms [9]. These system-level approaches allow us to study, not only the individual parts of a biological system, but also their interactions. *In silico* approaches are essential to build these mathematical models, and advances in the field of computer science have allowed for more complex models to be used [11].

System-level approaches can provide insights on four critical aspects of a biological system [10]:

- **Structure**: The various components of the system, such as genes, proteins, biochemical reactions and metabolites, as well as the interaction between them.
- **Dynamics**: Variation of the system over a period of time relative to parameters, such as enzyme kinetics and gene expression.
- **Control**: Regulatory mechanisms that can alter the state of the organism and can potentially be exploited to achieve a desired phenotype.
- **Design**: Strategies to manipulate cell phenotypes into achieving desired properties. These are based on simulations rather than random hypothesis testing.

The focus of this work will be on system-wide models of metabolism, specifically for humans. These models attempt to predict phenotypes based on the enzymatic content of the cell, with more advanced methods exploiting other layers to enhance these predictions. There are currently two main types of metabolic models depending on how they represent time, namely:

- Kinetic (or dynamic) models: Time is an independent variable which affects enzyme properties, leading to variable intracellular metabolite concentration. Information regarding enzyme kinetics is needed, often through a rate law and additional parameters.
- Constraint-based models: This approach discards the time component by assuming metabolite concentrations remain stable (pseudo steady-state). The main layer included in this type of model is the structure of the biochemical reaction network, namely, the stoichiometry.

Kinetic models are usually restricted by the amount of available information concerning enzyme dynamics. The need for fine-tuning of enzyme parameters also hinders the use of this modelling approach and increases the computational demand when predicting phenotypes. For the reasons mentioned above, constraint-based models appear as a less demanding approach towards modelling metabolism at the genome-scale. The concepts underlying these models will be discussed in detail throughout the next section.

Table 1: Main data sources for various omics data types with experimental measurements available for humans. The marks represent whether a given database contains a type of omics data, shown in the table with letters: **G**enomics, **T**ranscriptomics, **P**roteomics, **M**etabolomics, **F**luxomics

Database	***G***	***T***	***P***	***M***	***F***	**Reference**
European Nucleotide Archive	•					[12]
DNA Data Bank of Japan	•					[13]
GenBank	•					[14]
Genomic Data Commons	•	•				[15]
Human Protein Atlas		•	•			[16]
Genotype-Tissue Expression		•				[17]
Functional Annotation of Mammalian Genomes		•				[18]
ArrayExpress		•				[19]
Gene Expression Omnibus		•				[20]
RNA-Seq Atlas		•				[21]
Human Proteome Map			•			[22]
Human Protein Reference Database			•			[23]
Human Metabolome Database				•		[24]
Metabolomics Workbench				•		[25]
MetaboLights				•		[26]
Central Carbon Metabolic Flux Database					•	[27]

2.1.2 Omics data

High-throughput molecular biology techniques currently generate large amounts of quantitative and qualitative information about various molecules present in living organisms. The suffix has been commonly used to describe each individual field of study concerned with gathering large amounts of data from one type of biological molecule. Five main types can be identified and this section will provide a brief description of each along with relevant experimental methods and data sources.

Genomics This layer encompasses genomes with annotated sub-sequences that represent genes. The Human Genome Project, a worldwide effort to sequence the entire human genome, concluded in 2003 costing heavily in both time and resources. In present times, next-generation sequencing, as opposed to Sanger-based technologies, yields whole-genome sequences at much lower expense, which has greatly contributed to the large amount of publicly available genomes present in databases such as GenBank, EMBL's European Nucleotide Archive and the DNA Data Bank of Japan. Genomic variation is also an important object of study, given the importance of copy number variation and loss of function mutations in oncogenesis.

Transcriptomics Messenger RNA (mRNA) quantification is usually performed through DNA microarrays or RNA-sequencing, a critical step in providing a gene expression profile for a given cell. This is especially important in organisms with multiple types of differentiated cells (such as humans) in which proteins can be expressed with different rates and alternative splicing can occur, leading to multiple protein isoforms. Due to the large amount of information, studies on human cells have been published on a multitude of databases such as ArrayExpress, Human Protein Atlas, as well as others that are highlighted on Table 1.

Proteomics Protein content can be quantified and identified with mass spectrometry or western blotting. This level of information is useful in determining protein function after post-translational modification and to truly assert the expression of a protein, since mRNA quantification may not correlate with it. The availability and coverage of these datasets is usually smaller than transcriptomics, although it is, theoretically, a better measurement of the presence of a given enzyme, and thus, an important data source for integration of omics data in genome-scale metabolic models.

Metabolomics Metabolomics deals with molecules present in the external media and within cells that are subject to biochemical conversions originating cell metabolism, comprising a wide variety of lipids, amino acids, carbohydrates and other small organic molecules. Gas and liquid chromatography, mass spectrometry and nuclear magnetic resonance spectroscopy can be used for detection, separation and quantification of metabolites.

Fluxomics Fluxomics attempts to determine the rates at which metabolic conversions occur. Although closely related to the metabolome, fluxomics provides knowledge on the kinetic properties of a biochemical reaction system rather than presenting a single snapshot (as is the case of metabolomics). Fluxome experimental data is mostly obtained using ^{13}C-marked substrates that are subsequently measured with mass spectrometry.

There are currently a number of databases providing experimental data on various omics layers in humans, as shown on Table 1, but there is a significant challenge in creating methods to integrate different layers of data to generate useful predictive models. In this report, the focus lies on the usage of genome-scale metabolic models as a means of combining multiple layers of information.

2.1.3 Genome-scale metabolic reconstructions

A genome-scale metabolic model (GSMM) is a mathematical construct that can be used to predict phenotypes from whole-genotypes of an organism. A single genome-scale network reconstruction (GENRE) is the first step towards building a GSMM. A GENRE is a collection of the biochemical transformations occurring in the cell (metabolic network) that have been determined through functional genome annotation of the organism, as well as literature review (or bibliome). Although there are highly-detailed protocols for creating GENREs as well as tools that assist or automate certain processes, the basic reconstruction pipeline can be summarized in three important steps:

- **Draft reconstruction**: The genome of the organism is annotated so that metabolic genes can be identified. The set of biochemical reactions can then be identified by matching the metabolic gene products with Enzyme Commission (EC) numbers, which can then be matched to reactions (using databases such as Kyoto Encyclopedia of Genes and Genomes (KEGG) or Brenda).

- **Curation and refinement**: The aim of this stage is to refine the reconstruction through inclusion of more organism-specific data and validation of each individual entry. Typical steps include the addition of formulae, reaction stoichiometry and thermodynamic constraints, gene-protein-reaction relationships and the estimated biomass composition for the target organism.

- **Model creation**: This stage is mostly automated and concerns the creation of mathematical structures that can be used to predict phenotypes. In constraint-based models, this step mostly consists on the creation of the mathematical formalisms representing the properties curated in the previous step.

- **Validation**: This step includes tests to ensure the model represents the target organism in the best way possible. Metabolomics and fluxomics data are crucial for this step, as they provide the metabolite profile and reaction rate values to assess the model's predictive accuracy. Gaps between pathways are also identified, as well as metabolic dead-ends. The model is also tested for biomass production, and if it can achieve that under standard environmental conditions.

The process of building a GENRE is not confined to a single iteration, and it is expected to repeat the manual reconstruction steps multiple times until the final model's performance is satisfactory.

2.1.4 Human metabolic models

Human metabolic models have been reconstructed for a variety of purposes in the past two decades. Early attempts at representing human metabolism through models only included limited parts of the human metabolism. Wiback et al. employed a small metabolic model to extract the metabolic functions of a human red blood cell through usage of extreme pathways, predicting similar results to previous kinetic models of this cell [28]. The results of this analysis were further used to assess the regulatory demands for the obtained metabolic functions [29].

Vo et al. also presented a small-scale human mitochondrial metabolic network characterizing maximum ATP yields and available metabolic functions that match the available literature [30]. Further contributions include a model of human hepatoma cells and subsequent analysis of intracellular fluxes and growth under different combinations of amino acids in the growth medium [31].

A few years after the end of the Human Genome Project, the availability of the entire human genome led to the creation of the first genome-scale human metabolic models. There are currently four main models that represent the entire metabolic landscape of human cells and can be defined as generic. The most relevant of these are summarized on Table 2, as well as case studies highlighting their application. Recon1 [32] appeared in 2007 as the first of these models shortly followed by the introduction of the Edinburgh Human Metabolic Network (EHMN) [33]. These two efforts formed the basis for more recent reconstructions, namely, the HMR database [34, 35] and Recon2 [36].

Recon2 has had some revisions that have slightly diverged over the past few years. Thiele and colleagues have released a major Recon2 update in 2015 (Recon2.04), correcting the GPR associations to comply with a more recent build of the human genome.

Another branch of revisions begins with Recon2.1, which was able to ensure complete carbon and elemental balancing of the various metabolites at the cost of drastically increasing model size [37]. Further developments on this revision led to Recon2.2 which increases prediction accuracy of ATP yields under various carbon sources [38]. Despite the various revisions, each one includes improvements from their predecessors, highlighting the importance of community efforts in building GENREs that lead to more accurate GSMMs.

The Recon2 and the HMR2 database were used by Brunk and co-workers as a basis for the Recon3D model reconstruction [39], further extending human GSMMs to integrate single nucleotide polymorphism (SNP)s. Thiele et al. have used this model as a basis for building an innovative whole-body metabolic model capable of representing a wide variety of human tissues and associated microbial communities [40], as

Table 2: Summary of the presently available genome-scale metabolic models for humans. Only generic models are considered in this table.

Name	Genes	Reactions	Description	References
Recon1	1496	3743	First genome-scale metabolic model of a generic human cell. Integrates gene information from KEGG and National Center for Biotechnology Information (NCBI). Validated using flux balance analysis to predict 288 reviewed metabolic functions	[32]
Edinburgh (EHMN)	2322	2823	Network combining literature data from a proprietary database (EMP) and enzyme annotation from KEGG, UniProt and HGNC	[33]
HMR database	1512	5535	Generic model derived from a database containing the Recon 1 and EHMN networks as well as information from HumanCyc and KEGG (*i*Human1502).	[34, 35]
Recon2	1789	5063	Revised reconstruction derived from Recon1 as well as inputs from HepatoNet1, EHMN and others. 9 new pathways were added and 62 others were extended. Subject to many revisions and extensions from other groups.	[36]
Recon3	3288	13543	Derived from Recon2 and HMR2.0 models. Provides information for modelling host-microbe interactions and dietary compound metabolism.	[39]
Human1	3628	13520	Includes parts of Recon3D, HMR2.0 and many other components from all previous reconstruction efforts.	[41]

well as differentiating between male and female metabolism (Harvey and Harvetta models, respectively).

The most recent human GSMM stems from the efforts of Robinson et al. who have presented the Human1 reconstruction [41]. This model appears to provide the highest predictive accuracy for gene essentiality and phenotype predictions when compared with its predecessors from which a considerable amount of information was integrated. Furthermore, this consensus reconstruction is available in a repository with version control, which enables greater transparency in the reconstruction process, as well as inputs from the community which are constantly integrated in the model.

Since humans are multicellular organisms, the generic model is not capable of accurately predicting the metabolism of a differentiated cell, whose enzymatic content may be drastically different. To tackle this problem, context-specific models can be created, either manually (such as HepatoNet1 [42]) or in an automated way, such that their enzymatic content matches that of the cell to be studied. The concepts pertaining the methods used to build these models will be explained further in the context-specific model reconstruction section.

2.2 Constraint-based modelling

Constraint-based modelling is currently the most adequate tool to simulate metabolism over the entire genome of a living organism. Despite being a relatively recent approach, first presented just over two decades ago, it has proven itself useful in predicting phenotype from genotypes and currently features a wide array of analysis and design methods for numerous applications. The key principles and methods surrounding constraint-based modelling (CBM) and their integration with omics data will be covered in the following sections.

2.2.1 Principles of constraint-based modelling

2.2.1.1 Main assumptions

CBMs are based on a few assumptions that distinguish them from kinetic models, namely:

1. Biochemical reactions are considered instantaneous;
2. Time is not considered in the system (invariant);
3. Reaction rates and metabolite concentrations do not vary;
4. Each reaction must consume the same amount of mass than what it produces (mass balance).

Considering the three assumptions above, the variation of metabolite concentrations (c) over time (t) can be expressed through Equation 2.1 [43].

$$\frac{dc(t)}{dt} = 0 \tag{2.1}$$

Equation 2.2 can be derived by discarding enzyme kinetics and assuming metabolite concentrations vary according to the reaction rates (on vector v) that produce or consume a given metabolite, each multiplied by its corresponding stoichiometric coefficient present in the stoichiometric matrix (S).

$$S \cdot v = 0 \tag{2.2}$$

Considering the rows and columns in S represent, respectively, equations and variables, this leads to a linear system of equations that is underdetermined for most realistic cases. This means that there are more variables (n) than equations (m). As such, there are multiple valid solutions to this problem. A valid solution for Equation 2.2 (or flux distribution) is any point in the multidimensional space described by a vector $v \in \Re^n$ that satisfies the equations in S.

2.2.1.2 Model components

The basic principles of constraint-based modelling have been defined and thus, it is relevant to further highlight the main components of a CBM. In the previous section, the reconstruction process was presented as incorporating experimental data from various layers of information. An overview of these components is represented on Figure 1. Although biologically relevant and typically present in many GSMMs, gene associations are not strictly required for core constraint-based methods, since the base formalisms merely require a set of metabolites connected by reactions (defined in the stoichiometric matrix).

A key consideration when dealing with CBMs is that some reactions are not associated with enzymes. This is due to the presence of spontaneous reactions that do not need a catalyst and complex cellular processes that do not have an explicit enzyme association and are thus represented as pseudo-reactions. These are useful to model mechanisms that are otherwise too complex to depict in CBMs. Cell growth, as an example, depends on many processes that fall beyond the scope of CBMs and is, thus, represented as a reaction that converts metabolites required for growth and produces an fictional entity (biomass). Figure 2 represents an example of a small metabolic network extracted from the work of Klamt et al. [44], which will be used further in this chapter as a practical example.

Stoichiometric matrix The S stoichiometric matrix of size m-by-n, represents the reactions in the system and their stoichiometry. Each row i and column j represent, respectively, one of the m metabolites and n reactions and each value S_{ij} represents the stoichiometric coefficient for metabolite i in reaction j. The stoichiometric matrix based on the network depicted on Figure 2 is depicted on Figure 3.

Capacities Reaction rates can be constrained to represent reversibility/directionality or specific capacities from fluxomics data. To this end, linear inequalities are added to the model defining a range for each

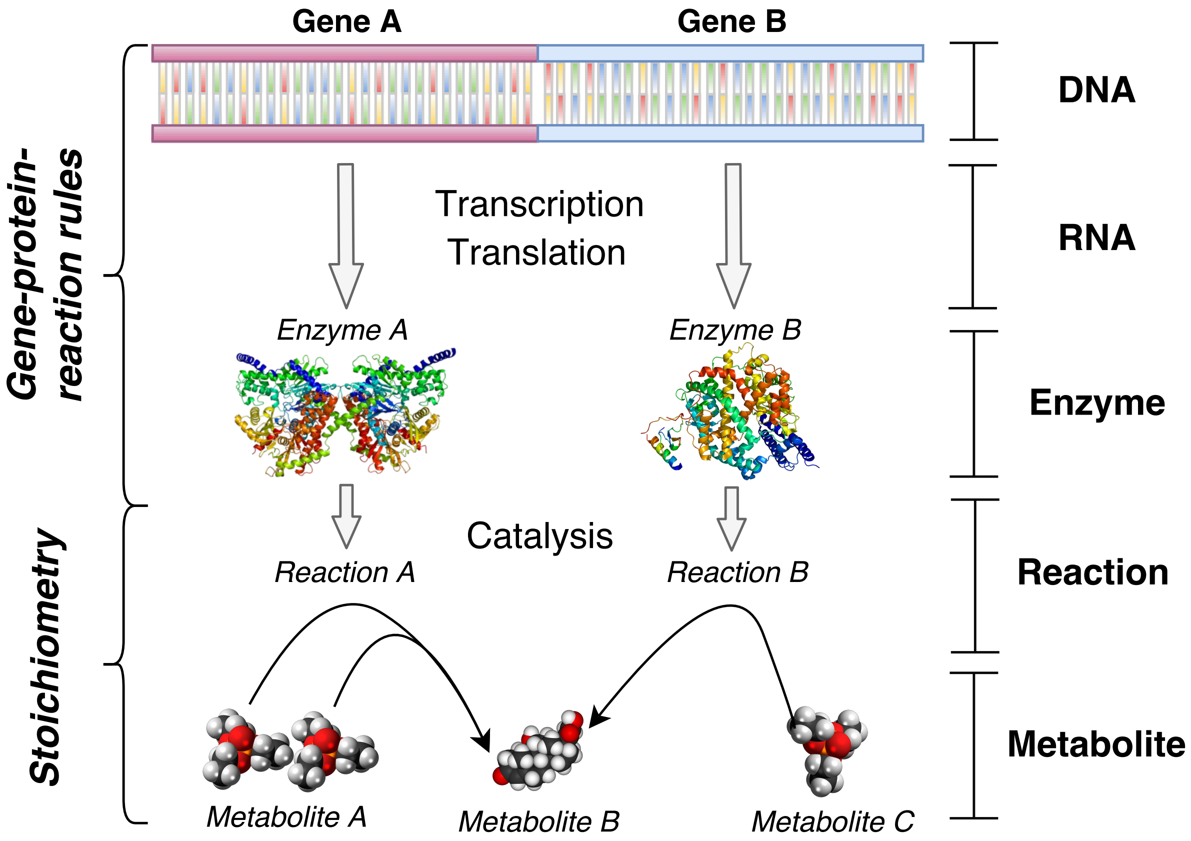


Figure 1: Overview of the five main layers included in a genome-scale metabolic model. Functional genomics (DNA), transcriptomics (RNA) and proteomics (enzymes) data are used to derive gene-protein-reaction rules and identify the enzyme content of the cell. With this information, the set of metabolites can be inferred by looking at the reactions catalysed by enzymes. The model is then used to predict the fluxome (reaction rates) of the cell and the metabolome (metabolites) to some extent.

flux. For any given flux v_i, a maximum u_i and minimum l_i value are set (Equation 2.3). If $l_i \geq 0$, reaction i is irreversible and directed forward. If $l_i < 0$ and $u_i > 0$, the reaction can occur on both directions (reversible). Flux capacities are particularly important for drain reactions (explained below) as this component is required to set uptake rates or to allow production of any given metabolite. Figure 3 provides a visual representation of the linear system including the steady-state assumption and flux capacities for the network depicted on Figure 2.

$$l_i \leq v_i \leq u_i \tag{2.3}$$

Model boundaries A fully mass-balanced CBM would be a closed system and thus, all reactions would have a set of products synthesized from a set of reactants. The steady-state assumption implies that the only feasible flux distribution would be that in which all fluxes are null. To allow exchanges in an otherwise

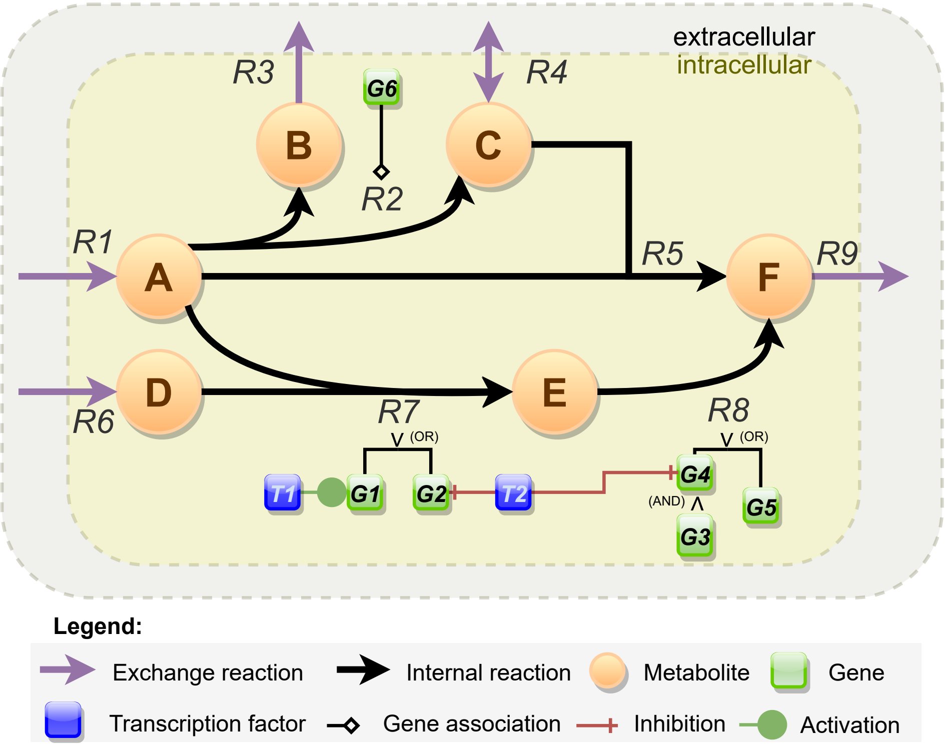


Figure 2: Toy metabolic network adapted from the work of Klamt et al. [44], with arrows representing reactions (with defined directionality) and circles representing metabolites. Reactions $R1$,$R3$,$R4$,$R6$,$R9$ are exchange reactions (depicted in purple) due to the unbalanced uptake/secretion of products. Gene-protein-reaction rules are represented through genes (green squares) connected with Boolean operators. Further information can also be associated with enzyme-coding genes such as interactions involving transcription factors (blue squares).

closed system, some metabolites are imported or eliminated at no cost through **exchange reactions**. In the network depicted on Figure 2, reactions $R1$, $R3$, $R4$, $R6$ and $R9$ are exchange reactions, which either produce a metabolite without any reactants or consume them without generating any product. In a realistic setting, exchange reactions are required to mimic the availability of chemical compounds in the extracellular media or to allow their secretion. Through manipulation of these reactions' flux capacities, environmental conditions and growth media formulations can be integrated into these models to generate meaningful predictions.

Gene-protein-reaction association A GSMM may include annotations that identify the relationship between genes encoding metabolic enzymes for a given reaction. GPR reaction rules encode this information as a Boolean expression using conjunction (AND) and disjunction (OR) operators. When expressed in disjunctive normal form (DNF), such rules can be generalized as a collection of AND operations between several genes, which then become operands of an OR operation. In this form, each AND expression can

Stoichiometric matrix

$$\underbrace{\begin{array}{c|ccccccccc} & R1 & R2 & R3 & R4 & R5 & R6 & R7 & R8 & R9 \\ \hline A & 1 & -1 & 0 & 0 & -1 & 0 & -1 & 0 & 0 \\ B & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ C & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ D & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ E & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ F & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -1 \end{array}}_{S} \times \underbrace{\begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \\ v_8 \\ v_9 \end{bmatrix}}_{v} = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}}_{0}$$

Figure 3: Representation of the basic structure of the metabolic model for the toy network in Figure 2, as defined in Equation 2.2 (left side) which defines the steady-state constraints.

be considered as a single isozyme requiring one or more expressed genes to be synthesized and the OR expression groups all of these isozymes, denoting they are capable of catalysing the reaction to which the GPR rule belongs to. These expressions are often used to enable the simulation of mutants, since GPR rules can be evaluated to determine, for a given expression state, whether there is a valid combination of genes that catalyse an enzyme needed for a given reaction.

2.2.2 Constraint-based phenotype prediction

The first group of methods to be reviewed concerns the prediction of reaction rates using CBMs. Three well known methods will be presented and described in this section.

CBMs are usually undetermined systems and thus, numerical optimization methods must be used to find one or more solutions. This occurs since the system defined through Equations 2.2 and 2.3 represents a large solution space from which infinite points would solve it. A guided search towards the solution is required since the vast majority of the solution space might not be biologically relevant.

A linear programming (LP) optimization problem can be formulated using the aforementioned equations as constraints. Given any objective, LP solvers attempt to find a single solution in the entire space that maximizes or minimizes a given objective. In CBMs, one can artificially create a reaction representing such an objective and define the LP as its maximization, subject to Equations 2.2 and 2.3, leading to a problem similar to the following,:

$$\begin{aligned} \underset{v}{\text{maximize}} \quad & Z(v) = c^T \cdot v \\ \text{subject to} \quad & S \cdot v = 0 \\ & l_i \leq v_i \leq u_i, \forall i \in \{1, \ldots, n\} \\ & c, v, l, u \in \Re^n, \; S \in \Re^{m,n} \end{aligned} \tag{2.4}$$

where Z is the objective function, defined as a linear combination of c and the v vector of variables (the flux vector in Equation 2.2); c specifies the weight of each variable towards the objective function; l and u are, respectively, the lower and upper bound vectors.

2.2.2.1 Flux balance analysis

The most used approach allowing phenotype prediction with CBMs is flux balance analysis (FBA), which has been used extensively [45], since it was presented by Varma and Palsson [3].

FBA essentially solves the system in Equation 2.4 and attempts to maximize a relevant flux. In the original publication, the biomass pseudo-reaction is maximized, yielding a single flux distribution that is valid for maximum cell growth. In stress-free conditions, this is a generally valid assumption for microbes. The basic principles behind the constraints used for FBA and many of its derivative methods are represented on Figure 4.

Due to the existence of alternative optima that allow maximum cell growth, FBA typically yields a space of possible solutions, rather than a single flux distribution. parsimonious flux balance analysis (pFBA) attempts to solve this issue by assuming enzyme usage in cells is minimal. Mathematically, this requires an initial optimization towards the desired objective and subsequent minimization of the sum of all fluxes in the cell [46]. This further constrains the possible optima.

Predicting phenotypes in mutants holds an additional challenge since the growth maximization assumption may not hold true. Some derivatives of FBA have tried to tackle this issue by assuming the cell is optimized to try and maintain its previous state, rather than optimizing for a new growth optimum.

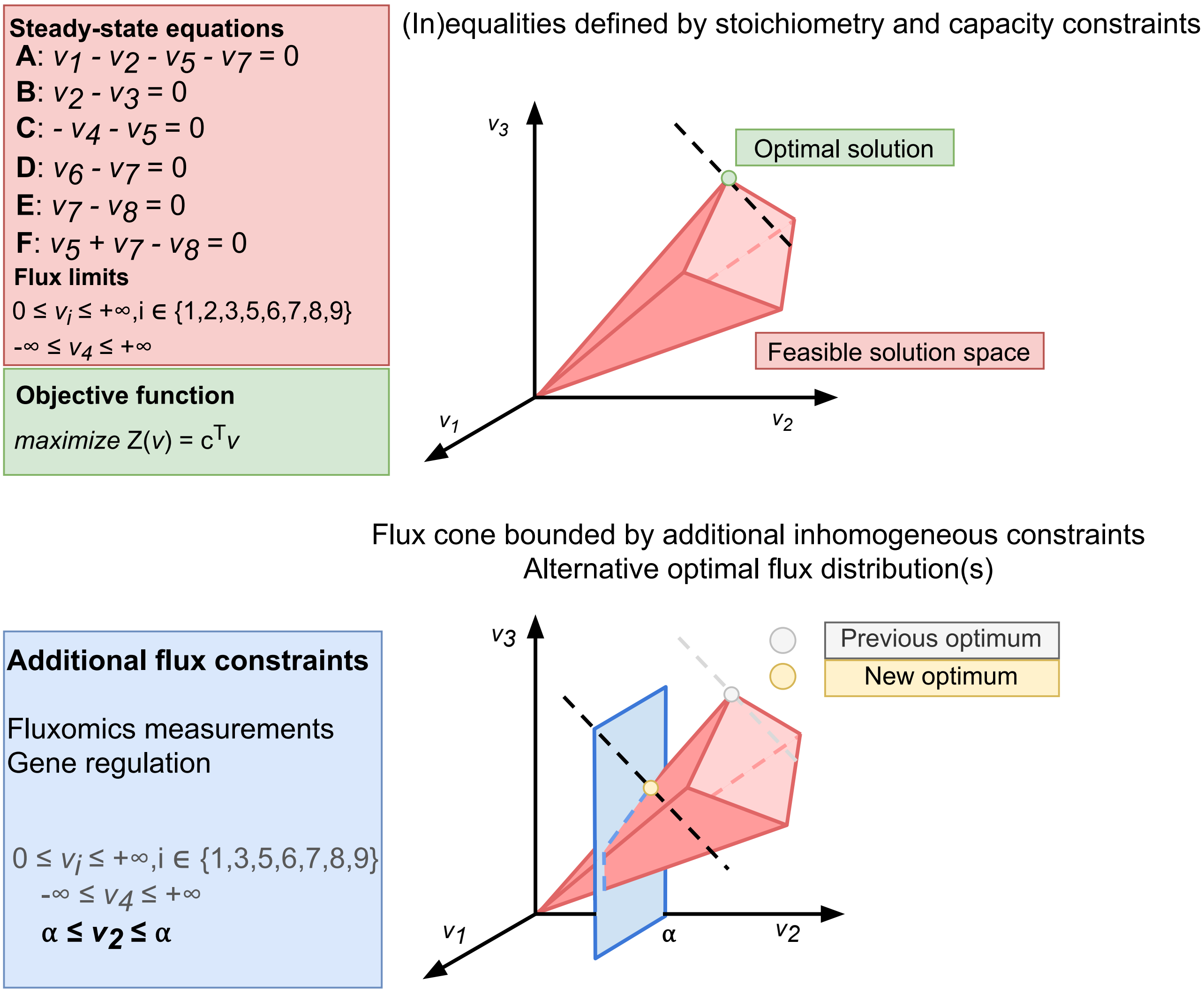
Minimization Of Metabolic Adjustment (MOMA) introduces a quadratic programming problem maximizing the similarity between the simulated fluxes and reference flux distribution [47]. Regulatory On/Off Minimization (ROOM) uses a mixed integer linear programming (MILP) problem to formulate a similar problem, but the amount of significantly altered reactions is minimized rather than globally maximizing the similarity between simulated and reference fluxes [48].

Applications FBA and its derivatives have been used extensively for a wide array of applications [45], with growing interest in human studies, even though defining a cellular objective for a differentiated human cell is not trivial. Cancer studies, on the other hand, are more suitable for FBA since one of the hallmarks of the disease is uncontrolled growth, which can be modelled as the maximization of the biomass pseudo-reaction. Shlomi and colleagues used FBA to determine the causes of the Warburg effect that allows cancer cells to increase their growth rates [49]. Recently, Sahoo et al. used FBA in an effort to model the effect of 18 major drug compounds and found interesting associations between dietary patterns and drug metabolism, as well as novel mechanisms for interactions between statins and cyclosporine [50].

2.2.3 Pathway analysis

Constraint-based methods can be used to analyse the functions present in a metabolic model. Despite the usefulness of phenotype prediction methods such as FBA, these only display a single optimal metabolic state. Pathway analysis methods, on the other hand, can determine which functional units are present in the model, relying on unbiased methods that yield the theoretical limits of the entire module [51]. Such methods can be based on **nullspace** or **convex** analysis.

Nullspace analysis The nullspace of a CBM is defined as a matrix K satisfying $S \cdot K = 0$. The n columns of K contain the vectors (flux distributions) which fully describe the solution space of the CBM, where n is determined by the nullity of S, defined as $Nullity(S) = n - rank(S)$ [51]. These columns cannot be obtained through a linear combination of other columns in K, but any feasible flux distribution can be generated through linear combinations of the columns in K. Nullspace analysis reveals important aspects of a CBM, particularly for model reduction, such as identifying reactions unable to carry non-zero flux (blocked reactions) and sets of reactions whose flux is correlated (enzyme subsets) [52, 53]. These analyses, however, do not consider capacity constraints, revealing some limitations when considering the biological feasibility of their results.



Steady-state equations

A: $v_1 - v_2 - v_5 - v_7 = 0$

B: $v_2 - v_3 = 0$

C: $- v_4 - v_5 = 0$

D: $v_6 - v_7 = 0$

E: $v_7 - v_8 = 0$

F: $v_5 + v_7 - v_8 = 0$

Flux limits

$0 \leq v_i \leq +\infty, i \in \{1,2,3,5,6,7,8,9\}$

$-\infty \leq v_4 \leq +\infty$

Objective function

maximize $Z(v) = c^T v$

(In)equalities defined by stoichiometry and capacity constraints

Flux cone bounded by additional inhomogeneous constraints
Alternative optimal flux distribution(s)

Additional flux constraints

Fluxomics measurements
Gene regulation

$0 \leq v_i \leq +\infty, i \in \{1,3,5,6,7,8,9\}$

$-\infty \leq v_4 \leq +\infty$

$\boldsymbol{\alpha \leq v_2 \leq \alpha}$

Figure 4: Representation of the solution space defined using steady-state, stoichiometry, capacity and arbitrary flux constraints in a constraint-based model. The solution space in the top part is constrained by the steady-state equations derived from reaction stoichiometry. Phenotype prediction methods usually attempt to find optimal solutions lying in the vertices of the flux cone. Additional flux constraints can reduce this space and potentially change the model's optima.

Convex analysis The solution space of a CBM with n reactions can be represented in an n-dimensional space as a flux cone P [51]. The defining vectors of P, or extreme ray (ER)s are the essential pathways describing the CBM. State of the art methods for convex analysis of CBM solution spaces are commonly referred to as being elementary flux mode (EFM) analysis methods. The methods based on ERs and leading up to EFM analysis will be explored in the following sections.

2.2.3.1 Convex analysis approaches

The extreme ray approach towards analysing CBMs depends upon its geometrical representation in an n-dimensional space. For all subsequent methods, consider the $m \times n$ stoichiometric matrix S, the flux vector $v \in \Re^n$, D as a diagonal matrix where $D_{ii} = 1, \forall i \in Irrev$ and $Irrev$ as the set of indices for irreversible reactions. Equation 2.5 defines the flux space P containing all valid flux distributions

$$P = \{v \in \Re^n : S \cdot v = 0 \ ; \ D \cdot v \geq 0\} \tag{2.5}$$

If one assumes all reactions in the model to be irreversible in the forward sense ($v_i \geq 0 \ \forall i \in \{1, ..., n\}$), P is said to be a pointed convex polyhedral cone. This cone is characterized by a set of generating vectors that fully define it where each vector is an ER. ERs are feasible flux vectors exhibiting interesting properties that can be exploited for CBM analysis [54]:

- **Generators**: They fully describe P and as such, any flux distribution v is a combination of weighted ERs;
- **Minimal**: ERs are the smallest set of vectors that generate P;
- **Non-decomposable**: No ER of P can be represented as a combination of other ERs in the same set.

These properties translate to a concept of essential pathway sets that attempt to fully represent a cell's metabolism. Additionally, all possible phenotypes are contained within this definition, leading to unbiased analysis and design methods that are not dependent on any objective. Despite this, ER enumeration requires all reactions to be irreversible for the three properties to remain true.

A possible solution is to calculate ERs in an augmented flux space P' that can represent P as a pointed cone, as shown on Equation 2.6. Two early approaches for network-based pathways have been presented using different methods to build P'.

$$P' = \{v' \in \Re^{n+r}_{0+} : S' \cdot v' = 0 \; ; \; I \cdot v' \geq 0\} \tag{2.6}$$

These transformations can involve several layers to include various CBM components. The concept of extreme current (EC) first featured this through **decoupling of reversible reactions** [55]. This results in an augmented solution space with a forward and reverse reaction (with non-negative flux) for each reversible reaction in P. This decoupling, however, leads to a large number of possible ERs and pathways containing spurious fluxes in which both split reactions mapped to a single reversible reaction carry flux. The concept of extreme pathway (EP) further extends this by separating exchange and internal fluxes and decoupling reversible reactions in the latter group [56].

Given the shortcomings described above, EFMs were presented as a possible unifying definition of a biologically relevant pathway [57–59]. Similarly to ECs and EPs, EFMs attempt to fully define the flux space, but introduce the concept of support for flux vectors. The support of any given flux vector v, *supp(v)*, is a vector containing the indices of active reactions in v. Two conditions are required for a flux vector e_i , $\forall i \in \{1, ..., k\}$ to be considered a part of the set of EFMs, E:

- e_i must be contained within P and any flux vector $v \in P$ can be described as a non-negative linear combination of the elements in E.

- There can be no feasible flux vector v where $supp(v) \subset supp(e_i)$. This ensures e_i cannot be decomposed and remain admissible in P.

Any flux distribution can be obtained by carrying a weighted sum of EFMs and together with non-decomposability, the entire network can be defined by them. A key advantage over other concepts is the fact that fluxes from various EFMs cannot cancel each other out (non-cancellation), leading to a more intuitive analysis. EFMs have some shortcomings, particularly when it comes to the amount of modes that are generated. As a result of the non-cancellation condition imposed on EFMs, the amount of enumerated pathways is much higher, and indeed it has been shown that for metabolic networks, the EPs of a network are almost always a subset of EFMs and never a superset.

Several derivatives of these concepts have appeared to address certain challenges associated with convex analysis. The concept of a elementary flux pattern (EFP), presented by Kaleta et al. [60] attempts to combine the valuable insights provided by EFMs with an enumeration method that only takes into account a set of predefined reactions. Similarly to EFMs, an EFP is also a minimal set of reactions describing a network conversion with the added constraint that restricts the pool of reactions to those belonging in the predefined subset. This drastically reduces the search space, which confers EFPs with a significant advantage in scalability.

The minimal generating set (MGS) concept was presented as an attempt to find the smallest possible pathways that can fully generate the flux cone in P. MGSs are similar to EFMs but do not require the non-cancellation property, allowing non-negative combinations of vectors within MGSs to describe any $v \in P$ even if two cancelling fluxes are involved. This leads to less enumerated pathways of smaller size, which is beneficial for computational demand, but lacks biological relevance. Furthermore, there may be multiple different MGSs capable of describing P, unlike EFMs which are unique for a given CBM [61].

The concepts referred above still lack the ability to include inhomogeneous constraints (with non-zero bound values), since accounting for these would lead P to become a polyhedron that is not a pointed cone. However, some authors have recognized the potential of elementary flux vectors (EFVs) [62, 63], presented by Urbanczik in the past decade [64], to further enhance the applicability of pathway analysis. EFVs share the same properties as EFMs, but their computation is performed in an augmented space converting the flux polyhedron into a pointed cone. Indeed, when P is originally a pointed cone, EFVs coincide with EFMs.

2.2.3.2 Minimal cut sets

One interesting application of EFMs is in finding intervention strategies to cancel certain phenotypes. These phenotypes can be selected by choosing all of the EFMs that include it (e.g. biomass producing phenotypes in bacteria). The set of reactions that, when deleted, disables the EFMs in question, is named a cut set. Additionally, if all reactions in the cut set are essential for it to be able to block the EFMs, it is also termed a minimal cut set (MCS). Formally, an minimal cut set (MCS) d in a set D must not be a subset of any other MCS in D [44]. MCSs are useful since they provide intervention strategies that are minimal, unbiased by cellular objectives and guaranteed to fully eliminate the undesired phenotype.

As a practical example, assume X in the toy network depicted in Figure 2 is an undesired compound, and $R9$ catalyses its synthesis. A subset U of the complete EFMs contains this reaction. To find the MCSs for $R9$, one must simply find sets of reactions that disable the EFMs in U. Table 3 highlights this example, adapted from [44].

Some extensions to the original definition have made it possible to increase the biological relevance of MCS based intervention strategies. The introduction of constrained minimal cut sets ensure that, despite blocking undesired behaviours, a set of desired ones are still active [65]. Constrained regulatory minimal cut sets allow over/underexpression constraints to be combined with reaction deletions, which increases the amount of valid strategies while decreasing their size [66]. Finally, genetic MCSs allow direct enumeration of strategies containing genes instead of reactions [67].

Table 3: Elementary modes and minimal cut sets that block $R9$ relative to the toy network on Figure 2. Elementary modes carrying flux through $R9$ are highlighted in grey. Adapted from [44]

		R1	R2	R3	R4	R5	R6	R7	R8	R9
Elementary modes	EM1	1	1	1	-1	0	0	0	0	0
	EM2	1	0	0	0	0	1	1	1	1
	EM3	2	1	1	0	1	0	0	0	1
	EM4	1	0	0	1	1	0	0	0	1
Minimal cut sets	MCS1									•
	MCS2	•								
	MCS3					•	•			
	MCS4					•		•		
	MCS5					•			•	
	MCS6		•		•		•			
	MCS7			•	•		•			
	MCS8		•		•			•		
	MCS9			•	•			•		
	MCS10		•		•				•	
	MCS11			•	•				•	

2.2.3.3 Applications

Having reviewed the most prominent concepts in pathway analysis, the overall consensus is that EFMs are currently the most popular approach with noticeable advantages over MGSs, EPs and ECs. Generically, EFMs can prove useful in a variety of analyses [54]:

- Finding novel metabolite conversions - at least one EFM with the associated exchange reactions should be active;
- Determining which reactions are required for a set of undesired EFMs and use it as a possible optimization target;
- Investigating the effect of environmental conditions on the cell's possible phenotypes by comparing EFMs before and after a perturbation.

