FROM CONDITION-SPECIFIC INTERACTIONS TOWARDS THE DIFFERENTIAL COMPLEXOME OF PROTEINS

Dissertation
zur Erlangung des Grades
des Doktors der Naturwissenschaften
der Fakultät für Mathematik und Informatik
der Universität des Saarlandes

vorgelegt von
Thorsten Alexander Will

Saarbrücken
2020

Tag des Kolloquiums:	22. September 2020
Dekan der Fakultät:	Prof. Dr. Thomas Schuster
Prüfungsausschuss:	
Vorsitzende:	Prof. Dr. Verena Wolf
Erstgutachter:	Prof. Dr. Volkhard Helms
Zweitgutachterin:	Prof. Dr. Olga Kalinina
Beisitz:	Dr. Alexander Gress

ABSTRACT

While capturing the transcriptomic state of a cell is a comparably simple effort with modern sequencing techniques, mapping protein interactomes and complexomes in a sample-specific manner is currently not feasible on a large scale. To understand crucial biological processes, however, knowledge on the physical interplay between proteins can be more interesting than just their mere expression. In this thesis, we present and demonstrate four software tools that unlock the cellular wiring in a condition-specific manner and promise a deeper understanding of what happens upon cell fate transitions.

PPIXpress allows to exploit the abundance of existing expression data to generate specific interactomes, which can even consider alternative splicing events when protein isoforms can be related to the presence of causative protein domain interactions of an underlying model. As an addition to this work, we developed the convenient differential analysis tool PPICompare to determine rewiring events and their causes within the inferred interaction networks between grouped samples.

Furthermore, we present a new implementation of the combinatorial protein complex prediction algorithm DACO that features a significantly reduced runtime. This improvement facilitates an application of the method for a large number of samples and the resulting sample-specific complexes can ultimately be assessed quantitatively with our novel differential protein complex analysis tool CompleXChange.

ZUSAMMENFASSUNG

Das Transkriptom einer Zelle ist mit modernen Sequenzierungstechniken vergleichsweise einfach zu erfassen. Die Ermittlung von Proteininteraktionen und -komplexen wiederum ist in großem Maßstab derzeit nicht möglich. Um wichtige biologische Prozesse zu verstehen, kann das Zusammenspiel von Proteinen jedoch erheblich interessanter sein als deren reine Expression. In dieser Arbeit stellen wir vier Software-Tools vor, die es ermöglichen solche Interaktionen zustandsbezogen zu betrachten und damit ein tieferes Verständnis darüber versprechen, was in der Zelle bei Veränderungen passiert.

PPIXpress ermöglicht es vorhandene Expressionsdaten zu nutzen, um die aktiven Interaktionen in einem biologischen Kontext zu ermitteln. Wenn Proteinvarianten mit Interaktionen von Proteindomänen in Verbindung gebracht werden können, kann hierbei sogar alternatives Spleißen berücksichtigen werden. Als Ergänzung dazu haben wir das komfortable Differenzialanalyse-Tool PPICompare entwickelt, welches Veränderungen des Interaktoms und deren Ursachen zwischen gruppierten Proben bestimmen kann.

Darüber hinaus stellen wir eine neue Implementierung des Proteinkomplex-Vorhersagealgorithmus DACO vor, die eine deutlich reduzierte Laufzeit aufweist. Diese Verbesserung ermöglicht die Anwendung der Methode auf eine große Anzahl von Proben. Die damit bestimmten probenspezifischen Komplexe können schließlich mit unserem neuartigen Differenzialanalyse-Tool CompleX-Change quantitativ bewertet werden.

Computers are magnificent tools for the realization of our dreams,
but no machine can replace the human spark of
spirit, compassion, love, and understanding.

Louis V. Gerstner, Jr.

ACKNOWLEDGMENTS

In the last almost six years, my life changed considerably in so many regards. I would like to take this opportunity to thank all those people who have supported me in a variety of ways during this journey and my time in Saarbrücken, especially all those who have contributed significantly to making this a great stage of my life, friends and colleagues, and my entire family, whose support and assistance have made an equally invaluable contribution to this era and the work that came with it.

In the first place, I would like to thank my advisor Prof. Dr. Volkhard Helms for his continuous support and mentorship, for giving me fairly free rein with defining and shaping projects, and for always having an open ear and a helpful advice when needed. Your steady enthusiasm and unbroken motivation is unmatched.

I also want to thank Prof. Dr. Olga Kalinina for taking the time to review my thesis and all my collaborators and friends from the SFB 1027 such as Prof. Dr. Ivan Bogeski, Jun.-Prof. Dr. Bianca Schrul and Dr. Björn Becker.