Despite the relevance of these applications, usage of EFMs and related concepts in human studies is still limited. Table 4 highlights key studies in the field. Two main limitations concern the large scale of human GSMMs and the computational demand and low scalability of these methods, although recent developments such as the *k*-shortest EFM algorithm have improved this dramatically. There is great potential for drug target detection with minimal cut sets (MCSs) and similar algorithms, with at least one study by Apaolaza et al. pointing out this application [68].

Table 4: Table with important human health studies involving various convex analysis methods.

Author	**Approach**	**Description**	**Reference**
Wiback et al.	EP	Calculation of the extreme pathways in a human red blood cell model	[28]
Price et al.	EP	Application of singular value decomposition to interpret extreme pathway matrices	[29]
Kaleta et al.	GPR	Finding evidence of gluconeogenesis using fatty acids as carbon source by calculating GPRs on Recon1	[69]
Gebauer et al.	EFM	Detection of a large amount of energy wasting cycles involving cofactors through analysis of short EFMs	[70]
Rezola et al.	EFM	Filtering EFMs from human tissue-specific models with high-throughput transcriptomics, correctly depicting liver metabolism	[71]
Rezola et al.	EFM	Integration of gene expression data to identify differentially expressed and characteristic EFMs in lung cancer cells	[72]
Apaolaza et al.	gMCS	Calculation of MCS for genetic intervention strategies to discover metabolic targets in cancer cells	[68]

2.2.3.4 Algorithms and computational tools

The development of efficient algorithms for pathway analysis is an important aspect for proper application of these concepts in realistic case studies. Most of these methods rely on enumerating a large number of possible pathways in the CBM, and an algorithm based on testing every combination of active reactions is not computationally feasible. To this end, several approaches have been developed to narrow the search for pathways, specifically EFMs. Additionally, some recent approaches for MCS enumeration are also presented.

Double description algorithm The most common algorithm for full EFM enumeration is the double description (DD) algorithm. In essence, this algorithm attempts to generate new EFMs, and checks whether these have already been found (elementarity tests) [73], until all EFMs are enumerated. Several improvements have been proposed to reduce the problem's complexity, whether through improved tests, CBM compression or enumeration in multiple sub-networks derived from the CBM. Despite all of these efforts, the DD method is unsuitable for GSMMs since computational demand rises exponentially and deems EFM calculation infeasible. Nevertheless, *efmtool*, presented by Terzer and co-workers [74] remains the most efficient approach to calculate the entire set of EFMs.

Partial enumeration approaches There are currently some recent approaches that do not require full EFM enumeration. With such algorithms, a stopping criterion can be defined so that enumeration is possible in GSMMs, at the cost of obtaining only a subset of all EFMs. One example is the EFMEvolver approach, in which Kaleta et al. employ genetic algorithms to enumerate EFMs in a stochastic manner [60]. Another interesting approach is the k-shortest EFM algorithm, in which EFMs are enumerated iteratively by finding the smallest possible pathway and constraining the problem so that the next EFM does not contain active reactions from previous EFMs [7]. Despite their shortcomings, these approaches allow EFM enumeration at the genome-scale and their use is required in studies involving human metabolism.

Minimal cut set enumeration MCS enumeration is highly dependant on EFMs, as was previously mentioned. One alternative to enumerate them is to enumerate all EFMs and, similarly to the DD algorithm for EFM enumeration, check whether a set of reactions is a cut set for a given target, and whether it is minimal [75]. This is unsuitable for GSMMs, as the entire set of EFMs is difficult to enumerate. Recently, the k-shortest EFM method has been applied to a dual network based on a CBM where its EFMs are MCS [76], and the resulting algorithm developed by von Kamp and colleagues [8], MCS Enumerator, has enabled the enumeration of MCS at the genome-scale. This ultimately presents an interesting opportunity to apply such methods on human GSMMs.

2.2.4 Constraint-based modelling frameworks

Constraint-based modelling methods rely on mathematical methods that are too complex for humans to solve in an acceptable time frame. Although linear programming optimization methods have been available for several decades, computational frameworks specifically developed for constraint-based modelling have only begun to appear recently.

Although some of these software packages were introduced over a decade ago, there are still ongoing developments and updates. Arguably the most comprehensive resource among them is the COBRA Toolbox for MATLAB which, at its third major revision, has amassed 14 years of updates and provides implementations for a wide majority of constraint-based methods.

However, packages bound to proprietary software, such as MATLAB, greatly hinder the integration of these methods in other resources, such as executables and web services, as well as their distribution. Tools such as YANA and OptFlux have solved this issue through usage of the Java programming language.

Table 5: Overview of previously developed computational frameworks designed to load, manipulate, analyse or simulate constraint-based metabolic models.

Name	Platform	Year	Description	Ref.
METATOOL	GNU Octave / MATLAB	1999	Scripts implementing metabolic pathway analysis algorithms	[53]
FluxAnalyzer	MATLAB	2003	Framework for metabolic network analysis with a graphical user interface (GUI)	[77]
YANA	Java	2005	Multi-platform tool for graph-based and constraint-based modelling analysis with a user friendly GUI	[78]
COBRA Toolbox	MATLAB	2007	Modular constraint-based modelling framework with several phenotype prediction and flux analysis methods	[79]
OptFlux	Java	2010	Metabolic engineering framework (MEW) with a GUI packaged as a standalone tool (OptFlux) with various constraint-based analysis and optimization methods	[80]
COBRApy	Python	2013	Framework implementing part of the methods in COBRA Toolbox while offering a class-oriented design to implement new methods	[81]
RAVEN	MATLAB	2013	Toolbox for constraint-based model reconstruction and analysis with support for multi-omics data integration	[82]
MONGOOSE	MATLAB	2014	Metabolic network modelling toolbox offering support for exact arithmetic optimization methods	[83]
ReFramed	Python	2019	Modular constraint-based modelling framework with FBA-based simulation methods from which several omics integration and community modelling methods have been implemented	[84]
MEWpy	Python	2021	Open-source framework with a modular and extensible architecture with features ranging from phenotype prediction and strain optimisation algorithms	[85]

In recent years, a considerable amount of effort has been redirected towards the development of constraint-based modelling packages and frameworks in higher-level languages such as Python and Julia. COBRApy and ReFramed are based on the former and leverage the existence of high performance numerical computation and data analysis libraries (such as *numpy* and *pandas*) to provide a framework that is easy to use and provides good performance.

2.3 Enriching constraint-based models with omics data

CBMs present a very flexible framework for integrating fluxomics data. However, there is an insufficient amount of experimental measurements to successfully account for all possible perturbations. Furthermore, the process of integrating data beyond the flux layer is not trivial in CBMs, and several methods are being developed towards this purpose [86]. Due to wide availability of transcriptomics data for a large variety of organisms, there have been multiple efforts to accurately integrate RNA expression data in CBMs and these have been thoroughly compared in terms of predictive performance [87]. These methods are particularly useful for studies in human cells, both for reconstructing context-specific models of various tissues with different expression profiles and enhance phenotype prediction methods with RNA expression. The main focus of this section is to provide a brief overview of these methods and their applications in health.

Transcriptomics integration methods differ mainly on their ability to provide a numerical result such as one or various flux distributions, or a model. Some algorithms, due to implementation strategies, return both. Additionally, these can also be divided depending on how expression levels are processed [87] and whether an objective function is required.

2.3.1 Transcriptomics-enhanced phenotype prediction

Phenotype prediction with transcriptomics data was initially proposed through the pioneering work of Akesson et al. by disabling reactions whose associated gene expression levels were low. The authors used transcriptomics data to determine these lowly expressed genes and presented results involving a *Saccharomyces cerevisiae* model with improved prediction accuracy using this approach [88].

Another approach is to constrain fluxes with continuous limits determined by gene expression levels, as proposed in E-Flux, presented by Colijn et al. [89]. The method was used to successfully determine the impact of several environmental perturbations on the biosynthesis of mycolic acid in *Mycobacterium tuberculosis*.

If multiple gene expression datasets are available, as well as a gene regulatory network (discussed in the following chapter), a probabilistic model of expression can be created, and the resulting probabilities can be incorporated as flux bounds in a CBM. Chandrasekaran and Price introduced such a method, named Probabilistic Regulation of Metabolism, and validated it with gene knockout datasets for *Mycobacterium tuberculosis* [90]. Differential expression between two or more conditions can also be used to constrain the CBM through usage of statistical tests. This was the approach employed in the Metabolic Adjustment by Differential Expression method [91].

Although the authors claim transcriptomics-enhanced simulations improve phenotype prediction in their respective case studies, Machado and Herrgard have demonstrated through a systematic evaluation on micro-organisms that pFBA performs similarly to these methods [87]. There are additional algorithms that use transcriptomics to enhance phenotype predictions, such as Gene Inactivity Moderated by Metabolism and Expression (GIMME) [92] and Integrative Metabolic Analysis Tool (iMAT), but since they are also used to build models based on experimental data, these will be reviewed in the next section.

A noteworthy approach to mention is COBRAme [93], a framework developed by Lloyd et al. that allows the creation of constraint-based models combining the typical steady-state metabolic constraints, as well as artificial reactions and metabolites representing gene expression. These additional constructs rely on curated information of amino acid and nucleotide sequences, as well as the composition of enzymes associated with transcription and translation to add a highly detailed gene expression layer. These representations, named ME-models, led to improved gene essentiality predictions in an *E. coli* model. The applicability of this approach for large-scale models, however, is not yet proven, with computational demand and curated data requirements presenting as major limitations when applying this approach to model human metabolism.

2.3.2 Context-specific model reconstruction

Integrating omics data directly in the simulation algorithm is an important problem, but it can also be relevant to contextualize a model according to these same data. Several context-specific reconstruction algorithms have been developed to deal with this task with different approaches depending on how omics data are processed and how reactions are selected. Estévez et al. classify CSMR algorithms in three families that are summarized on Figure 5. These families are closely associated with their required inputs, namely[94]:

- **Metabolic tasks (GIMME-like)**: One way of representing omics data in the model is by defining a set of required metabolic function (RMF)s. An optimization problem containing the basic CBM assumptions and the objective function can then be solved, including RMFs as constraints.
- **Expression thresholds (iMAT-like)**: Alternatively, gene expression level thresholds can be set to filter the data and determine active and inactive reactions. The basic CBM optimization problem can be extended by forcing flux in active reactions and setting an upper limit on the inactive ones.
- **Core reactions (MBA-like)**: A core of required reactions is determined through experimental data (evidences of activity in literature or measurements) and is kept in the model, while non-core reactions are removed if not necessary. The model's consistency must be checked at each removal step

The GIMME algorithm [92] assumes a cellular objective and a set of required metabolic functions (RMFs) in its formulation, minimizing the variation between simulated fluxes and experimental data. Several variations have been proposed to deal with data from the proteome (GIMMEp) [95] and metabolome (GIM3E), as well as reversibility constraints [96].

The iMAT algorithm [97] does not require objective functions or RMFs, relying instead on an optimization problem that constrains reactions based on expression thresholds. The Task Integrative Network Inference for Tissues (INIT) [34] builds upon this algorithm by proposing weighing functions, rather than grouping genes in categories and allowing metabolomics data integration [6]. Task-driven Integrative Network Inference for Tissues (tINIT) further extends this latter method by adding RMFs as a requirement to build the context-specific model [6].

The Model Building Algorithm was the first to employ an approach based on core reactions, defining high and moderate confidence cores, as well as the non-core set of reactions. Reactions are iteratively removed from the latter set along with blocked reactions not present in the high confidence set that result from this modification [98]. The removal process is stochastic and thus the algorithm must be run multiple times to obtain meaningful results.

The metabolic Context-specificity Assessed by Deterministic Reaction Evaluation (mCADRE) algorithm instead assigns a score to each reaction based on expression measurements, network properties and confidence. These scores determine whether a reaction is part of the core. Similarly to MBA, non-core reactions are successively removed and the model is checked for consistency in the same manner. However,

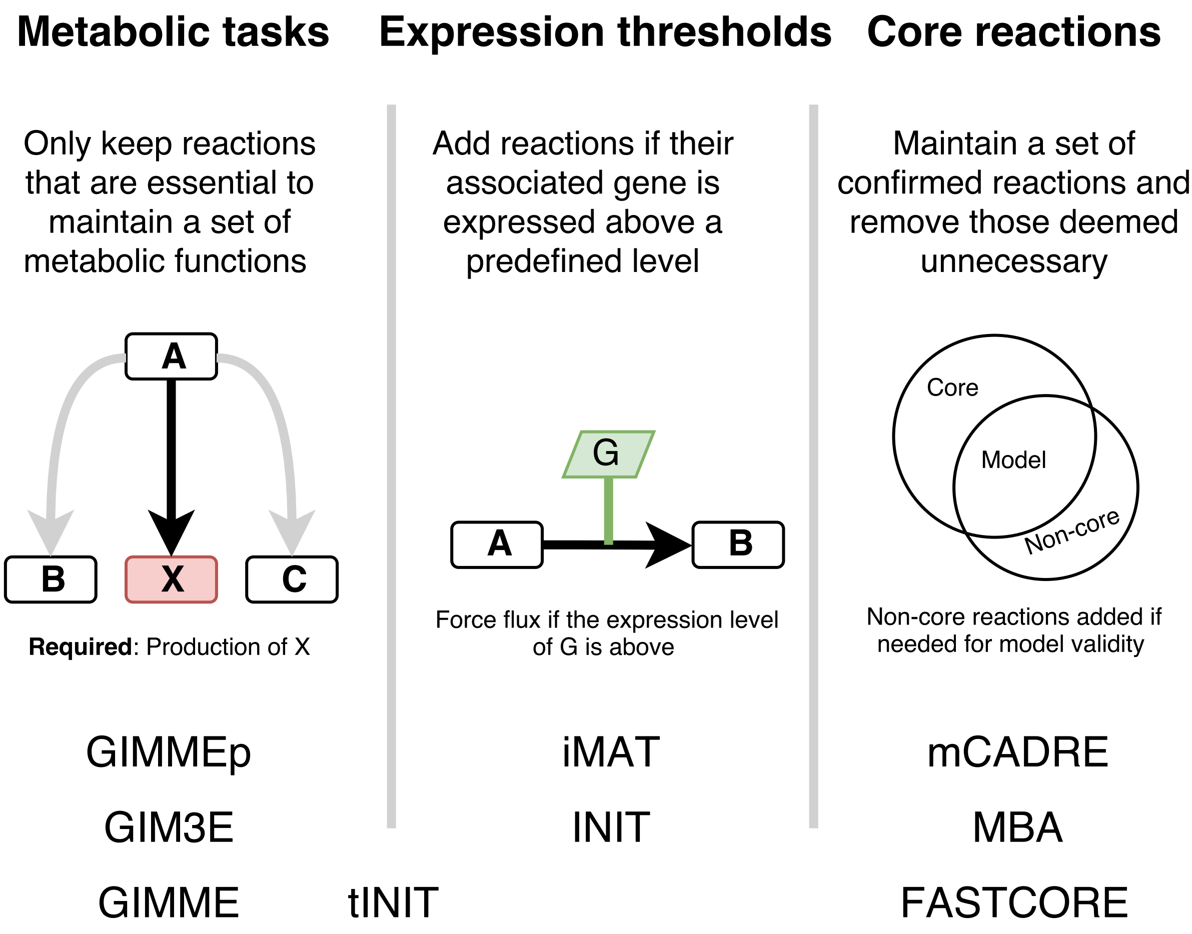

Metabolic tasks
Expression thresholds
Core reactions
Only keep reactions that are essential to maintain a set of metabolic functions
Add reactions if their associated gene is expressed above a predefined level
Maintain a set of confirmed reactions and remove those deemed unnecessary
A
B
X
C
Required: Production of X
G
A
B
Force flux if the expression level of G is above
Core
Model
Non-core
Non-core reactions added if needed for model validity
GIMMEp
GIM3E
GIMME
tINIT
iMAT
INIT
mCADRE
MBA
FASTCORE

Figure 5: Overview of the three major families of CSMR algorithms according to their input requirements.

mCADRE does not require the entire set of core reactions to remain in the model and a set of essential metabolites can be defined so that the model must account for their production [99].

FastCORE presented a more computationally efficient proposal by turning the non-core reaction removal step into an optimization problem. This problem maximizes the amount of core reactions in the model, while a second problem minimizes the amount of non-core reactions. These are repeated in cycles until the core reaction set is stable. FastCORMICS further extends this method by adding a microarray data processing workflow, which integrates seamlessly within the FastCORE approach [4].

Finally, the Cost Optimization Reaction Dependency Assessment (CORDA) algorithm defines similar sets of reactions to those in MBA but initially builds a core model with highly expressed reactions and uses a cost function represented as an artificial metabolite produced by the reaction candidates to be added [5].

CSMR algorithms should be generic enough to be used with any model organism and omics datasets

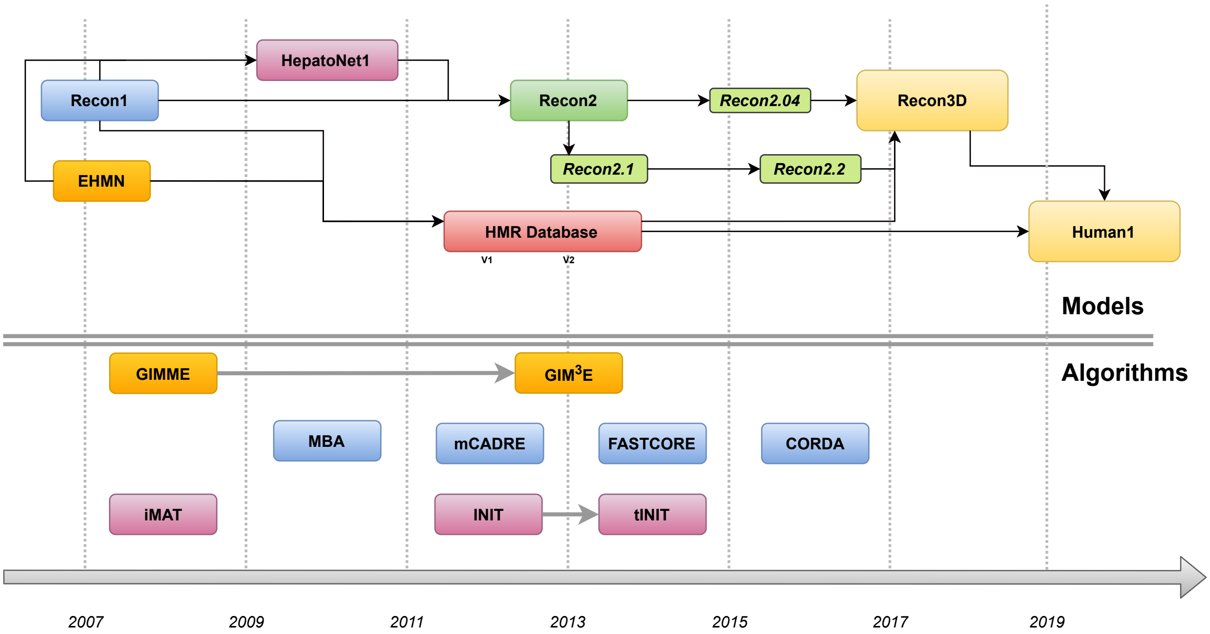


Figure 6: Timeline of the currently available generic genome-scale metabolic models of human cells and context-specific model reconstruction algorithms.

that can be mapped to biological entities in that model. However, their development was a key contribution to several metabolic modelling based studies involving human metabolism. Figure 6 shows the co-evolution of these methods along with contemporary human metabolic models.

2.3.2.1 Applications

Context-specific reconstruction enables the creation of models for multicellular organisms with differentiated cells, and is thus, an invaluable asset for systems biology applied to humans. A large number of cell-specific models of human tissues have been reconstructed with automated methods, including several types of cancer. Table 6 highlights some of these models and associated studies.

Common applications of cell-specific models usually focus on comparative analysis of healthy and perturbed models and application of constraint-based analysis methods can help uncover the mechanisms and effects of a given condition. This enables a rational approach towards drug target detection. A great potential that has yet to come concerns the usage of affordable sequencing techniques and automated cell-specific model reconstruction to quickly build a personalized model of a patient's affected tissue. The best therapy could then be identified by determining drug targets for the patient in question.

Table 6: Summary of relevant studies using context-specific model generation.

Description	**Model**	**Reference**
Development of the first cancer-specific human GSMM, used to identify target pathways to treat renal cancer	Recon1	[100]
Reconstruction of a renal cancer GSMM and detection of 5 alternative hypotheses for treatment	HMR	[101]
Application of MCSs to identify lethal sets of genes. Authors assessed the lethality of *RRM1* on myeloma cells	Recon2	[68]
Automated reconstruction of models for 69 normal tissues and 16 cancer cell lines	HMR	[34]
Reconstruction of 60 cancer cell lines with prediction of drug targets and limited experimental validation	Recon1	[102]
Reconstruction of 27 patient-specific models of hepatocellular carcinoma, revealing not just common but also individual therapeutic options	HMR	[6]
Determination of a human biomass equation to investigate cancer metabolism - Warburg effect	Recon1	[49]
Identification of characteristic metabolic functions in two non-small lung cancer subtypes using EFMs	Recon1	[72]
Reconstruction of the human hepatocyte (HepatoNet1) to demonstrate liver responses to altered metabolic states	Recon1	[42]
Reconstruction of human adipocyte, hepatocyte and myocyte models. A multi-tissue model was created to simulate the Alanine and Cori cycles as well as nutrient absorption	Recon1	[103]
Automated reconstruction of hundreds of tissue-specific models of healthy and cancer cells, with drug target discovery for the latter	Recon1	[104]
Integration of multi-omics data and two generic models to automatically reconstruct cell-specific models. Cancer drug targets were determined through comparative analysis	HMR	[34]
Reconstruction of a new hepatocyte model to reveal metabolic aspects of fatty liver disease not associated with alcoholism	HMR2	[35]

2.4 Gene regulatory models

2.4.1 Gene regulatory networks

A Gene regulatory network (GRN) is a construct describing the interactions between genes (DNA) and their transcription and subsequent translation products (RNA and proteins, respectively). These are often called transcriptional regulatory networks since, typically, the main focus of these networks is to study transcription factors (TFs), which are proteins that regulate the expression of other genes. These ultimately control many aspects of the transcription process and can affect the expression of other TFs. This leads to complex cascades of regulation (such as feedback loops), that are hard to analyse without resorting to mathematical and computational methods.

2.4.2 Model structure

Schlitt and Brazma suggest a hierarchical classification for GRN -based models, according to the level of detail and features included [105].

Network components A network contains elements from the organism .The first step towards building a GRN model is to compile a list of components, such as TFs and promoter genes (and relevant biochemical properties) regulating gene expression in the selected organism and/or pathways. Genome annotation is a way to obtain lists of molecules such as genes or TFs, which can then be inserted in a database with cross-references to ontologies describing more generic regulation events. The scale of these lists in eukaryotic organisms varies from the hundreds to the thousands.

Topology Network components can then be connected to represent interactions between the identified components. This is usually represented as a graph containing nodes (molecular entities) connected to each other. These connections can be directed to represent specific regulatory relationships by defining regulatory genes as sources pointing (unidirectionally) to the genes they regulate, or targets. Undirected connections can be used to represent, for instance, binding of two proteins.

Control logic With topological models, the relationships between entities are described, but its effect is not specified. An additional layer can be added to describe these effects through Boolean functions, including combinations of logical operations (AND, OR and NOT) to describe conditional activation or repression events. These functions can be discrete, representing gene states as active or inactive, or assigned to continuous values by adding weights to quantify the effect (linearly) of each gene in an interaction.

Kinetics A temporal dimension can be added to the model to describe changes in gene expression over discrete time points. The simplest approach is to expand upon the logical model (based on a Boolean network) and explore changes in state over time.

It is worth noting that the scope of GRN models decreases as more layers are included, since detailed logical relationships and temporal parameters for TFs are not always known. As such, topological models are preferred for larger networks, as opposed to kinetic models.

2.4.3 Modelling approaches and applications

A large number of GRN modelling approaches have been reviewed by Karlebach and Shamir [106], who propose a binary distinction between discrete (logical) and continuous models.

2.4.3.1 Logical models

Boolean networks Logical models are arguably the most basic approach towards GRN modelling. Glass and Kaufmann presented an early approach with Boolean networks (BNs), proposing the usage of on/off states for gene expression in regulatory models [107]. Despite appearing in the early 1970s, this approach has proven useful in yeast studies to assess the robustness of its cell functions [108] in yeast through GRN analysis and to investigate the kinetics involved in cell-cycles [109].

Probabilistic Boolean networks Shmulevich et al. further presented Probabilistic BNs as an extension to deal with uncertainty in the data [110]. The assignments between entities and functions are chosen randomly, based on a probability distribution for various functions. The global system states are then generated according to the initial assignment and elicit the usage of sampling methods to deal with the stochastic nature of the algorithm.

MetaReg Gat-Viks and colleagues have further improved upon the BN formalism by filtering regulatory functions that would only fit the network's steady-state [111]. This approach, named MetaReg, was used to discover previously unknown regulatory mechanisms for amino acid biosynthesis. A probabilistic version was also presented and further developments on this model allow addition and improvements on existing GRN s, leading to the discovery of new modules and interactions between existing ones.

Petri nets Other approaches such as Petri nets have also been used to successfully replicate results from BNs, as demonstrated by Steggles et al. through a regulatory model of *Bacillus subtilis*, whose results have been proven to match experimental measurements [112].

2.4.3.2 Continuous models

Linear models Linear models of GRN s assume the contribution of each gene in a regulatory function is proportional to a given weight. Yeung and colleagues presented an approach where a large number of models is created through singular-value decomposition and the best ones are selected, an useful approach when data is scarce. Some extensions of linear models have been presented, dealing with alternative functions to determine gene contributions, and accounting for delays in the occurrence of regulatory events.

ordinary differential equations (ODEs) While linear models are simple and relatively effective, more complex models may provide better insights in some cases where regulation is controlled through mechanisms governed by complex dynamics. Li presented a nonlinear ODE model to predict the cell-cycle regulation dynamics in *Caulobacter crescentus*, and a similar study was also presented for yeast cells by Chen et al. [113]. These models require a greater amount of parametrization and calibration, which might render them infeasible for larger scale contexts.

2.5 Signalling networks

Signalling networks are important systems that dynamically interpret extracellular signals and relay this information so that a proper response can be triggered, allowing cells to display different behaviours that ensure the long-term survival of an organism. The complexity associated with these systems elicits the usage of mathematical formalisms and models that can represent its behaviour and enable analyses regarding its functionality under different contexts.

Two of the most relevant signalling cascades involved in cancer studies are the mitogen-activated protein kinase (MAPK) and class I phosphoinositide-3' kinase (PI3K) pathways. These are comprised of multiple proteins and other molecules present in the cell, that regulate transcription through binding, modification and translocation, and deeply influence cell growth, differentiation and other important events [114].

There is a great level of complexity associated with the MAPK/PI3K pathways, as there are multiple receptor proteins that can feed signals into it, feedback loops, highly dynamic behaviour and multiple subcellular locations. Mutations in several proteins of the pathways have been found to promote cancer with multiple mechanisms, such as growth factor overexpression, and cell cycle deregulation. Thus, it is important to include a signalling layer in *in silico* predictions of cancer biology to provide a better depiction of the underlying mechanisms leading towards cancer [114].

2.5.1 Network structure

Each small molecule participating in a signalling network is usually represented as a node that can be interlinked with various other nodes. Pathways can be identified as paths linking nodes associated with inputs (such as external stimuli) to those associated with outputs (signals for cellular responses). Past studies focused on these linear structures, usually relaying signals from the cell membrane to the cytosol and other organelles.

With the considerable improvements in high-throughput techniques over the last two decades, more recent studies have proposed a modular structure encompassing cell signals. Such modules are highly connected, regulate multiple functions and form a large and complex network of signalling molecules.

Nodes, pathways and modules provide a generic view on the various interactions between nodes, but the nature of these interactions can be described further. A **causal reconstruction** of the signalling network includes information regarding the effects nodes cause on each other. Causal networks are usually inferred from experimental data observing cellular responses under various stimuli.

A **mechanistic reconstruction** of the network can further expand the causal view by representing simple interactions through the biochemical reactions that cause them (e.g. degradation, phosphorylation).

Causal and mechanistic reconstructions of signalling networks are represented mathematically through distinct formalisms that will be briefly covered in the following sections.

2.5.2 Causal modelling

Causal networks can be mathematically represented with a variety of methods. With a Boolean representation, each node is assigned with an on/off (active/inactive) state and rules involving logical operations can be inferred from the network. As an example, in the module represented on Figure 7, one could infer

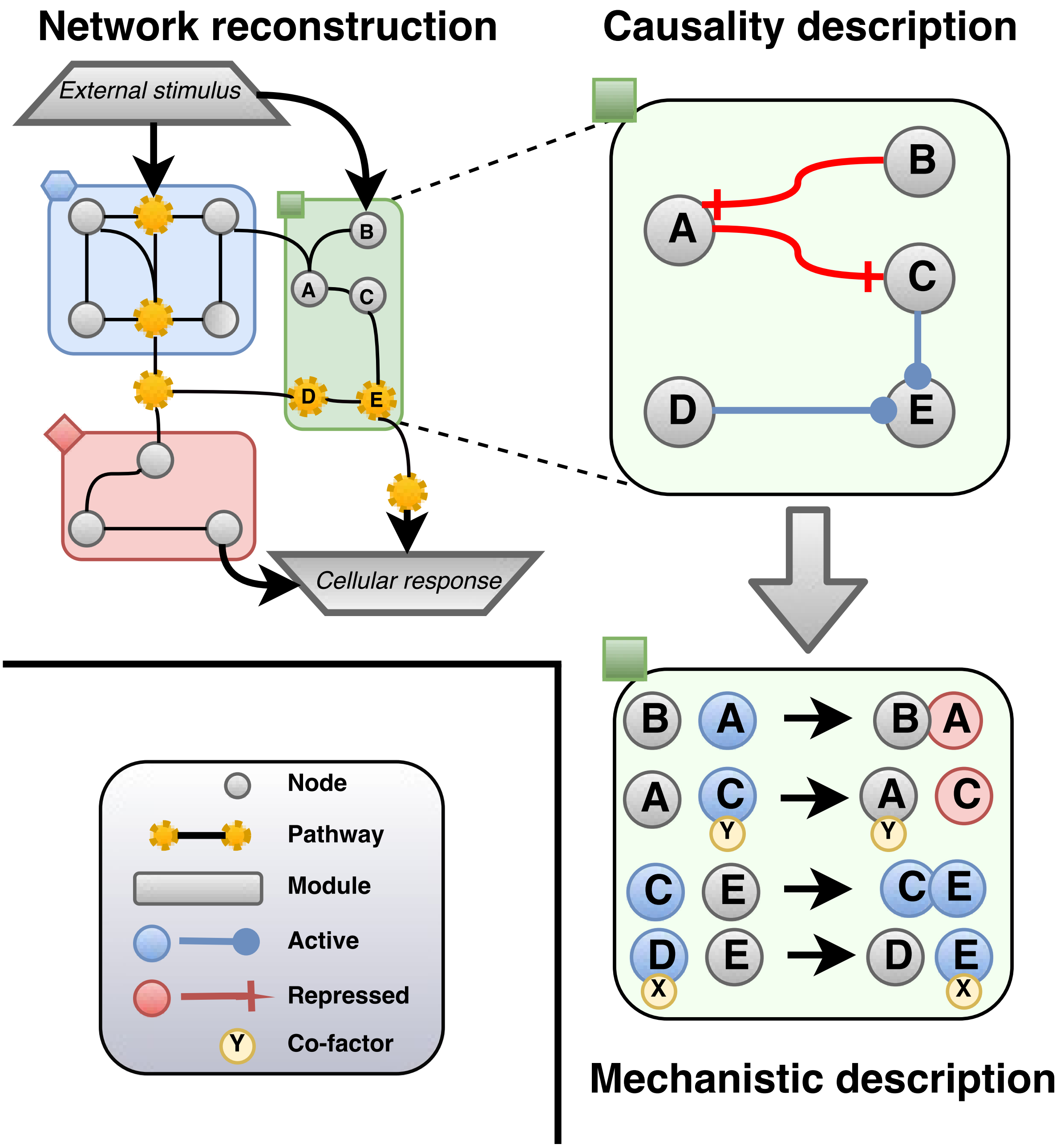


Figure 7: Representation of a fictional signalling network along with the two major formalisms. The network reconstruction merely represents generic interactions between molecules. With the **causality description**, the interactions are assigned with a given effect, allowing representation as Boolean networks. Finally, these relationships can be represented by more complex reactions between entities (**mechanistic description**)

that E is active if C or D are active, A is inactive when B is active and C is inactive when A is active. Despite representing signalling interactions through simple rules, causal models can also provide reliable insights on signalling mechanisms.

Models based on Boolean networks can be dynamic, simulating the nodes' state over individual time points. Such models have been used to predict signalling events associated with cell differentiation in *Drosophila melanogaster* [115]. Alternatively, logical steady-state analysis can be used to uncover structural and functional properties of the network without requiring dynamic parameters. Klamt and associates have used these models to simulate the mechanisms involved in T-cell activation [116].

An alternative to Boolean networks is modular response analysis. With this approach, experimental measurements for various perturbations are used to create a global response matrix containing coefficients for each perturbation-node pair. A local response matrix is then derived, this time representing weights for each node-node interaction. Using this approach, Klinger et al. constructed models of the Ras/PI3K signalling pathways to discover inhibition targets aimed at blocking colorectal cancer cell growth [117].

2.5.3 Mechanistic modelling

Mechanistic models are usually built as dynamic (kinetic) systems of biochemical reactions, which can be based on rules representing interactions between different signalling molecules. Such kinetic models are usually formalized with ODEs that require a large set of parameters.

Since these models are sets of biochemical reactions, the rate equations defining their behaviour over time must be determined, often through literature. Typical equations are already commonly known in biology, such as mass-action and Michaelis-Menten laws [118], and are already applied in mechanistic models.

An early approach was presented by Moehren and associates, in which a kinetic model of the epidermal growth factor receptor (EGFR) signalling pathway was used to determine its dependence on temperature. The authors were able to represent this dependence by adjusting the parameters on each rate equation to the ones corresponding to discrete temperature points and achieved similar predictions to those found *in vivo* [119].

Schoeberl and colleagues have also modeled part of the MAPK signalling pathway with mass-action kinetics to find drug targets for cancer therapies [120]. By using sensitivity analysis, which is commonly used in ODE models to quantify the impact of each parameter in the simulation, the authors were able to find *ERBB* proteins as possible drug targets. This protein was not previously considered as a suitable target and prompted the development of a monoclonal antibody that proved useful as a cancer therapy and is currently on Phase 2 clinical trials.

Although quantitatively accurate, mechanistic models are restricted to well-known signalling pathways of smaller sizes as a result of the large number of kinetic parameters required for each reaction (often not available) and computational demand.

2.6 Integration with constraint-based modelling

In the previous section, some transcriptomics integration methods were reviewed, covering few aspects of gene regulation. Despite the variety within these methods, the task of integrating multiple models with different formalisms is not straightforward. A representation of the intricacies of this integration is shown on Figure 8. In this section, an overview on both gene regulatory and signalling layers is presented, as well as current perspectives on fully integrated cell models.

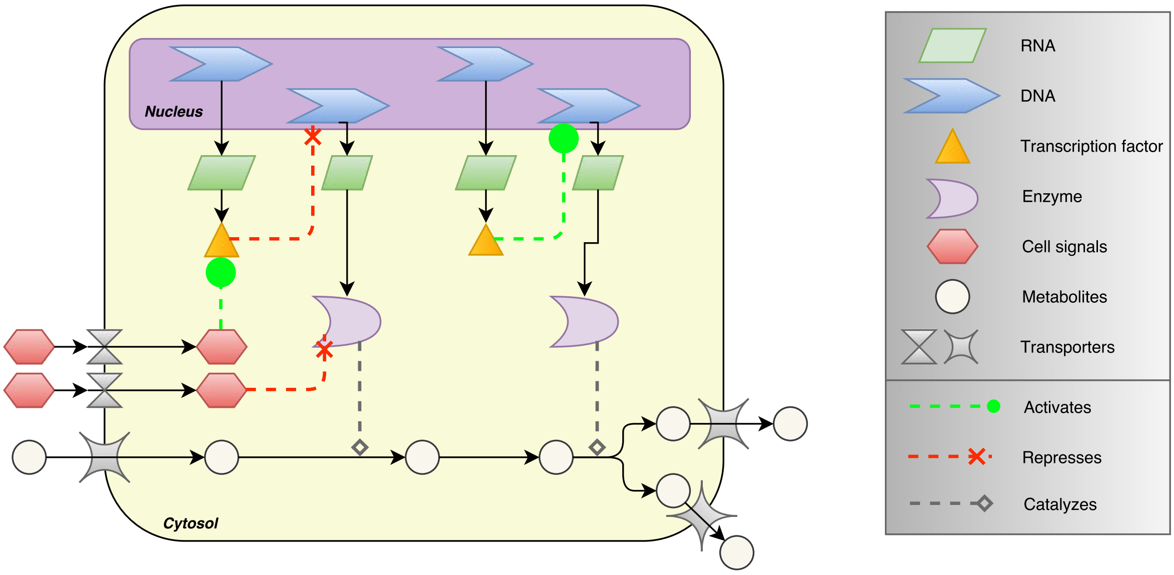


Figure 8: Representation of a conceptual network integrating signalling, regulatory and metabolic layers.

2.6.1 Signalling network integration

The integration of signalling and metabolic networks is currently limited to simple approaches that are small in scale. Gonçalves et. al hypothesize that the lack of integrated models combining these two layers can be attributed to the rarity of direct interactions between them, since these mostly occur indirectly through gene regulation [121]. Despite this, recent work concerning signalling pathways in cancer has shown that their disruption can lead to regulatory changes that affect metabolic processes [122], which confirms the importance of integrating the metabolic and signalling layers.

König and colleagues have presented a kinetic model based on ordinary differential equations comprising glucose metabolism in the liver and integrating sugar regulating hormones by implying different effects on certain enzymes [123]. This means that a single signalling state had be translated into enzymatic changes. Similarly, the PI3K/AKT/mTOR signalling pathway was integrated on a HeLa cell (immortalized human cell line) dynamic metabolic model by defining their effects as changes in the maximum rates of certain reactions [124]. These two studies do not describe the signalling pathways mathematically, which elicitis the development of more advanced methods that can fully represent the two networks and their interactions.

2.6.2 Gene regulatory network integration

The integration of gene regulatory networks on CBMs is not a new concept. The integration of transcriptomics data, as discussed in the previous chapter, either manipulates the computational method or modifies the structure of the CBM according to the gene expression profile which is governed by regulatory events, among others. However, without a fully integrated GRN , the model can only capture the effect of regulation on metabolism in a single direction.

Methods such as regulatory FBA [125] and steady-state regulatory FBA (srFBA) [126] map a single regulatory state as constraints in a CBM. With rFBA, the GRN is used to determine the regulatory state of the cells, which translates into flux bounds that block reactions whose expressing genes are inactive. srFBA instead adds Boolean constraints seamlessly within the optimization problem to represent the GRN and GPR rules. The growth pseudo-reaction is then maximized, subject to these constraints. This forces the usage of logical GRN modelling approaches and while useful, merely provides a single optimal state for the network. The PROM [90] approach that was previously described somewhat improves this by representing regulatory events as probabilities rather than definite binary states.

2.6.3 Multi-layer approaches

Signalling events do not usually interfere directly with entities associated with metabolism, such as enzymes or metabolites themselves. This is usually achieved through regulation of gene expression and as such, the interaction between signalling and metabolism is poorly explored through integrative CBM approaches.

However, there have been methods to develop a fully integrated model containing metabolic, regulatory and signalling layers. Covert et al. proposes the integrated FBA algorithm which uses a similar approach to regulatory FBA, but extends it with an independent kinetic model (ODE-based) representing certain pathways [127]. After specifying the initial conditions, a Boolean state of the GRN can be obtained, defining gene and protein expression. Additionally, the kinetic model is simulated up until a defined end time point. Both of these inputs are then considered when building flux constraints, which are added to the final FBA problem. The result is a flux distribution which is used as an initial condition so that the process can be repeated.

The integrated dynamic FBA (idFBA) algorithm presented by Min Lee et al. adds signalling and regulatory layers within the stoichiometric matrix of a FBA problem [128]. idFBA includes the definition of *fast* reactions that comply with the steady-state assumption and *slow* reactions with alternate kinetic behaviour (such as those involved in signalling and regulation). The FBA problem is then simulated on a single time step, and the resulting flux distribution is used to constrain the problem and the stoichiometric matrix. Slow reactions are activated or deactivated according to their activation delay and duration by modifying the problem iteratively through each time step.

A major development by Karr and colleagues was the creation of a whole-cell model for *Mycoplasma genitalium*. The authors reconstructed the organism's chromosome structure including encoding genes, promoters, among other important DNA sequences. Each gene was annotated, and each gene product was assigned with a structure, revealing post-transcriptional and translational modifications. This network was represented as a set of modules performing a certain cellular process and each was modelled independently with an adequate formalism. Simulation is performed over a time course, in a similar manner to ODE-based models [129].

Fully mechanistic approaches, however, are not the only examples of network-based methods capable of predicting genotype and phenotype relationships, while integrating multiple biological entities. A review by Dugourd and Saez-Rodriguez [130] recently shown examples of relevant approaches for multi-scale network analysis and contextualisation. With the wide availability of biological network resources, there are several databases from which a prior knowledge network (PKN) - a collection of interactions between biological entities - can be generated. When coupled with diffusion algorithms such as TieDIE [131] or CAusal Reasoning for Network identification using Integer VALue programming (CARNIVAL) [132], one can contextualise the PKN, using omics data as a means to filter and extract relevant sub-networks. Most efforts in this sense have been directed towards signalling networks.

Up to this date, there is still a gap concerning the development of these models in humans, both due to lack of experimental data and the immense computational demand and mathematical complexity posed by these problems. However, some approaches mentioned in this section such as idFBA could be leveraged to improve predicted phenotypes with human GSMMs. A recent contribution by Thiele et al. explores a whole-body metabolic model in which various components of the stoichiometric matrix represent different organs and structures. The addition of regulatory and signalling layers could further enhance this representation of human biology.

3

Software development

3.1 Introduction

In this chapter, three frameworks developed within the scope of this work are explored in detail, namely Constraint Based Analysis of Metabolic Pathways (CoBAMP), Tissue-specific RecOnstruction and Phenotype Prediction using Omics data (TROPPO) and Gene Regulation and Signalling Pathways (GRaSP). CoBAMP is a constraint-based modelling framework including basic simulation capabilities and a modular structure for pathway analysis methods. This framework also formed the basis for TROPPO, which uses CoBAMP's modelling routines to implement an omics data integration pipeline ranging from data preprocessing to context-specific model reconstruction, as well as refinement and validation. Finally, GRaSP extends CoBAMP models to accommodate signalling and gene regulatory mechanisms. The software tools and associated outcomes are represented on Figure 9.

With the large variety of toolboxes and frameworks, choosing a single framework to extend with novel methods is not a simple task. As such, the development of constraint-based methods for metabolic pathway analysis, integration of omics data and representation of signalling and gene regulatory networks was performed so that the new methods would be compatible with all frameworks, but remained independent from them.

The motivation behind this effort is also justified by the lack of pathway analysis enumeration and omics integration algorithms implemented in the Python programming language. Similarly, context-specific model reconstruction and omics integration methods are only scarcely available in ReFramed [84] and COBRApy [81], eliciting the development of modular software resources to make these methods available for future improvement with the addition of signalling and gene regulatory models.

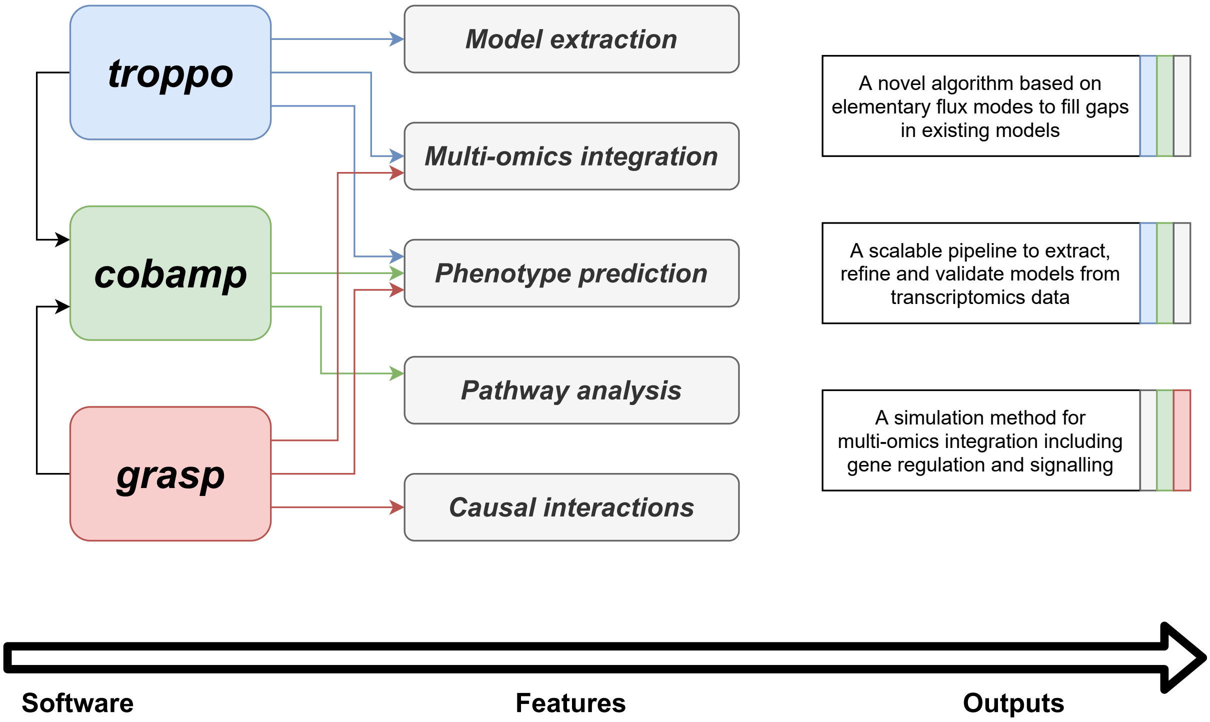

troppo
cobamp
grasp
Model extraction
Multi-omics integration
Phenotype prediction
Pathway analysis
Causal interactions
A novel algorithm based on elementary flux modes to fill gaps in existing models
A scalable pipeline to extract, refine and validate models from transcriptomics data
A simulation method for multi-omics integration including gene regulation and signalling
Software
Features
Outputs

Figure 9: Overview of the software packages and features implemented within the scope of this work. The outputs on the right-hand side are colour coded to match the packages used in the methods and results associated with each box.

3.2 COBAMP: A framework for constraint-based model simulation and analysis

The first software tool developed throughout this work is CoBAMP. This tool was developed with the purpose of providing a basis to implement constraint-based modelling (CBM) methods. This includes model representation and manipulation as well as simulation and analysis through linear programming (LP) and mixed integer linear programming (MILP) formulations.

Linear programming problems An important feature of this framework was the development of tools to easily define, modify and optimize LP and MILP problems through matrix inputs. Since it is based on the Opt (optlang) package, multiple solvers are supported although some improvements were made to speed up the process of creating a model when compared to the default syntax of optlang.

Parallel processing Modern solvers are capable of using multiple processor cores to speed up linear optimisation problems. However, this is not as relevant when considering multiple optimisations where the optimisation is not the time-limiting factor. CoBAMP includes several routines to easily run batch optimisations on LP and MILP with varying constraints.

Model representation A simple model representation class was implemented, connecting the linear models described in the previous paragraph with reaction and metabolite identifiers as well as gene-protein-reaction (GPR) association. These models were built to include a linear programming problem built from the model's information and can thus be used with any CoBAMP method that involves LP problems, while also allowing direct manipulation of its variables and constraints.

Compatibility with modelling frameworks The developed package was built to re-use model data structures from existing modelling frameworks. Although an internal model representation was also developed to serve as a basic input for the simulation and analysis algorithms in CoBAMP, a model reader was implemented for COBRApy and ReFramed, allowing seamless interconversion between representations and ensuring these algorithms can be integrated with any framework.

Pathway analysis tools The main focus of this package was to provide scalable methods to enumerate pathway analysis concepts. This was addressed by implementing the k-shortest elementary flux mode (EFM) enumeration algorithm using the linear programming tools in CoBAMP. minimal cut set (MCS) and elementary flux pattern (EFP) enumeration were implemented simply by extending the EFM enumeration problem. A considerable amount of effort was put to generalise these algorithms as much as possible to easily accommodate for further extensions. Some analysis and plotting features are also offered by the package.

3.2.1 Linear programming framework

Constraint-based modelling methods are traditionally implemented using LP or MILP. Since many of these methods share common steps, a modular structure was required to facilitate the implementation of new methods based on the basic steady-state model optimization problem. The core module contains two important sub-modules to achieve this, namely `linear_systems` and `optimization`.

3.2.1.1 Linear system structure and representation

CoBAMP implements a linear system framework that simplifies the formulation of linear problems. It does so by encapsulating the underlying linear problem as represented within the solver through the usage of the optlang package. optlang is a Python framework providing a generic interface for various linear optimisation solvers. Furthermore, it also provides a language to easily define objectives, constraints and variables for optimisation problems [133]. The core component in CoBAMP that wraps the features in optlang is the `LinearSystem` class.

The `LinearSystem` abstract class (`cobamp.core.linear_systems`) represents a generic linear programming problem. Any instance of its subclasses contains a private `optlang Model` object that is populated by methods implemented in the parent class to easily and efficiently manipulate variables, constraints and bounds. This is represented on Figure 10. This structure takes advantage of the modular `optlang` framework to provide compatibility with multiple solvers since the appropriate solver interface is loaded when a `LinearSystem` object is created. Solver configuration is also handled by `optlang`.

Although `optlang Model` objects allow this by design, the implemented methods circumvent some performance limitations caused by the simplified problem definition features when using the standard methods provided by the library. Additionally, these methods were also designed to extend models with new constraints by providing `numpy` arrays as inputs.

A `GenericLinearSystem` class was then implemented to provide a standard implementation for the `LinearSystem` class, establishing the basic requirements for a linear programming problem, namely a linear coefficient matrix S with bounds for both constraints (right-hand side values) and variables, their names and types and a solver string specifying the appropriate optlang interface to load.

With a lower-level `LinearSystem` class, some higher-order abstractions could be created to improve the framework's modularity. One particular example of this was the implementation of the `KShortestCompatibleLinearSystem` which adds the concept of decision variables as a new parameter specifying which variables in a `LinearSystem` should be considered for EFM enumeration. This is useful as it allows for other concepts based on EFMs, such as EFPs, to be implemented by changing this parameter.

Several `GenericLinearSystem` subclasses have been implemented to serve as basis for several constraint-based modelling methods:

- `SteadyStateLinearSystem`: creates a problem assuming all constraints must equal 0 when the problem is created;

- `IrreversibleLinearSystem`: creates a `KShortestCompatibleLinearSystem` subclass where all constraints are equal to 0, but splits variables in their positive and negative components;

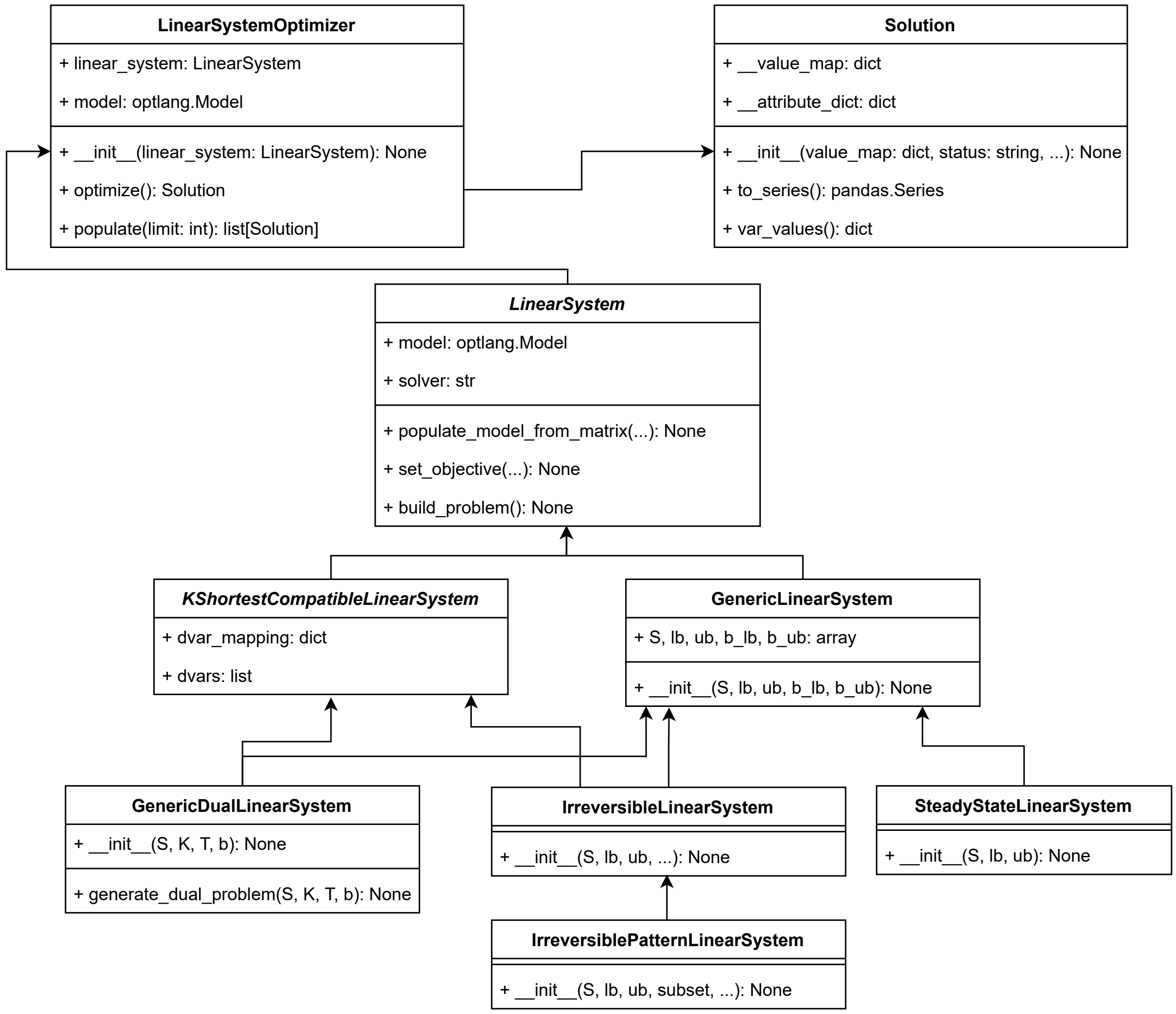

LinearSystemOptimizer
+ linear_system: LinearSystem
+ model: optlang.Model
+ __init__(linear_system: LinearSystem): None
+ optimize(): Solution
+ populate(limit: int): list[Solution]
Solution
+ __value_map: dict
+ __attribute_dict: dict
+ __init__(value_map: dict, status: string, ...): None
+ to_series(): pandas.Series
+ var_values(): dict
LinearSystem
+ model: optlang.Model
+ solver: str
+ populate_model_from_matrix(...): None
+ set_objective(...): None
+ build_problem(): None
KShortestCompatibleLinearSystem
+ dvar_mapping: dict
+ dvars: list
GenericLinearSystem
+ S, lb, ub, b_lb, b_ub: array
+ __init__(S, lb, ub, b_lb, b_ub): None
GenericDualLinearSystem
+ __init__(S, K, T, b): None
+ generate_dual_problem(S, K, T, b): None
IrreversibleLinearSystem
+ __init__(S, lb, ub, ...): None
SteadyStateLinearSystem
+ __init__(S, lb, ub): None
IrreversiblePatternLinearSystem
+ __init__(S, lb, ub, subset, ...): None

Figure 10: Overview of the class structure of the `linear_systems` and optimization modules in CoBAMP.

- `GenericDualLinearSystem`: creates a dual system to enumerate minimal cut sets of the provided linear problem matrix.

3.2.1.2 Optimization

A separate optimization module contains a `LinearSystemOptimizer` that takes `LinearSystem` instances as input and solves the underlying optimization problem, yielding `Solution` instances. Although the `LinearSystem`'s state is not altered by these classes, it can trigger solver population.

The `LinearSystemOptimizer` handles a single `LinearSystem` instance, containing an optimize method that calls `optlang` optimization routines. Additionally, a populate method was also implemented for `CPLEX` and `GUROBI` solvers on MILP problems to enumerate a fixed amount of alternative optima for a given problem. Both methods return one or more `Solution` objects, representing solutions to the optimization problem.

`Solution` objects hold the variable values, optimization status and an optional dictionary with annotations. These objects can be extended with subclasses for further processing of variable values as required by specific methods. One example of this is the KShortest`Solution` subclass that decodes the `Solution` of an elementary flux mode enumeration problem on a model with split reversible reactions, returning values for the original model reactions with appropriate sign.

A `BatchOptimizer` class was also implemented to allow optimization of a given `LinearSystem` in parallel using the `pathos` library. Although modern solvers are already capable of using multiple processor cores to solve optimization problems, some performance gains can be achieved on less computationally intensive problems by running multiple optimizations at once. Given a `LinearSystem` instance, a set of variable bound changes and associated objectives, `BatchOptimizer` is able to return the respective `Solutions` much faster than iteratively performing the same task.

3.2.2 Constraint-based model representation

CoBAMP was built primarily to accept representations of constraint-based models based on matrices and vectors rather than using standard formats such as Systems Biology Markup Language (SBML), a design choice that favors the implementation of methods using LP problems rather than dealing with more abstract concepts, such as reactions and metabolites. Nevertheless, these abstractions are still useful for certain basic features, such as performing flux balance analysis and dealing with other components, such as GPR rules which are associated with reactions but are not modeled directly as LPs. A `core.models` sub-module was added to include these abstractions.

A `ConstraintBasedModel` class was implemented, which does not substitute the model manipulation features featured in other constraint-based modelling frameworks, but allows for modifications in the underlying LP through reaction and metabolite abstractions rather than dealing with constraints and variables directly. This class can be instantiated using the same inputs as required by `SteadyState-LinearSystem` but can additionally interpret lists of reaction and metabolite names that are associated with their respective variables and constraints. Furthermore, GPR rules can also be added through string representations of their Boolean expressions. When instantiated, a `SteadyStateLinearSystem` and associated `LinearSystemOptimizer` are also created, formulating the LP problem as defined by flux balance analysis (FBA).

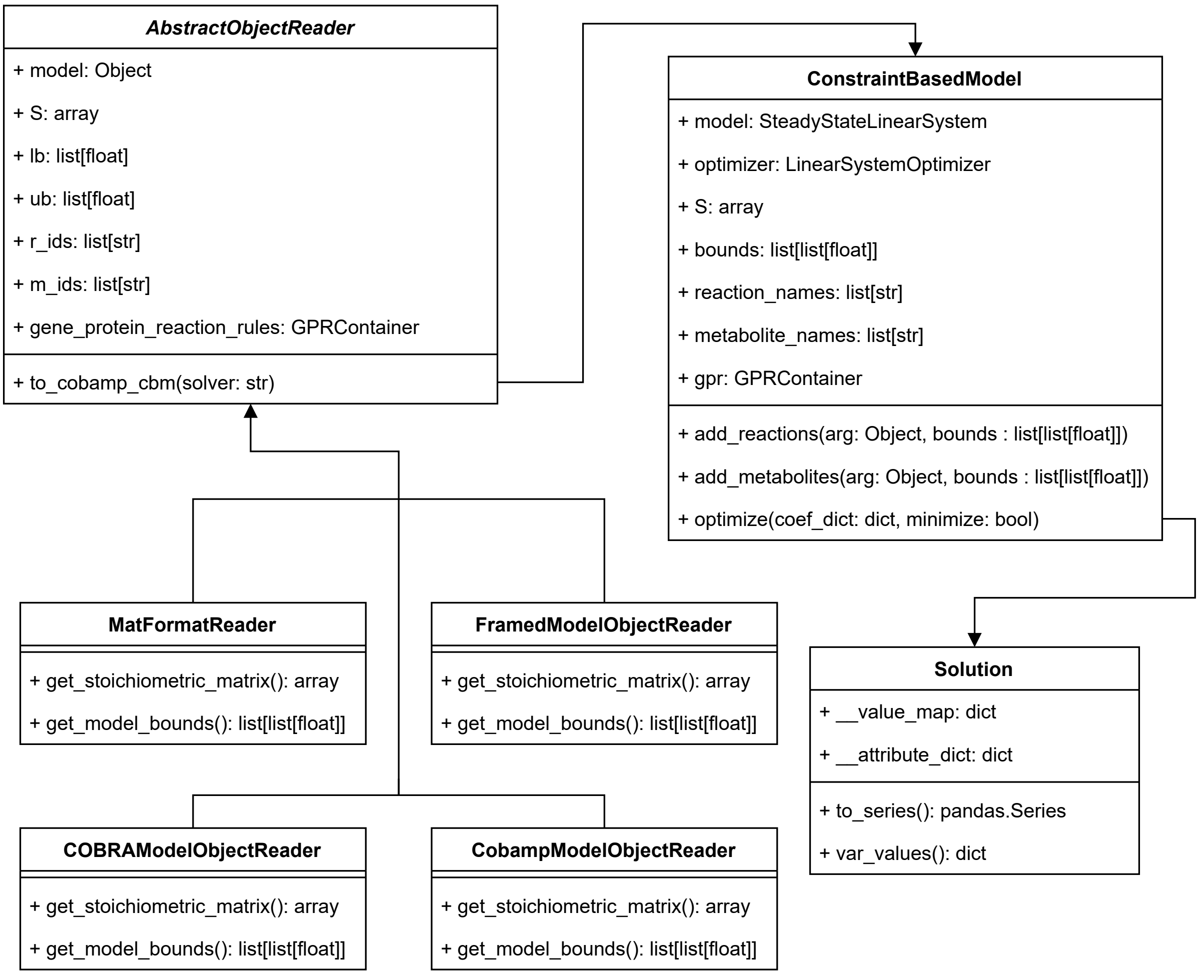

AbstractObjectReader
+ model: Object
+ S: array
+ lb: list[float]
+ ub: list[float]
+ r_ids: list[str]
+ m_ids: list[str]
+ gene_protein_reaction_rules: GPRContainer
+ to_cobamp_cbm(solver: str)
ConstraintBasedModel
+ model: SteadyStateLinearSystem
+ optimizer: LinearSystemOptimizer
+ S: array
+ bounds: list[list[float]]
+ reaction_names: list[str]
+ metabolite_names: list[str]
+ gpr: GPRContainer
+ add_reactions(arg: Object, bounds : list[list[float]])
+ add_metabolites(arg: Object, bounds : list[list[float]])
+ optimize(coef_dict: dict, minimize: bool)
MatFormatReader
+ get_stoichiometric_matrix(): array
+ get_model_bounds(): list[list[float]]
FramedModelObjectReader
+ get_stoichiometric_matrix(): array
+ get_model_bounds(): list[list[float]]
Solution
+ __value_map: dict
+ __attribute_dict: dict
+ to_series(): pandas.Series
+ var_values(): dict
COBRAModelObjectReader
+ get_stoichiometric_matrix(): array
+ get_model_bounds(): list[list[float]]
CobampModelObjectReader
+ get_stoichiometric_matrix(): array
+ get_model_bounds(): list[list[float]]

Figure 11: Overview of the class hierarchy for model representation structures in CoBAMP.

3.2.2.1 Gene-protein-reaction rules

A `GPRContainer` class was implemented to handle parsing and evaluation of Boolean rules. This class accepts a list of GPR rule strings representing a Boolean expression with AND/OR operators. To facilitate evaluation and interpretation, these are parsed using the Boolean.py library which is able to convert these expressions into disjunctive normal form.

This also allows representing and storing a single GPR rule as a list where each element is itself also a list of operands in an AND expression. Each of these sub-expressions is a part of a greater OR expression and, thus, evaluation is performed by replacing variables with a truth value and applying appropriate operators to the operands in each rule.

3.2.2.2 Model input/output

The framework agnostic design that is a main feature of CoBAMP elicited the need to implement adapter routines that could extract information from other models. Instead of implementing new parsers, adapter classes were implemented to extract this information from objects that already contain it in other frameworks. This allowed CoBAMP to be fully independent from established tools such as `cobrapy` or `reframed` while also being compatible with them.

An `AbstractModelObjectReader` abstract class was implemented as part of the wrappers.core submodule, defining a generic constructor with a model object from an external framework as the only mandatory input. This class enforces the implementation of several methods in its subclasses that are able to retrieve the reactions and metabolites data necessary to recreate a standard flux balance analysis LP problem. This process is then achieved through the `to_cobamp_cbm` method that generates a `ConstraintBasedModel` instance. Alternatively, the extracted data can be accessed separately.

The subclasses of `AbstractModelObjectReader` are implemented in separate submodules within the wrappers module, each corresponding to a constraint-based modelling framework. Figure 11 represents the definition of `AbstractModelObjectReader` and its subclasses as well as its role in generating the inputs needed to represent metabolic models. Additionally, support for these frameworks' simulation methods is guaranteed by a `ConstraintBasedModelSimulator` class where CoBAMP `Solutions` can be obtained from simulation methods outside of it.

3.2.3 Metabolic pathway analysis algorithms

CoBAMP features routines for the enumeration of metabolic pathway analysis concepts based on the k-shortest EFM algorithm [7]. Although this is the only algorithm whose computational demand is compatible with genome-scale models of human metabolism, the framework is extensible and follows a design model split in three parts:

- `Enumerator` object: Accepts a `LinearSystem` as input and other optional parameters as needed and returns one or more `Solution` objects. It should implement all routines required to enumerate the expected pathways.

- `Algorithm` object: Requires a `PropertyDict` object as input, containing the optional parameters required for the Enumerator object and implements a `get_enumerator` method, which accepts a LinearSystem object as input, validates the problem and returns an iterator that can call enumeration routines from the Enumerator class.

- Wrapper object: Higher-level object that accepts a model instance from any compatible CBM framework and several parameters with default settings. This class should include methods to decode Solutions from the LinearSystem into appropriate identifiers as featured in the metabolic model and a get_enumerator method that yields an iterator that can iteratively return these Solutions in native *Python* formats (such as dictionaries or lists) instead of Solution instances.

3.2.3.1 K-shortest algorithm

The K-shortest algorithm is able to enumerate elementary flux modes and can also be used by extension to enumerate minimal cut sets and elementary flux patterns. The implementation of this method reflects the modularity of this algorithm in the way that the KShortestEFMAlgorithm and KShortestEnumerator classes were used for all of these pathway concepts.

A KShortestEnumeratorWrapper abstract class provides a higher-level interface to create a pathway enumeration problem from a metabolic model object, validating parameters using the KShortestEFMAlgorithm class and providing an implemented get_enumerator method that is generic to all pathway concepts and returns an iterator that yields one or more KShortestSolution objects. The get_linear_system abstract method must then be implemented for each subclass, creating an appropriate LinearSystem object given the parameters supplied to the class' constructor. Figure 12 represents the three main classes implementing the K-shortest algorithm, method wrappers based on this architecture and the usage of the model reading features to simplify the inputs needed to run the algorithm.

The main difference between elementary flux modes and patterns, as described on Section 2.2.3.1, lies on a supplied subset of reactions upon which the latter concept is restricted to. This subset is applied by changing the dvars attribute on KShortestCompatibleLinearSystem. When enumerating EFMs, the IrreversibleLinearSystem class is used, including all available variables corresponding to fluxes into the dvars attribute, while the IrreversibleLinearPatternSystem allows for an additional subset parameter defining which fluxes to propagate into the dvars attribute. When this latter class is used, the KShortestEnumerator is initialized differently, including auxiliary constraints needed for EFP enumeration.

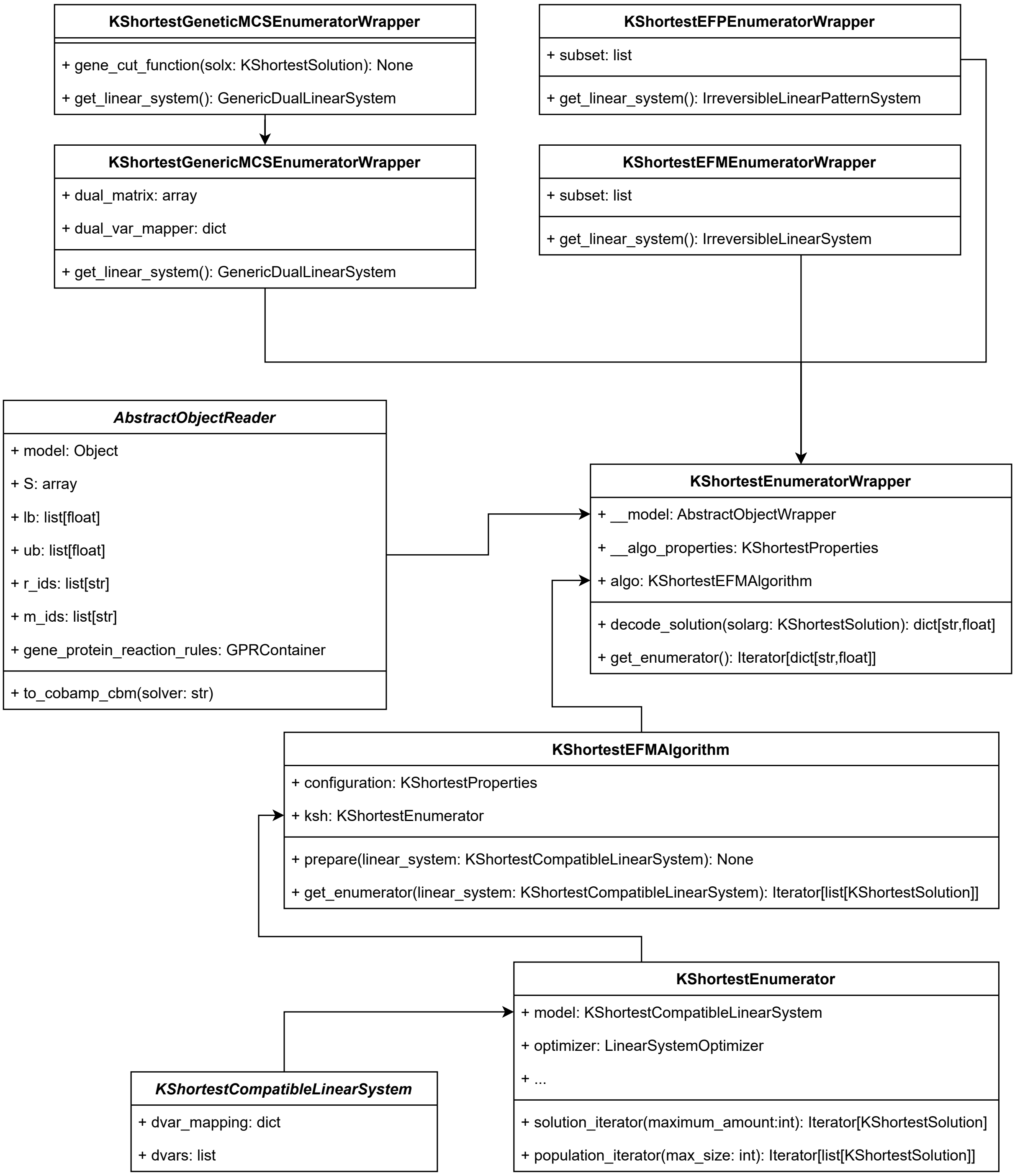

KShortestGeneticMCSEnumeratorWrapper
+ gene_cut_function(solx: KShortestSolution): None
+ get_linear_system(): GenericDualLinearSystem
KShortestEFPEnumeratorWrapper
+ subset: list
+ get_linear_system(): IrreversibleLinearPatternSystem
KShortestGenericMCSEnumeratorWrapper
+ dual_matrix: array
+ dual_var_mapper: dict
+ get_linear_system(): GenericDualLinearSystem
KShortestEFMEnumeratorWrapper
+ subset: list
+ get_linear_system(): IrreversibleLinearSystem
AbstractObjectReader
+ model: Object
+ S: array
+ lb: list[float]
+ ub: list[float]
+ r_ids: list[str]
+ m_ids: list[str]
+ gene_protein_reaction_rules: GPRContainer
+ to_cobamp_cbm(solver: str)
KShortestEnumeratorWrapper
+ __model: AbstractObjectWrapper
+ __algo_properties: KShortestProperties
+ algo: KShortestEFMAlgorithm
+ decode_solution(solarg: KShortestSolution): dict[str,float]
+ get_enumerator(): Iterator[dict[str,float]]
KShortestEFMAlgorithm
+ configuration: KShortestProperties
+ ksh: KShortestEnumerator
+ prepare(linear_system: KShortestCompatibleLinearSystem): None
+ get_enumerator(linear_system: KShortestCompatibleLinearSystem): Iterator[list[KShortestSolution]]
KShortestEnumerator
+ model: KShortestCompatibleLinearSystem
+ optimizer: LinearSystemOptimizer
+ ...
+ solution_iterator(maximum_amount:int): Iterator[KShortestSolution]
+ population_iterator(max_size: int): Iterator[list[KShortestSolution]]
KShortestCompatibleLinearSystem
+ dvar_mapping: dict
+ dvars: list

Figure 12: Overview of the class hierarchy for pathway analysis approaches based on the k-shortest elementary flux mode enumeration algorithm as implemented within CoBAMP.

The MCS enumeration features are implemented with a slightly more complex architecture but follows a similar structure as EFM and EFP enumeration. In this case, the `GenericDualLinearSystem` class is used as the `LinearSystem` to supply to the Enumerator class. To accommodate for both classical and genetic variations of MCSs, the generic definition presented by Apaolaza et al. [67] was implemented. This definition implies three main inputs, namely a metabolic model, a target phenotype and a dual matrix (DM) mapping primal with dual variables.

Additional data structures were created to easily translate these inputs into linear programming, namely the `AbstractConstraint` abstract class whose subclasses `DefaultFluxbound` and `DefaultYieldbound` can be used to define target phenotypes using, respectively, absolute flux and flux ratio constraints. These implement a materialize method that supplies an appropriate constraint as a matrix.

Finally, the DM matrices define the type of MCSs to enumerate. Classical MCSs map primal variables with two dual variables. For genetic minimal cut set (gMCS)s, the `gpr.integrate` module includes a `GeneMatrixBuilder` class that can build a dual-primal mapping matrix from an existing `GPRContainer`. This modular definition enables further extensions to the concept of MCSs by modifying the DM matrix.

3.3 TROPPO: A framework for context-specific metabolic model extraction and omics data integration

TROPPO is a novel library implementing several context-specific model extraction methods, essential for modelling human metabolism through omics data integration, in the Python programming language. This collaborative effort was necessary since many of these methods are available only through MATLAB, either as a part of the COBRA Toolbox or as separate scripts. The library can be divided in three main parts, namely, algorithms, omics data processing and model validation.

Omics data platform TROPPO implements features to process data with transcriptomics and proteomics measurements. It provides simple commands to import the data through the `pandas` library [134], an external *Python* package that assists in loading and manipulating tabular data from various formats. Omics measurements are standardised into data structures that function as inputs for the algorithms implemented in TROPPO. Identifier conversion features are also provided and genes present on the HUGO Gene Nomenclature Committee (HGNC) database can be easily converted into different nomenclatures.

Context-specific model reconstruction The main feature of TROPPO is the implementation of context-specific metabolic reconstruction (CSMR) algorithms in a *Python* environment. Several context-specific model extraction methods were reimplemented as a part of this framework in a modular architecture where new algorithms can easily be added, using standardised inputs. CoBAMP is used for these implementations and thus, TROPPO also shares the same compatibility with other frameworks. Model refinement is also achieved through gap filling methods.

Model validation The package also provides methods to validate context-specific models, leveraging the efficient batch simulation routines provided by CoBAMP to predict phenotypes for multiple models and biological scenarios with simple commands. Moreover, TROPPO also includes a task evaluation module and with an intuitive metabolic task definition. Tasks can also be loaded and exported into JavaScript Object Notation (JSON) files for later use.

3.3.1 Omics data processing

The class architecture for omics data handling in TROPPO is based on previous works by Correia et al. [135]. An omics module contains the data structures and routines required to load and process omics datasets. The main classes and data flow implemented in the `omics` module are represented on Figure 13 Firstly, each omics data sample is stored in an `OmicsContainer` object, which is a generic container with four fields:

- `data`: A dictionary mapping valid biological database identifiers with a numeric value;
- `condition`: A string identifying the sample;
- `nomenclature`: A string identifying the biological database to which the identifiers belong;
- `omicstype`: A string identifying the type omics data (e.g., transcriptomics).

This object contains several methods to deal with missing values, apply transformations and extract meaningful sets of genes or metabolites according to a user defined threshold. When dealing with transcriptomics, gene/transcript names can be converted into a desired nomenclature. The most important feature added to this class, however, is the `get_integrated_data_map` class method that generates `OmicsDataMap` objects containing reaction scores that can be integrated into constraint-based models. This method requires an `AbstractModelObjectReader` instance from CoBAMP which is then used to match the model's gene identifiers and GPR rules with the ones present in the OmicsContainer data field, finally returning an OmicsDataMap object with integrated scores.

The omics module also contains a readers sub-module implementing parsers for generic tabular data, as well as microarray and Human Protein Atlas proteomics data. Since most datasets of this sort are stored as 2-dimensional arrays, a TabularContainer class was also implemented to facilitate loading and storage of measurements and their associated biological entity and sample identifiers. The OmicsMeasurementSet subclass further expands it to allow conversion of dataset entries into OmicsContainer objects, while its TypedOmicsMeasurementSet subclass further allows for conversion of feature identifiers when provided with an IdentifierMap object, which is capable of converting biological identifiers for various databases when supplied with a mapping from resources such as HGNC [136].

Finally, an integration sub-module was added to handle OmicsDataMap objects and integrate them as inputs for context-specific model reconstruction algorithms. The ScoreIntegrationStrategy abstract class defines a template for that purpose, forcing the implementation of an integrate method that takes an OmicsDataMap object as input and returns an appropriate score input depending on the algorithm to be used. Five of these strategies were implemented as part of the context-specific metabolic reconstruction pipeline featured in this work.

3.3.2 Context-specific reconstruction and validation methods

The implementation of context-specific model reconstruction methods was organised in two distinct modules. The methods module contains algorithm implementations that are independent of metabolic model abstractions and mainly deal with inputs in numerical formats. A methods_wrappers module contains classes and mappings that allow the usage of these algorithms with higher-level containers such as objects representing entire metabolic models.

3.3.2.1 Model extraction and refinement algorithms

In TROPPO, the methods module contains algorithms that directly operate on constraint-based models or rely on the usage of linear programming formulations that use network topology and stoichiometry as inputs. The class structure and modular organisation of the algorithms implemented in troppo can be visualised on Figure 14. Both the ContextSpecificModelReconstructionAlgorithm and GapfillAlgorithm wrappers share a common constructor template, requiring a stoichiometric matrix,

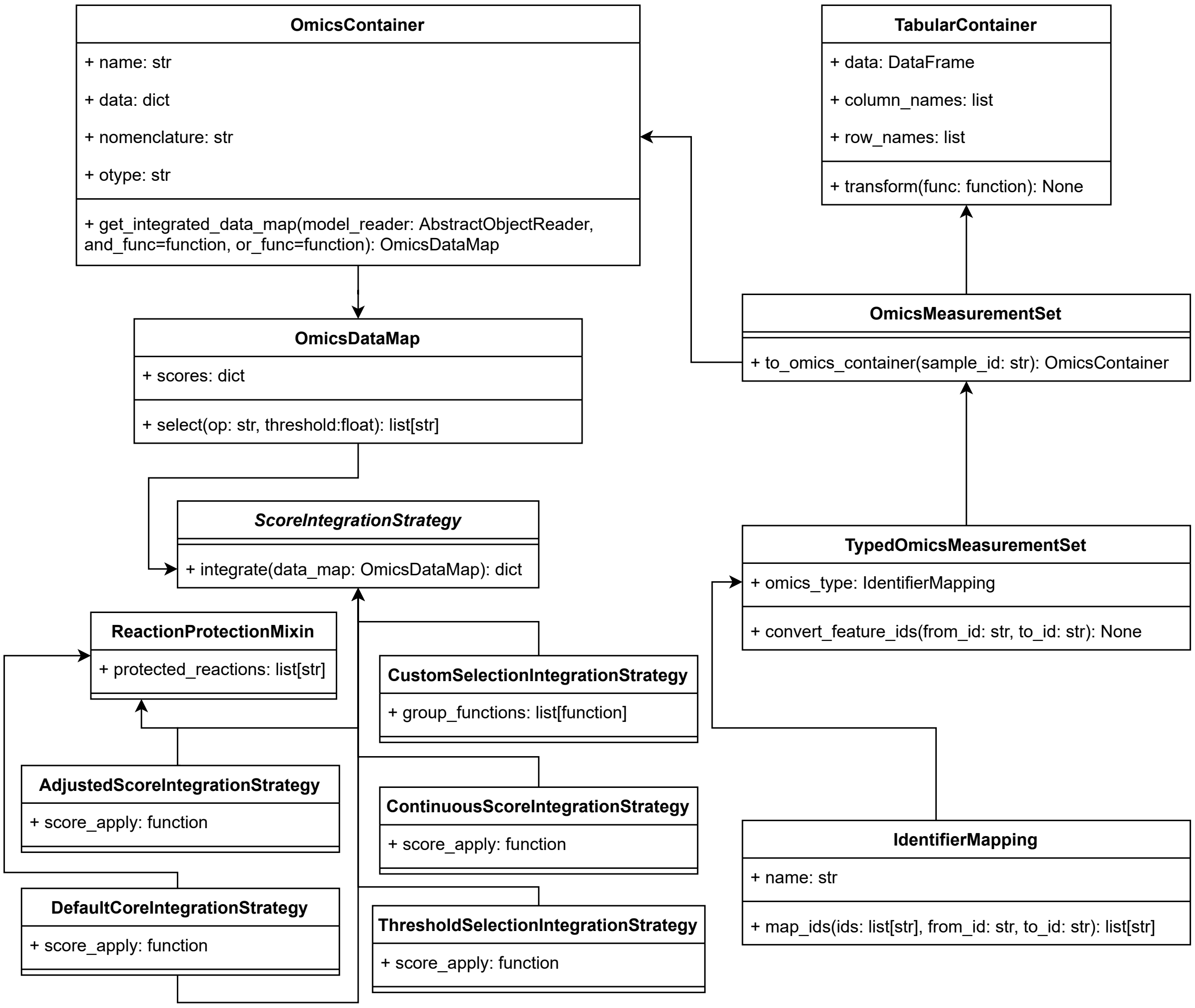

OmicsContainer
+ name: str
+ data: dict
+ nomenclature: str
+ otype: str
+ get_integrated_data_map(model_reader: AbstractObjectReader, and_func=function, or_func=function): OmicsDataMap
TabularContainer
+ data: DataFrame
+ column_names: list
+ row_names: list
+ transform(func: function): None
OmicsMeasurementSet
+ to_omics_container(sample_id: str): OmicsContainer
OmicsDataMap
+ scores: dict
+ select(op: str, threshold:float): list[str]
ScoreIntegrationStrategy
+ integrate(data_map: OmicsDataMap): dict
TypedOmicsMeasurementSet
+ omics_type: IdentifierMapping
+ convert_feature_ids(from_id: str, to_id: str): None
ReactionProtectionMixin
+ protected_reactions: list[str]
CustomSelectionIntegrationStrategy
+ group_functions: list[function]
AdjustedScoreIntegrationStrategy
+ score_apply: function
ContinuousScoreIntegrationStrategy
+ score_apply: function
IdentifierMapping
+ name: str
+ map_ids(ids: list[str], from_id: str, to_id: str): list[str]
DefaultCoreIntegrationStrategy
+ score_apply: function
ThresholdSelectionIntegrationStrategy
+ score_apply: function

Figure 13: Overview of the omics data processing layer implemented in TROPPO.

flux bound vectors and an object containing algorithm specific properties. The output of the mandatory run class method is always a vector of Boolean values, one for each flux in the input model, representing a reaction's state as absent (False) or present (True) in the reconstructed model for the supplied context, which is passed as a property in the appropriate class.