I wish to thank the whole Chair of Computational Biology and also all other members of the Center for Bioinformatics, including all Ph.D., master and bachelor students during my time, and, of course, also the outstanding administrative support by our great secretary Kerstin. Of course, many people deserve to be mentioned individually, like my long-time study and work colleagues that became good friends Kerstin, Jan, the Alexanders, Christoph, Mat, Nick, Michael, Christian, Markus, Florian, Andreas, Daria, Lara, Tim, Siba and Maryam. Thank you very much Kerstin and Markus for also proofreading parts of the thesis and providing valuable input.

Needless to say, I want to thank my family for their unconditional support throughout my studies and during my time working in academia. . . an educational journey that, starting with Biophysics in Kaiserslautern, basically took longer than my complete time in school.

Without doubt, nobody has been more important in this pursuit and, at times, carried a heavier burden than my wife Geza. During my doctoral studies, we experienced the best but also mastered the worst times of our lifes. I am more grateful than ever to have you by my side. At last, please remain the sunshine that you are for ever, Clementine.

I love you two.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

NOMENCLATURE

(N)CM	(non-)classical monocyte
AA	amino acid
AP-MS	affinity purification - mass spectrometry
AS	alternative splicing
BED	Browser Extensible Data
CC	connected component
CD4	naive CD4 T cell
cDNA	complementary DNA
ChIP-seq	chromatin immunoprecipitation with DNA-sequencing
CLP	common lymphoid progenitors
CMP	common myeloid progenitor
CREBBP	CREB-binding protein
CV	cross-validation
DDI(N)	domain-domain interaction (network)
DE	differential expression
DNA	deoxyribonucleic acid
EB	erythroblast
ENCODE	ENCyclopedia Of DNA Elements project
ESEA	Edge Set Enrichment Analysis
FDR	false discovery rate
FIFO	first in - first out
FPKM	fragments per kilobase per million reads
FTP	File Transfer Protocol
GMP	granulocyte monocyte progenitor
GO	Gene Ontology
GPU	Graphics Processing Unit
GRC	Genome Reference Consortium

GSEA	Gene Set Enrichment Analysis
GTEx	Genotype-Tissue Expression project
HC	hierarchical clustering
HGNC	HUGO Gene Nomenclature Committee
HPM	Human Proteome Map
HSC	hematopoetic stem cell
HTTP	Hypertext Transfer Protocol
KEGG	Kyoto Encyclopedia of Genes and Genomes
LCL	lymphoblastoid cell line
LEF1	Lymphoid enhancer-binding factor 1
LP	linear programming
M	monocyte
MEP	megakaryocyte erythrocyte progenitor
MK	megakaryocyte
MPP	multipotent progenitor
mRNA	messenger RNA
MS	mass spectrometry
N	neutrophil
NCBI	National Center for Biotechnology Information
NGS	next-generation sequencing
NMD	nonsense-mediated decay
PDB	Protein Data Bank
Pol III	RNA polymerase III
PPI(N)	protein-protein interaction (network)
RNA	ribonucleic acid acid
RNA-seq	RNA-sequencing
RPKM	reads per kilobase per million reads
SQL	Structured Query Language
TCGA	The Cancer Genome Atlas

TF(C)	transcription factor (complex)
TFBS	transcription factor binding site
TPM	transcripts per million
UCSC	University of California, Santa Cruz
Y2H	Yeast two-hybrid

1 INTRODUCTION

1.1 MOTIVATION

Life is a complex matter. It begins with a single cell that, in a seemingly endless succession of iterations that appear rather chaotic, grows and divides into clusters of cells. A healthy human being comprises between 10^{12} and 10^{16} cells of around 200 cell types with distinct phenotypes at the end of this development, which required a complex interplay of factors in time and space [1, 2]. Both the blueprints for all possible utilities an organism might want to make use of during its lifetime and the complete regulatory information that controls this endeavor are encoded as a sequence of four letters in a double-strand of deoxyribonucleic acid (DNA). Although individual cells differ dramatically in morphology and function, this sequence is the same in almost all cells of an organism [2].

Despite the crucial role of the genomic DNA in storing this information, the differences in morphological and physiological complexity, often quantified as the number of possible gene expression patterns, is barely reflected in the size of the genomes or the number of protein-coding genes when different organisms are compared [3, 4]. Whereas the nematode *Caenorhabditis elegans* has around 19,000-20,000 protein-coding genes, the fruitfly *Drosophila melanogaster* possesses more cell types and tissues with only around 14,000 genes. More so, this is not even three times the number of genes found in the single-celled yeast *Saccharomyces cerevisiae* ($\sim 6,000$ protein-coding genes) [5]. Still, depending on its exact type and life cycle stage, every single cell in a multicellular organism has certain base requirements of proteins that should be expressed and also requires the ability to dynamically adapt to other factors, such as environmental stimuli or physiological needs. Given this discrepancy in expected genome sizes, how can cells in higher eukaryotes govern their potentially large number of very diverse states?