Algorithm properties are represented as `PropertiesReconstruction` or `GapfillProperties` instances which are subclasses of CoBAMP's `PropertyDictionary` class, which encapsulates a dictionary mapping property identifiers with their respective values. This implementation allows the definition of mandatory and optional properties as well as type checking. Every algorithm in TROPPO requires the implementation of the appropriate subclass to accommodate for the specific inputs of each method. By defining the omics data input as a property, the same parent class can be used for algorithms with vastly different input formats, which effectively detaches the input processing step from the model extraction

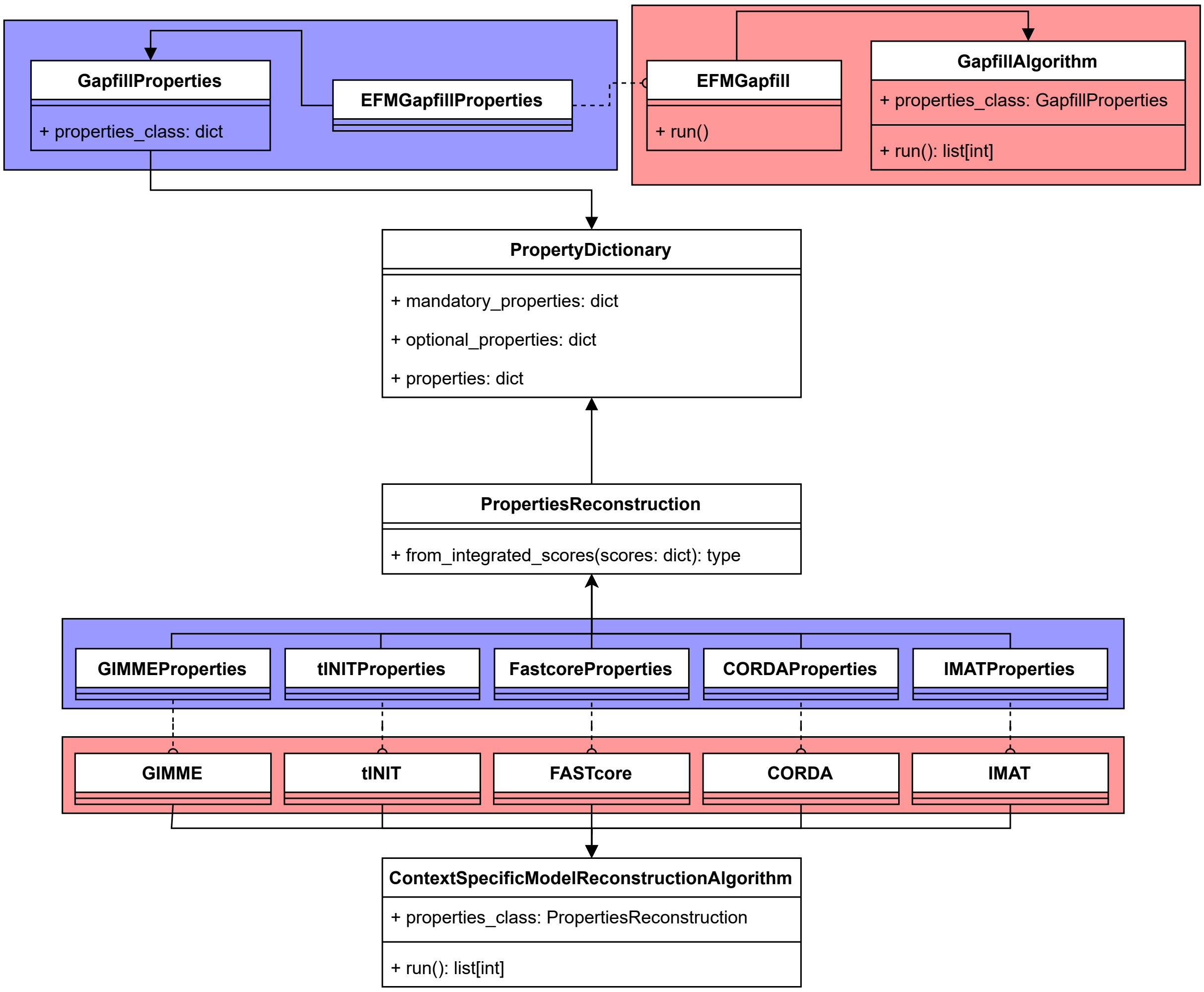


Figure 14: Overview of the class structure for the model extraction and refinement methods implemented in TROPPO.

step.

`PropertiesReconstruction` defines a single mandatory property, a string that represents the linear programming solver to be used for optimizations. `GapfillProperties` is similar, but additionally implements the `lsystem_args` optional property that influences the steady-state balance of the model (useful in task-based gap filling) and `avbl_fluxes` which is an optional list of fluxes that should be considered as present prior to the execution of the gap filling algorithm.

3.3.2.2 Context-specific method wrappers

With a generic definition of a context-specific model extraction algorithm, the entire pipeline can be made available through a higher-level wrapper class that includes routines to use an `AbstractModelObjectReader`, an `OmicsContainer` and algorithm specific properties to return an algorithm's output.

Firstly, a `ModelBasedWrapper` class is defined, containing routines to extract appropriate inputs from an `AbstractModelObjectReader` class, namely the stoichiometric matrix and flux bound vectors, which will be needed to create algorithm instances. The subclasses of this abstract class must then implement run methods that can interpret `PropertyDictionary` subclass instances and associate their type with the correct algorithm. To do this, the methods_wrappers module contains several dictionaries which are constantly updated as more methods are implemented. These dictionaries are:

- `map_properties_algorithms`: Maps `PropertyDictionary` subclasses with `ContextSpecificModelReconstructionAlgorithm` subclasses;
- `algorithm_instance_map`: Maps strings with the algorithms' names with the appropriate `ContextSpecificModelReconstructionAlgorithm` subclass;
- `integration_strategy_map`: Maps strings with a scoring strategy's name with the associated `ScoreIntegrationStrategy` from the `omics.integration` module.

Context-specific model extraction methods can be used through the `ReconstructionWrapper` class, a subclass of `ModelBasedWrapper` which implements a `run_from_omics` method whose mandatory arguments are:

- `omics_data`: An `OmicsContainer` instance with loaded data, an iterable with a numeric score for each of the model's reactions or a dictionary mapping reaction identifiers with their activity scores;
- `algorithm`: A string representing an algorithm within the keys of `algorithm_instance_map`;
- `integration_strategy`: A `ScoreIntegrationStrategy` instance or a tuple with two elements, namely the integration strategy name found in `integration_strategy_map` and the mandatory parameters for its construction;
- `and_or_funcs`: A tuple with two functions to replace, respectively, the and and or Boolean operators.

This method encompasses part of the input preprocessing and model reconstruction steps in the pipeline detailed in Chapter 4. Its output is a processed dictionary mapping string identifiers for each reaction in the model with the associated Boolean presence flag. A similar class architecture is employed

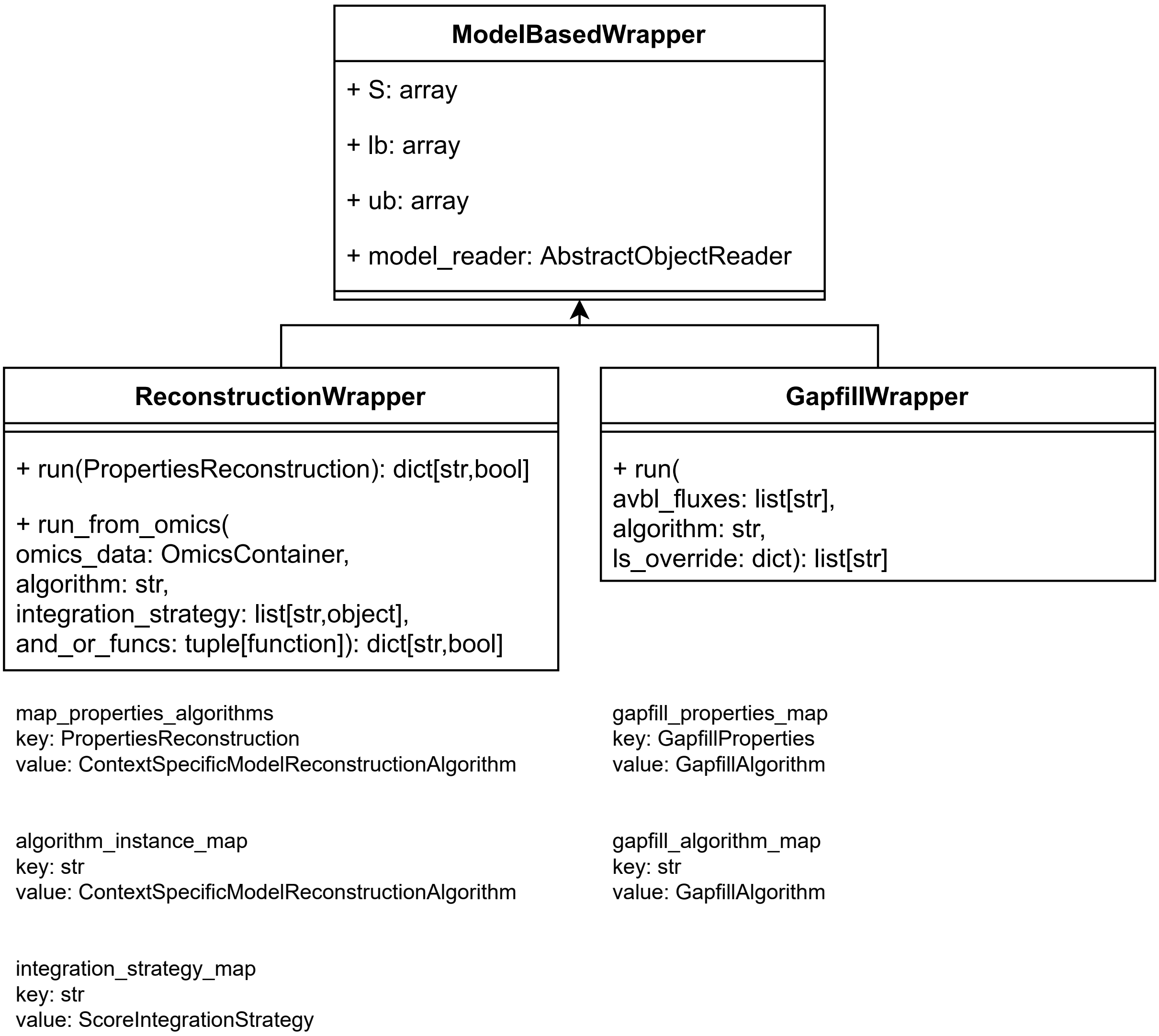

ModelBasedWrapper
+ S: array
+ lb: array
+ ub: array
+ model_reader: AbstractObjectReader
ReconstructionWrapper
+ run(PropertiesReconstruction): dict[str,bool]
+ run_from_omics(
omics_data: OmicsContainer,
algorithm: str,
integration_strategy: list[str,object],
and_or_funcs: tuple[function]): dict[str,bool]
GapfillWrapper
+ run(
avbl_fluxes: list[str],
algorithm: str,
ls_override: dict): list[str]
map_properties_algorithms
key: PropertiesReconstruction
value: ContextSpecificModelReconstructionAlgorithm
algorithm_instance_map
key: str
value: ContextSpecificModelReconstructionAlgorithm
integration_strategy_map
key: str
value: ScoreIntegrationStrategy
gapfill_properties_map
key: GapfillProperties
value: GapfillAlgorithm
gapfill_algorithm_map
key: str
value: GapfillAlgorithm

Figure 15: Overview of the class structure for the higher-level wrappers used to simplify the process of reconstructing and refining models with TROPPO.

for gapfilling methods, where a `GapfillWrapper` class implements a run method with the same input parameters as those defined in the `GapfillProperties` class along with an algorithm string present in which defines the gapfilling algorithm present the keys of a `gapfill_algorithm_map` that maps algorithm names with their classes. The organisation of these wrappers, parameters and data structures registering the various algorithms are depicted on Figure 15

3.3.3 Model validation

Model validation in TROPPO can be performed either through simulation with a phenotype prediction method (validation sub-module) or through metabolic tasks (`tasks` module).

3.3.3.1 Phenotype prediction

Constraint-based model simulation tools are already available through Python frameworks such as COBRApy, ReFramed and even CoBAMP, and thus, TROPPO only implements a ContextSpecificModelSimulator class with a simulate method to provide an abstraction for the flux bound contexts from context-specific model extraction methods and environmental conditions.

The ContextSpecificModelSimulator class requires a ConstraintBasedModelSimulator object from the CoBAMP framework, a scenarios dictionary that maps environmental condition names with dictionaries mapping flux identifiers with upper and lower bounds and a post_process optional argument where a function can be passed to further process the simulation results into a desired format. The implemented simulate method is similar to that found in ConstraintBasedModelSimulator's batch_simulate method, requiring objective coefficients and a simulation function (phenotype prediction method) as well as an additional contexts dictionary that maps sample names (strings) to the outputs of ReconstructionWrapper's run_from_omics method (dictionaries mapping flux identifiers with Boolean values). Since batch_simulate is used, these optimizations are always run in parallel.

3.3.3.2 Task evaluation

Metabolic task evaluation is a feature that is present in both the COBRA and RAVEN toolboxes for MATLAB, but is missing in *Python* frameworks. Furthermore, the definition of metabolic tasks for these frameworks differ, which elicited the development of a generic data structure for loading and evaluating them. The implemented tasks module contains a core sub-module that implements a Task object where the metabolic task definition is stored and a TaskEvaluator object that uses CoBAMP to efficiently evaluate metabolic tasks in parallel and a task_io object that implements routines to read and write tasks from JSON and Microsoft Excel ©formatted files.

Parsers and writers are combined in the same TaskIO abstract class, requiring the implementation of read_from_string and write_to_string which accept a string and a task as input and return a Task object and a string, respectively. This abstract class already implements the read_task and write_task methods that call the methods described above to perform reading and writing operations according to the formatting implemented on their subclasses. JSON was chosen as the primary format for reading and storage as it can be easily mapped into Python data structures, although a Microsoft Excel ©parser is also implemented since most available task lists are present in this format.

The Task class is the main component of this architecture. It defines a task as a set of flux conditions that are then converted into commands to modify an existing `ConstraintBasedModel` instance from CoBAMP to properly assess the metabolic task. From the definition of metabolic tasks by both Thiele, Agren and Richelle [34, 137, 138], a `Task` object requires, optionally, any combination of the following inputs:

- `name`: String containing the task's name;
- `annotations`: A dictionary with optional annotations such as extended descriptions;
- `should_fail`: Boolean flag that determines whether the task is supposed to fail;
- `reaction_dict`: Reactions to add to the metabolic model represented as a dictionary mapping the new reactions' names with a dictionary containing the stoichiometric coefficients for each metabolite and a pair of numerical lower and upper bounds;
- `flow_dict`: A dictionary with names of metabolites that interact with the extracellular medium mapped to lower and upper bounds. This parameter is implemented as two separate dictionaries, namely `inflow_dict` and `outflow_dict` for uptake and secretion, respectively;
- `mandatory_activity`: A list of reactions that are expected to be active when evaluating a task. While this does not affect the evaluation itself, it can be used for further analysis.

The Task object contains auxiliary methods for manipulation of reaction identifiers and evaluation of previously determined flux distributions. Although there are methods to evaluate a single task on a `ConstraintBasedModel` object, its main purpose is to provide a way to store the task's definition and instructions about how the metabolic model must be modified for it to be evaluated. This is achieved through the `get_add_reaction_cmds` which uses partial functions to queue calls to `ConstraintBasedModel`'s reaction addition functions and the `get_task_bounds` function which returns a dictionary mapping flux identifiers with flux bounds to be changed.

With at least one Task object and a `ConstraintBasedModel` or `AbstractModelObjectReader`, a `TaskEvaluator` can be instantiated, using a CoBAMP metabolic model as a basis for the optimizations required for task evaluation. Although this object loads all supplied tasks, only one can be evaluated at once. For this reason, a `current_task` class attribute holds a string that represents the `Task` object's name field and can be set by the user to change the task to be tested. All modifications required for each task are introduced into the model upon calling the constructor, a process that would otherwise incur in a significant amount of time if it was done iteratively for each task.

Any given task can then be evaluated by setting the `current_task` parameter to the name string associated with the `Task` object. While this is inherently single-threaded, for large sets of flux bounds to be tested, a `batch_evaluate` function is supplied, allowing parallel evaluation of these contexts for each task, which, although not truly parallel, significantly improves performance.

The output for any `TaskEvaluation` result is a tuple with 3 elements, namely the task evaluation status in relation with the expected outcome (`should_fail` parameter), a dictionary mapping flux identifiers in the `mandatory_activity` parameter with their status as active or inactive, as well as the flux distribution generated as part of this evaluation.

3.4 GRASP: Integrating causal and Boolean network logic in constraint-based models

The implementation of CoBAMP as a generic constraint-based modelling framework is also complemented by the GRaSP package, which adds the ability to load causal or Boolean networks as graph representations for further integration. Although the only phenotype prediction method, designed as a result of this work, is also reliant on omics data, the mathematical representations used for regulatory and signalling networks are fundamentally different than those found on constraint-based models. As such, GRaSP is a separate entity that is completely optional and not a part of either CoBAMP or TROPPO.

3.4.1 Graph data structures

The first step towards the inclusion of causal interaction graphs in constraint-based models was the implementation of classes that could represent a graph and allow for its manipulation.

3.4.1.1 Graph representation

Formally, a graph $G = (V, E)$ is assumed as a collection V of vertices (or nodes) connected through a set of edges E. In GRaSP, both edges and nodes are represented as objects that always implement two abstract classes that ensure a string identifier (`IdentifiableEntity`) and an annotations field with miscellaneous information (`AnnotateableEntity`) are present.

A `Node` object appropriately represents a single vertex, and can only hold the same fields defined in its parent classes. An `Edge` object, however, contains two additional mandatory properties, namely the source and incident vertices, which must be `Node` instances. `Edge` objects can be subclassed, with a `WeightedEdge` subclass representing an edge with an associated weight.

The two data structures for representing vertices and edges allow us to define a `GenericGraph` class that attempts to represent any simple graph such as G. Since these representations are only needed to store information, there is no distinction between directed and undirected edges. The `GenericGraph` class can thus be instantiated using only a collection of `Node` and `Edge` objects as posed by the simple definition $G = (V, E)$. However, this graph representation includes `add` and `remove` methods to manipulate the graph and perform critical checks such as verifying whether valid Node and Edge objects are being added. Furthermore, a `remove_unconnected_edges` method ensures the graph is consistent so that every edge connects vertices that have been added to the graph.

3.4.1.2 Input methods

A parser was implemented to store graph structures into organized files. The chosen format for these operations is a simple variation of the Simple Interaction Format (SIF) format already used in other network analysis tools. This tabular format assumes three columns, namely `source`, `weight` and `target` where each row is a single graph edge. In each row, the `source` and `target` columns are, respectively, the source and incident nodes of the edge while the `weight` column is meant to be a numerical value which can be used, as an example, to encode the interaction type. A `SIFParser` class implements a `parse` method that accepts the table as a `pandas DataFrame` instance and returns a `GenericGraph` object with new `Edge` instances as encoded in the table and new `Node` objects inferred from the unique elements of the `source` and `target` columns.

3.4.2 Integrated modelling of signalling, gene regulation and metabolism

The input data structures implemented in GRaSP, as well as the constraint-based model representation available in CoBAMP, provide the necessary tools to build models with extended model representations spanning several biological layers. These features are included in the `integration` module containing two important sub-modules: (1) the `models` sub-module extends CoBAMP's constraint-based models to include additional layers while the (2) `simulation` sub-module implements novel phenotype prediction methods based on them.

3.4.2.1 Modelling framework

In this work, integrated models are formalised using the same mathematical principles as constraint-based models. As such, extensive use of the CoBAMP framework was essential to simplify the implementation of integrative approaches since its `ConstraintBasedModel` class already implements several features essential to manipulate and optimize the linear programming problems it represents. A two-step hierarchy was devised so that there is an `IntegratedGPRModel` class that integrates gene expression and an `IntegratedGPRCausalModel` subclass that further expands it to include causal interactions.

The `IntegratedGPRModel` class is itself a subclass of `ConstraintBasedModel` and operates in a similar fashion. However, its constructor accepts a `ConstraintBasedModel` instance with gene-protein-reaction rules which then serves as a template to build a new LP problem integrating gene expression. It additionally contains several fields that are inferred from the input model, such as the metabolites and reactions that are exclusively part of the metabolic model. A `get_integrated_gpr_model` function implements the process described in 5.2.1, accepting a `ConstraintBasedModel` and returning a new instance with the expanded system. The output of this function is then used by the constructor to initialize the `IntegratedGPRModel` with the new pseudo-reactions and pseudo-metabolites.

Similarly, the `IntegratedGPRCausalModel` class has a constructor with the exception of an additional `graph` parameter which must be a `GenericGraph` instance. Also similarly to the previous description, there is a `merge_linear_with_causal_model` function which takes the same parameters but this time modifies the existing model instance to include the causal interactions present in `graph`, integrating them as reactions and metabolites. The constructor of `IntegratedGPRCausalModel` calls its parent class' constructor and then calls the `merge_linear_with_causal_model` referencing itself as the model input. This class defines an additional property named `causal_interaction_edges` encoding the causal interactions added to the model as a dictionary mapping reaction identifiers with the gene pseudo-metabolites it connects.

3.4.2.2 Phenotype prediction

The model representations in this work were developed primarily to be used with the integrated parsimonious flux balance analysis (ipFBA) approach presented in Chapter 5. The routines necessary to implement this method were integrated in the `simulation.ipfba` sub-module within the `integration` module of the GRaSP framework.

The computational requirements of the LP problem formulated in `IntegratedGPRCausalModel` instances, as well as the additional constraints of ipFBA may exceed the memory limits of many consumer grade computers. The `in silico` experiments carried out using the Human-GEM metabolic model often surpassed 10GB. To mitigate this, all ipFBA simulations are performed using the LP problem that is generated through the `IntegratedGPRCausalModel` so that it is not duplicated in the process of performing a new simulation.

The ipFBA approach is implemented by the `IPFBA` class whose mandatory constructor arguments imply the existence of an `IntegratedGPRCausalModel` instance and a collection of `SimulationObjective` classes encoding objective constraints as defined by the method. The constructor then stores this model in a `model` field and calls the `prepare_ipFBA_model` routine which adds appropriate constraints as per the definition of ipFBA

Objective constraints are stored as `SimulationObjective` instances. This class accepts LP objectives as input, encoded using the following arguments:

- A dictionary of variables (reactions) mapped to floating-point values denoting its coefficient in the objective function.
- A Boolean flag indicating whether this is a minimization or maximization objective
- A string with the objective's name

The `SimulationObjective` class then implements a `get_bounds` method which accepts a `ConstraintBasedModel` instance and optimizes the LP problem contained in the instance, yielding the objective's value. These methods are used by the `IPFBA` upon instantiation to pre-calculate the bounds for the objective constraints so that they can be then adjusted before simulating.

Finally, the `run` method accepts a set of objective bounds, which are lists of tuples encoding the range of optima allowed in the simulation, gene and expression constraints as dictionaries mapping gene names to positive floating point values and finally, optimization weights as dictionaries mapping model reactions to objective function coefficients which are then use to set the global objective function of the LP problem.

4

A pipeline for large-scale reconstruction and validation of context-specific models

The work presented in this chapter was developed in co-authorship with Jorge Ferreira as part of a publication that was submitted and is now pending review.

4.1 Introduction

Context-specific models can be particularly useful for researching cancer metabolism, since they have been shown to be able to simulate rapid growth, mutations in metabolic genes and the Warburg effect (aerobic glycolysis) [139]. Several methods were developed to build draft cell/tissue-specific metabolic models throughout the past years, taking as inputs a template generic GSMM and different types of omics data.

The steps involved in the reconstruction of tissue-specific metabolic models cover various tasks ranging from omics data preprocessing to model reconstruction and validation. The complexity of these steps elicited the development of pipeline to successfully integrate omics data and create models capable of successfully predicting cell metabolism. Previous works by Richelle have described methods and key considerations to take into account when building such models [138, 140].

The challenges and multiplicity of algorithms and parametrisation involved in this process elicited the development of a generic pipeline for context-specific model reconstruction, allowing the assessment on the impact of different data preprocessing and algorithm choices by validating models against two datasets with

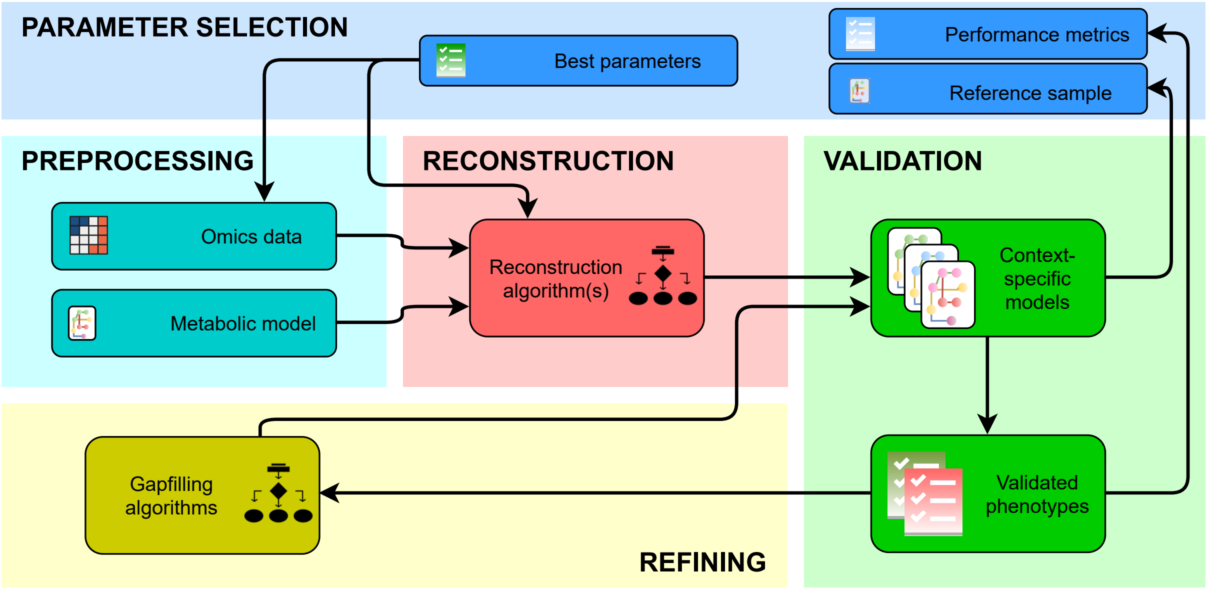


Figure 16: Overview of the context-specific model reconstruction pipeline implemented as part of this work. The inputs for any reconstruction start with a metabolic model and omics data which are integrated through reconstruction algorithms into context-specific models that can optionally be refined. When more than one sample is available, a reference sample can be used to fine tune parameters according to predefined performance metrics.

phenotypic data. In a first set of experiments, this pipeline was applied to generate multiple reconstructions of the MCF-7 breast cancer cell line using recent transcriptomics data and knockout screenings from the Cancer Cell Line Encyclopedia (CCLE) [141–143], as well as fluxomics and proteomics data from a recent work by Katzir et al [144].

4.2 Methods

The methods implemented as part of this work consist in a context-specific metabolic reconstruction (CSMR) pipeline containing four essential steps: input preprocessing (1), context-specific reconstruction (2), refinement (3), validation (4) and an optional parameter calibration step that is only used for large-scale reconstructions. The pipeline and interactions between its different components are highlighted on Figure 16.

4.2.1 Input preprocessing

The inputs for any context-specific reconstruction always involve a template genome-scale metabolic model capable of yielding a non-zero flux distribution and only containing reactions capable of carrying non-zero flux, as well as a set of omics measurements integrated in the model via CSMR algorithms.

4.2.1.1 Model preprocessing

We first ensure the model is consistent by identifying blocked reactions - whose maximum and minimum fluxes are null under open exchange conditions - with flux variability analysis, using the `find_blocked_reactions` function from the COBRApy package. We also remove any boundary metabolites - usually added to balance exchange reactions - prior to running any reconstruction, gapfill, or analysis method. This model must be feasible in steady-state conditions and must be capable of allowing flux through the biomass pseudo-reaction.

4.2.1.2 Transcriptomics data as a proxy of enzyme activity

In this pipeline, we focus specifically on transcriptomics data that is mappable with the model's gene associations, although some methods allow integration of other data types.

Similarly to previous approaches [41, 138, 140], we use transcriptomics data as a proxy for enzyme presence and flux activity from which we can calculate reaction activity score (RAS) to serve as inputs for CSMR algorithms. These scores should ideally reflect whether a given reaction in the model is likely to be present in the context represented by the transcriptomics data. The work of Richelle et al. details the implications of several ways to infer glsRASs and provides several thresholding options [140], which we adapted as a part of our work. Out of the parameterisation choices highlighted by the authors, we focused on varying the thresholding approach and GPR integration functions.

Using transcriptomics to characterize enzyme activity is not trivial, since the relationship between messenger RNA and protein expression is not fully understood, despite ongoing progress in quantifying both of these biological entities. However, Nusinow et al. have recently quantified the proteome for a subset of the CCLE panel and found a moderately positive correlation (mean Pearson c.c. = 0.48) between mRNA and protein abundance [145]. In this work, we assume a linear relationship so that RASs are calculated based on gene expression measurements.

4.2.1.3 Scoring transcript activity from expression measurements

RNA-Seq technologies typically produce transcript-level measurements represented as proportions of the entire transcriptome. However, we intend, for the RASs used in this work, to obtain a positive or negative value relative to reference thresholds calculated across all samples. To this end, we process expression measurements into a transcript activity score transcript activity score (TAS) that can better represent this dichotomy.

We first define the concept of a global threshold, where the expression of all genes contribute. This is useful in filtering out transcripts whose expression is low or high enough for them to be assigned as inactive or active, respectively, with a high degree of confidence. However, this type of thresholding does not take into account the variability of measurements between transcripts. While originally available for microarray-based transcriptome quantification [146], a generic expression barcode for all cell types is hard to define and apply in RNA-Seq measurements, due to the different conditions in which experiments are performed.

A local thresholding approach can also be considered to mitigate the aforementioned problems. Rather than condensing the entire measurements into a single value, thresholding can be performed on a per gene basis, yielding a value for each gene independently. Similarly to the global thresholding approach, local thresholds can also be used as a reference for fold change calculations.

In this pipeline, both thresholds are calculated by first determining transcript-wise quantiles for various percentages (from 10% to 90%), yielding multiple sets of local thresholds, one for each percentage. To convert the latter to global thresholds, we use the mean value of a local threshold set to obtain a single representative value for the entire expression dataset.

After establishing appropriate thresholds, we can then combine them to establish rules that can be used to determine whether a transcript is active and calculate its TAS to better represent that activity level. We implemented an approach based on the work of Richelle et al [140], where transcript activity can be represented in two main states, namely:

- Inactive: transcript expression is inactive with a high degree of confidence - assigned when expression values are lower than a global lower threshold (g_{min}). The expected TAS values will always be negative.

- Active: transcript expression is active with a high degree of confidence since its value exceeds a global upper threshold (g_{max}). The expected TAS values are always positive.

This also implies the existence of an intermediate state of uncertainty for cases where transcript expression lies between the two thresholds. In these cases, we distinguish between active and inactive transcripts by comparing the transcript's expression with its transcript-specific local threshold ($l(y)$) and

Table 7: Functions used to convert transcript expression values into transcript activity scores, assuming x as a vector of expression levels for each transcript. The "Expression value" column represents the condition that values in x must meet for the corresponding reference threshold t used to calculate a ratio with the formula $log(\frac{x_i}{t})$. Finally, the range of TAS values for each condition is detailed on the last column. * In the local 2-state strategy, the TAS range does not start at 0 since 1 is added to the formula in the specific case when the expression value is greater than g_{max}

Strategy	**Expression value**	**Reference threshold**	**TAS range**
Global	$x \geq 0$	g_{max}	$]-\infty, \infty[$
Local 1-state	$x \leq g_{max}$	g_{max}	$]-\infty, 0[$
	$x \geq g_{max}$	$l(y)$	$]-\infty, \infty[$
Local 2-state	$x \geq g_{max}$	g_{max}	$[1, \infty]^*$
	$x \leq g_{min}$	g_{min}	$]-\infty, 0[$
	$g_{min}x < g_{max}$	$l(y)$	$[-1, 1]$

the expected TAS value is constrained between -1 and 1, with its sign reflecting whether it is considered active.

We implemented three thresholding strategies based on the work of Richelle et al [140] with some minor changes to accommodate for the expected distribution of TAS values across multiple states. A global thresholding strategy implies a single global threshold to distinguish between active and inactive transcripts. An extension of this strategy, named local 1-state, includes local thresholding for transcripts that would otherwise be considered as inactive, and assigns TASs based on the ratio between expression and the transcript's local threshold. Finally, we also included a local 2-state strategy defined by the usage of two thresholds and an intermediate state, as defined above.

TASs are then generated by calculating the ratio between measured expression values and an appropriate threshold which is chosen according to the state in which the transcript is assigned. A detailed description of the formulae used in each state for the three employed strategies can be found on Table 7.

4.2.1.4 Inferring reaction activity from transcript scores

The TASs from the aforementioned strategies are then converted to RASs using the GPR rules provided with the model. GPR rules are Boolean expressions that describe, for a given reaction, which combinations of transcripts are involved with the synthesis of one or more enzymes capable of catalysing it. These rules are often expressed or can be converted into disjunctive normal form, where multiple conjunctions (expressions with the AND operator) denoting the various enzymes or isoforms involved are bound by a disjunction (OR operator).

RASs must be presented as continuous scores and, thus, the Boolean operators in GPR rules must be replaced with numerical values. We replaced AND operators with a *minimum* function - an enzyme's activity is limited by the lowest expressed transcript/subunit - while OR operations could be replaced with either *sum* or *maximum* functions. When using *sum*, we assume the reaction activity correlates with the combined activity of all enzymes and isoforms catalysing it, while the *maximum* function equates reaction activity with the highest expressed enzyme's score.

4.2.2 Model reconstruction

4.2.2.1 Normalizing inputs for context-specific reconstruction algorithms

This step includes conversion of RASs into inputs accepted by the different CSMR algorithms, given their diverse nature, and we have implemented two alternatives to perform this conversion in our routines. Although our pipeline is generic, we have chosen the FASTCORE [99] and Task-driven Integrative Network Inference for Tissues (tINIT) [34] algorithms for context-specific model reconstruction.

Methods such as tINIT , where scores mirror the reactions' states as present or absent, can take RAS as input without any further processing. On the other hand, algorithms such as FASTCORE require a set of core reactions as input. In this case, a further threshold must be applied for these to be obtained. In our work, we emphasized the division between positive and negative scores to represent activity, and as such, core reactions are those with a RAS above 0.

4.2.2.2 Algorithm output and post-processing

With the inputs appropriately adapted, the output of each algorithm is always a binary vector r of size n (equal to the number of reactions in the template model), indicating reactions' presence. Indeed, this vector includes Boolean flags indicating whether each reaction should be kept or removed in the context-specific model.

The models generated by the CSMR algorithms are then checked for consistency with expected phenotypes. For each of these models, we knockout (set lower and upper bounds to 0) reactions flagged for removal before to perform any simulation or analysis. We first check whether the model is capable of allowing non-zero flux through the biomass reaction, to ensure lethality can be tested. When growth medium formulations are available, we can additionally ensure that the model is feasible and capable of growth if the compounds present in growth media are the only ones allowed to be consumed. This is achieved by constraining exchange reactions that do not involve medium metabolites to only allow positive flux values, thus only allowing medium metabolites to be consumed by the model.

4.2.3 Refinement

When the preliminary checks described above fail, gap fill approaches can be employed to infer sets of missing reactions that can expand the solution space and enable expected phenotypes.

4.2.3.1 Elementary flux mode-based gap filling approach

Gap filling was performed using a novel EFMGapfill approach, which was implemented in the *troppo* Python package. This algorithm leverages efficient elementary flux mode (EFM) enumeration algorithms to find minimal sets of active fluxes required for feasibility under a specific condition. Assuming a stoichiometric matrix S of m metabolites and n fluxes, the flux vector v and an identically sized vector y, and a set K of reactions available to fill gaps, the LP formulation employed in EFMGapfill can be defined as follows:

$$\begin{aligned}
\text{min} \quad & \sum_{p \in K} y_k && && && (4.1)\\
\text{s.t.} \quad & \sum_{j=1}^{n} S_{i,j} \cdot v_j &=& \; 0 && (\forall i \in \{1, ..., m\}) && (4.2)\\
& y_k - M v_k &\geq& \; 0 && (\forall k \in 1, ..., n) && (4.3)\\
& v_k - y_k &\geq& \; 0 && (\forall k \in 1, ..., n) && (4.4)\\
& v_k &\geq& \; 0 && (\forall k \in 1, ..., n) && (4.5)\\
& v \in \Re^n_{0+}, y \in \{0, 1\}, M = 10^6 && && && (4.6)
\end{aligned} \qquad \text{(LP1)}$$

In the formulation represented in LP 4.1, constraint 1 defines the steady-state constraint, similarly to other constraint-based approaches, such as FBA. Constraints 2 and 3 associate the binary variables in y to the fluxes in the vector v. In this expanded solution space, the variables in y will hold a value of 1 if their associated flux in the vector v is greater than 1. Otherwise, both variables are set to 0. These variables and indicator constraints are then used to discretize fluxes into active and inactive states. The objective function is dependent on the set of reactions K available for the algorithm to add as a gap filling solution, although the objective is always to minimize the sum of a subset of the vector y whose indices are contained in K.

We adapt the input K according to a Boolean vector r (a set of reaction indices). K will have all reactions from the template model not included in r. The vector r is typically the output of a previously determined CSMR reconstruction. Furthermore, we also define and constrain an objective reaction u (usually the biomass pseudo-reaction) to always carry non-zero flux, representing a phenotype that is expected to be maintained upon tailoring the model to the subset of reactions in r.

We identified two possible gap filling scenarios that can be accomplished using this approach. In the first, we do not assume constraints on external metabolite exchanges and thus, we also exclude these reactions from K. The resulting solution from our gap filling approach is the smallest set of intracellular reactions not found in r that should be included so that the context-specific model is capable of carrying flux through u.

An alternative scenario may arise where the set of reactions r must not be manipulated, but the model still requires gap filling to predict growth. The growth medium, rather than the enzyme content of this model must be the target for manipulation. In this case, all intracellular reactions not in r must be constrained and exchange reactions must be split into forward and reverse reactions carrying flux in opposite directions. To find the minimal set of extracellular metabolites required for the model to carry flux through u, the set K must be defined as the set of reverse exchange reactions in the model.

4.2.4 Validation

An important question arising from any CSMR process is ensuring the reconstructed models are capable of capturing the metabolic context of the cell or tissue, as represented by their corresponding omics measurements. Although literature review may reveal expected behaviours and phenotypes associated with the specific context to be modeled, a truly systematic validation of these models can only be achieved with large-scale datasets covering a wide range of measured biological entities. Such experiments should clearly point out the effect of perturbations that can be mapped onto the model on cell metabolism so that simulated fluxes become directly comparable. In this section, we describe how gene knockout screens and fluxomics can be integrated in our pipeline to validate these models.