Organisms that are more complex generally possess more regulatory proteins and regulatory sequence regions in their DNA in relation to their genome size [4]. Furthermore, through evolution they acquired and developed additional mechanisms that allow for further means of altering the expression of individual genes in various ways, such as expanding the proteome by being able to distinguish between different isoforms of a protein [6, 7], adjustment of the three-dimensional conformation of the DNA [8, 9] or expanding the encoding of regulatory information by posttranslational modifications of histones [10] or DNA [11]. But the true achievement that ultimately enables the drastic gain in controllable states is the interplay between all those layers and players. Distinct mechanisms of combinatorial control enable cells of higher eukaryotes to leverage their repertoire of machinery in a way that allows for the exponential growth of its regulatory capabilities. A better understanding of this regulatory interplay is crucial to be able to understand developmental processes but also irregularities that cause diseases [3, 4, 8]. For this reason, a paradigm shift is required to move our focus from the classical view of genes and proteins as the units of biological functionality to the collaboration of relevant entities, e.g. the context-specific relationships among the regulatory proteins, instead [8, 12–15].

While several mechanisms are important for the regulation of the specific protein abundance, which is decisive for the phenotype, transcriptional control is the earliest and likely the paramount step in the relevant cascade [2, 3, 16]. Multiprotein complexes comprising transcription factors (TFs), which are DNA-binding regulatory proteins, have emerged as a foundation of signal integration in eukaryotic transcriptional regulation [8, 13, 14]. Their potentially tremendous information content renders them valuable targets of research. First of all, the TFs included in such a complex can indicate cooperative interplay of the regulators, which ideally translates to a much more specific selection of genes that are likely targeted by the complex. Second, further regulatory proteins that are recruited by a TF complex can aid in clarifying the regulatory effect that is exerted by the assembly in a certain context [8, 13, 14, 17].

Since the experimental determination of protein complexomes is tedious and error-prone, TF complexes are an ideal system to be studied by computational methods. Although the prediction of protein complexes from data on protein-protein interactions is considered a well-established problem, integral standard tools that we have in the areas of individual molecules, such as differential analysis pipelines in particular, are not available for the study of protein interactomes and their assemblies. This thesis aims at closing this gap by presenting software tools that cover these quintessential research questions.

1.2 OVERVIEW AND OBJECTIVES OF THIS THESIS

In my master thesis (awarded and published by Springer in their BestMasters series [18]) and in the condensed manuscript ”Identifying transcription factor complexes and their roles” [17] that emerged from the thesis, we followed the idea that by predicting protein complexes that involve TFs, one would be able to gather regulatory information from the knowledge of the exact assemblies that would allow to construct gene regulatory networks at a new level of dealing with TF combinatorics. The results in yeast encouraged us to pursue the topic and work towards an application to human data, where the combinatorial interplay should be even more pronounced. This is not simply a repetition of the same basic workflow. Processing higher eukaryotes correctly is a task that requires more effort and care than for *S. cerevisiae*, for example. Besides the obvious overall increase in scale, new tasks had to be addressed and new opportunities became apparent.

Figure 1.1 presents an overview of all projects covered in this thesis in their respective biological context.

In the case of multicellular organisms, the prediction of protein complexes from protein-protein interaction data requires knowledge on which interactions