4.2.4.1 Gene essentiality

Gene essentiality screens, such as those performed with CRISPR, provide a directly quantifiable measurement of the impact of gene deletions on cell viability, which can be modeled on CBMs through metabolic tasks [41]. The biomass objective function, included in most human models, groups most of these tasks' demands by aggregating the necessary components for cell division and maintenance. Given the computational demand of checking multiple gene knockouts for each task and each omics sample, we will focus on predicting lethal gene knockouts using the biomass objective function as a measure of cell growth.

The CBM workflow used to predict essential genes uses GPRs to determine the set of reactions to exclude given a knocked-out gene g. To this end, we first obtain a mapping $\omega(g,r)$, which evaluates the GPR expression of reaction r with every gene marked as active, except for g. To apply the gene knockout, we must first determine the set $K = \{r|r \in R, \forall\omega(g,r)\}$, which identifies the reactions that are disabled upon deletion of g; then, we set the lower and upper bounds of each reaction in K to 0. Adding these constraints to the model, the simulation can be run using FBA, yielding predicted growth rates for each gene deletion.

Finally, flux distributions resulting from gene knockouts can be evaluated. It is useful to always compare predicted mutant growth rates with wild-type levels. We considered several growth rate thresholds based on the wild-type value to represent viability, although previous studies have considered growth rates below 0.1% of the predicted wild-type rate to imply lethality. Additionally, infeasible solutions are considered as non-viable. We then discretize each gene knockout's result as essential or non-essential and compare them with the experimental screening.

We used Matthews' correlation coefficient (MCC) to assess the predictive ability of our models. The multiclass definition of MCC as implemented in the *scikit-learn* package is presented on Equation 4.7, assuming a generic classifier to predict K classes, t as a vector with the amount of true positives and p the vector with the amount of predictions each class k, while c is the total amount of true positive samples for all classes and s is the number of samples.

$$MCC = \frac{c \times s - \sum_k^K p_k \times t_k}{\sqrt{(s^2 - \sum_k^K p_k^2) \times (s^2 - \sum_k^K t_k^2)}} \tag{4.7}$$

4.2.4.2 Predicted fluxes

Alternatively, flux distributions obtained from the model using an appropriate phenotype prediction method can be directly compared with experimentally measured fluxes, obtained from techniques such as isotope labeling coupled with metabolic flux analysis. In this work, we employ parsimonious flux balance analysis (pFBA) to predict phenotypes using our context-specific reconstructions. We have chosen this method since it requires no prior knowledge and reduces the admissible solution space of FBA by assuming cells not only attempt to achieve the predefined cell objective, but also minimise the overall sum of metabolic fluxes to do so.

A key limitation in using constraint-based models to predict flux values is the lack of reliable measurements for substrate uptake fluxes. This directly influences the predicted growth rate and intracellular fluxes. A more reliable comparison can be made by discretizing flux values into three classes: *forward active*, if the flux is positive, *reverse active* if it is negative (flux is active, but carried in the reverse direction), or *null* when there is no flux. Although less precise, this discards the usage of experimentally measured external metabolite consumption rates. The model's predictive ability can then be ascertained by using metrics suitable for multiclass predictive models such as Matthews' correlation coefficient or weighted F1 scores.

4.2.5 Flux analysis

Models reconstructed using our pipeline yielded flux distributions obtained frompFBA that were used for further analyses. Before applying decomposition methods, statistical tests or using these data for classification tasks, we first scaled flux values to avoid numerical issues. To achieve this, we transformed the entire dataset by applying a sigmoid function $s(x)$ (Equation 4.8) that maintains flux signs, but brings very large values closer. Standardization was not performed as keeping the flux sign intact allows for proper interpretation of these values regarding alternative flux modes associated with the same reaction.

$$s(x) = a \cdot \left(\frac{k}{k + e^{ix}} \right) + 1 \tag{4.8}$$

Relevant fluxes were selected before using supervised or unsupervised algorithms by eliminating fluxes with low variance. Furthermore, we also select an arbitrary number of features ranked by their significance in explaining the variance of the data relative to a discrete clinical feature using one-way analysis of variance (ANOVA) tests.

One of the methods used to analyse predicted fluxes is principal component analysis (PCA), which we used to further reduce the high-dimensionality of the metabolic model's solution space. We also inspected principal component loadings to identify groups of fluxes that were relevant with the clinical features in the biological samples from which the models were reconstructed.

Finally, we also used predicted fluxes to train supervised learning classifiers. We used Random Forest classifiers with varying number of Decision Tree estimators. K-fold cross-validation was used to assess the classifiers' predictive performance using Matthews' correlation coefficient as our metric. In some instances, we trained classifiers using severalpFBA flux distributions for the same cell line. To avoid the inclusion of models from the same cell line in both training and testing folds, we have implemented a custom cross validation routine that splits datasets by cell lines rather than by individual flux distributions.

4.2.6 Software availability

The software featured in this work was developed using the Python programming language. Although compatibility between language sub-versions should not cause any problems, we recommend using Python 3.6 and above. The entire source-code to perform all steps of the pipeline featured in this work is accessible through the GitHub repository at `https://github.com/BioSystemsUM/human_ts_models/`. The packages *cobamp*, *troppo*, *cobrapy*, *pandas*, *seaborn*, *scikit-learn* and *matplotlib* libraries are required to replicate the results and analysis featured in this work.

A significant part of our model reconstruction pipeline has been implemented using the *troppo* framework [147], developed in-house but freely available for the community. This software package provides an environment for omics data processing and subsequent integration with constraint-based metabolic models. This software is structured around two main parts: the omics layer handles data parsing, labeling and normalization, as well as mappings to previously loaded constraint-based metabolic models; the reconstruction layer contains routines to easily adapt omics inputs into appropriate reaction-level scores and run context-specific model reconstruction algorithms using novel implementations of existing methods.

We also used the *cobrapy* package [81] to read genome-scale metabolic models in the standardized SBML format, manipulate their content and predict phenotypes usingpFBA. The IBM® ILOG® CPLEX®[1] (version 12.8) solver was used for all constraint-based analyses and CSMR methods involving linear programming optimization problems, with or without mixed-integer constraints. Some parts of the omics data processing pipeline were performed using the *pandas* package. These routines have been generalized and included in the source-code of this work as auxiliary functions, although most parts of the input preprocessing pipeline are fully accessible through *troppo*.

The remaining parts of the context-specific model reconstruction have also been implemented in several components of *troppo*. Both fastCORE and tINIT algorithms used in this work were run using in-house implementations, which had been validated in a previous work [147]. The EFMGapfill approach is a novel addition to this software package and was implemented using an in-house implementation of the k-shortest

[1] `https://www.ibm.com/analytics/cplex-optimizer`

EFM enumeration already available as part of *cobamp* [148]. This package was also used to run these routines with multiprocessing support whenever applicable.

The plots featured in this work were generated using the *matplotlib* and *seaborn* libraries.

4.3 Results

Our first study evaluated the influence of different input processing methods on model reconstruction, using the MCF7 breast cancer cell line as a case study. We validated the predictive ability of each parameter setup through a comparison of predictions from the reconstructed models with expected phenotypes from gene deletion screens and fluxomics measurements. This cell line was chosen due to its common use in many previous studies, from which a large quantity of knowledge and omics data can be accessed.

In a second stage, using knowledge from these MCF7 models, we selected the best performing pre-processing options for each algorithm, and reconstructed various models for all cell lines, validating them with gene essentiality predictions. In the absence of fluxomics measurements, we also assessed whether such models could be used to generate relevant information for other tasks by using the result of several pFBA simulations as features for supervised machine learning approaches.

4.3.1 Case-study setup

We used the Human-GEM (version 1.5.0) genome-scale metabolic reconstruction as our template model, stemming from a recent effort by Robinson et al. to provide a consensus metabolic model for Homo sapiens [41]. The model consists of 13417 reactions associated with a total of 3625 genes and 4164 unique metabolites, integrating knowledge from previous reconstructions. The model and auxiliary reaction and metabolite tables were downloaded from the corresponding version release on the GitHub repository at `https://github.com/SysBioChalmers/Human-GEM`.

The experimental data used in this work was obtained from two different sources. The Cancer Cell Line Encyclopedia provides a pre-processed and standardised RNA-seq transcriptomics dataset for over 56000 genes across 1270 unique cell lines, with measurements expressed in transcripts per million (TPM). TAS calculations were performed across the entire dataset, although the only integrated scores were those whose associated genes were mapped to the template metabolic model.

These datasets are complemented with the Achilles dataset, characterizing lethal effects of over 18000 gene knockouts through CRISPR experiments [142, 143]. Gene essentiality scores from this experiment

Table 8: Required parameters for model reconstruction and possible options from which to choose from (separated by commas).

Parameter		Options
Algorithm		FASTCORE, tINIT
g_{min} quantile		10th, 25th, 50th, 75th, 90th
g_{max} quantile		25th, 50th, 75th, 90th
Local quantile		10th, 25th, 50th, 75th, 90th
Integration functions	AND	minimum
	OR	maximum, sum

were generated using CERES [143] and processed further until these scores are normalized so that -1 and 0 represent, respectively, the median essential and non-essential gene knockout effects. For this work, we considered five essentiality thresholds evenly distributed across the range between -1.5 and -0.5. As of the first quarter of 2020, 739 cell lines had been included in the Achilles dataset [149], which we then selected as the candidates for our large-scale model reconstruction effort. Gene/transcript nomenclature was converted using the latest HUGO Gene Nomenclature Committee approved symbol mappings whenever needed [136].

Fluxomics measurements for MCF7 cell lines were obtained as part of the dataset used in the analysis of the work of Katzir et al. [144], where time-series metabolomics acquired through Liquid Chromatography-Mass Spectrometry were used to estimate the rates of 44 reactions in three growth media conditions. Despite being originally mapped to the reactions in the Recon 1 genome-scale metabolic model (GSMM), we processed the data and matched these flux measurements and reaction directionality with the Human-GEM template model.

4.3.2 Reconstruction of MCF7 cell line models

We first reconstructed models of the MCF7 cancer cell line by considering every possible combination of the parameters displayed on Table 8, excluding invalid combinations. In addition to these models, we included a MCF7 cell line reconstruction featured in the work of Robinson et al. as a baseline comparison [41].

We obtained 320 models from this reconstruction effort and assessed their ability to correctly predict essential genes and flux activity. To understand the influence of parameterization on the models' performance, we observed the distribution of values across multiple parameter options and evaluated parameter importance numerically using a linear regression.

4.3.2.1 Gene essentiality predictions

The results summarized on Figure 17 show that global thresholds have a greater impact on gene essentiality predictions, since the local 1-state strategy, which places a greater emphasis on local thresholding leads to worse gene essentiality predictions. Additionally, the average of all models reconstructed using the global thresholding strategy is close to that found in local 2-state models.

Despite this similarity when comparing the average of all models for each strategy, the best predictions were achieved using the local 2-state strategy, which is a clear indicator that a combination of both thresholding approaches are useful to estimate RASs. Although the performance achieved using the local 2-state strategy could be attributed to the usage of two (rather than one) global thresholds, we observed that the g_{min} parameter has a negligible influence on the models' predictive performance, which supports our claim that, in fact, both local and global thresholds have a positive effect when combined into the same TAS calculation strategy.

We were also able to infer some of the properties associated with the dataset, where g_{max} and local thresholds at the 25th percentile seem to have the most positive effect on predictive ability in all models. Although the CERES score threshold representing the median essential gene knockout was set at -1, our models show slightly increased predictive power at -0.75.

Aside from data preprocessing related parameters, we have found the best parameter combinations are to use the tINIT algorithm in conjunction with the maximum function as replacement for the AND operator. We also observed that refining the model with EFMGapfill to allow growth using only the defined growth media metabolites as substrate did not result in better gene essentiality predictions.

4.3.2.2 Flux activity predictions

We also performed a similar assessment on the ability of our models to correctly predict reaction activity and directions for the MCF7 cell line under three growth medium compositions with associated fluxomics measurements. For each parameter combination, we generated three corresponding predictions using the growth medium as an additional flux constraint and calculated the MCC between the measured and predicted flux activities.

In Figure 18, we can see that most parameters affect flux and essentiality predictions similarly. The best performing strategies are still those based on global and local thresholding with 2 states. In both algorithms, we also observed that constraining nutrient uptake to the metabolites that could be matched with the growth medium led to higher predictive power.

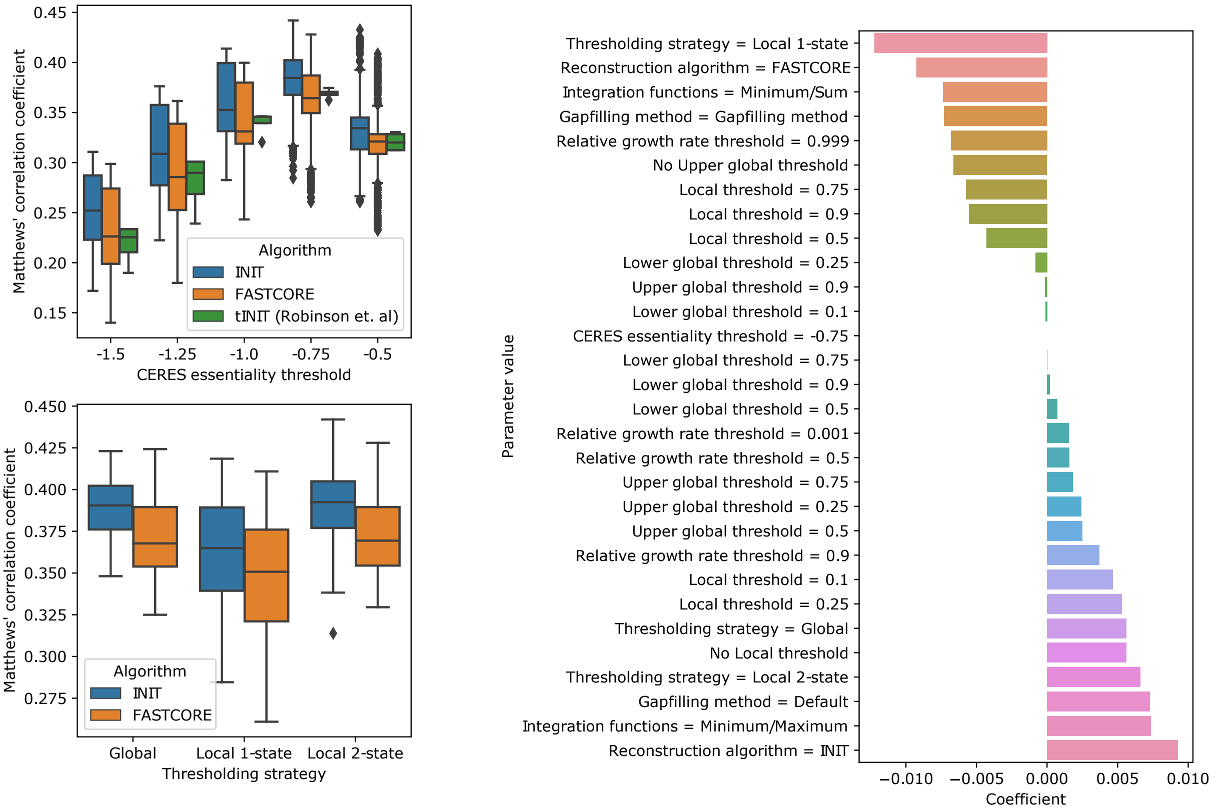


Figure 17: Overview of the influence of parameterization on the models' performance when predicting essential genes as determined by their MCC. **Bottom left**: MCC value distribution for each thresholding strategy (horizontal axis) and algorithm combination (coloured box and whiskers). **Top left**: MCC value distribution for each CERES score threshold (horizontal axis) and algorithm combination (coloured box and whiskers). **Right**: Linear coefficients for each individual parameter value on a regression model aimed at predicting MCC values. Each parameter variable was one-hot encoded as multiple binary variables.

The relationship between average MCC and its standard deviation is shown in Figure 19 where we can observe that FASTCORE reconstructed models were able to reach the highest correlation with the experimental fluxomics data, although they are more sensitive to parameterization. INIT, on the other hand, showed higher average MCC values across all parameter combinations with less dispersion and yielded models that rank closer when evaluated with this metric. We also compared our models with a baseline MCF7 cell line model featured in the work of Robinson et al.[41], which ranks significantly lower than our best FASTCORE and INIT models.

Overall, we did not significantly improve the predictions when considering a direct comparison of active fluxes between simulation and experimental quantification, although we identified some key reconstruction parameters that influence the context-specific model's performance.

FASTCORE appears as the more consistent tool to extract context-appropriate sets of reactions from a template model. Due to its low computational demand, these reconstructions can be repeated with

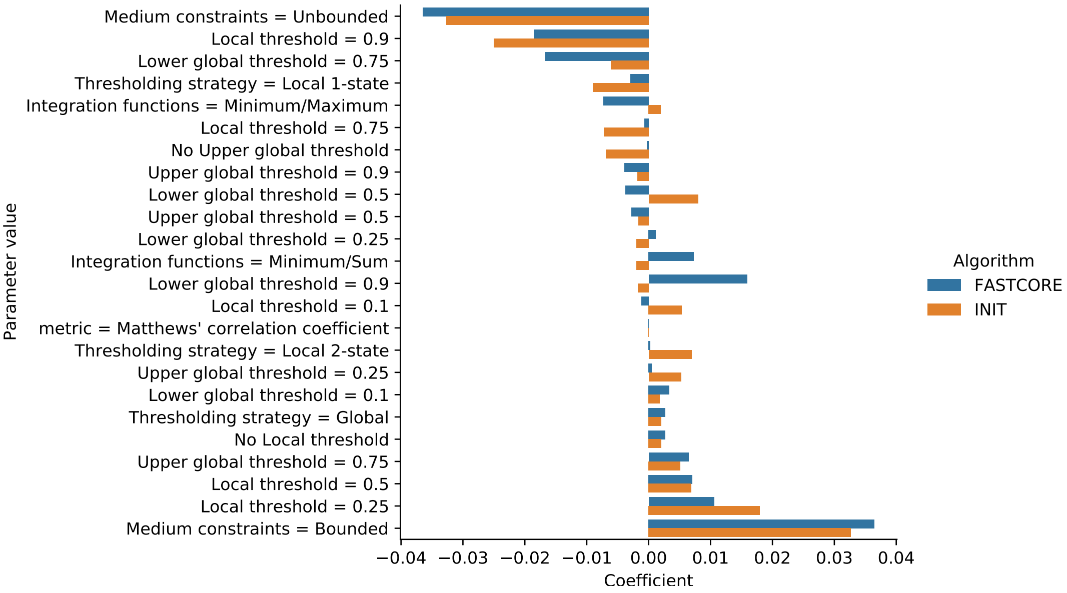


Figure 18: Linear coefficients for the parameter values tested in the MCF7 context-specific reconstruction case study on a regression model aimed at predicting MCC values from each parameter combination. Each parameter variable was one-hot encoded as multiple binary variables.

alternative parameters or to sample large amounts of models. tINIT , on the other hand, shows great potential to yield high-quality context-specific models, but thresholding parameters seem to heavily affect predictive ability.

4.3.3 Large-scale metabolism reconstructions of cancer cell lines

We used 10 of the highest scoring parameter combinations from the MCF7 cell line case study to reconstruct the entire panel of cell lines available in CCLE with associated gene knockout effect screens (n=739). A similar reconstruction pipeline was employed in this larger case study, although we did not perform gap filling relative to the growth medium, due to heavy computational demand and an expected negative impact in phenotype predictions.

4.3.3.1 Predictive performance assessment

The results summarized on Figure 20 depict our large-scale results which show similar predictive accuracy to those reconstructed for MCF7 cell lines. There are slight differences in gene essentiality prediction performance between the 10 selected parameter combinations, with tINIT models reconstructed displaying slightly higher scores. In all of these scenarios, the selected pipeline parameterization choices improved gene essentiality predictions, when comparing with the models reconstructed in the work of Robinson

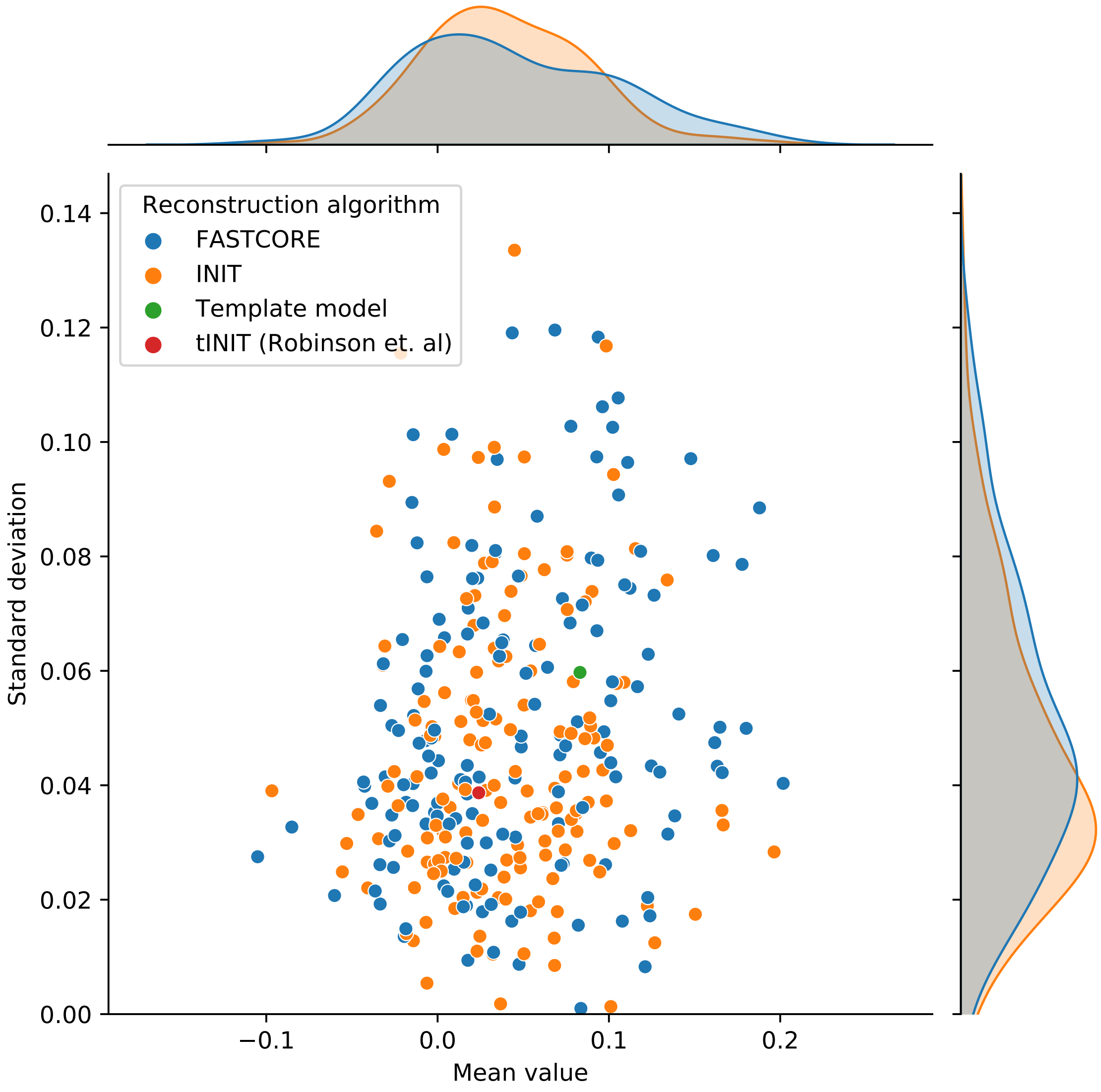


Figure 19: Relationship between average MCC value and standard deviation for each group of 3 simulations (conditions) that make up a single parameter combination in flux predictions for the MCF7 cell line case study. Different colors represent different algorithms and/or baseline comparison models.

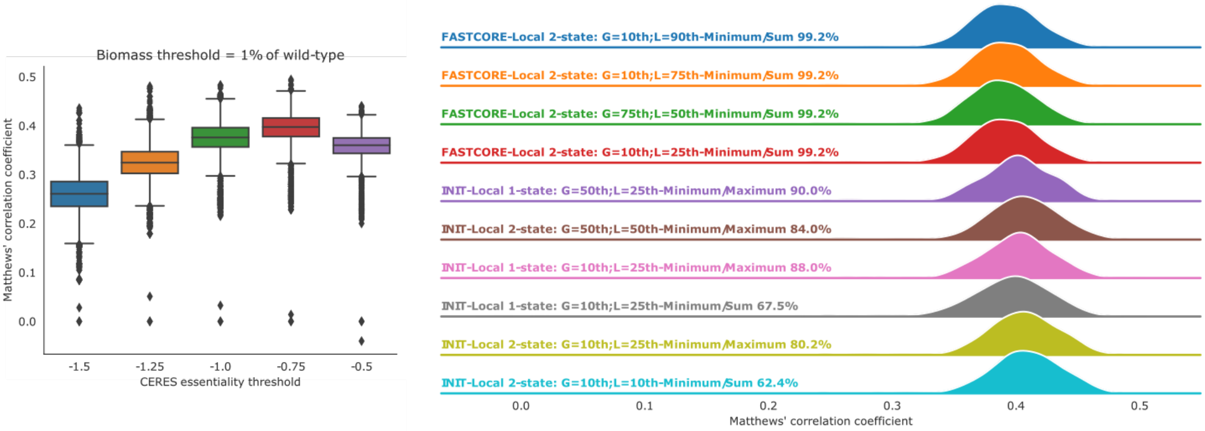


Figure 20: Overview of the predictive capability of the models reconstructed for each CCLE cell line. **Left**: MCC value distribution for all models in each lethal gene effect threshold. **Right**: Distribution of MCC values for each parameter combination selected for large scale reconstruction of CCLE models. The percentage value at the end of each thresholding strategy description represents the proportion of cell line models that could be successfully reconstructed.

et al, which asserts the importance of using more complex scoring strategies involving global and local thresholds.

The CERES score characterizing gene essentiality in the CRISPR experiments is undoubtedly the parameter that affects these predictions the most, with the best results obtained at a threshold of -0.75. This finding along with the overall low MCC values found with our approaches can be due to several factors. On the one hand, the biomass equation is a generalized assumption of the metabolites needed for cell growth, and thus, is not tailored for each specific cell line. The lack of more exact constraints on model uptake also results in gene knockouts that either do not affect flux through the biomass pseudo-reaction or completely inhibit it, which by itself elicits the usage of a threshold since a direct correlation between CERES scores and growth can not be found using constraint-based models.

4.3.3.2 Exploring metabolic variability in breast cancer

We used the models and their respective predicted fluxes to explore the metabolic heterogeneity among various breast cancer cell lines. To do so, we retrieved molecular subtype annotations from the DepMap repository and used PCA to project these flux distributions using reduced features and obtain relevant information on the flux patterns present in different breast cancer subtypes. These results are summarized on Figure 21.

The subset of models belonging to breast cancer were extracted and the 200 fluxes with most statistically significant differences among subtypes were used to decompose the dataset. We included all reconstructed models, regardless of their parameters, and reduced the latent space to 3 principal components (PCs) capable of explaining 33% of the observed variance. The loadings of each principal component were obtained, with each flux being summarised in their corresponding pathways by calculating the average of the absolute ratios between the weights of each flux and the maximum observed values in the PC loadings.

Firstly, we can conclude that the decomposition of predicted fluxes can adequately distinguish between major molecular subtypes of breast cancer. The second PC (PC2) marks a good distinction between basal and luminal cell lines and correlates with reported prognosis and aggressiveness [150], with luminal BCs with better prognosis assigned to positive values as opposed to basal breast cancers which appear in this PC as negative values.

Lipid metabolism is typically deregulated in breast cancer and we were able to identify changes in fatty acid synthesis, with long-chain fatty-acid CoA ligase (encoded by the *ACSL1* gene) and fatty acid desaturase (encoded by *FADS*) activity, as well as increased arachidonic acid production negatively correlated with the values in PC2. ACSL1 in particular has been reported as being transcriptionally upregulated in several breast cancer subtypes [151], and although it is not specific to basal BC, its expression has been shown to negatively impact survival rates. Moreover, previous studies have also shown that arachidonic acid promotes tumor cell migration in a basal B BC cell line [152], which further reinforces the association of PC2 with poor prognosis.

Another important finding was the negative correlation of PC2 with mitochondrial citrate carrier (*SLC25A1* gene) activity. This transporter plays a fundamental role in maintaining mitochondrial activity in high proliferating cells [153] and is highly expressed in triple-negative breast cancer (TNBC), corroborating the hypothesis that PC2 depicts a gradient of cancer aggressiveness.

4.3.3.3 Predicted metabolic fluxes as relevant features

The lack of fluxomics data for the whole set of cell lines featured in DepMap does not allow to carry out a large-scale systematic comparison of the pFBA flux distribution predictions with experimental data. However, we set out to assess whether or not these predicted fluxes could be useful in predicting several clinical features associated with each sample. To do so, we established a supervised classification task, where the disease's primary location would be predicted using various datasets, namely, (1) standardized

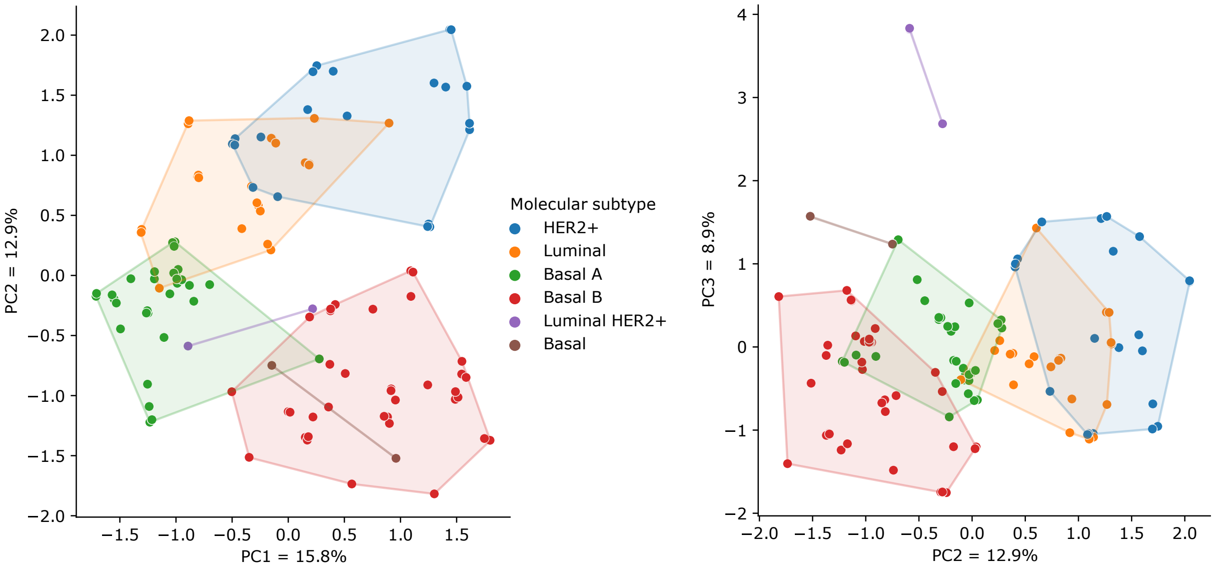


Figure 21: Cell line models reconstructed for breast cancer cell lines projected in lower dimensions through the usage of PCA. The left figure shows the first PC against the second, while the right figure displays the second PC against the third, in the horizontal and vertical axes, respectively.

expression values (TPM from RNASeq) using the entire gene set, as well as only those genes that can be integrated in the metabolic model, (2) TASs generated for each sample, (3) predicted fluxes (using pFBA over reconstructed models) and (4) the reaction presence (binary) from the CSMR algorithm outputs. Our results on this task are summarized on Figure 22.

Classifiers trained with standardized transcriptomics data showed good relative performance (MCC mean=0.570, sd=0.038) with the subset corresponding to metabolic genes only slightly outperforming it. Processing these data and generating TASs slightly increased predictive capabilities (MCC mean=0.594, sd=0.030), which further justifies applying our preprocessing workflows before analysing and integrating omics data. However, the outputs of CSMR algorithms, namely, the presence or absence of each reaction resulted in models capable of predicting a cancer cell line's primary site with an average MCC of 0.525 (sd=0.031). Furthermore, pFBA simulations resulted in even worse classifiers that could only reach an average MCC of 0.298 (sd=0.042).

Overall, our results show that context-specific model reconstruction and flux balance analysis approaches are not yet consistent enough for accurate quantitative flux predictions, as predicted metabolic fluxes by themselves did not appear to be relevant features for complex classification tasks.

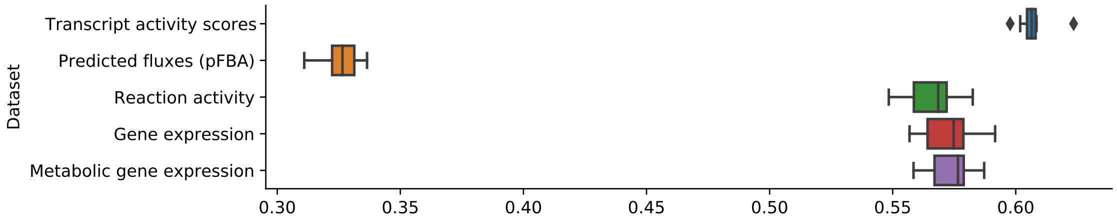


Figure 22: Distribution of average Matthews' correlation coefficient values for cross-validated classifiers trained with various datasets

4.4 Conclusions

We built upon several previous efforts to generate constraint-based models of differentiated human tissues, being capable of assembling a generic pipeline that can be useful in standardizing the process of integrating transcriptomics data into human metabolic models for the scientific community. Furthermore, this pipeline is available as part of an open-source software tool providing a generic framework for the implementation of context-specific model reconstruction tasks.

We were able to leverage large-scale multi-omics experiments with cancer cell lines and a state-of-the-art human metabolic reconstruction to generate meaningful models capable of capturing the metabolic diversity among, and within, multiple types of cancer. We were also able to validate the models using experimentally determined essential genes and fluxomics data.

The usage of decomposition methods to understand flux predictions allowed us to establish a link between metabolic phenotypes and breast cancer prognosis, and by making use of the interpretability of constraint-based models, we were also able to pinpoint key enzymes and metabolites associated with deregulated growth. This elicits the potential for similar approaches to assist in contextualizing transcriptomics profiles into metabolic phenotypes, with the purpose of understanding the intricate mechanisms responsible for human diseases, especially for personalized medicine applications.

The availability of metabolomics and proteomics data still pales in comparison with RNA-Seq technologies used for transcriptomics quantification. As such, we have developed this work to only consider the latter omics type, and we argue that the reconstruction of models based on transcriptomics data results in computational tools that can be more easily adapted to a clinical setting since they do not rely on generating multiple omics datasets. However, we have also built the computational tools, namely `troppo`, in a way that these datasets can be easily integrated and used with appropriate methods.

Although encouraging, our results show the difficulty in closing the gap between experimentally measured and predicted fluxes. We argue that there is value in building representative models using gene expression alone, since the techniques used to obtain these measurements are far more ubiquitous and less costly. However, naturally, this lack of information implies some limitations when interpreting the model. This was evident when using model simulations to predict a cell line's disease, where classifiers trained with these predictions displayed poor predictive performance.

In the absence of precise exo-metabolome uptake or secretion rates, CBMs in their original definition, are merely capable of predicting metabolic pathways on a discrete level, and thus, flux distributions must always be interpreted relative to a given original state or model context rather than assuming these fluxes are numerically comparable. A related challenge also appears when considering the biomass objective function, which is usually too generic to describe different tissue types, and hinders the ability for these approaches to generate meaningful models for cells whose metabolic objective is difficult to define.

Recent works that have incorporated exo-metabolite measurements [140], metabolic task protection and alternative formalisms to include more complex parameters [41], have reached better Pearson correlation coefficients with fluxomics measurements, although with smaller case studies. Another important aspect would be to expand the scope of constraint-based models to also include regulation and signal transduction enabling predictions of metabolic fluxes that can be contextualized with their corresponding regulators.

We must, additionally, acknowledge the importance of using a fluxomics data source for a reference cell line to serve as a basis for subsequent reconstructions, since it allows us to find ideal sets of parameters for larger-scale efforts. In this work, this led to a significant decrease in computational resource usage, as well as a better choice of parameters without exhaustive reconstructions.

The implementation of this complex pipeline in a modular framework allows for the usage of different methods that might fit a particular purpose. Previous works have reported the heterogeneity in outputs from various CSMR algorithms and our case study clearly shows that this choice impacts the type of phenotypes to predict and, as such, we extended *troppo* in such a way that reconstructing a context-specific metabolic model is a simple task, even for users with limited programming skills

5

Integrative constraint-based models of metabolism, gene regulation and cell signalling

5.1 Introduction

The previous chapter describes how context-specific modelling approaches yield models that can capture some metabolic phenotypes associated with a certain molecular background. Overall, it was possible to conclude that gene essentiality predictions from context-specific models were acceptable, unlike fluxomics which did not correlate with the reconstructed models in a level that could be deemed significant.

Several factors could be hindering the predictive ability of constraint-based models. Firstly, the basic mathematical formulation of such models adopts the steady-state assumption, which simplifies the complex dynamics of enzymes. Moreover, the only entities that are usually represented with constraint-based models are reactions and metabolites which, as was described in the previous chapter, can be complemented with omics measurements.

Within the scope of this work, there was a clear gap concerning the lack of integration of regulatory and signalling interactions in constraint-based models. This prompted the development of a method to expand the constraint-based framework to directly represent entities, such as genes or regulatory proteins and an associated phenotype prediction method that can leverage this representation to integrate multi-omics measurements.

Recently, Dugourd and colleagues, including myself, have presented Causal Oriented Search of Multi-Omic Space (COSMOS), an approach capable of integrating multi-omics datasets to generate contextualised multi-scale networks [154]. COSMOS uses a MetaPKN, which is a comprehensive network connecting signalling, metabolic and gene regulatory interactions. By using the CAusal Reasoning for Network identification using Integer VALue programming (CARNIVAL) algorithm, COSMOS is able to extract a connected subnetwork of a PKN that is both consistent with omics data and spans several biological layers. The CARNIVAL algorithm is an integer linear programming (ILP) formulation that deals with prior knowledge network (PKN)s as networks of causal interactions whose biological entities are assigned a down-regulated, up-regulated or neutral state according to the omics data.

Inspired by the the approach presented in COSMOS using causal links to model mechanistic principles of metabolism, we propose a novel method to perform the same type of integration. The ipFBA approach presented in this chapter combines a novel representation that models metabolic fluxes, metabolic gene expression and regulatory interactions in a global formalism, similarly to COSMOS. However, ipFBA does this using an LP formulation that maintains the steady-state assumption, but prioritizes compliance with metabolic gene expression, to deliver flux predictions that can provide reliable hypotheses for the heterogeneity between different cells and tissues, as well as predictions that are compliant with the context provided by omics measurements.

Similarly to Chapter 4, results were focused on validating the method and demonstrating that metabolic models can be used to predict phenotypes and metabolic patterns across multiple cellular contexts. To this end, phenotypes were again predicted for the samples on the Cancer Cell Line Encyclopedia (CCLE) cell line panel after parameter calibration using the MCF7 cell line. This allows a more direct comparison with the results from the previous chapter.

5.2 Methods

5.2.1 Multi-layer constraint-based models

Biological networks have been made available through a number of databases and other resources describing the connections between several types of biological entities. In this work, we combine these networks into a graph capable of including multiple biological networks in a single representation. Although we focus heavily on the connections between gene expression and regulation with metabolism, this approach is extensible to other omics biological layers such as epigenetics or phosphoproteomics. Figure 23 depicts the layers included in our approach with their corresponding biological entities and associated omics types as well as the networks that connect these entities together.

We first include cell metabolism as depicted in genome-scale metabolic networks detailing a set of

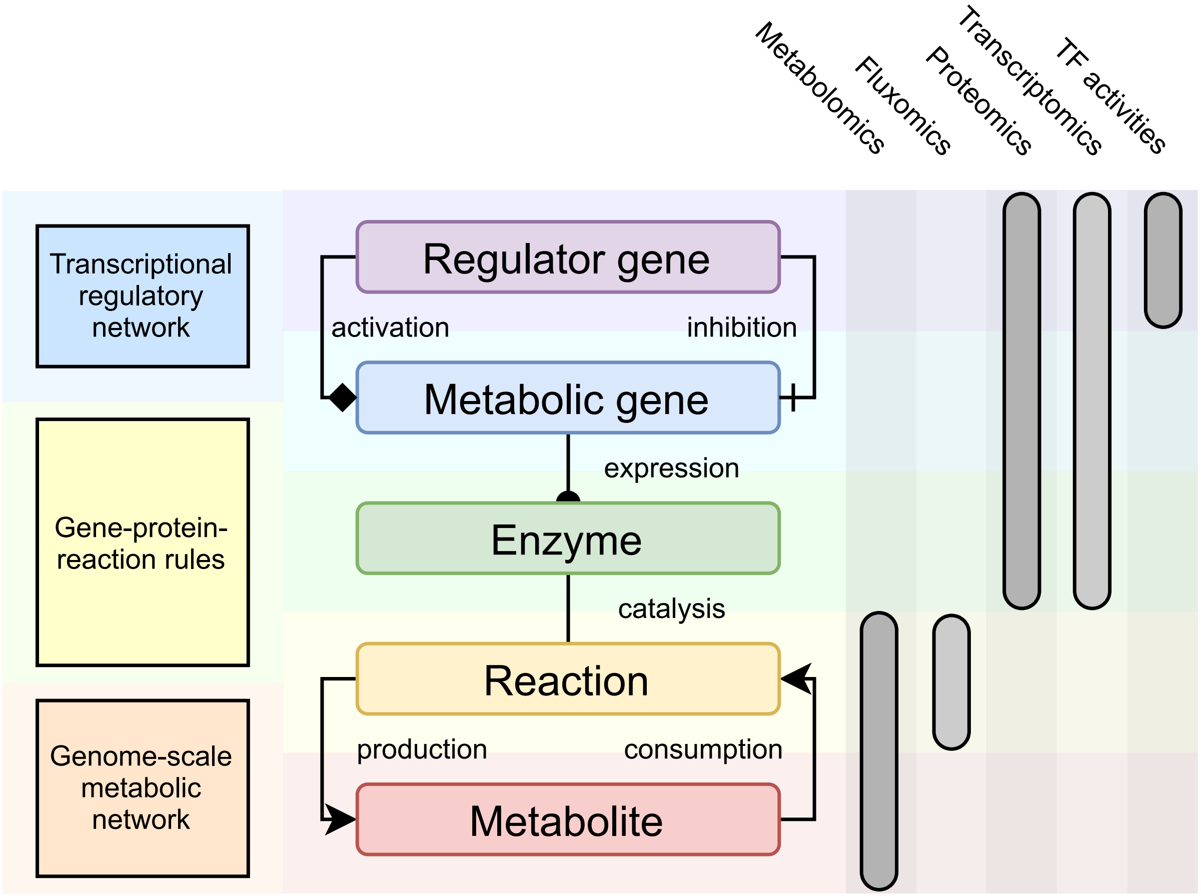


Figure 23: Overview of the layers capable of being integrated in the approach presented throughout this section. The leftmost part of the figure represents the networks encoding links between the entities as well as the interactions represented in the center. The various omics inputs that can be integrated are represented in the right part, associated with their corresponding biological entities.

metabolic reactions mostly inferred from genome annotation. This, in turn, also defines the set of metabolites that could be found within the organism, and the stoichiometric coefficients for each metabolite and reaction, indicating whether a metabolite is consumed or produced.

The reactions in the genome-scale metabolic network are usually complemented with GPR rules defining which combinations of genes encode the enzymes responsible for catalysing a given reaction. This establishes links between the metabolite and reaction layer to the enzyme and gene layers.

Finally, interactions between genes are represented through a transcriptional regulatory network (TRN), connecting them with interactions that might be enriched with information on their effect, which is typically either activation or inhibition.

Connecting all of these layers within the same modelling framework poses several challenges which require heavy parametrisation or assumptions that simplify the resulting model and simulation approaches.

Firstly, model predictions were centred around the metabolic layers, using the enzyme, gene and regulatory layers to expand the amount of information that could be integrated so that it could then be used to provide a flexible scaffold to then constrain predictions to omics inputs.

Additionally, the modelling paradigms typically used for representing the various layers also differ greatly. Metabolic models are usually formulated as ordinary differential equations or constraints in linear programming, while gene expression and regulation are often described and modelled as causal relationships (e.g. Boolean rules).

Linear programming was used as the underlying mathematical framework as it allows us to expand existing constraint-based metabolic models with other constraints capable of modelling additional layers. However, this requires that the layers outside of metabolism to also be represented as linear equations. The generic integration process involves representing the various biological molecules as additional pseudo-metabolites, with the interactions between them modelled as reactions. This process will be detailed in the following sections.

5.2.1.1 Extending metabolic models with gene expression

The first step in our approach is to include enzymes and their encoding genes as a part of the model. To do so, we first expand the model to include reaction usage as an entity in the model. Inspired in the work of Machado et al. [155], we use the basic idea of representing enzymes as metabolites.

We first replace each reversible reaction into two irreversible reactions representing the forward and reverse flux. In this same model, a set E of N pseudo-metabolites are also added, one for each reaction in the original model, created to represent the reaction's availability. For each reaction in this expanded model containing irreversible and duplicated reversible reactions we add the corresponding pseudo-metabolite in E as a reactant to be consumed. These transformations are represented on Figure 24.

With this expanded representation, fluxes are not only rate-limited by chemical reactants, but also by enzyme availability, which can then be made available to the model through an exchange reaction, or by creating additional reactions that produce metabolites in E. It is worth mentioning that metabolic reactions in this representation must be irreversible so that the metabolites in E are always consumed and never produced through metabolic reactions.

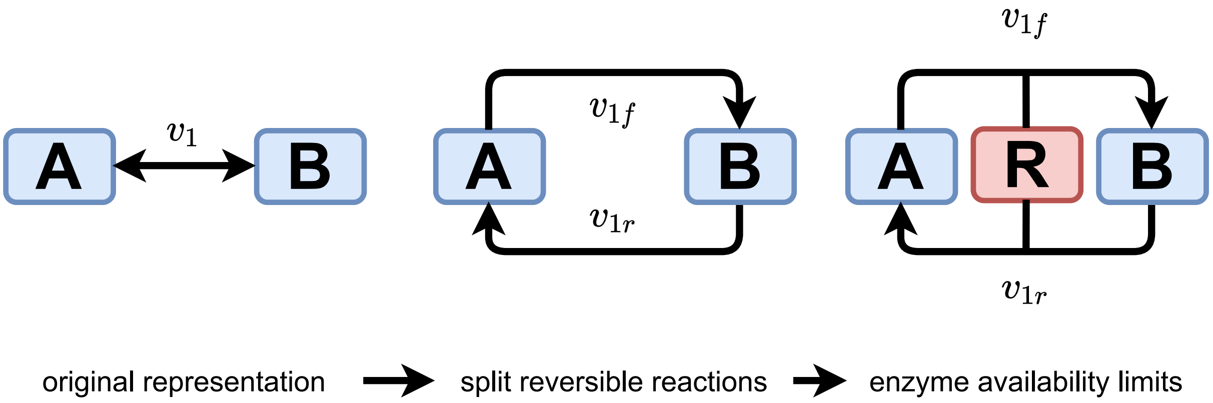


Figure 24: Representation of constraint-based model reactions with limited enzyme availability. The approach presented in this work splits reactions into their forward and reverse components and adds a reactant (represented as R) to each component. This reactant can then be made available using other pseudo-reactions to ensure it is produced.

5.2.1.2 Gene-protein-reaction rules as linear constraints

In Section 2.2.1.2, GPR rules are presented as the GSMM components that link genes with the reactions they are associated with. These Boolean expression can represent the isozymes capable of catalysing a given reaction as well as the protein subunits in cases where an enzyme is a protein complex. Converting the Boolean representation into a formulation that can be directly integrated into constraint-based models requires operands (genes) to be defined as continuous variables and the conjunction and disjunction operators replaced with operations suitable for numeric values, such as those obtained through transcriptome quantification experiments.

The work of Richelle et al. establishes two approaches to extract continuous values from GPR Boolean expressions, namely by replacing the AND operator with a minimum function and the OR operator with a sum or maximum function [138]. This leads to two important assumptions: (1) the availability of a given isozyme with various protein subunits will be limited by the subunit with the least expression; (2) the availability of a given reaction is defined either by the total amount of isozymes as determined by the minimum value of the genes in a given isozyme complex or by the most expressed isozyme.

The operations shown above can be replicated in constraint-based models with pseudo-metabolites and reactions. Since any reaction in a constraint-based model is bound by stoichiometry and rate limited by the concentration of its reactants, this property can be exploited to generate reactions that produce isozyme species using gene species in the isozyme complex as reactants. Similarly, one can also create

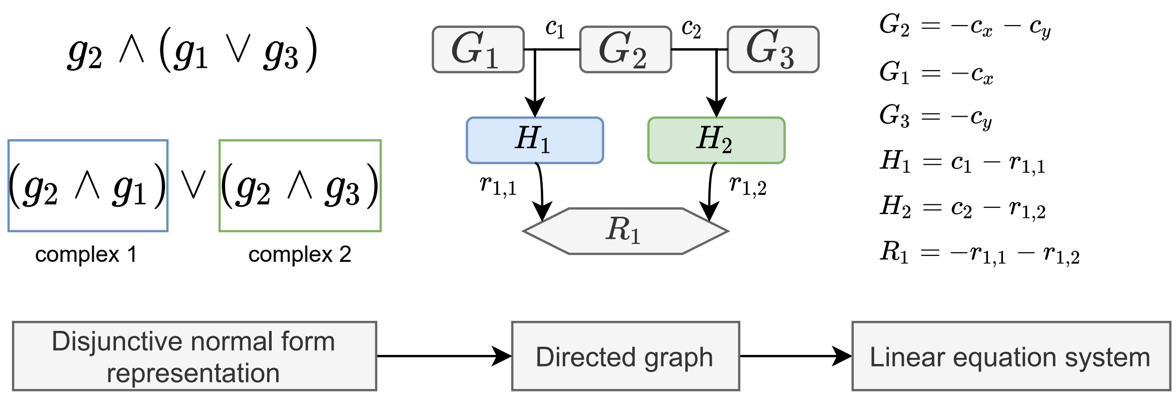


Figure 25: Representation of GPR rules as a linear system of equation. Boolean expressions are first converted to DNF form and represented as a graph which can then be translated into a system of linear equations.

simple reactions converting these isozyme species into a reaction pseudo-metabolite, such as the ones detailed in the previous section, where these metabolites limit the amount of flux.

Formally, we can encode the information provided by the GPR rules into two matrices E and L which represent, respectively, reaction-to-complex and complex-to-gene relationships.

Assuming a set of GPR rules in disjunctive normal form (DNF), we define a function $\omega(i)$ yielding a set of sets X. Each set in X contains indices denoting the genes necessary for a single isozyme capable of catalysing reaction i to be expressed (operands of AND expressions). The set C of isozyme complexes can then be obtained by retrieving all distinct elements $c \in \omega(i) \forall i \in 1, ..., N$. We can then define a matrix E where each row represents a metabolic reaction and the columns represent the complexes in C. Each element $E_{i,j}$ is set to 1 if the complex C_j is present in $\omega(i)$ and 0 otherwise. Similarly, a matrix L is also defined to represent the relationship between complexes and the genes present in them. Assuming a set of genes G represented in the columns of L, each element $G_{i,j}$ of this matrix is -1 when a gene g_j is a part of the complex C_i or 0 otherwise. This process can be visualised on Figure 25

5.2.1.3 Representing causal interactions as steady-state flows

The link between transcription and metabolism has been established through the formalisms defined in the previous sections. The final step towards a fully integrated model is the addition of a regulatory layer capable of affecting the availability of metabolic genes. In this work, we attempt to integrate causal interactions between genes and proteins described using simple mechanisms, such as activation and inhibition. Although this is a simplified view on the complex processes behind signalling and transcriptional control, it requires less parametrisation than more complex mechanistic models which cover less genes and usually involve kinetics, which are not represented using constraint-based modelling.

In this work, we assume regulatory inputs are stored in a directed graph $S = (V, A)$ where each vertex in the set V represents a gene and the set of edges A, each connecting a source gene x which regulates a target gene y, assuming $x, y \in V$. An additional function $w(x, y)$ maps each edge to a weight that is either positive if x activates y or negative if x inhibits y. Interactions without a defined weight are not considered for this graph and as such, the edges of S can be defined as the set $A \subseteq \{(a, b) | (a, b) \in V^2 \wedge a \neq b \wedge w(a, b) \neq 0\}$.

The gene metabolites added in the previous section only account for genes associated with metabolic interactions. As such, to represent regulatory interactions, we must also expand the set of genes G to also include the vertices in the set V. Furthermore, we also add a secondary set of gene metabolites H, each representing a gene in G, which we will call "pool metabolites". These pool metabolites simulate the availability of a gene while the set G are meant to represent the activity of the proteins they encode. To connect these reactions, we add one reaction for each gene $i \in G$ converting z units of H_i into one unit of H_i. For the purpose of this work, we set $z = 1$, although additional information pertaining to the ratio between mRNA and protein quantities can be expressed through this parameter.

We then simplify regulatory interactions where a given gene x regulates a gene y by representing them as reactions that consume a regulator gene pseudo-metabolite G_x and produce or consume its regulatory target gene pool metabolite H_y. An example of these transformations is represented on Figure 26 This allows us to model regulatory interactions by adding regulator genes as sources for the target genes, that, when produced, can also carry flux towards the layers responsible for connecting them with metabolic reactions. If a regulatory interaction becomes active, it will either produce or deplete its target gene pool metabolite, thus regulating their availability to regulate or express other enzymes.

Formally, we express these relationships in a $2p$ by q R matrix where p is the number of genes and q regulatory interactions. R can be subdivided into two p-by-q submatrices R' and R'' for gene activity and gene pool metabolites, respectively. For each interaction $A_j = (x, y) \forall j \in 1, ..., q$, the matrix element $R'_{x,j}$ is set to -1, while $R''_{y,j}$ is set to 1 or -1, for activation and inhibition interactions, respectively.

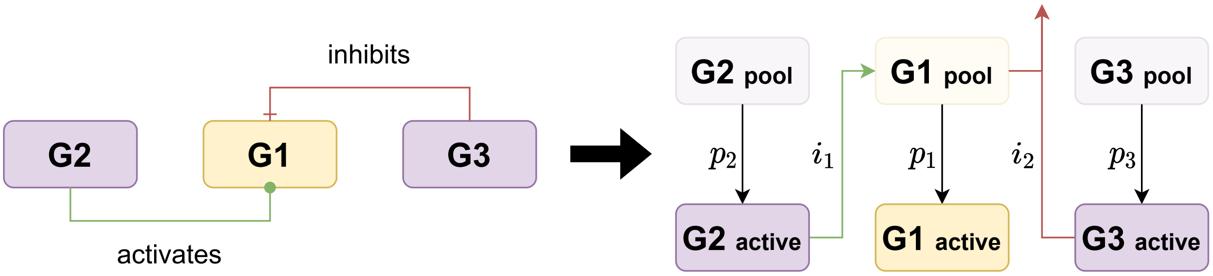


Figure 26: Representation of regulatory interactions as constraint-based model reactions. In this example, two regulator genes have effects on $G1$. These regulatory interactions are represented as reactions that consume a gene pseudo-metabolite to either produce or deplete the target pool metabolite associated with the regulated gene, depending if the interaction activates or inhibits the gene. Note that inhibition reactions are exchange reactions consuming both metabolites (without any product).

5.2.1.4 Multi-layer constraint-based model

The system described in the former sections can be summed up in Equation 5.1.

$$\begin{bmatrix} S & -S_{rev} & 0 & 0 & 0 & 0 & 0 \\ -I & -I_{rev} & E & 0 & 0 & 0 & 0 \\ 0 & 0 & -I & I & 0 & 0 & 0 \\ 0 & 0 & 0 & L & R' & I & 0 \\ 0 & 0 & 0 & 0 & R'' & -I & I \end{bmatrix} \cdot \begin{bmatrix} v \\ u \\ r \\ c \\ i \\ p \\ d \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.1}$$

$$v \in \Re^n, u \in \Re^{|Rev|}, r \in \Re^{|Enz|}, c \in \Re^{|And|}, i \in \Re^{|A|}, p \in \Re^{|G|}, d \in \Re^{|G|}$$

The matrices represented above have been defined in the previous sections. Additionally, we also assume n as the number of reactions in the metabolic model (columns of S), Rev as the indices of S corresponding to reversible reactions, And as the amount of individual associations between genes, A as the set of regulatory graph edges and G as the set of genes represented in the model.

The system shown in Equation 5.1 represents a steady-state model that can be simulated using LP solvers in a similar manner as that employed with FBA. Since gene-associated conversions are all irreversible and directed towards producing gene metabolites, they are only used to constrain metabolic fluxes without allowing flows towards genes. However, to fully explore the potential of these models, we developed novel objective functions and associated constraints to better guide the search for flux distributions that can capture phenotypes that are consistent with transcriptomics.

5.2.2 Integrated parsimonious flux balance analysis

In this work, we also present the ipFBA approach, an alternative phenotype prediction method based on FBA that includes an objective function and constraints adapted to our novel expanded networks, yielding optimal flux distributions relative to their similarity with observed transcriptomics measurements. The ipFBA algorithm consists of a two-step LP optimisation routine based on the extended model representation presented on the previous sections.

The system of equations defined in Equation 5.1 is iteratively optimised to find limits for secondary objectives, a concept that represents assumptions on the model organism such as growth rate limits or substrates that are guaranteed to exist in the growth medium. The limits for these objectives are subsequently added to the existing LP problem to constrain the solution space prior to obtaining a flux distribution. This system is then optimised with a main objective function that represents consistency with omics measurements, redirecting flux that is associated with inactive genes or proteins. The resulting flux distribution will comply with metabolic assumptions while representing a metabolic state that is more closely tied to the omics measurements.

5.2.2.1 Linear programming formulation

Assuming the system represented on Equation 5.1, we define the following LP problem.

$$\begin{aligned}
\text{min} \quad & \sum_{i \in g} w_i \cdot d_i & & & & (5.2)\\
\text{s.t.} \quad & \sum_{j=1}^{n'} N_{i,j} \cdot y_j & = \quad & 0 & (\forall i \in \{1, ..., m'\}) \quad & (5.3)\\
& & & & \text{(LP2)} & \\
& \sum_{j=1}^{n'} T_{i,j} * y_j & \leq \quad & t & (\forall j \in \{1, ..., n'\}) \quad & (5.4)\\
& & y \in \Re^{n'}, & t \in \Re^{|T|} & & (5.5)
\end{aligned}$$

The objective definition in 5.1 of LP 2 assumes the definition of a vector w with a value for each gene in the set g. This major objective minimizes the sum of values for gene pool metabolite conversion reactions - indexed in the vector d - corresponding with the reactions that produce the gene pools multiplied by each gene's weight as defined in w. This weight can be derived from various data sources, such as transcriptomics or transcription factor activities for genes associated with regulatory events, and should represent the relative activity of each gene in a given sample as a continuous value that is positive or negative depending whether the gene is expressed in high or low relative quantities. This results in flux distributions whose active fluxes will tend to be those associated with reactions encoded by highly expressed genes.

Additionally, we can also include further constraints to narrow down specific phenotypes, such as growth media demands and minimum growth rates. For these purposes, we extend the system defined in Equation 5.1 with arbitrary inhomogeneous constraints. To do so, one must define a $k \times n$ matrix T where each row represents a single constraint and each column represents a conversion in the expanded system. Constraint 4 defines the constraints defined in this matrix as inequalities that bound sums of one or more fluxes.

A vector t of length k is also defined, with upper bounds for the constraints expressed in the matrix T. The signs of values in T and t also allow for lower bounds to be set. As a practical example, the definition of an arbitrary lower bound constraint i on the minimum growth rate (flux through $y_{biomass}$) is performed by setting $T_{i,biomass} = -1$ and setting an upper bound on this constraint $t_i = -\mu$, where μ is the minimum growth rate value. Note that this upper bound constraint would be expressed as $-1y_{biomass} < mu$, which can also be expressed as $y_{biomass} \geq \mu$, which represents a lower bound on the biomass flux.

As part of the ipFBA algorithm, we also devised an algorithm to determine the values of t. Using the system defined in Equation 5.1, one can define its objective as an arbitrary function $Z(y, i) = \sum_{j=1}^{n} T_{i,j} \cdot y_j$, where c is a single row from the matrix T. The limits of Z can be obtained by maximizing and minimizing this function. We will call this system LP 2. The system can then be further constrained using Constraint 4 of the LP 2 formulation, where the vector t can be updated with an appropriate value, to which we recommend using a proportion of the maximum theoretical objective value as an upper bound. This process can be repeated to further constrain the system.

5.2.2.2 Integration of omics inputs through objective function constraints

We employed our objective constraint approach to integrate omics inputs into our simulation. In this work, we focus on three types of omics constraints, namely:

- Transcriptomics
- Endometabolomics
- Exometabolomics or growth media formulations

In a generic integration scenario, each omics integration input is mapped to one row with index i added to the T matrix where the non-zero elements of row i of T are defined by the expression $T_{i,z} = w$. w is a weight vector encoding the positive or negative contribution of a given flux in y. Using the formulation in LP 2 along with the objective coefficients encoded in $T_{i,z} \forall z | y_z \neq 0$, the lower bound or upper bounds of this objective function in the system can be determined. When using the formulation in LP 2, these values can then be used to generate appropriate ranges for the objectives, encoded in the vector t.

This generic constraint allows to provide omics inputs by carefully selecting appropriate reaction sets and weights. Since these are objective functions, the idea is to reach solutions that maximize the usage of highly active biological entities while also minimizing the usage of those with low or absent measurements. As such, the weights defined in w should be negative or positive, respectively, for low and high activity molecules.

In transcriptomics integration, the weights are calculated using the logarithm of the fold change value relative to the mean, and these weights are mapped directly to the gene pool conversions, similarly to how the global objective function is defined on LP 2. For metabolomics inputs, we first compute fold change logarithms for each metabolite and then map these values into the metabolic reactions that either produce or consume them. In the case where there are multiple inputs for the same metabolic reaction, the mean value of all inputs is used. Finally, to integrate the growth media, we attribute a positive weight to all substrate import reactions whose associated metabolites are not a part of the medium formulation and a null weight to those that are and then adjust the upper bound of this objective constraint to only allow a small percentage of flux to be carried in absent import reactions.

5.2.2.3 Software availability

The software featured in this work was developed using the Python programming language. Source-code for the ipFBA and routines to build extended models of metabolism compatible with this algorithm can be found in the GitHub repository at `https://github.com/BioSystemsUM/cobamp-grasp/`. The *cobamp* package is required as a dependency to use ipFBA. The case studies featured in this work are also available as Jupyter Notebooks that can be accessed in the `examples` folder of the repository. Plots were built using the `seaborn` and `matplotlib` packages.

5.3 Results

The Human-GEM (version 1.5.0) genome-scale metabolic reconstruction [41] was used as a template model for ipFBA in the breast cancer cell line analyses. The model and auxiliary reaction and metabolite tables were downloaded from the corresponding version release on the GitHub repository at `https://github.com/SysBioChalmers/Human-GEM`. For the renal cancer case study, the redHUMAN Recon3 model reconstruction developed by Masid et al. was used [156]. This model is a significantly reduced version of the Recon3 human generic model that attempts to provide a better representation of human core metabolism through reaction lumping and curated reaction constraints [156].

In all case studies, the same TRN from OmniPath was used. OmniPath is a large database of molecular biology resources and aggregates over 100 sources [157]. In this work, OmniPath was used to retrieve moderate to high confidence transcription factor interactions. Interactions without a consensus inhibition or stimulation activity were removed, resulting in a TRN with 2997 interactions which was integrated in the ipFBA extended model.

Fluxomics and transcriptomics data were obtained are those presented on Section 4.3.1

Transcriptomics data were then converted to TAS, first by assessing multiple thresholds and then selecting the best combinations for larger-scale cases. The 739 cell lines used in this study match those for which a CRISPR knockout screen was available. Gene/transcript nomenclature was converted using the latest HUGO Gene Nomenclature Committee approved symbol mappings whenever needed [136].

5.3.1 Metabolic flux predictions of the MCF7 cell line

The ipFBA approach was validated using fluxomics estimates from metabolomics data acquired for the MCF7 cell line. To ensure the method is capable of predicting fluxes from transcriptomics data, several flux predictions are generated using multiple parameter combinations. This is an essential step in which the optimal algorithm parameters will be obtained by comparing predicted flux distributions with experimental data and generating an appropriate value to quantify the predictive accuracy. In this analysis, the MCC is used as a classification metric.

The results achieved with ipFBA largely surpass the predictive ability of pFBA performed on context-specific models integrating the same type of data. When performing a direct comparison with the case study featured on Chapter 4, there is a large improvement with ipFBA predictions achieving average MCCS between 0.28 and 0.46, while the best context-specific model using FASTCORE was only able to achieve

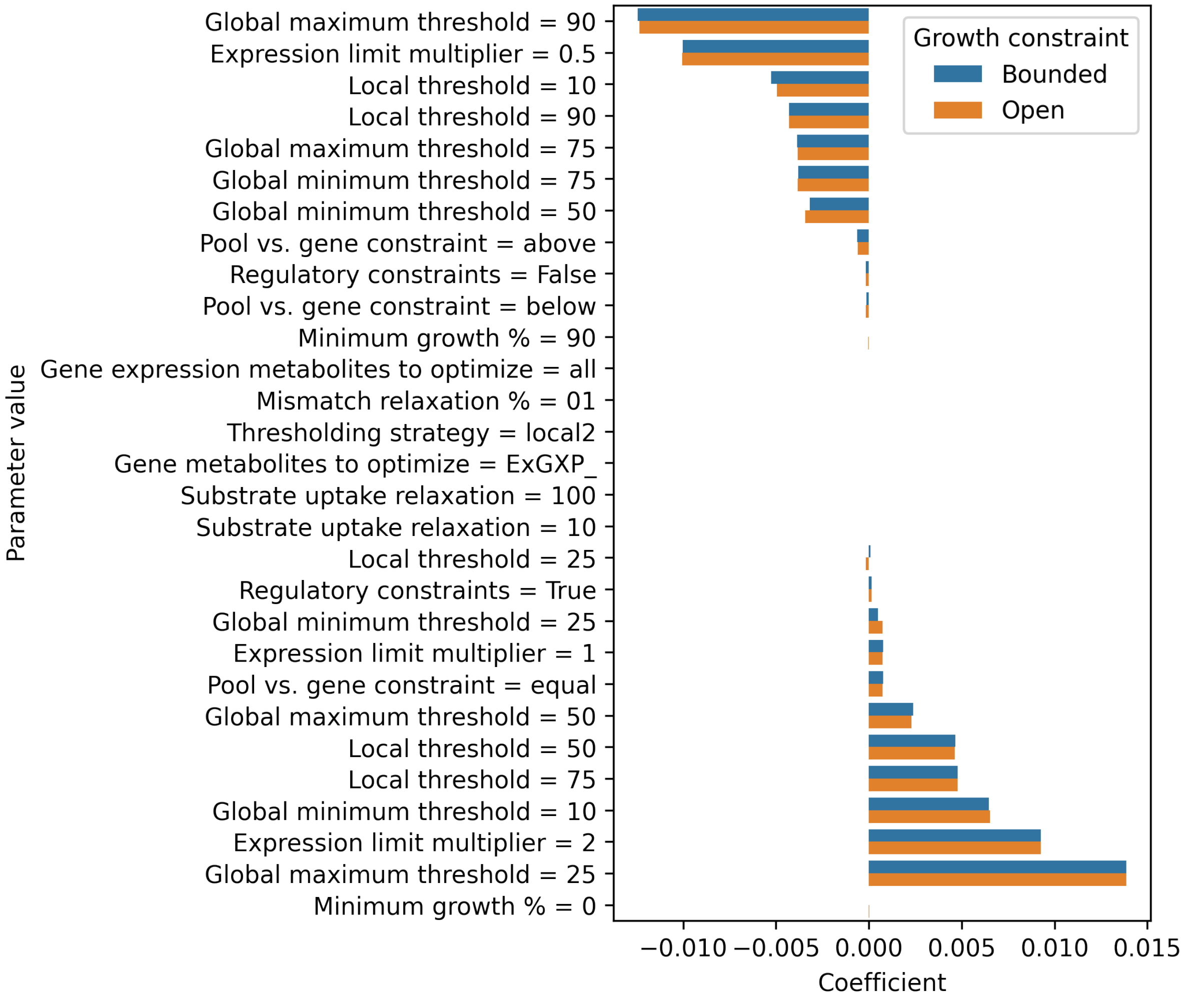


Figure 27: Overview of the impact of different parameter choices in the average MCC values. Each these values were obtained by fitting linear regression models with the results from the MCF7 case study. Each parameter choice is associated with two values, respectively, with (blue, bounded) or without (orange, open) bounds on the growth pseudo-reaction.

an average MCC of 0.20. This means that even using the worst parameter combinations, ipFBA leads to better predictions, thus validating the presented flux prediction approach. However, for the purpose of adapting this method for use with large-scale datasets, it is also important to analyse the sensitivity to alternative parametrisation choices.

The overall impact of parametrisation choices was assessed by fitting a linear regression model to predict average MCC values from the chosen parameters. The results from this analysis are depicted on Figure 27.

The most important parameter influencing MCF7 flux predictions was undoubtedly the global maximum threshold applied when converting gene expression values from TPM to TAS. Setting this parameter

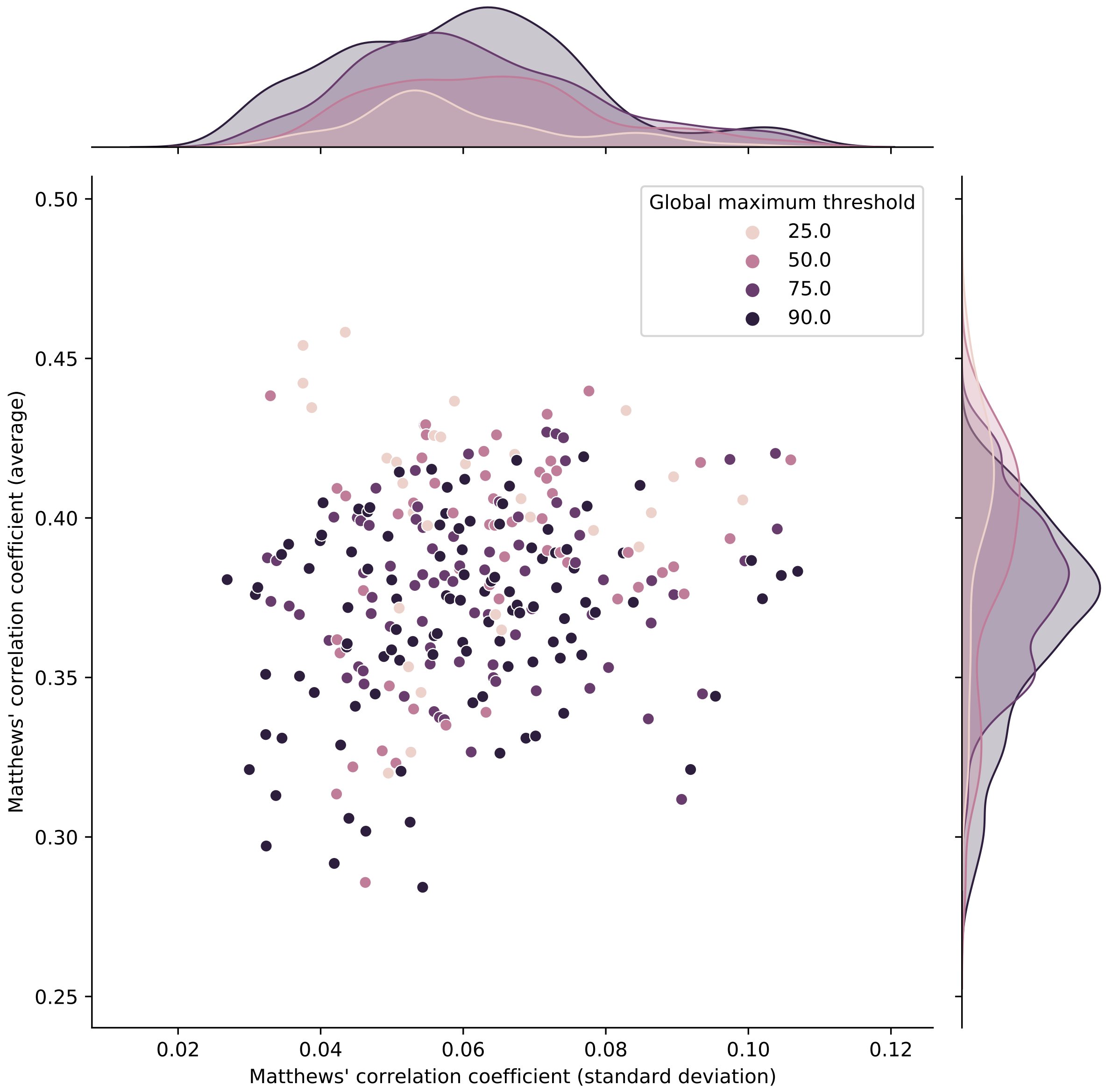


Figure 28: Overview of the average and standard deviation of MCC values for the MCF7 flux prediction case study and associated global maximum thresholds. The vertical axis shows increasing MCC average values while the horizontal axis shows their standard deviation. Each group of simulations is represented by a dot with increasingly darker colors for higher thresholds.

to the 25th percentile led to improved predictions, while the 90th percentile leads to worse results. These results can be observed on Figure 28. This demonstrates the importance of finding appropriate cutoff values to determine whether a given transcript can be deemed as active. As a result, the best global minimum threshold is the 10th percentile, since all combinations with a global maximum threshold at the 25th percentile can only be found with this combination. The best local thresholds were found at the 50th and 75th percentiles.

One important question answered through this analysis was the impact of regulatory interactions in this approach. The linear coefficients for this parameter could not demonstrate that these interactions contributed signficantly towards better predictions, appearing with negligible contributions. Upon closer inspection, however, we can identify a cluster of five solutions with high MCC and relatively low standard deviation, out of which only one was not generated with active regulatory interactions. This finding, although not confirmatory, indicates that this parameter does not negatively affect the predictive capabilities of ipFBA, and expands the solution space and the amount of information that can be overlayed in the model.

There are, however, some parameters with an expected impact that did not match the findings in this analysis. Both growth rate and substrate uptake objective constraints had negligible effects on the predictive ability of ipFBA which raises concerns about the ability to simulate certain autotrophies or dependence on certain growth media components. On the other hand, it is also possible that gene expression itself is the most important factor when attempting to predict metabolic activity.

We also did not observe any significant difference between constraining gene pool or gene expression metabolites. However, these constraints appear to be more effective when their respective bounds are set to double the expression value. Throughout the development of this method, one important concern was the correlation between fluxes associated with gene expression and metabolic fluxes themselves. Since this information is hard to determine, all gene expression fluxes contribute at the same rate towards the reactions they are associated with. The result of higher expression bounds is the expansion of the solution space which might be necessary for reactions with higher absolute flux values.

The resulting parameter combination with the highest average MCC was identified in this analysis and was used as a reference for the subsequent large-scale flux prediction efforts carried out using the entire CCLE panel.

5.3.2 Large-scale assessment of cancer phenotypes

The previous case study allowed the identification of optimal parameters for the prediction of intracellular fluxes in the MCF7 breast cancer cell line. Using these parameters, a large-scale evaluation of the CCLE cell line panel was carried out with the purpose of identifying patterns in predicted phenotypes and their association with clinical variables. In a first analysis, breast cancer phenotypes were grouped according to their reported breast cancer subtypes, revealing key differences in metabolic flux activity. Additionally, these simulations were also used to characterize the differences between primary and metastatic cell lines.

5.3.2.1 Metabolic heterogeneity in breast cancer subtypes

Principal component analysis was used to analyse breast cancer simulations and identify genes, reactions and signalling interactions that could be associated with the molecular subtypes of breast cancer represented in the CCLE. On Figure 29, simulations from ipFBA are projected in the first three principal components (PCs) and although the explained variance is relatively low for these components (17.2%, 6.2% and 4.0% for PCs 1,2 and 3, respectively), a clear separation between all subtypes can be observed. Negative and positive values for PC1 separate, respectively, luminal from basal cell lines, while PC2 further distinguishes between basal A and B subtypes and PC3 separates between luminal and HER2-positive subtypes.

After establishing that ipFBA was capable of predicting phenotypes that capture the molecular differences in breast cancer cell lines, the eigenvalues of each reaction, gene and interaction modelled by ipFBA were inspected to identify patterns associated with these subtypes. These results are shown in greater detail on Appendix I on Figures 35 (gene expression), 36 (reactions), and 37 (signalling interactions).

GALNT2 and *PLTP* were found to be highly associated with positive values on the first principal component, which are also correlated with the aggressive basal molecular subtypes. *GALNT2* has been identified as upregulated in malignant breast tissue [158] and is involved in the glycosylation of *PLTP*, which is ubiquitously expressed in both cancer and normal tissue [159, 160]. *SNAI2* was also found to be associated with the aggressive breast cancer subtypes found in PC1, which is consistent with the existing literature [161].

Interestingly, there was a striking difference in the mevalonate pathway subcellular localization. Basal breast cancer cell lines exhibited higher contributions from the cytosolic hydroxymethylglutaryl-CoA synthase encoded by *HMGCS1*, while luminal cell lines show higher fluxes with the mitochondrial enzyme encoded by the *HMGCS2* gene. The findings are supported by a recent experiment with several breast

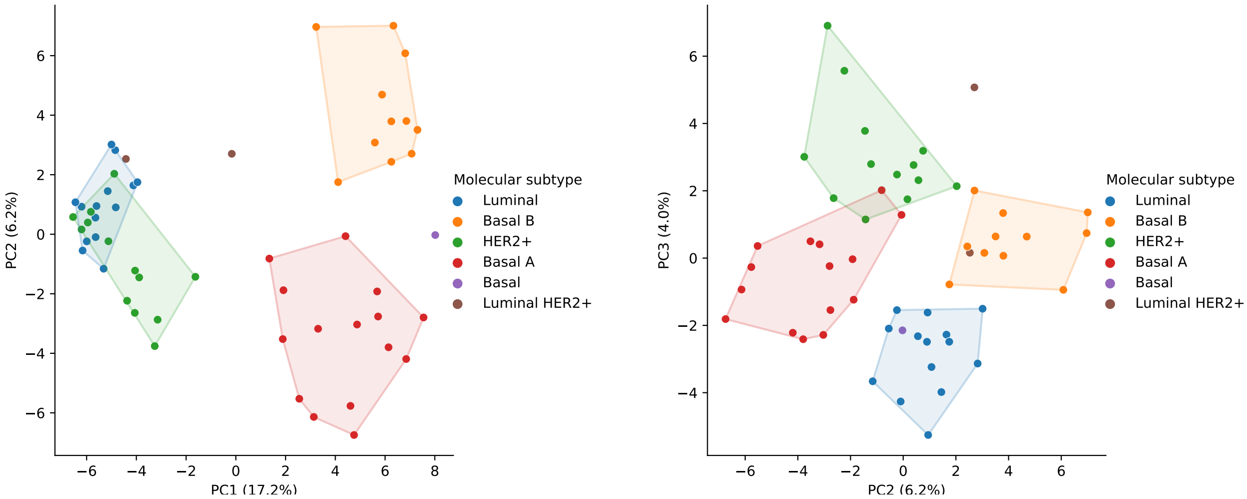


Figure 29: Principal component analysis projections of the flux distributions obtained for breast cancer cell lines.

cancer stem cell lines where *HMGCS2* correlates with the tumourigenic potential of each cell line [162].

Both synthases have identical metabolic activity, converting acetoacetyl-CoA and acetyl-CoA derived from fatty acids or branched-chain amino acids into 3-Hydroxy-3-Methylglutaryl-CoA (HMG-CoA) which is a precursor for mevalonate and acetoacetate. The enzyme encoded by *HMGCS1* present in the cytosol provides precursors that are mainly used for the production of mevalonate which then leads to the production of terpenoids, critical for the activation of signalling pathways that enhance cell growth and proliferation [163]. Walsh et al. have also recently found *HMGCS1* to be upregulated in breast cancer stem cells and that its activity correlates with the tumourigenic potential of each cell line [162].

The enzyme encoded by *HMGCS2* in the mitochondria synthesises HMG-CoA that is mobilized towards ketogenesis. Wang et al. recently shown that interrupting this process inhibits cell proliferation [164], which explains the negative association of this gene and enzyme with more aggressive cell lines.

These findings are also supported by the high contributions of *BCAT1* towards basal breast cancer cell metabolism. This gene encodes the cytosolic branched-chain-amino-acid aminotransferase, which catabolises leucine and α-ketoglutarate (AKG) into 4-methyl-2-oxopentanoate and L-glutamate. This enzyme is associated with autophagy as a regulator of the mechanistic target of rapamycin kinase (mTOR) pathway [165], cell proliferation and invasion in tumours from relapsed breast cancer patients [166] which explain its usage as a biomarker for unfavourable progression in triple negative breast cancer (TNBC) [167].

Glucose-3-phosphate conversion to dihydroxyacetone phosphate (DHAP) by the enzyme encoded by *GPD2* was also associated with basal breast cancer cells, although there is no significant evidence to claim this prediction is accurate. Nevertheless, Wu et al. have successfully used a GPD2 inhibitor to halt tumour growth in a mouse model [168].

Overall, the simulations generated with ipFBA show some degree of agreement with the existing literature and PCA was able to identify meaningful reactions and genes that are capable of distinguishing between distinct breast cancer phenotypes. Furthermore, some of these genes and reactions have already been touted as potential metabolic targets, demonstrating the potential of integrated metabolic models for drug target identification.