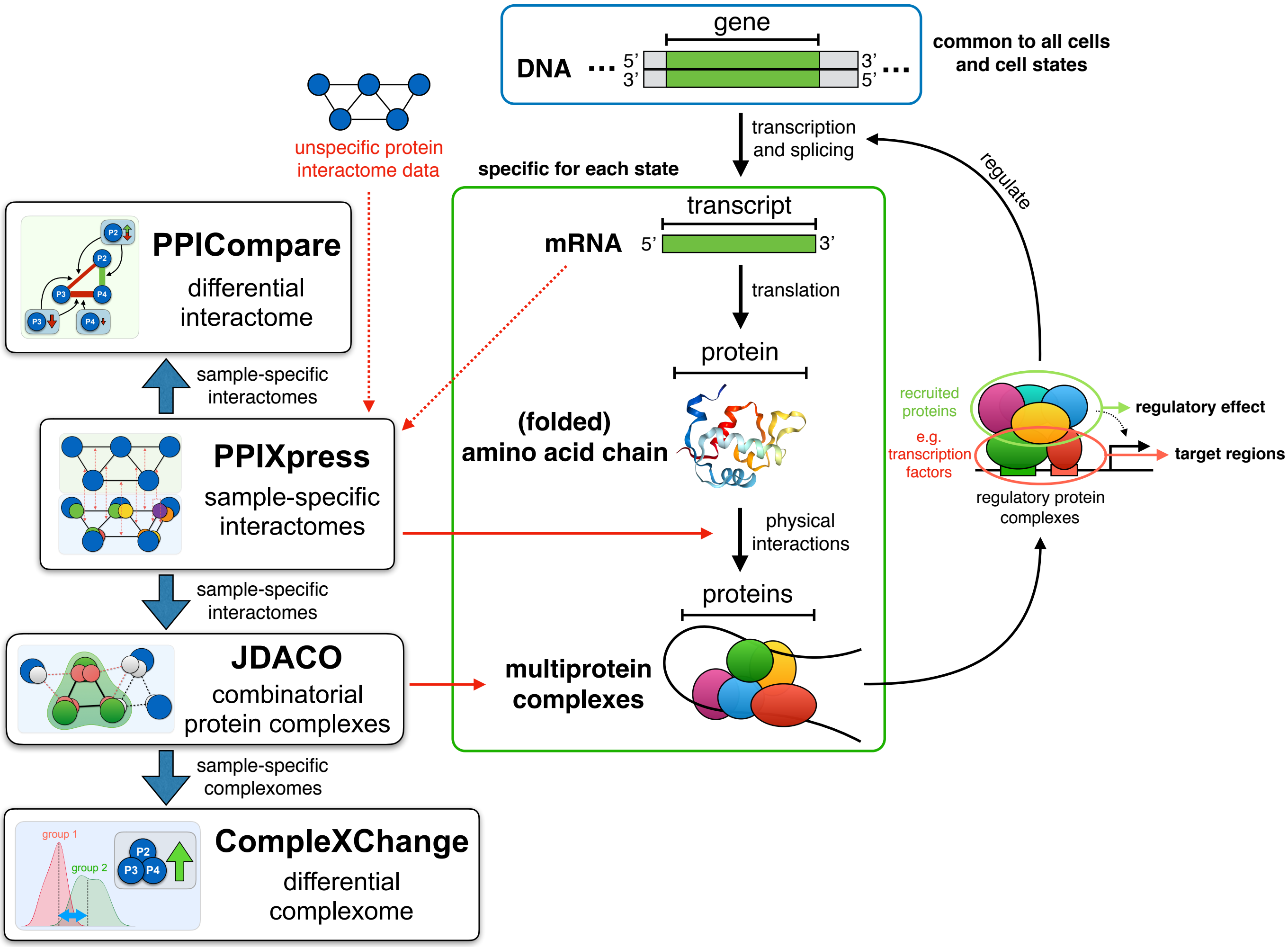

gene
DNA
5'
3'
3'
5'
common to all cells and cell states
unspecific protein interactome data
transcription and splicing
regulate
specific for each state
transcript
mRNA
5'
3'
translation
protein
(folded) amino acid chain
physical interactions
proteins
multiprotein complexes
recruited proteins
e.g. transcription factors
regulatory effect
target regions
regulatory protein complexes
PPICompare
differential interactome
sample-specific interactomes
PPIXpress
sample-specific interactomes
sample-specific interactomes
JDACO
combinatorial protein complexes
sample-specific complexomes
CompleXChange
differential complexome
group 1
group 2
P2
P3
P4

Figure 1.1: *Overview of the projects presented in this dissertation in the biological context. For the software tools, dotted red arrows reference the biological input data whereas the continuous red arrows point to the output of the tools. The folded protein structure was generated from Protein Data Bank (PDB) [19] entry 6Q9O (version 1.1) [20] using the NGL viewer [21].*

are active in a specific cellular context. Although a wealth of experimental data on protein interactions is stored in public databases, the aggregated knowledge therein is not representative for particular living cells. Usually, a contextualization step is conducted on the basis of gene expression data to detect the potentially engaged interactions as those where both partners are expressed in the sample of interest. Building on the idea that interactions between protein domains facilitate protein interactions and under the premise that current sequencing techniques are by default able to yield expression data at transcript resolution, we went a step further than existing approaches and devised a method that employs data integration to make use of this increase in resolution. The resulting tool PPIXpress is thus able to infer protein interactomes that consider the effects of protein isoforms on individual interactions.

With the availability of such specific interactomes, the opportunity became apparent to use the novel information on isoforms in a differential analysis methodology. PPICompare, our application towards this general task, implements a classical differential analysis approach to study the rewiring of isoform-specific networks between groups, and additionally captures what drives these changes, e.g. the deregulation of one or both of the interaction partners which can be caused by either switching the protein-coding gene on/off, or, on the other hand, by switching between two isoforms of the corresponding gene.

Concurrent to all other projects, our complex prediction algorithm DACO was reimplemented in Java. The focus of the new software design concentrated on substantially increasing its performance and scalability to enable the processing of many samples in appropriate time.

Ultimately, all efforts in the thesis culminate in the idea to analyze the complexomes derived from contextualized protein interactomes in a quantitative way. The differential protein complex analysis software CompleXChange is therefore the final part of a complete pipeline from isoform-specific interactomes inferred on the basis of transcript expression data to differential protein complexes. Hence the title of the thesis became ”From condition-specific interactions towards the differential complexome of proteins”.