5.3.2.2 Predicting cell line primary disease with simulated fluxes

The identification of several differences in metabolism and gene expression through ipFBA prompted its usage to attempt to generate relevant features for classification tasks. Similarly to the results from the approach presented on Section 4.3.3.3, the goal is to verify if predicted fluxes can be used to predict a cell line's primary disease. The same flux analysis and supervised learning workflow presented on 4.2.5 was used for this task.

Standardized expression values and TASs generated from these data were used as baseline datasets. The simulations from ipFBA were divided into three datasets containing different sets of reactions, namely, the entire model representation, gene expression reactions and finally, only metabolic fluxes. Results for this classification task are shown on Figure 30.

Firstly, it is clear that transcriptomics measurements and their derived scores are much better than ipFBA when training with a low number of features, with average MCC values starting at 0.476 for standardized expression values and 0.51 for the associated TAS using only 50 features, while ipFBA solutions only reach 0.30 considering the complete set of reactions, 0.38 for the gene regulation and expression reactions and 0.275 for metabolic fluxes. However, these values drastically change as the amount of features increases and ipFBA simulations ultimately achieve nearly identical performance as the transcriptomics data that originated them, with MCCs of around 0.58.

The performance of ipFBA simulated fluxes appear to be the result of gene expression and regulatory interactions since ipFBA predictions have similar performance using all reactions or just those associated with genes rather than metabolic reactions. Nevertheless, models trained on the metabolic flux part of the ipFBA solution led to an average MCC of 0.47, which is a significant increase from the same case study using pFBA to predict phenotypes on models whose omics data was integrated using CSMR algorithms.

The results from this task demonstrate the ability for ipFBA to integrate omics data and generate

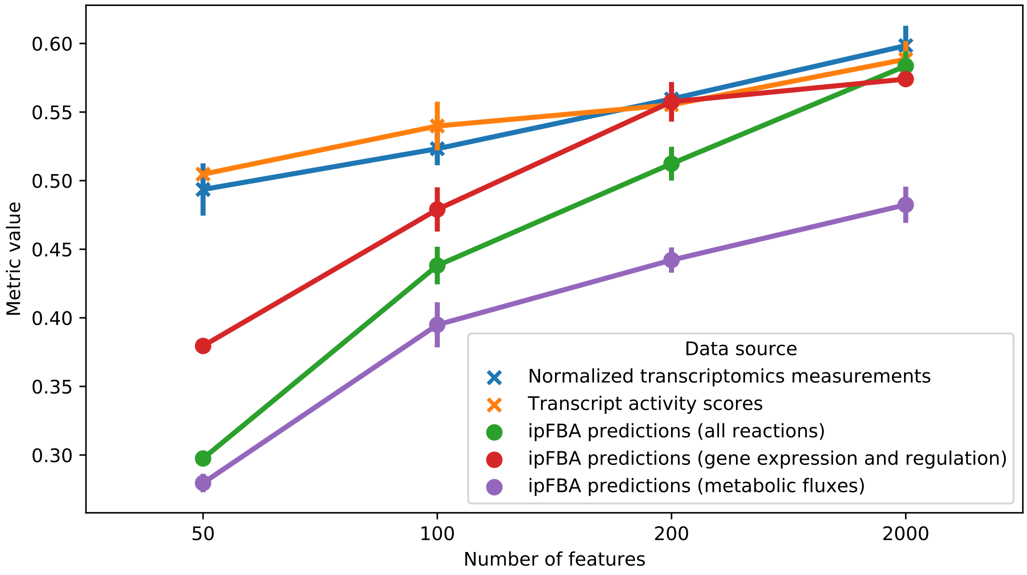


Figure 30: Predictive performance of random forest classifiers for prediction of cell line primary diseases using ipFBA and transcriptomics datasets. The vertical axis represents the Matthews' correlation coefficient value while the horizontal axis denotes the amount of features kept in the datasets. The cross symbols denote the transcriptomics measurements and scores derived from them while the dots represent predictions obtained with ipFBA

simulated fluxes for distinct cell lines and capture their differences better than CSMR methods.

5.3.3 Identification of metabolic patterns in renal cancer

The previous case studies involving cell lines on the CCLE were critical to assess the influence of parametrisation and to establish the predictive ability of ipFBA. To further demonstrate the applicability of this method in other datasets, a renal cancer cohort was obtained from the work of Dugourd et al. [154] where the COSMOS multi-omics integration tool is successfully applied.

The aim is to use flux and gene activity predictions to establish a multi-layer network, similarly to COSMOS, that can provide insights into the metabolic patterns and differences associated with renal cancer. Unlike the previous case studies, a control (healthy) group is available, which allows for differential analysis of the obtained predictions. Flux predictions were obtained for both healthy and tumour samples using ipFBA. In this case study, both metabolomics and transcriptomics were given as inputs for the simulations.

The predictions from the renal cancer cohort were also used to generate differential networks that summarize the variation in activity of various biomolecules represented by the ipFBA extended modelling approach.

The predictions obtained for this case study were divided in their corresponding sample groups, namely healthy and tumour. Reactions that were found to be inactive in all samples were removed from the analysis. Using these values, the sample groups were summarised by obtaining the average value of each flux within its cohort group, resulting in two flux distributions.

A final flux distribution was calculated by dividing average flux values from the tumour cohort by those from the healthy tissue. This is similar to obtaining a fold change between both conditions and a logarithm was also applied to further establish a distinction between fluxes, enzymes and regulators that have increased (positive values) or decreased (negative values) presence in renal cancer.

5.3.3.1 Metabolic patterns in renal cancer

The renal cancer cell flux predictions were analysed using PCA, similarly to the previous case study. In a preliminary analysis, the distribution of healthy and tumour samples was assessed to assert whether predicted fluxes can separate between both types of samples. This is depicted on Figure 31, where it is possible to identify quadrants where healthy and tumour samples are well represented.

Overall, PCA led to a reduced space where healthy and tumour samples are separated by principal component 1, which explains a high amount of the fluxes' variance (87.03%). Negative values for principal components 1 and 2 do appear to associate with tumour samples. However, the separation between sample sites is not as reliable as the one found for the breast cancer case study.

This result led to the development of a different analysis pipeline to attempt to extract knowledge from ipFBA simulations by calculating a differential flux distribution for the entire dataset.

Firstly, *NR2F1* was identified as a regulator gene with links to valine metabolism and mitochondrial β-oxidation of fatty acids. This is demonstrated on Figure 33, where *NR2F1* inhibits the *ACADM* gene. Interestingly, this pattern is already identified in Human Protein Atlas (HPA), where *NR2F1* expression correlated negatively with survival while *ACADM* correlates positively [159].

Furthermore, it is also possible to observe a downregulation of genes such as *HADHA*, *EHHADH* and *ECHS1*, which are subunits of the hydroxyl-coenzyme A dehydrogenase enzyme. Overexpression of these three genes has been shown to halt tumour growth in clear cell renal carcinoma, and also touted as biomarkers capable of separating healthy from normal tissue as well as expected prognosis of the disease

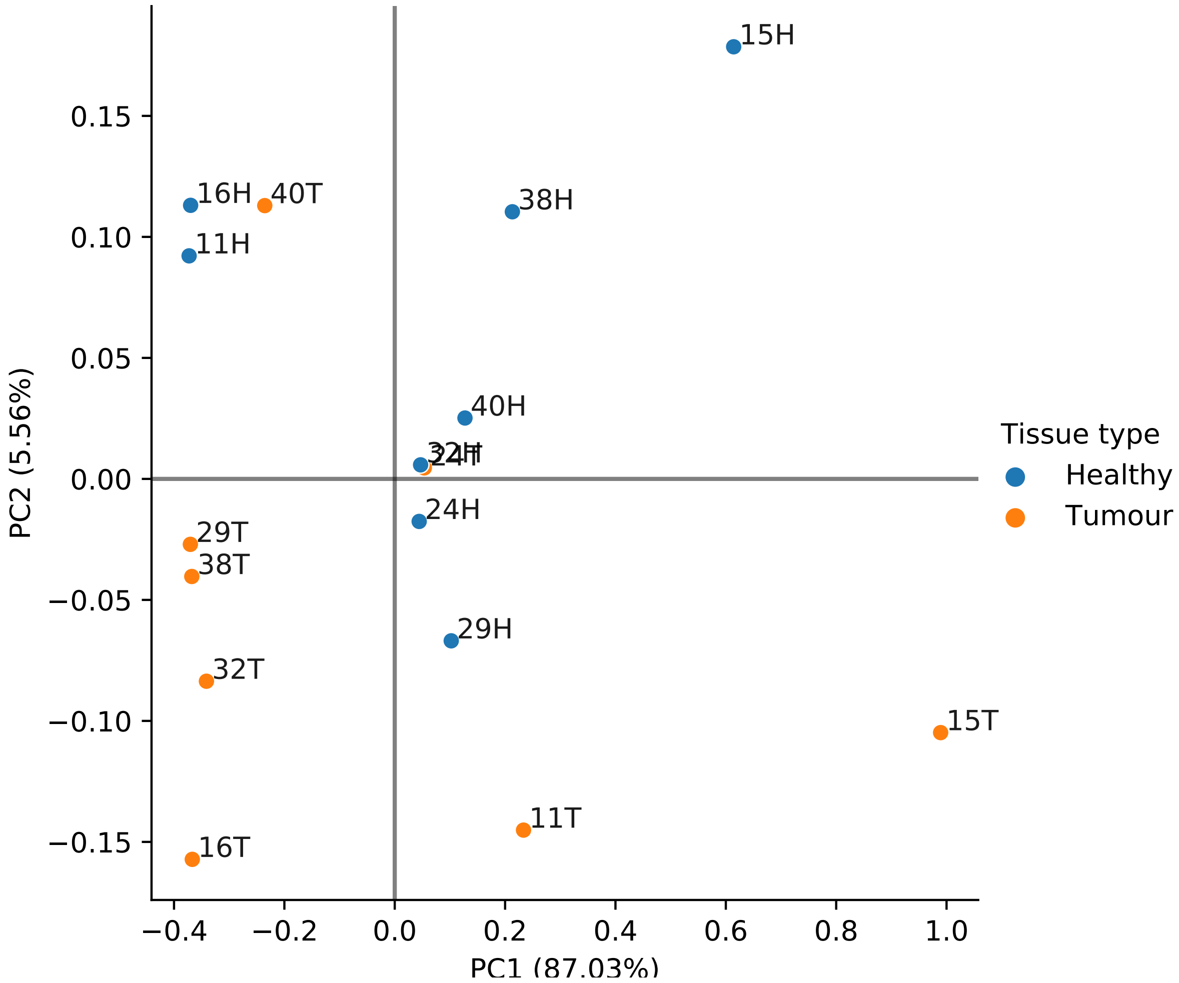


Figure 31: Principal component analysis of predicted fluxes from ipFBA of healthy and tumour samples from renal cancer patients. The horizontal and vertical lines mark the origins for the first and second principal component axes. Samples are coded with the patient identifier and a H or T character for healthy and tumour samples, respectively.

[169–172].

Despite these encouraging results, the mechanisms for this downregulation found in literature could not be predicted by ipFBA. The mTOR signalling pathway, for example, is assumed to drive the proliferative phenotypes that decrease mitochondrial β-oxidation. This pathway was not found to be differentially associated with these enzymes in this analysis.

In the glycolysis pathway, several differentially active enzymes and isoforms were found. As a critical pathway responsible for generating a considerable amount of adenosine triphosphate (ATP) in several cancers, the initial part of this pathway was found to be upregulated. The first step in the catalysis of glucose is hexokinase, whose *HK2* isoform was found to be highly active in tumour samples, unlike its

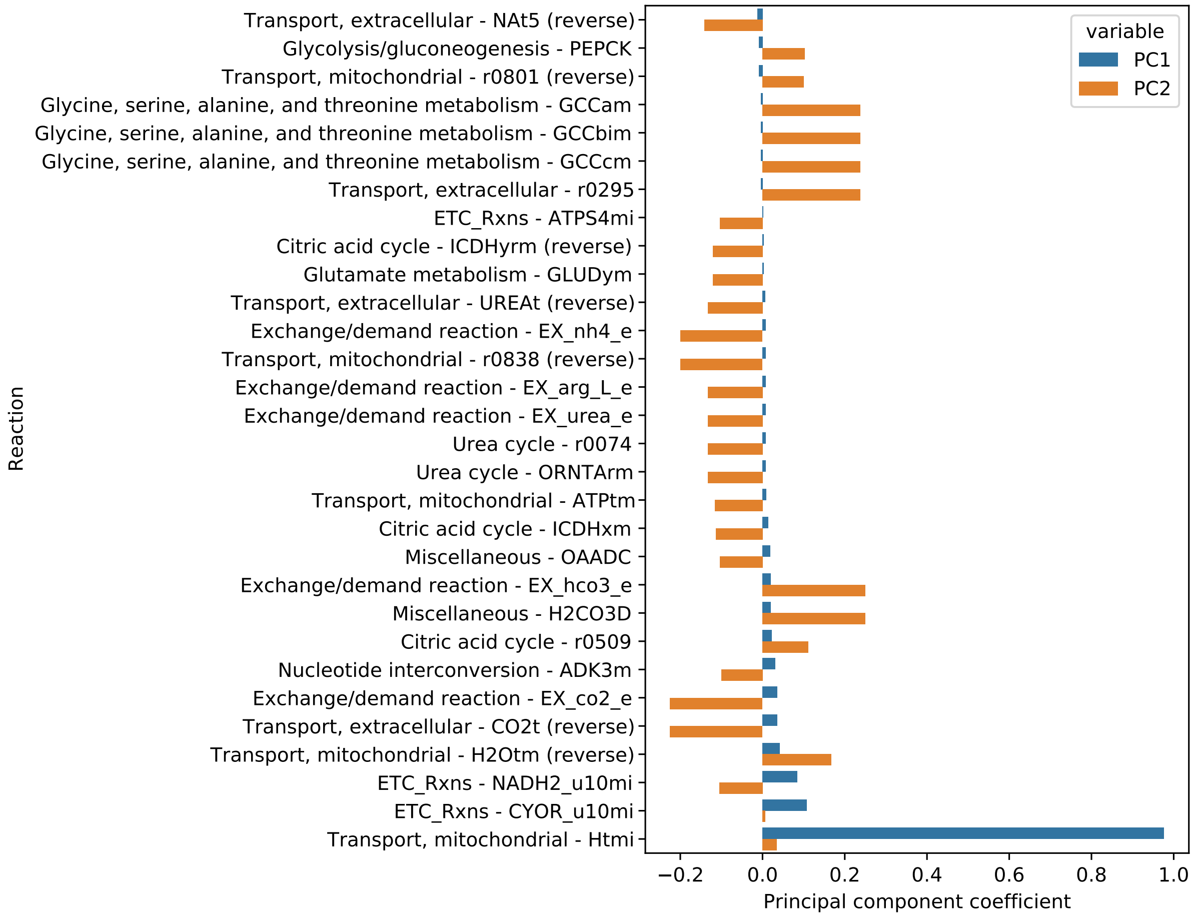


Figure 32: Reactions with the highest observed eigenvalues after principal component analysis using predicted fluxes of renal cancer samples. Each reactions contains two bars which represent the eigenvalues for the first two principal components.

HK1 counter part. This is an expected change, which has been found to drive the Warburg effect [173].

phosphofructokinase (PFK) was also highly active in tumour samples, with the enzyme encoded by the *PFKP* gene (platelet-type) appearing as the dominant isoform, unlike *PFKL* (liver-type). These are the main two enzymes catalysing the PFK reaction and although *PFKL* is also typically expressed in cancer, *PFKP* expression is correlated with worse outcomes, as described by Umar et al. [174]. Interestingly, *ZBTB7A* appeared as a negative regulator of *PFKP*. This versatile protein has been reported to function as both oncogene and oncosuppressor, with its effects varying with cancer types [175]. ipFBA results are also unclear for this effect as its inhibition of the *PKM* gene appears to occur in healthy cells but not in tumours due to the high level of *PFKP* expression, although *ZBTB7A* appears as a promising target for cancer therapies.

pyruvate kinase (PYK) and phosphoenolpyruvate carboxykinase (PEPCK) display overall lower flux in

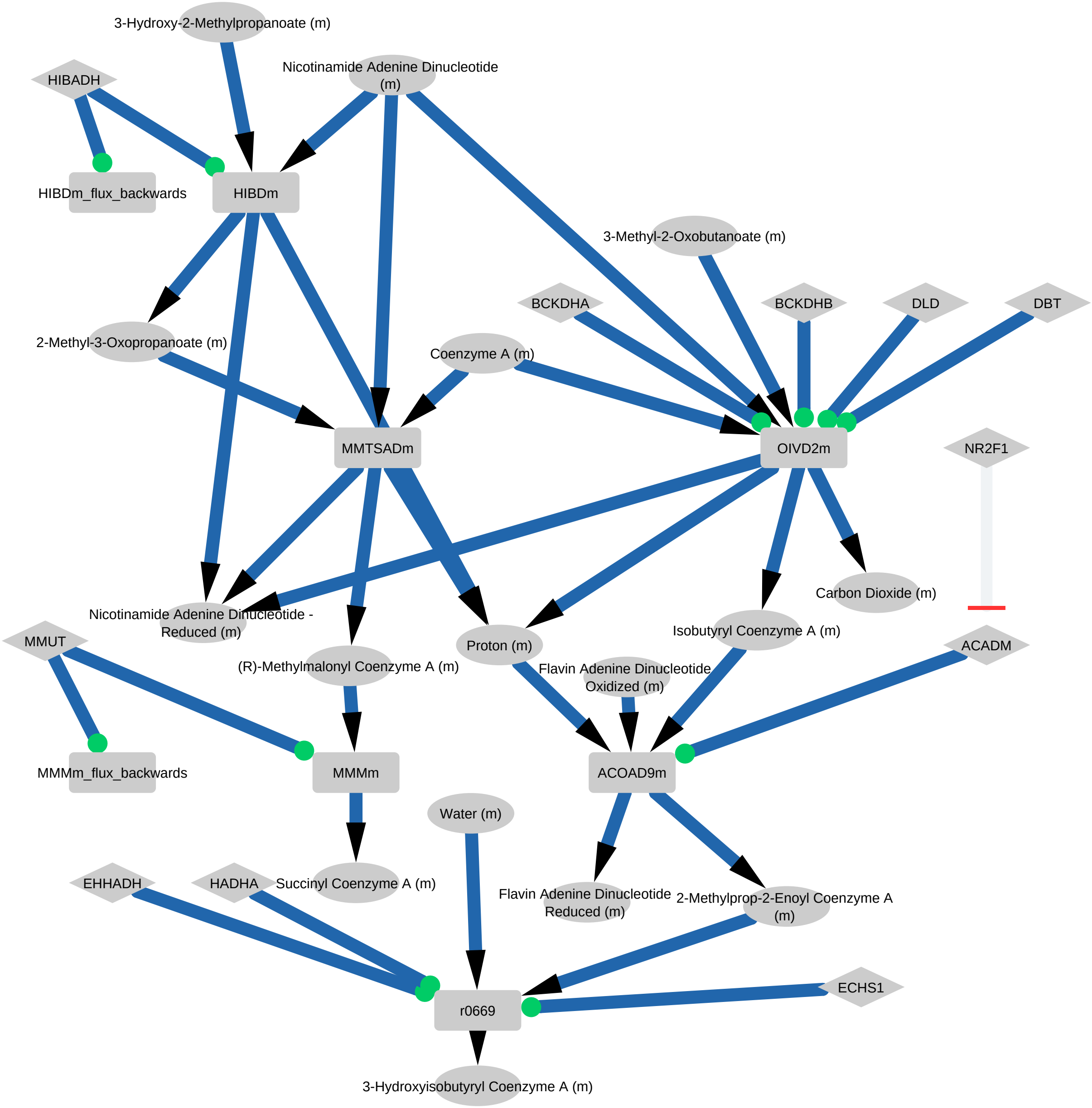

3-Hydroxy-2-Methylpropanoate (m)
HIBADH
Nicotinamide Adenine Dinucleotide (m)
HIBDm_flux_backwards
HIBDm
3-Methyl-2-Oxobutanoate (m)
BCKDHA
BCKDHB
DLD
DBT
2-Methyl-3-Oxopropanoate (m)
Coenzyme A (m)
MMTSADm
OIVD2m
NR2F1
Carbon Dioxide (m)
Nicotinamide Adenine Dinucleotide - Reduced (m)
Isobutyryl Coenzyme A (m)
MMUT
Proton (m)
ACADM
(R)-Methylmalonyl Coenzyme A (m)
Flavin Adenine Dinucleotide Oxidized (m)
MMMm_flux_backwards
MMMm
ACOAD9m
Water (m)
EHHADH
HADHA
Succinyl Coenzyme A (m)
Flavin Adenine Dinucleotide Reduced (m)
2-Methylprop-2-Enoyl Coenzyme A (m)
ECHS1
r0669
3-Hydroxyisobutyryl Coenzyme A (m)

Figure 33: Impact of *NR2F1* in the metabolism of branched chain amino acids and fatty acid oxidation. The blue arrows indicate a downregulation of the displayed pathways in renal cancer when compared to healthy tissue. Arrows represent consumption and production of metabolites (ellipses) by reactions (rounded rectangles) which are associated to proteins (diamond shape). An interaction with a green circle indicates the source enzyme catalyses the target reaction. Inhibition is represented by the red dash.

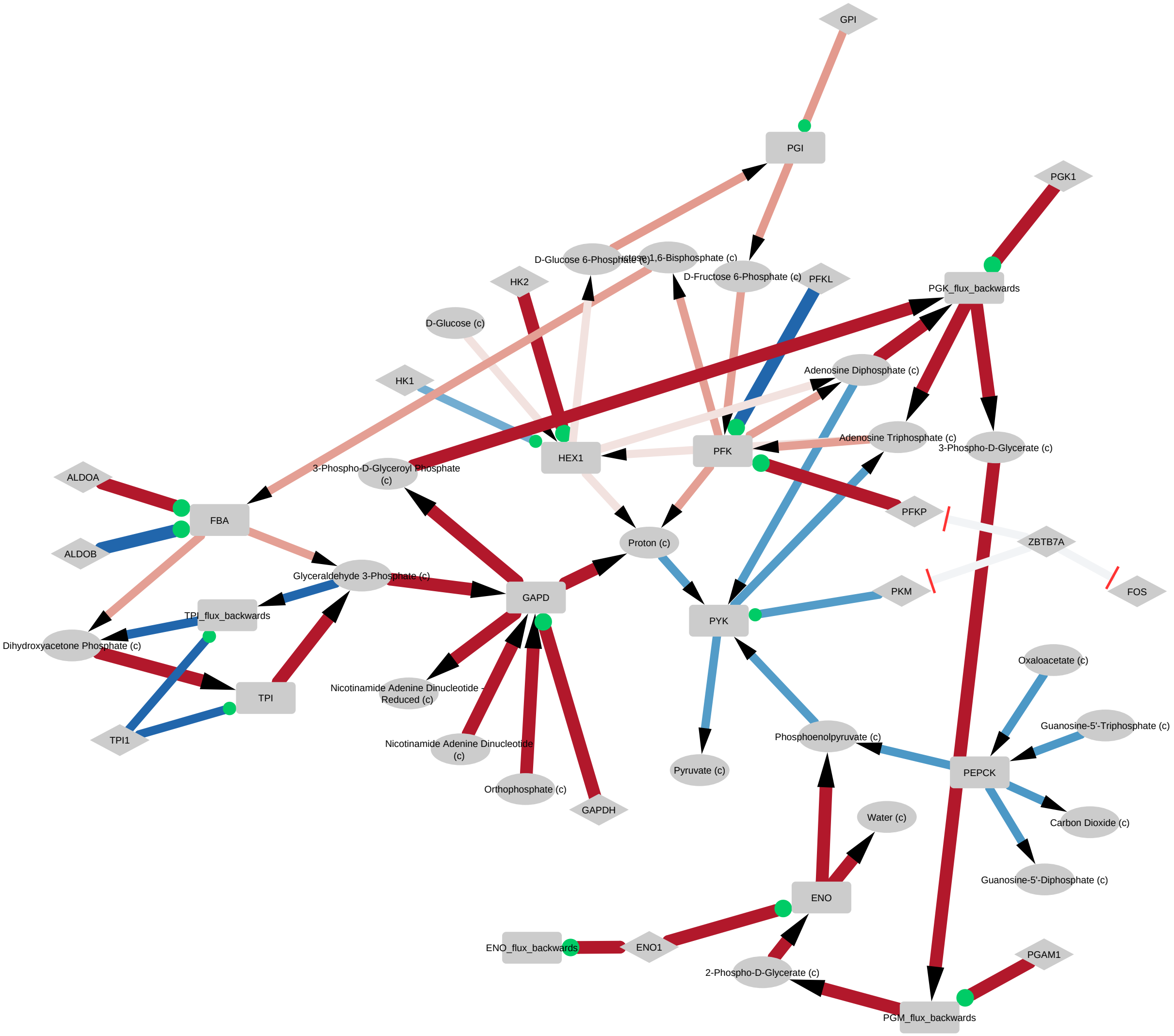

GPI
PGI
PGK1
D-Glucose 6-Phosphate (c)
D-Fructose 1,6-Bisphosphate (c)
D-Fructose 6-Phosphate (c)
PFKL
HK2
PGK_flux_backwards
D-Glucose (c)
HK1
Adenosine Diphosphate (c)
Adenosine Triphosphate (c)
3-Phospho-D-Glycerate (c)
PFK
HEX1
ALDOA
3-Phospho-D-Glyceroyl Phosphate (c)
FBA
PFKP
ZBTB7A
ALDOB
Proton (c)
Glyceraldehyde 3-Phosphate (c)
GAPD
PKM
FOS
TPI_flux_backwards
PYK
Dihydroxyacetone Phosphate (c)
Oxaloacetate (c)
Nicotinamide Adenine Dinucleotide - Reduced (c)
TPI
TPI1
Guanosine-5'-Triphosphate (c)
Phosphoenolpyruvate (c)
Nicotinamide Adenine Dinucleotide (c)
PEPCK
Pyruvate (c)
Orthophosphate (c)
GAPDH
Water (c)
Carbon Dioxide (c)
Guanosine-5'-Diphosphate (c)
ENO
ENO_flux_backwards
ENO1
PGAM1
2-Phospho-D-Glycerate (c)
PGM_flux_backwards

Figure 34: ipFBA predictions for differential glycolytic activity in renal cancer. Blue arrows indicate a down-regulation of the displayed pathways while red arrows represent up-regulation, both relative to healthy tissue. Arrows represent consumption and production of metabolites (ellipses) by reactions (rounded rectangles) which are associated to proteins (diamond shape). An interaction with a green circle indicates the source enzyme catalyses the target reaction.

renal cancer. These enzymes have an opposite activity since PYK is the enzyme involved in the final step of glucose catalysis to pyruvate, while PEPCK initiates gluconeogenesis with oxaloacetate as its substrate. Israelsen et al. describe that the reduction in expression of *PKM* gene encoding the PYK enzyme and decrease in enzymatic activity allows metabolic fluxes to be diverted to other biosynthetic pathways, allowing cancer cells to adapt to overcome nutrient deprivation [176].

5.4 Conclusions

The constraint-based modelling framework is a simple mathematical representation capable of modelling certain metabolic phenotypes. Through this work, an extended model representation was devised to seamlessly include overarching layers surrounding metabolism by converting transcriptional regulatory networks and the functional annotation represented by GPR rules into a unified network.

The top-down structure of this network allows gene expression and regulation to be modelled as reactions by only allowing interactions between two layers at once. Regulatory genes control the expression of metabolic genes, which then affect the flux through reactions catalysed by them.

Combining an expanded network representation with a novel omics integration method capable of including multiple inputs improved predictions when compared with CSMR algorithms. This was demonstrated by comparing this method with both fluxomics data, as was the case with the MCF7 case study, and a large-scale cell line panel leading to a clear picture on the heterogeneity of breast cancer. ipFBA was also useful as a method to generate multi-omics networks to characterise and identify patterns in clear cell renal carcinoma.

Several deregulated fluxes were found to be associated with breast cancer, with many of them recently described as possible therapeutic targets. This demonstrates the ability of the developed method to provide mechanistic answers to understand cancer metabolism. Using reduced models also allowed the creation of differential metabolic maps for valine metabolism, as well as glycolysis and the pentose phosphate pathway (PPP).

Despite the positive aspects of this method, some limitations are evident. Firstly, there is a clear reliance on a broad transcriptomics dataset since ipFBA mainly drives its predictions by penalising flux associated with down-regulated genes. This might invalidate other predictions on mutant phenotypes for which the gene expression state has not been measured. A particularly important use case for this is the prediction of essential genes, which might not provide insightful results if determined with ipFBA.

The ability to predict gene regulatory interactions driving certain phenotypes was also lacking. The differential network analysis performed in the renal cancer case study showed very few regulatory targets which could explain the observed changes in gene expression and flux. On one hand, the model used in this analysis was small, with far less metabolic genes to be affected by the interactions in the causal network. Furthermore, these interactions are represented as fluxes which imply a consumption of the regulator to produce the regulated gene which might not be able to carry flux due to steady-state violations.

Parametrisation is also a limiting factor in the predictive ability of this method. Firstly, omics data are integrated in ipFBA using thresholds to distinguish between high and low expression, which requires a calibration step to identify. Furthermore, gene expression is directly correlated with flux and regulatory interactions since genes, regulators and fluxes are interconverted at a ratio of 1, although it is certainly impossible to assume that this is biologically accurate.

The model representation devised in this approach leads to very large LP problems. Although modern solvers are capable of handling large problems such as this one, running multiple simulations with multiple parameter choices can quickly become a computationally intensive task.

Despite the former limitation, the method is still useful considering it can be used to generate a flux distribution that can then be discretised. It is therefore necessary to claim that ipFBA should be primarily used to generate pathways rather than flux distributions with numerically accurate flux values.

With the decreasing costs of high-throughput omics technologies, tools such as ipFBA could be used to contextualise this bulk of information to provide mechanistic insights on metabolic patterns, especially in a personalised medicine setting where each patient's omics signature can be used to establish a metabolic profile which is then useful to guide further therapies.

6 Conclusions

The work presented in the previous chapters resulted in three essential contributions, namely:

- Three computational frameworks for constraint-based modelling analysis and integration of omics data;
- A generic and scalable pipeline to extract context-specific metabolic models from omics data;
- Extended metabolic model representations enabling integration of multi-omics measurements to improve flux predictions.

Although the results shown in this work are indicative of the potential of combining constraint-based modelling and omics quantification in providing insights into cancer metabolism and pointing targets for drug development, several limitations can be identified. In this final chapter, some concluding remarks on the achieved outcomes are presented as well as targets for improvement as part of future work.

6.1 Overall outcomes

The work that resulted from this thesis culminated in the development of computational methods providing metabolic pathway analysis, model extraction and phenotype prediction methods capable of leveraging omics data and biological networks to simulate cancer metabolism.

The first major outcome was the development of three computational frameworks that are modular, extensible and provided through open-source platforms. CoBAMP provides a scalable architecture for the implementation of advanced constraint-based methods based on LP or MILP, while also containing pathway analysis tools that were previously only available in proprietary software platforms. TROPPO builds on this framework to include features to handle omics data inputs and methods to integrate this information into constraint-based models through efficient implementations of model extraction and phenotype prediction methods. Finally, GRaSP also extends the functionalities of CoBAMP by implementing methods to represent causal networks of biological molecules and algorithms to convert them into representations that can be directly integrated into constraint-based models.

With a rich set of software tools, it was then possible to devise a model extraction pipeline capable of processing transcriptomics data to reconstruct tissue-specific models of human cells. This pipeline builds on previous efforts and includes validation steps to assert the quality of context-specific models, leveraging fluxomics, knockout screens or metabolic tasks to generate useful metrics for model evaluation. The pipeline also includes model correction steps to which a significant contribution was made with the implementation of a novel gapfilling algorithm that uses EFM enumeration to quickly identify reactions that can improve model consistency. The issues posed by parametrisation were addressed by establishing the concept of a reference sample for calibration which uses validation metrics to decide the best combinations of parameters to apply at a larger scale.

The pipeline developed in this work improved the predictive ability of previously built context-specific models using transcriptomics and a GSMM as the only inputs. The performance of these models was demonstrated through the prediction of lethal genes for more than 700 cancer cell lines, as well as prediction of flux activity in MCF7 cell line models. Using pFBA together with a flux analysis pipeline involving PCA, several metabolic patterns in breast cancer could be identified. Simulated fluxes were also sufficient to establish a division between various molecular subtypes of breast cancer, with predictions capable of identifying enzymes that are critical for the development of aggressive cancer subtypes. Finally, our CSMR pipeline and TAS calculation approach were able to generate predictions that can be used with relative success in other classification tasks, extending the usage of these approaches as feature extraction methods for supervised learning.

The developed CSMR pipeline yielded positive results, but also revealed some limitations in the types of omics data to integrate and the inexistent representation of regulatory and signalling interactions within these models. As such, in the final part of this work, a novel model representation and phenotype prediction algorithm was developed. The ipFBA approach developed in this work integrates multiple biological network layers into a single model, which can be used as a scaffold to directly integrate omics measurements to enhance flux predictions, unlike CSMR algorithms. To this end, a novel simulation method was also developed, predicting phenotypes by prioritising consistency with gene expression.

ipFBA was successfully applied in flux prediction tasks, outperforming the models reconstructed in our CSMR pipeline for the MCF7 cell line. The method was also used to provide a much clearer separation of breast cancer subtypes and phenotypes, where it was possible to assert the role of pathways such as mevalonate biosynthesis which drive tumour growth and aggressiveness in TNBC. A renal cancer case study was also analysed, where it was possible to generate differential metabolic pathway maps which explained the involvement of the valine and glycolysis pathways in this disease. Finally, flux predictions from ipFBA were also significantly better at providing relevant features for the supervised learning tasks.

In summary, the methods developed in this work reveal a greater role for modelling approaches with high-throughput omics data as we approach a future where precision medicine could become the norm. Although the predictive ability of these methods still lacks the rigour demanded in medical applications, both the CSMR pipeline and ipFBA are capable of generating useful knowledge towards a better understanding of human disease as well as the identification of targets for future therapies.

6.2 Publications

Several publications were written during the development of this doctoral thesis. Five of these publications have been accepted and contributed to part of this work, namely:

- V. Vieira and M. Rocha. "CoBAMP: a Python framework for metabolic pathway analysis in constraint-based models". In: *Bioinformatics* 35.24 (2019), pp. 5361–5362
- J. Ferreira et al. "Troppo - A Python Framework for the Reconstruction of Context-Specific Metabolic Models". In: *Practical Applications of Computational Biology and Bioinformatics, 13th International Conference*. Ed. by F. Fdez-Riverola et al. Cham: Springer International Publishing, 2020, pp. 146–153
- J. Ferreira, V. Vieira, and M. Rocha. "Genome-Scale Metabolic Models". In: *Systems Medicine*. Ed. by O. Wolkenhauer. Oxford: Academic Press, 2021, pp. 420–428
- A. Dugourd et al. "Causal integration of multi□omics data with prior knowledge to generate mechanistic hypotheses". In: *Molecular Systems Biology* 17.1 (Jan. 2021), e9730

Additionally, the manuscript with the work contained on Chapter 4 has been submitted and is pending review. It is available as a pre-print, whose reference is the following:

- V. Vieira, J. Ferreira, and M. Rocha. "A pipeline for the reconstruction and evaluation of context-specific human metabolic models at a large-scale". In: *bioRxiv* (2021)

At the time of writing, the manuscript with the methods and results detailed on 5 is currently in preparation.

Finally, the following publications were also developed within the scope of the doctoral programme, but were not a core part of this thesis, namely:

- V. Vieira et al. *A Model Integration Pipeline for the Improvement of Human Genome-Scale Metabolic Reconstructions*. 2018

- V. Vieira et al. "Comparison of pathway analysis and constraint-based methods for cell factory design". In: *BMC bioinformatics* 20.1 (2019), pp. 1–15

6.3 Future work

The software tools and analyses featured in this thesis prompted several questions and shown development paths that could improve both the accessibility and accuracy of the approaches presented throughout this thesis.

The software presented in this work is mostly aimed at moderately advanced users with a good understanding of programming as well as systems biology. Despite a considerable amount of effort to simplify the usage of the CSMR pipeline and ipFBA, these methods may not be accessible to a significant part of the community. A graphical user interface could be implemented in the future to bridge this gap and eliminate the need for scripting.

The CSMR pipeline could also be expanded with more algorithms from which to choose. However, this should also be compounded with the development of a consensus model building algorithm capable of merging several models from various algorithms and data preprocessing parameters. This would ultimately facilitate the reconstruction of a single representative model a given omics context.

The CSMR pipeline could also include an optional design step to propose gene modifications with the purpose of achieving a given phenotype, such as cell death in the case of tumour cells. Furthermore, the integration of a drug database containing information on toxicity and mode of action could be used to probe a number of cancer cell lines and predict their drug sensitivity profiles.

The ipFBA approach was capable of generating relatively accurate flux predictions. However, due to the nature of the method's implementation, it is not suited to the simulation of mutant phenotypes, since the objective function is reliant on gene expression data. A new method or extension to this algorithm could be implemented to solve this issue as it would allow for the correct identification of lethal genes.

Bibliography

[1] D. Hanahan and R. A. Weinberg. “Hallmarks of cancer: the next generation”. In: *cell* 144.5 (2011), pp. 646–674.

[2] P. P. Hsu and D. M. Sabatini. “Cancer cell metabolism: Warburg and beyond.” In: *Cell* 134.5 (Sept. 2008), pp. 703–7.

[3] A. Varma and B. O. Palsson. “Stoichiometric flux balance models quantitatively predict growth and metabolic by-product secretion in wild-type Escherichia coli W3110.” In: *Applied and environmental microbiology* 60.10 (Oct. 1994), pp. 3724–31.

[4] M. P. Pacheco et al. “Integrated metabolic modelling reveals cell-type specific epigenetic control points of the macrophage metabolic network”. In: *BMC Genomics* 16.1 (2015), pp. 1–24.

[5] A. Schultz and A. A. Qutub. “Reconstruction of Tissue-Specific Metabolic Networks Using CORDA”. In: *PLoS Computational Biology* 12.3 (2016), pp. 1–33.

[6] R. Agren et al. “Identification of anticancer drugs for hepatocellular carcinoma through personalized genome-scale metabolic modeling”. In: *Molecular Systems Biology* 10.3 (Mar. 2014), p. 721.

[7] L. F. de Figueiredo et al. “Computing the shortest elementary flux modes in genome-scale metabolic networks”. In: *Bioinformatics* 25.23 (2009), pp. 3158–3165.

[8] A. von Kamp and S. Klamt. “Enumeration of smallest intervention strategies in genome-scale metabolic networks”. In: *PLoS Comput Biol* 10.1 (2014), e1003378.

[9] B. Palsson. “The challenges of in silico biology”. In: *Nature Biotechnology* 18.11 (Nov. 2000), pp. 1147–1150.

[10] H. Kitano. “Systems Biology: A Brief Overview”. In: *Science* 295.5560 (2002), pp. 1662–1664.

[11] H. Kitano. “Computational systems biology”. In: *Nature* 420.6912 (2002), pp. 206–210.

[12] R. Leinonen et al. “The European nucleotide archive”. In: *Nucleic Acids Research* 39.SUPPL. 1 (2011), pp. 31–34.

[13] J. Mashima et al. “DNA Data Bank of Japan”. In: *Nucleic Acids Research* 45.D1 (2017), pp. D25–D31.

[14] D. A. Benson et al. “GenBank”. In: *Nucleic Acids Research* 41.D1 (2013), pp. 36–42.

[15] R. L. Grossman et al. “Toward a Shared Vision for Cancer Genomic Data”. In: *New England Journal of Medicine* 375.12 (Sept. 2016), pp. 1109–1112.

[16] M. Uhlen et al. “Tissue-based map of the human proteome”. In: *Science* 347.6220 (2015), p. 1260419.

[17] J. Lonsdale et al. “The Genotype-Tissue Expression (GTEx) project”. In: *Nature Genetics* 45.6 (May 2013), pp. 580–585.

[18] M. Lizio et al. “Gateways to the FANTOM5 promoter level mammalian expression atlas”. In: *Genome Biology* 16.1 (2015), pp. 1–14.

[19] N. Kolesnikov et al. “ArrayExpress update-simplifying data submissions”. In: *Nucleic Acids Research* 43.D1 (2015), pp. D1113–D1116.

[20] T. Barrett et al. “NCBI GEO: Archive for functional genomics data sets - Update”. In: *Nucleic Acids Research* 41.D1 (2013), pp. 991–995.

[21] M. Krupp et al. “RNA-Seq Atlas-a reference database for gene expression profiling in normal tissue by next-generation sequencing”. In: *Bioinformatics* 28.8 (2012), pp. 1184–1185.

[22] M.-S. Kim et al. “A draft map of the human proteome”. In: *Nature* 509.7502 (May 2014), pp. 575–581.

[23] T. S. Keshava Prasad et al. “Human Protein Reference Database - 2009 update”. In: *Nucleic Acids Research* 37.SUPPL. 1 (2009), pp. 767–772.

[24] D. S. Wishart et al. “HMDB 4.0: the human metabolome database for 2018”. In: *Nucleic Acids Research* 46.November 2017 (2017), pp. 608–617.