All tools that I developed during my time as a doctoral candidate at the Center for Bioinformatics were constructed on the basis of a shared framework of classes in Java that I updated steadily. The complete codebase of this framework, including the analyses conducted using Java, is openly accessible in my GitHub repository at https://github.com/edeltoaster/jdaco_dev and all software is distributed under the open-source licence GNU General Public License 3 (GPLv3)[1].

[1] https://www.gnu.org/licenses/gpl-3.0.txt

1.2.1 *First author publications included in this thesis*

Will, T. and Helms, V., **"PPIXpress: construction of condition-specific protein interaction networks based on transcript expression"**, *Bioinformatics*, vol. 32, no. 4, pp. 571, Feb. 2016.

Abstract: Protein-protein interaction networks are an important component of modern systems biology. Yet, comparatively few efforts have been made to tailor their topology to the actual cellular condition being studied. Here, we present a network construction method that exploits expression data at the transcript-level and thus reveals alterations in protein connectivity not only caused by differential gene expression but also by alternative splicing. We achieved this by establishing a direct correspondence between individual protein interactions and underlying domain interactions in a complete but condition-unspecific protein interaction network. This knowledge was then used to infer the condition-specific presence of interactions from the dominant protein isoforms. When we compared contextualized interaction networks of matched normal and tumor samples in breast cancer, our transcript-based construction identified more significant alterations that affected proteins associated with cancerogenesis than a method that only uses gene expression data. The approach is provided as the user-friendly tool PPIXpress which is available at `https://sourceforge.net/projects/ppixpress/`.

Will, T. and Helms, V., **"Rewiring of the inferred protein interactome during blood development studied with the tool PPICompare"**, *BMC Systems Biology*, vol. 11, no. 1, p. 44, Apr. 2017.

Abstract: Differential analysis of cellular conditions is a key approach towards understanding the consequences and driving causes behind biological processes such as developmental transitions or diseases. The progress of whole-genome expression profiling enabled to conveniently capture the state of a cell's transcriptome and to detect the characteristic features that distinguish cells in specific conditions. In contrast, mapping the physical protein interactome for many samples is experimentally infeasible at the moment. For the understanding of the whole system, however, it is equally important how the interactions of proteins are rewired between cellular states. To overcome this deficiency, we recently showed how condition-specific protein interaction networks that even consider alternative splicing can be inferred from transcript expression data. Here, we present the differential network analysis tool PPICompare that was specifically designed for isoform-sensitive protein interaction networks. Besides detecting significant rewiring events between the interactomes of grouped samples, PPICompare infers which alterations to the transcriptome caused each rewiring event and what is the minimal set of alterations necessary to explain all between-group changes. When applied to the development of blood cells, we verified that a reasonable amount of rewiring events were reported by the tool and found that differential gene expression was the major determinant of cellular adjustments to the interactome. Alternative splicing events were consistently necessary in each developmental step to explain all significant alterations and were especially important for rewiring in the context of transcriptional control. Applying PPICompare enabled us to investigate the dynamics of the human protein interactome during developmental transitions. A platform-independent implementation of the tool PPICompare is available at https://sourceforge.net/projects/ppicompare/.

Will, T. and Helms, V., **"Differential analysis of combinatorial protein complexes with CompleXChange"**, *BMC Bioinformatics*, vol. 20, no. 1, p. 300, Jun. 2019.

Abstract: Although a considerable number of proteins operate as multiprotein complexes and not on their own, organism-wide studies so far are only able to quantify individual proteins or protein-coding genes in a condition-specific manner for a sizeable number of samples, but not their assemblies. Consequently, there exist large amounts of transcriptomic data and an increasing amount of data on proteome abundance, but quantitative knowledge on complexomes is missing. This deficiency impedes the applicability of the powerful tool of differential analysis in the realm of macromolecular complexes. Here, we present a pipeline for differential analysis of protein complexes based on predicted or manually assigned complexes and inferred complex abundances, which can be easily applied on a whole-genome scale. We observed for simulated data that results obtained by our complex abundance estimation algorithm were in better agreement with the ground truth and physicochemically more reasonable compared to previous efforts that used linear programming while running in a fraction of the time. The practical usability of the method was assessed in the context of transcription factor complexes in human monocyte and lymphoblastoid samples. We demonstrated that our new method is robust against false-positive detection and reports deregulated complexomes that can only be partially explained by differential analysis of individual protein-coding genes. Furthermore we showed that deregulated complexes identified by the tool potentially harbor significant yet unused information content. CompleXChange allows to analyze deregulation of the protein complexome on a whole-genome scale by integrating a plethora of input data that is already available. A platform-independent Java binary, a user guide with example data and the source code are freely available at https://sourceforge.net/projects/complexchange/.