[25] *The Metabolomics Workbench*.

[26] K. Haug et al. “MetaboLights - An open-access general-purpose repository for metabolomics studies and associated meta-data”. In: *Nucleic Acids Research* 41.D1 (2013), pp. 781–786.

[27] Z. Zhang et al. “CeCaFDB: A curated database for the documentation, visualization and comparative analysis of central carbon metabolic flux distributions explored by 13C-fluxomics”. In: *Nucleic Acids Research* 43.D1 (2015), pp. D549–D557.

[28] S. J. Wiback and B. O. Palsson. “Extreme pathway analysis of human red blood cell metabolism”. In: *Biophysical Journal* 83.2 (2002), pp. 808–818.

[29] N. D. Price et al. “Network-based analysis of metabolic regulation in the human red blood cell”. In: *Journal of Theoretical Biology* 225.2 (2003), pp. 185–194.

[30] T. D. Vo, H. J. Greenberg, and B. O. Palsson. “Reconstruction and functional characterization of the human mitochondrial metabolic network based on proteomic and biochemical data”. In: *Journal of Biological Chemistry* 279.38 (2004), pp. 39532–39540.

[31] T. D. Vo et al. “Isotopomer analysis of cellular metabolism in tissue culture: A comparative study between the pathway and network-based methods”. In: *Metabolomics* 2.4 (2006), pp. 243–256.

[32] N. C. Duarte et al. “Global reconstruction of the human metabolic network based on genomic and bibliomic data”. In: *Proceedings of the National Academy of Sciences of the United States of America* 104.6 (2007), pp. 1777–1782.

[33] H. Ma et al. “The Edinburgh human metabolic network reconstruction and its functional analysis”. In: *Molecular systems biology* 3 (2007), p. 135.

[34] R. Agren et al. “Reconstruction of Genome-Scale Active Metabolic Networks for 69 Human Cell Types and 16 Cancer Types Using INIT”. In: *PLoS Computational Biology* 8.5 (May 2012). Ed. by C. D. Maranas, e1002518.

[35] A. Mardinoglu et al. “Genome-scale metabolic modelling of hepatocytes reveals serine deficiency in patients with non-alcoholic fatty liver disease”. In: *Nature Communications* 5.May 2013 (2014), pp. 1–11.

[36] I. Thiele et al. “A community-driven global reconstruction of human metabolism”. In: *Nature Biotechnology* 31.5 (2013), pp. 419–425.

[37] K. Smallbone. “Striking a balance with Recon 2.1”. In: (2013), pp. 14–17.

[38] N. Swainston et al. “Recon 2.2: from reconstruction to model of human metabolism”. In: *Metabolomics* (2016).

[39] E. Brunk et al. “Recon 3D: A resource enabling a three-dimensional view of human metabolism and disease”. In: *Nature biotechnology* ().

[40] I. Thiele. “When metabolism meets physiology: Harvey and Harvetta”. In: (2018).

[41] J. L. Robinson et al. “An atlas of human metabolism”. In: *Science Signaling* 13.624 (Mar. 2020), eaaz1482.

[42] C. Gille et al. “HepatoNet1: A comprehensive metabolic reconstruction of the human hepatocyte for the analysis of liver physiology”. In: *Molecular Systems Biology* 6.411 (2010).

[43] R. Heinrich and S. Schuster. “The regulation of cellular systems”. In: *Book* (1996), p. 416.

[44] S. Klamt and E. D. Gilles. “Minimal cut sets in biochemical reaction networks”. In: *Bioinformatics* 20.2 (2004), pp. 226–234.

[45] J. D. Orth, I. Thiele, and B. O. Ø. Palsson. “What is flux balance analysis?” In: *Nature biotechnology* 28.3 (2010), pp. 245–248.

[46] N. E. Lewis et al. “Omic data from evolved E. coli are consistent with computed optimal growth from genome-scale models”. In: *Molecular systems biology* 6.1 (2010), p. 390.

[47] D. Segre, D. Vitkup, and G. M. Church. “Analysis of optimality in natural and perturbed metabolic networks”. In: *Proceedings of the National Academy of Sciences* 99.23 (2002), pp. 15112–15117.

[48] T. Shlomi, O. O. Berkman, and E. Ruppin. “Regulatory on/off minimization of metabolic flux changes after genetic perturbations.” In: *Proceedings of the National Academy of Sciences of the United States of America* 102.21 (May 2005), pp. 7695–700.

[49] T. Shlomi et al. “Genome-scale metabolic modeling elucidates the role of proliferative adaptation in causing the Warburg effect”. In: *PLoS computational biology* 7.3 (2011), e1002018.

[50] S. Sahoo et al. “Modeling the effects of commonly used drugs on human metabolism”. In: *FEBS Journal* 282.2 (2015), pp. 297–317.

[51] C. H. Schilling et al. “Metabolic pathway analysis: basic concepts and scientific applications in the post-genomic era.” In: *Biotechnology progress* 15.3 (1999), pp. 296–303.

[52] J. M. Rohwer, S. Schuster, and H. V. Westerhoff. “How to recognize monofunctional units in a metabolic system.” In: *Journal of theoretical biology* 179 (1996), pp. 213–228.

[53] T. Pfeiffer et al. “METATOOL: For studying metabolic networks”. In: *Bioinformatics* 15.3 (1999), pp. 251–257.

[54] F. Llaneras and J. Picó. “Which metabolic pathways generate and characterize the flux space? A comparison among elementary modes, extreme pathways and minimal generators”. In: *Journal of Biomedicine and Biotechnology* 2010 (2010).

[55] B. L. Clarke. “Stoichiometric network analysis”. In: *Cell Biophysics* 12.1 (1988), pp. 237–253.

[56] C. H. SCHILLING, D. LETSCHER, and B. Ø. PALSSON. “Theory for the Systemic Definition of Metabolic Pathways and their use in Interpreting Metabolic Function from a Pathway-Oriented Perspective”. In: *Journal of Theoretical Biology* 203.3 (2000), pp. 229–248.

[57] S. Schuster and C. Hilgetag. “On elementary flux modes in biochemical reaction systems at steady state”. In: *Journal of Biological Systems* 02.02 (June 1994), pp. 165–182.

[58] S. Schuster, D. A. Fell, and T. Dandekar. “A general definition of metabolic pathways useful for systematic organization and analysis of complex metabolic networks”. In: *Nature Biotechnology* 18.3 (2000), pp. 326–332.

[59] S. Schuster et al. “Reaction routes in biochemical reaction systems: Algebraic properties, validated calculation procedure and example from nucleotide metabolism”. In: *Journal of Mathematical Biology* 45.2 (2002), pp. 153–181.

[60] C. Kaleta et al. “EFMEvolver: Computing elementary flux modes in genome-scale metabolic networks”. In: *P. 14th German Conf. Bioinformatics* (2009), pp. 180–190.

[61] R. Urbanczik and C. Wagner. “An improved algorithm for stoichiometric network analysis: theory and applications”. In: *Bioinformatics* 21.7 (2005), pp. 1203–1210.

[62] S. Müller and G. Regensburger. “Elementary vectors and conformal sums in polyhedral geometry and their relevance for metabolic pathway analysis”. In: *Frontiers in Genetics* 7.MAY (2016), pp. 1–11.

[63] S. Klamt et al. “From elementary flux modes to elementary flux vectors: Metabolic pathway analysis with arbitrary linear flux constraints”. In: *PLoS Computational Biology* 13.4 (2017), pp. 1–22.

[64] R. Urbanczik. “Enumerating constrained elementary flux vectors of metabolic networks.” In: *IET systems biology* 1.5 (Sept. 2007), pp. 274–9.

[65] O. Hädicke and S. Klamt. “Computing complex metabolic intervention strategies using constrained minimal cut sets”. In: *Metabolic Engineering* 13.2 (2011), pp. 204–213.

[66] R. Mahadevan, A. Von Kamp, and S. Klamt. “Genome-scale strain designs based on regulatory minimal cut sets”. In: *Bioinformatics* 31.17 (2015), pp. 2844–2851.

[67] I. Apaolaza, L. V. Valcarcel, and F. J. Planes. “GMCS: Fast computation of genetic minimal cut sets in large networks”. In: *Bioinformatics* 35.3 (Feb. 2019), pp. 535–537.

[68] I. Apaolaza et al. “An in-silico approach to predict and exploit synthetic lethality in cancer metabolism”. In: *Nature Communications* 8.1 (2017), pp. 1–9.

[69] C. Kaleta et al. “In Silico evidence for gluconeogenesis from fatty acids in humans”. In: *PLoS Computational Biology* 7.7 (2011).

[70] J. Gebauer et al. “Detecting and investigating substrate cycles in a genome-scale human metabolic network.” In: *The FEBS journal* 279.17 (Sept. 2012), pp. 3192–202.

[71] A. Rezola et al. “Selection of human tissue-specific elementary flux modes using gene expression data”. In: *Bioinformatics* 29.16 (2013), pp. 2009–2016.

[72] A. Rezola et al. “In-Silico prediction of key metabolic differences between two non-small cell lung cancer subtypes”. In: *PLoS ONE* 9.8 (2014), pp. 1–7.

[73] J. Gagneur and S. Klamt. “Computation of elementary modes: a unifying framework and the new binary approach.” In: *BMC bioinformatics* 5.1 (2004), p. 175.

[74] M. Terzer and J. Stelling. “Large-scale computation of elementary flux modes with bit pattern trees”. In: *Bioinformatics* 24.19 (2008), pp. 2229–2235.

[75] U.-U. Haus, S. Klamt, and T. Stephen. “Computing Knock-Out Strategies in Metabolic Networks”. In: *Journal of Computational Biology* 15.3 (2008), pp. 259–268.

[76] K. Ballerstein et al. “Minimal cut sets in a metabolic network are elementary modes in a dual network”. In: *Bioinformatics* 28.3 (2012), pp. 381–387.

[77] S. Klamt et al. “FluxAnalyzer: Exploring structure, pathways, and flux distributions in metabolic networks on interactive flux maps”. In: *Bioinformatics* 19.2 (Feb. 2003), pp. 261–269.

[78] R. Schwarz et al. “YANA - A software tool for analyzing flux modes, gene-expression and enzyme activites”. In: *BMC Bioinformatics* 6.1 (June 2005), pp. 1–12.

[79] S. A. Becker et al. “Quantitative prediction of cellular metabolism with constraint-based models: The COBRA Toolbox”. In: *Nature Protocols* 2.3 (Mar. 2007), pp. 727–738.

[80] I. Rocha et al. “OptFlux: an open-source software platform for in silico metabolic engineering”. In: *BMC systems biology* 4.1 (2010), pp. 1–12.

[81] A. Ebrahim et al. “COBRApy: COnstraints-Based Reconstruction and Analysis for Python”. In: *BMC Systems Biology* 7.1 (2013), p. 74.

[82] R. Agren et al. “The RAVEN Toolbox and Its Use for Generating a Genome-scale Metabolic Model for Penicillium chrysogenum”. In: *PLoS Computational Biology* 9.3 (Mar. 2013). Ed. by C. D. Maranas, e1002980.

[83] L. Chindelevitch et al. “An exact arithmetic toolbox for a consistent and reproducible structural analysis of metabolic network models”. In: *Nature Communications* 5 (2014), pp. 1–9.

[84] D. Machado. *ReFramed - First major release*. 2019.

[85] V. Pereira, F. Cruz, and M. Rocha. “MEWpy: a computational strain optimization workbench in Python”. In: *Bioinformatics* (Jan. 2021). btab013.

[86] J. L. Reed. “Shrinking the Metabolic Solution Space Using Experimental Datasets”. In: *PLoS Computational Biology* 8.8 (2012), pp. 1–5.

[87] D. Machado and M. Herrgård. “Systematic Evaluation of Methods for Integration of Transcriptomic Data into Constraint-Based Models of Metabolism”. In: *PLoS Computational Biology* 10.4 (2014).

[88] M. Åkesson, J. Förster, and J. Nielsen. “Integration of gene expression data into genome-scale metabolic models”. In: *Metabolic Engineering* 6.4 (Oct. 2004), pp. 285–293.

[89] C. Colijn et al. “Interpreting expression data with metabolic flux models: Predicting Mycobacterium tuberculosis mycolic acid production”. In: *PLoS Computational Biology* 5.8 (2009).

[90] S. Chandrasekaran and N. D. Price. “Probabilistic integrative modeling of genome-scale metabolic and regulatory networks in <i>Escherichia coli</i> and <i>Mycobacterium tuberculosis</i>”. In: *Proceedings of the National Academy of Sciences* 107.41 (2010), pp. 17845–17850.

[91] P. A. Jensen and J. A. Papin. “Functional integration of a metabolic network model and expression data without arbitrary thresholding”. In: *Bioinformatics* 27.4 (2011), pp. 541–547.

[92] S. A. Becker and B. O. Palsson. “Context-specific metabolic networks are consistent with experiments”. In: *PLoS computational biology* 4.5 (2008), e1000082.

[93] C. J. Lloyd et al. “COBRAme: A computational framework for genome-scale models of metabolism and gene expression”. In: *PLoS Computational Biology* 14.7 (July 2018), e1006302.

[94] S. Robaina Estévez and Z. Nikoloski. “Generalized framework for context-specific metabolic model extraction methods.” In: *Frontiers in plant science* 5.September (Sept. 2014), p. 491.

[95] A. Bordbar et al. “Model-driven multi-omic data analysis elucidates metabolic immunomodulators of macrophage activation”. In: *Molecular Systems Biology* 8 (2012), p. 558.

[96] B. J. Schmidt et al. “GIM3E: Condition-specific models of cellular metabolism developed from metabolomics and expression data”. In: *Bioinformatics* 29.22 (2013), pp. 2900–2908.

[97] T. Shlomi et al. “Network-based prediction of human tissue-specific metabolism”. In: *Nature biotechnology* 26.9 (2008), pp. 1003–1010.

[98] L. Jerby, T. Shlomi, and E. Ruppin. “Computational reconstruction of tissue-specific metabolic models: Application to human liver metabolism”. In: *Molecular Systems Biology* 6.401 (2010), pp. 1–9.

[99] N. Vlassis, M. P. Pacheco, and T. Sauter. “Fast reconstruction of compact context-specific metabolic network models.” In: *PLoS computational biology* 10.1 (Jan. 2014), e1003424.

[100] O. Folger et al. “Predicting selective drug targets in cancer through metabolic networks”. In: *Molecular systems biology* 7 (2011), p. 501.

[101] F. Gatto et al. “Flux balance analysis predicts essential genes in clear cell renal cell carcinoma metabolism”. In: *Scientific Reports* 5 (2015), pp. 1–18.

[102] K. Yizhak et al. “Phenotype-based cell-specific metabolic modeling reveals metabolic liabilities of cancer.” In: *eLife* 3 (Jan. 2014), pp. 1–23.

[103] A. Bordbar et al. “A multi-tissue type genome-scale metabolic network for analysis of whole-body systems physiology.” In: *BMC systems biology* 5.1 (Jan. 2011), p. 180.

[104] Y. Wang, J. a. Eddy, and N. D. Price. “Reconstruction of genome-scale metabolic models for 126 human tissues using mCADRE”. In: *BMC Systems Biology* 6.1 (2012), p. 153.

[105] T. Schlitt and A. Brazma. “Current approaches to gene regulatory network modelling”. In: *BMC Bioinformatics* 8.SUPPL. 6 (2007), pp. 1–22.

[106] G. Karlebach and R. Shamir. “Modelling and analysis of gene regulatory networks”. In: *Nature Reviews Molecular Cell Biology* 9.10 (2008), pp. 770–780.

[107] L. Glass and S. A. Kauffman. “The logical analysis of continuous, non-linear biochemical control networks”. In: *Journal of Theoretical Biology* 39.1 (1973), pp. 103–129.

[108] S. Kauffman et al. “Random Boolean network models and the yeast transcriptional network”. In: *Proceedings of the National Academy of Sciences* 100.25 (2003), pp. 14796–14799.

[109] F. Li et al. “The yeast cell-cycle network is robustly designed”. In: *Proceedings of the National Academy of Sciences* 101.14 (Apr. 2004), pp. 4781–4786.

[110] I. Shmulevich et al. “Probabilistic Boolean Networks: a rule-based uncertainty model for gene regulatory networks.” In: *Bioinformatics (Oxford, England)* 18.2 (2002), pp. 261–274.

[111] I. Gat-Viks, A. Tanay, and R. Shamir. “Modeling and analysis of heterogeneous regulation in biological networks”. In: *Journal of Computational Biology* 11.6 (2004), pp. 1034–1049.

[112] L. J. Steggles et al. “Qualitatively modelling and analysing genetic regulatory networks: A Petri net approach”. In: *Bioinformatics* 23.3 (2007), pp. 336–343.

[113] K. C. Chen. “Integrative Analysis of Cell Cycle Control in Budding Yeast”. In: *Molecular Biology of the Cell* 15.8 (May 2004), pp. 3841–3862.

[114] J. Saez-Rodriguez, A. MacNamara, and S. Cook. “Modeling Signaling Networks to Advance New Cancer Therapies”. In: *Annual Review of Biomedical Engineering* 17.1 (2015), pp. 143–163.

[115] A. Gonzalez, C. Chaouiya, and D. Thieffry. “Logical modelling of the role of the Hh pathway in the patterning of the Drosophila wing disc”. In: *Bioinformatics* 24.16 (Aug. 2008), pp. i234–i240.

[116] S. Klamt et al. “A methodology for the structural and functional analysis of signaling and regulatory networks.” In: *BMC bioinformatics* 7.1 (2006), p. 56.

[117] B. Klinger et al. “Network quantification of EGFR signaling unveils potential for targeted combination therapy”. In: *Molecular Systems Biology* 9.1 (2013).

[118] B. N. Kholodenko. “Cell-signalling dynamics in time and space”. In: *Nature Reviews Molecular Cell Biology* 7.3 (Mar. 2006), pp. 165–176.

[119] G. Moehren et al. “Temperature dependence of the epidermal growth factor receptor signaling network can be accounted for by a kinetic model.” In: *Biochemistry* 41.1 (2002), pp. 306–20.

[120] B. Schoeberl et al. “Therapeutically Targeting ErbB3: A Key Node in Ligand-Induced Activation of the ErbB Receptor-PI3K Axis”. In: *Science Signaling* 2.77 (June 2009), ra31–ra31.

[121] E. Gonçalves et al. “Bridging the layers: towards integration of signal transduction, regulation and metabolism into mathematical models”. In: *Molecular BioSystems* 9.7 (2013), p. 1576.

[122] E. Gonçalves et al. “Post-translational regulation of metabolism in fumarate hydratase deficient cancer cells”. In: *Metabolic Engineering* 45 (2018), pp. 149–157.

[123] M. König, S. Bulik, and H.-G. Holzhütter. “Quantifying the Contribution of the Liver to Glucose Homeostasis: A Detailed Kinetic Model of Human Hepatic Glucose Metabolism”. In: *PLoS Computational Biology* 8.6 (2012), e1002577.

[124] E. Mosca et al. “Computational modeling of the metabolic states regulated by the kinase Akt”. In: *Frontiers in Physiology* 3 NOV.November (2012), pp. 1–26.

[125] M. W. Covert, C. H. Schilling, and B. Palsson. “Regulation of gene expression in flux balance models of metabolism”. In: *Journal of Theoretical Biology* 213.1 (2001), pp. 73–88.

[126] T. Shlomi et al. “A genome-scale computational study of the interplay between transcriptional regulation and metabolism”. In: *Molecular Systems Biology* 3.101 (Apr. 2007).

[127] M. W. Covert et al. “Integrating metabolic, transcriptional regulatory and signal transduction models in Escherichia coli”. In: *Bioinformatics* 24.18 (2008), pp. 2044–2050.

[128] J. Min Lee et al. “Dynamic analysis of integrated signaling, metabolic, and regulatory networks”. In: *PLoS Computational Biology* 4.5 (May 2008). Ed. by C. A. Ouzounis, e1000086.

[129] J. J. R. Karr et al. “A whole-cell computational model predicts phenotype from genotype”. In: *Cell* 150.2 (2012), pp. 389–401.

[130] A. Dugourd and J. Saez-Rodriguez. *Footprint-based functional analysis of multiomic data*. June 2019.

[131] J. M. Drake et al. “Phosphoproteome Integration Reveals Patient-Specific Networks in Prostate Cancer”. In: *Cell* 166.4 (Aug. 2016), pp. 1041–1054.

[132] A. Liu et al. “From expression footprints to causal pathways: Contextualizing large signaling networks with CARNIVAL”. In: *bioRxiv* (June 2019), p. 541888.

[133] K. Jensen, J. G.R. Cardoso, and N. Sonnenschein. “Optlang: An algebraic modeling language for mathematical optimization”. In: *The Journal of Open Source Software* 2.9 (2017), p. 139.

[134] T. pandas development team. *pandas-dev/pandas: Pandas*. Version latest. Feb. 2020.

[135] S. Correia and M. Rocha. “A critical evaluation of methods for the reconstruction of tissue-specific models”. In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 9273. Springer Verlag, 2015, pp. 340–352.

[136] S. Tweedie et al. “Genenames.org: the HGNC and VGNC resources in 2021”. In: *Nucleic Acids Research* 49.D1 (Jan. 2020), pp. D939–D946.

[137] I. Thiele et al. “A community-driven global reconstruction of human metabolism”. In: *Nature Biotechnology* 31.5 (2013), pp. 419–425.

[138] A. Richelle et al. “Increasing consensus of context-specific metabolic models by integrating data-inferred cell functions”. In: *PLoS computational biology* 15.4 (Apr. 2019). Ed. by C. A. Ouzounis, e1006867.

[139] J. Y. Ryu, H. U. Kim, and S. Y. Lee. “Reconstruction of genome-scale human metabolic models using omics data”. In: *Integr. Biol.* 7.8 (2015), pp. 859–868.

[140] A. Richelle, C. Joshi, and N. E. Lewis. “Assessing key decisions for transcriptomic data integration in biochemical networks”. In: *PLOS Computational Biology* 15.7 (July 2019). Ed. by V. Hatzimanikatis, pp. 1–18.

[141] M. Ghandi et al. “Next-generation characterization of the cancer cell line encyclopedia”. In: *Nature* 569.7757 (2019), pp. 503–508.

[142] R. M. Meyers et al. “Computational correction of copy number effect improves specificity of CRISPR–Cas9 essentiality screens in cancer cells”. In: *Nature genetics* 49.12 (2017), pp. 1779–1784.

[143] J. M. Dempster et al. “Extracting biological insights from the project achilles genome-scale CRISPR screens in cancer cell lines”. In: *BioRxiv* (2019), p. 720243.

[144] R. Katzir et al. “The landscape of tiered regulation of breast cancer cell metabolism”. In: *Scientific Reports* 9.1 (2019), pp. 1–12.

[145] D. P. Nusinow et al. “Quantitative Proteomics of the Cancer Cell Line Encyclopedia”. In: *Cell* 180.2 (Jan. 2020), 387–402.e16.

[146] M. N. McCall et al. “The Gene Expression Barcode 3.0: improved data processing and mining tools”. In: *Nucleic Acids Research* 42.D1 (2013), pp. D938–D943.

[147] J. Ferreira et al. “Troppo - A Python Framework for the Reconstruction of Context-Specific Metabolic Models”. In: *Practical Applications of Computational Biology and Bioinformatics, 13th International Conference*. Ed. by F. Fdez-Riverola et al. Cham: Springer International Publishing, 2020, pp. 146–153.

[148] V. Vieira and M. Rocha. “CoBAMP: a Python framework for metabolic pathway analysis in constraint-based models”. In: *Bioinformatics* 35.24 (2019), pp. 5361–5362.

[149] B. DepMap. *DepMap 20Q1 Public*. Feb. 2020.

[150] X. Dai et al. *Breast cancer cell line classification and Its relevance with breast tumor subtyping*. 2017.

[151] M. R. Sebastiano and G. Konstantinidou. *Targeting long chain acyl-coa synthetases for cancer therapy*. Aug. 2019.

[152] A. De Chatterjee et al. “Arachidonic Acid Induces the Migration of MDA-MB-231 Cells by Activating Raft-associated Leukotriene B4 Receptors”. In: *Clinical Cancer Drugs* 5.1 (Apr. 2018), pp. 28–41.

[153] O. Catalina-Rodriguez et al. *The mitochondrial citrate transporter, CIC, is essential for mitochondrial homeostasis*. Tech. rep. 10. 2012, pp. 1220–1235.

[154] A. Dugourd et al. “Causal integration of multi□omics data with prior knowledge to generate mechanistic hypotheses”. In: *Molecular Systems Biology* 17.1 (Jan. 2021), e9730.

[155] D. Machado, M. J. Herrgård, and I. Rocha. “Stoichiometric Representation of Gene–Protein–Reaction Associations Leverages Constraint-Based Analysis from Reaction to Gene-Level Phenotype Prediction”. In: *PLoS Computational Biology* 12.10 (Oct. 2016), e1005140.

[156] M. Masid, M. Ataman, and V. Hatzimanikatis. “Analysis of human metabolism by reducing the complexity of the genome-scale models using redHUMAN”. In: *Nature communications* 11.1 (2020), pp. 1–12.

[157] D. Türei, T. Korcsmáros, and J. Saez-Rodriguez. “OmniPath: guidelines and gateway for literature-curated signaling pathway resources”. In: *Nature Methods* 13.12 (Dec. 2016), pp. 966–967.

[158] I. O. Potapenko et al. *Glycan gene expression signatures in normal and malignant breast tissue; possible role in diagnosis and progression*. Apr. 2010.

[159] M. Uhlen et al. *Towards a knowledge-based Human Protein Atlas*. Dec. 2010.

[160] S. Vuletic et al. “PLTP is present in the nucleus, and its nuclear export is CRM1-dependent”. In: *Biochimica et Biophysica Acta - Molecular Cell Research* 1793.3 (Mar. 2009), pp. 584–591.

[161] C. L. Alves et al. “SNAI2 upregulation is associated with an aggressive phenotype in fulvestrant-resistant breast cancer cells and is an indicator of poor response to endocrine therapy in estrogen receptor-positive metastatic breast cancer”. In: *Breast Cancer Research* 20.1 (June 2018).

[162] C. A. Walsh et al. “The mevalonate precursor enzyme HMGCS1 is a novel marker and key mediator of cancer stem cell enrichment in luminal and basal models of breast cancer”. In: *PLoS ONE* 15.7 (July 2020), e0236187–e0236187.

[163] N. Yoshikawa et al. “Isoprenoid geranylgeranylacetone inhibits human colon cancer cells through induction of apoptosis and cell cycle arrest”. In: *Anti-Cancer Drugs* 21.9 (Oct. 2010), pp. 850–860.

[164] Y. H. Wang et al. “HMGCS2 mediates ketone production and regulates the proliferation and metastasis of hepatocellular carcinoma”. In: *Cancers* 11.12 (Dec. 2019), p. 1876.

[165] L. Luo et al. “BCAT1 decreases the sensitivity of cancer cells to cisplatin by regulating mTOR-mediated autophagy via branched-chain amino acid metabolism”. In: *Cell Death and Disease* 12.2 (Feb. 2021), pp. 1–13.

[166] V. Thewes et al. “The branched-chain amino acid transaminase 1 sustains growth of antiestrogen-resistant and ERα-negative breast cancer”. In: *Oncogene* 36.29 (July 2017), pp. 4124–4134.

[167] J.-W. Song et al. “Omics-driven systems interrogation of metabolic dysregulation in COVID-19 pathogenesis.” In: *Cell Metabolism* 0.0 (June 2020).

[168] S. T. Wu et al. “Esculetin Inhibits Cancer Cell Glycolysis by Binding Tumor PGK2, GPD2, and GPI”. In: *Frontiers in Pharmacology* 11 (Mar. 2020), p. 379.

[169] S. Liu et al. “HADHA overexpression disrupts lipid metabolism and inhibits tumor growth in clear cell renal cell carcinoma”. In: *Experimental Cell Research* 384.1 (Nov. 2019), p. 111558.

[170] Z. Zhao et al. “Prognostic significance of two lipid metabolism enzymes, HADHA and ACAT2, in clear cell renal cell carcinoma”. In: *Tumor Biology* 37.6 (Dec. 2016), pp. 8121–8130.

[171] Y. Xu et al. “Identification of CPT1A as a Prognostic Biomarker and Potential Therapeutic Target for Kidney Renal Clear Cell Carcinoma and Establishment of a Risk Signature of CPT1A-Related Genes”. In: *International Journal of Genomics* 2020 (2020).

[172] H. Xiao et al. “Three novel hub genes and their clinical significance in clear cell renal cell carcinoma”. In: *Journal of Cancer* 10.27 (2019), pp. 6779–6791.

[173] R. Diaz-Ruiz, M. Rigoulet, and A. Devin. “The Warburg and Crabtree effects: On the origin of cancer cell energy metabolism and of yeast glucose repression”. In: *Biochimica et Biophysica Acta (BBA) - Bioenergetics* 1807.6 (2011). Bioenergetics of Cancer, pp. 568–576.

[174] S. M. Umar et al. “Prognostic and therapeutic relevance of phosphofructokinase platelet-type (PFKP) in breast cancer”. In: *Experimental Cell Research* 396.1 (2020), p. 112282.

[175] C. Constantinou et al. “The multi-faceted functioning portrait of LRF/ZBTB7A”. In: *Human genomics* 13.1 (2019), pp. 1–14.

[176] W. J. Israelsen and M. G. Vander Heiden. “Pyruvate kinase: Function, regulation and role in cancer”. In: *Seminars in Cell and Developmental Biology* 43 (2015). Metabolism in cancer cells and Transgenerational environmental and genetic effects, pp. 43–51.

[177] J. Ferreira, V. Vieira, and M. Rocha. “Genome-Scale Metabolic Models”. In: *Systems Medicine*. Ed. by O. Wolkenhauer. Oxford: Academic Press, 2021, pp. 420–428.

[178] V. Vieira, J. Ferreira, and M. Rocha. “A pipeline for the reconstruction and evaluation of context-specific human metabolic models at a large-scale”. In: *bioRxiv* (2021).

[179] V. Vieira et al. *A Model Integration Pipeline for the Improvement of Human Genome-Scale Metabolic Reconstructions*. 2018.

[180] V. Vieira et al. “Comparison of pathway analysis and constraint-based methods for cell factory design”. In: *BMC bioinformatics* 20.1 (2019), pp. 1–15.

I

Supplementary figures

I.1 Supplementary figures for Chapter 5

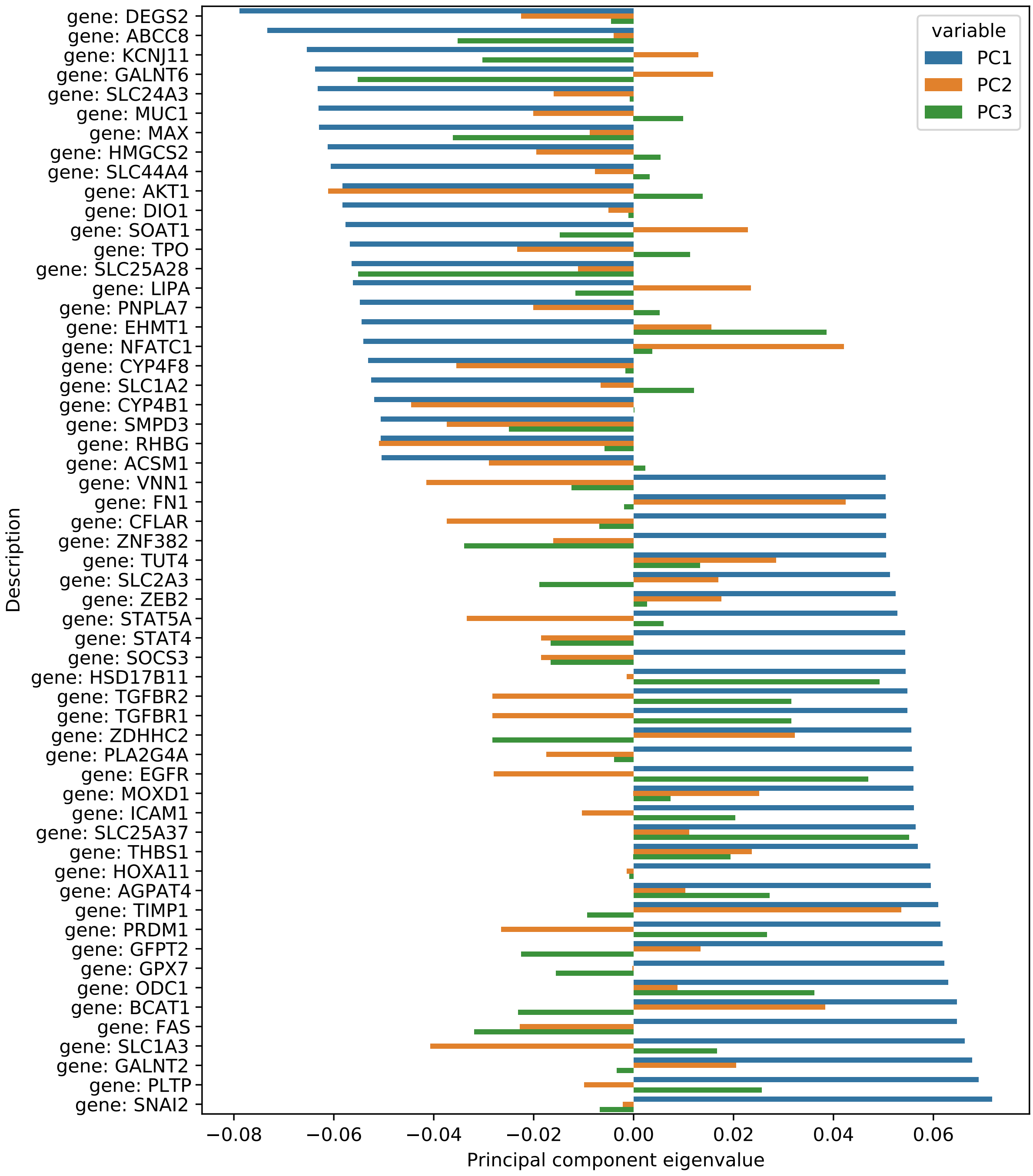


Figure 35: Gene expression reactions in ipFBA with the highest observed eigenvalues after principal component analysis using predicted fluxes for breast cancer cell lines. Each gene contains three bars which represent the eigenvalues for the first three principal components.

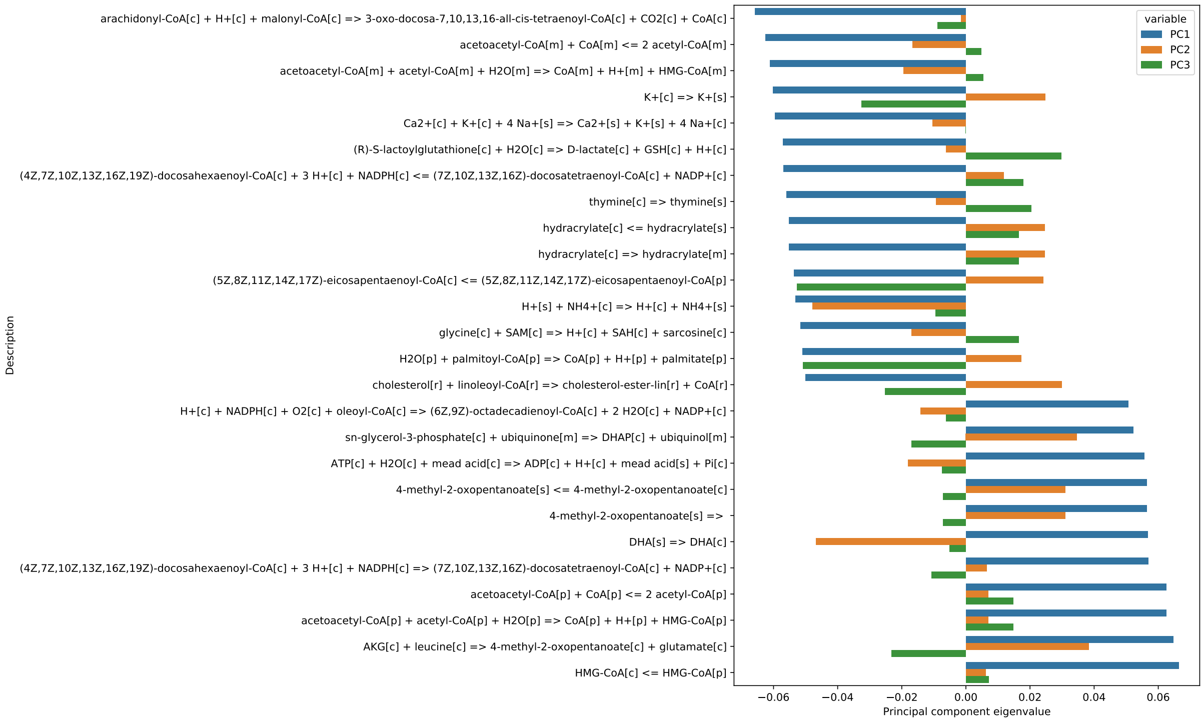


Figure 36: Metabolic reactions in ipFBA with the highest observed eigenvalues after principal component analysis using predicted fluxes for breast cancer cell lines. Each reaction contains three bars which represent the eigenvalues for the first three principal components.

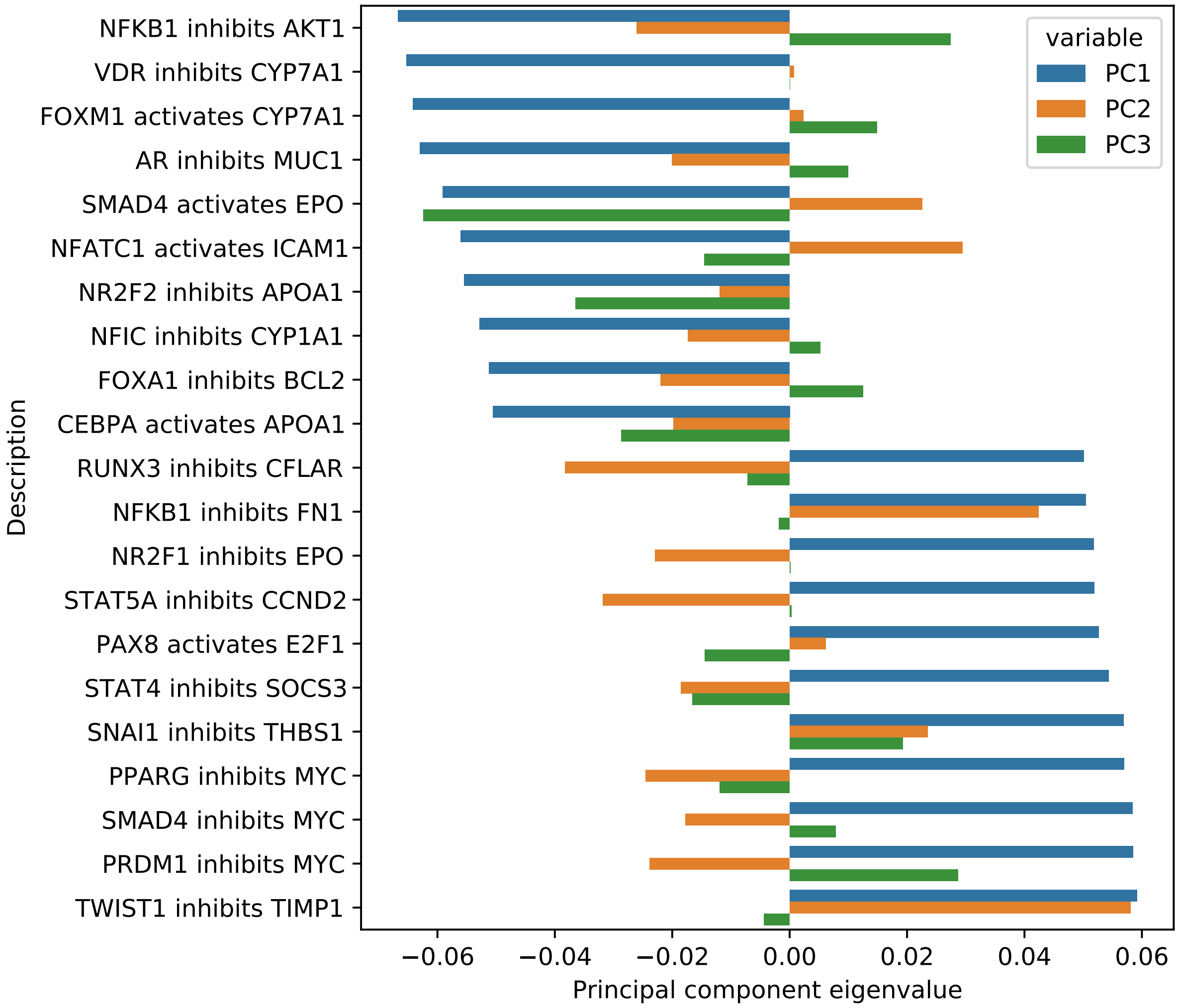


Figure 37: Signalling and regulatory interactions in ipFBA with the highest observed eigenvalues after principal component analysis using predicted fluxes for breast cancer cell lines. Each interactions contains three bars which represent the eigenvalues for the first three principal components.