1.2.2 *Coauthor publications during doctoral studies*

Nazarieh, M., Wiese, A., Will, T., Hamed, M., Helms, V., **"Identification of key player genes in gene regulatory networks"**, *BMC System Biology*, vol. 10, no. 1, p. 88, Sep. 2016.

Abstract: Identifying the gene regulatory networks governing the workings and identity of cells is one of the main challenges in understanding processes such as cellular differentiation, reprogramming or cancerogenesis. One particular challenge is to identify the main drivers and master regulatory genes that control such cell fate transitions. In this work, we reformulate this problem as the optimization problems of computing a Minimum Dominating Set and a Minimum Connected Dominating Set for directed graphs. Both MDS and MCDS are applied to the well-studied gene regulatory networks of the model organisms *E.*

coli and *S. cerevisiae* and to a pluripotency network for mouse embryonic stem cells. The results show that MCDS can capture most of the known key player genes identified so far in the model organisms. Moreover, this method suggests an additional small set of transcription factors as novel key players for governing the cell-specific gene regulatory network which can also be investigated with regard to diseases. To this aim, we investigated the ability of MCDS to define key drivers in breast cancer. The method identified many known drug targets as members of the MDS and MCDS. This paper proposes a new method to identify key player genes in gene regulatory networks. The Java implementation of the heuristic algorithm explained in this paper is available as a Cytoscape plugin at http://apps.cytoscape.org/apps/mcds. The SageMath programs for solving integer linear programming formulations used in the paper are available at https://github.com/maryamNazarieh/KeyRegulatoryGenes and as supplementary material.

My contribution: I improved the original implementation of the heuristic MCDS approach developed by Maryam Nazarieh and implemented the Cytoscape plugin. The complete code of the plugin is available in my GitHub repository at https://github.com/edeltoaster/cyto-MCDS.

Zhang, X., Gibhardt, C.S., Will, T., Stanisz, H., Körbel, C., Cappello, S., Dudek, J., Mitkovski, M., Laschke, M.W., Simmen, T., Schön, M.P., Helms, V., Niemeyer, B.A., Rehling, P., Vultur, A., Bogeski, I., **"Redox signals at the ER-mitochondria interface control melanoma progression"**, *EMBO Journal*, vol. 38, no. 1, p. e100871, Aug. 2019.

Abstract: Reactive oxygen species (ROS) are emerging as important regulators of cancer growth and metastatic spread. However, how cells integrate redox signals to affect cancer progression is not fully understood. Mitochondria are cellular redox hubs, which are highly regulated by interactions with neighboring organelles. Here, we investigated how ROS at the endoplasmic reticulum (ER)-mitochondria interface are generated and translated to affect melanoma outcome. We show that TMX1 and TMX3 oxidoreductases, which promote ER-mitochondria communication, are upregulated in melanoma cells and patient samples. TMX knockdown altered mitochondrial organization, enhanced bioenergetics, and elevated mitochondrial- and NOX4-derived ROS. The TMX-knockdown-induced oxidative stress suppressed melanoma proliferation, migration, and xenograft tumor growth by inhibiting NFAT1. Furthermore, we identified NFAT1-positive and NFAT1-negative melanoma subgroups, wherein NFAT1 expression correlates with melanoma stage and metastatic potential. Integrative bioinformatics revealed that genes coding for mitochondrial- and redox-related proteins are under NFAT1 control and indicated that TMX1, TMX3, and NFAT1 are associated with poor disease outcome. Our study unravels a novel redox-controlled ER-mitochondria-NFAT1 signaling loop that regulates melanoma pathobiology and provides biomarkers indicative of aggressive disease.

My contribution: The study contains several computational analyses that were conducted to support the experimental work in the Bogeski group. All computational contributions in the paper were designed, evaluated and reported by me. Using clinical records, gene expression and mutation data of melanoma patients in TCGA [22], we found that high expression of NFAT1, TMX1 or TMX3 negatively affected patient survival. Special care was taken regarding mutations of BRAF, which are apparent in every second melanoma patient [23, 24]. Furthermore, we performed a targeted enrichment analysis of differential gene expression results from two independent NFAT1 knockdown-studies [25, 26] to assess the role of NFAT1 as a regulator of redox processes and specific hallmarks of cancer. All evaluation code is available in my GitHub repository at `https://github.com/edeltoaster/TMX_NFAT_paper`.

Nazarieh, M., Hamed, M., Spaniol, C., Will, T., Helms, V., **"TFmiR2: Constructing and analyzing disease-, tissue- and process-specific transcription factor and microRNA co-regulatory networks"**, *Bioinformatics*, 2019.

Abstract: TFmiR2 is a freely available web server for constructing and analyzing integrated TF and miRNA coregulatory networks for human and mouse. TFmiR2 generates tissue- and biological process-specific networks for the set of deregulated genes and miRNAs provided by the user. Furthermore, the service can now identify key driver genes and miRNAs in the constructed networks by utilizing the graph theoretical concept of a MCDS. These putative key players as well as the newly implemented 4-node TF-miRNA motifs yield novel insights that may assist in developing new therapeutic approaches.

My contribution: I mainly generated the data that enables to adapt the networks in a tissue-specific manner. For human input data, the expression data by GTEx [27] was consulted to define which genes are likely abundant in the condition of interest and for use cases in mouse, data by ENCODE [28] was used. A smaller contribution by me was, for example, the main figure of the manuscript.

1.3 OUTLINE

Chapter 2 provides an introduction of biological and computational fundamentals relevant to all projects of this thesis.

Each project chapter begins with a “Prerequisites" section that covers the special background that is only of relevance in that particular project and ends with an “Addendum”, in which, when appropriate, I added some retrospective commentary on the work, gave an outlook on the potential future of the project or mentioned new developments in the research field and how that matters to the chapter. In between those parts, the transcript of the projects follows the classical scientific documentation structure of “Introduction”, “Materials and Methods”, “Results and Discussion” and “Conclusion”.

The order of the projects in the thesis was chosen such that all dependencies between projects are covered in prior chapters. Since it is an important basis for all later projects, Chapter 3 starts with PPIXpress, our approach to construct protein-protein interaction networks that are tailored by using transcript expression data and the application of this contextualization method to breast cancer data. Chapter 4 describes the tool PPICompare that builds upon the output of PPIXpress and enables differential analysis of protein-protein interaction networks whereby it also detects transcriptomic alterations that cause the rewiring events. The associated case study discusses modifications of the interactome during blood development. Chapter 5 introduces JDACO, a Java implementation of our domain-aware cohesiveness optimization algorithm DACO. This is the most technical chapter and explains how some small implementational details tremendously improve the execution speed of the algorithm and thus enable to scale the combinatorial protein complex prediction into high-throughput capability. This advancement was crucial for the applicability of CompleXChange, our solution to differential analysis of protein complexes, which is addressed in Chapter 6. Here, we show that CompleXChange outperforms the only comparable approach and report the differential complexome of two human monocyte subtypes.

At last, Chapter 7 ends the main text of the thesis with a conclusion of the research conducted during my studies and an outlook on the future of the projects.

2 BACKGROUND

This chapter serves to communicate the foundation for the projects of this thesis. Section 2.1 covers the biological background knowledge and summarizes relevant computational methods and databases. Section 2.2 then focuses on the theoretical and computational prerequisites for the subsequent chapters.

2.1 BIOLOGY, EXPERIMENTS AND PROCESSING

The biological introduction hereafter will provide an overview of the major biological processes and entities as they are shown in Figure 1.1. It should be noted that the content was selected to offer a focused view on what is relevant for the projects, of course, and the thesis certainly does not claim to provide a complete coverage of the topics. Notably, the regulatory pathways in the overall flow of information from DNA to proteins are intentionally simplified and mechanisms that are not directly relevant were left out.

Protein-coding genes are transcribed and processed into specific transcripts. These subjects of gene expression are covered in Section 2.1.1. Section 2.1.2 then proceeds with the translation of the transcripts into active proteins. The direct interactions of two or more proteins are treated in Section 2.1.3. At last, the special protein family of transcription factors and their role in complexes are the topics of Section 2.1.4.

2.1.1 *On genes and transcripts*

The information about the genome of an organism is stored in coiled double-strands of deoxyribonucleic acid (DNA). The genomic DNA both defines the blueprint of each cell's molecular machinery but also harbors regulatory sites that control the timed construction of this toolset. Each strand of DNA is a long sequence of the four nucleotides adenine, cytosine, guanine and thymine that are covalently bound to deoxyribose and a phosphate group. Cytosine and guanine as well as adenine and thymine can form nucleotide pairs via noncovalent hydrogen bonds and thus enable the characteristic double-helix structure of two complementary antiparallel strands of DNA in the nucleus, mitochondria and chloroplasts of eukaryotic cells [2, 29]. The genome of an organism is organized in one or several chromosomes that store the genetic material wrapped around histone proteins in a tightly packed and highly condensed state called chromatin [30].

Since the deciphering of the human genome sequence by the Human Genome Project [31] and Celera Genomics [32] in the early 2000s, the Genome Reference Consortium (GRC) aims to provide the best possible single consensus representation of the human genome [33]. Although this single reference sequence serves the community well and is tremendously helpful in guiding and thus accelerating the assembly of newly built human genomes, individual genome sequences can deviate significantly from the reference genome in regions of the genome with high genomic diversity. For this reason, alternative solutions that incorporate a broad distribution of idiosyncrasies across populations are discussed theoretically [34]. At the time of writing, the most current version of the human genome was GRCh38.p13 (patch 13 from March 1st, 2019)[1]. The expression data of breast cancer patients used in Chapter 3 were still mapped based on human genome assembly GRCh37 (also called hg19 in UCSC (University of California, Santa Cruz) releases) and thus on an earlier genomic coordinate-reference than the later projects that are all based on the newer major version GRCh38. Since only major versions have compatible coordinate frames, it is crucial to be aware of the exact reference that was used when heterogeneous data sources or data from different datasets are integrated.

Genes are defined segments of the genome which store construction plans that can be assembled into functional biochemical products by the process of gene expression. For protein-coding genes, the first step of gene expression requires the transcription of the protein-coding gene from the corresponding stretch of the DNA into messenger RNA (mRNA), a single-stranded transcript of ribonucleic acid (RNA). After further processing steps (see below) and exporting the mature mRNA from the nucleus to the cytoplasm (in eukaryotes), this transcript is then translated into an amino acid chain by the ribosome and finally folds into a mature protein [2, 35].

Around $35,000$ to $40,000$ human genes that are read out this way are known to date. Of these genes, $20,000$ to $21,000$ are estimated to code for proteins and equally many or slightly less, depending on the exact definitions and sources, encode non-coding genes [28, 36, 37]. The latter define functional non-coding RNAs that include, for example, transfer RNAs and ribosomal RNAs, which are important constituents of elementary cellular processes, but also long non-coding RNAs and a whole pool of small non-coding RNAs, such as microRNAs or small interfering RNAs [28, 38, 39]. In this thesis, we are only interested in the protein-coding portion of the genome.

Figure 2.1 shows the structure of a typical eukaryotic gene and its regulatory control mechanisms. The transcriptional start site (TSS) delineates the starting point of transcription, the dynamic readout of the genetic information that leads to specific cellular states. The TSS position is not defined by a distinct sequence motif but rather by functional motifs in the surrounding region. For phenotype-specific genes, the TATA-box motif (TATA), which is situated about 30bp upstream (in opposing direction to the transcription) of the TSS, is the most notable transcription initiating motif. TATA defines the binding site of the TATA-binding protein which serves the targeted alignment of the transcription preinitiation complex at the TSS. This multiprotein complex comprises the complete protein machinery that is necessary to facilitate transcription [40, 41]. On transcription initialization, the DNA double-strand is opened by complex members and the whole assembly moves in downstream direction ($5' \rightarrow 3'$). In this transcriptional elongation phase, RNA polymerase II builds a single

[1] https://www.ncbi.nlm.nih.gov/grc/human

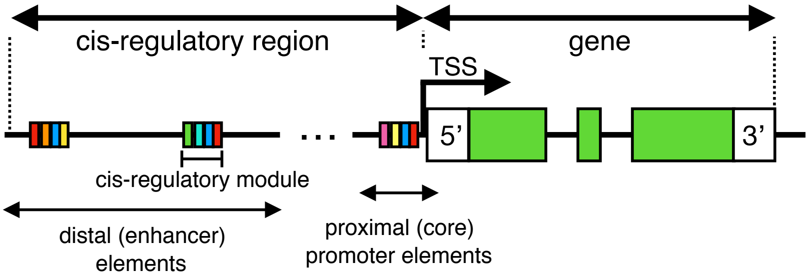


Figure 2.1: *Eukaryotic gene structure. The transcriptional start site (TSS) defines where the transcription from* $5' \rightarrow 3'$ *direction (defined orientation of the sugar backbone chain of the strand that is read) is started. Regulatory sites that control the rate of gene expression are usually positioned directly at the promoter (core elements), right next to the promoter (proximal elements) and more distant to the gene (distal elements, often accumulated in enhancer regions) [41]. Such binding sites for transcription factors are generally clustered into regulatory modules. Genes can be partitioned into untranslated regions (UTRs), which are not translated but serve regulatory purposes, exons (green boxes), which may later be translated into proteins, and introns, which are spliced out during mRNA processing as shown in Figure 2.2 [42]. Adapted and expanded from [14].*

strand of RNA that is exactly complementary to the DNA stretch that is read out by synthesizing the respective pre-mRNA of the gene, with a replacement of thymine by uracile. [2, 40].

The general machinery of transcription is omnipresent in all cells and available for all promoters. By itself, these proteins alone are neither sufficient to yield physiologically relevant levels of gene products, nor are they able to facilitate a state-specific control of the expression of each gene. The regulatory information about when a particular gene should be expressed by the cell is encoded by non-random sequence motifs that define binding sites for transcription factor (TF) proteins. TFs are DNA-binding proteins that, when bound in spatial vicinity to the promoter, have the ability to modulate the strength and speed of initialization and elongation of transcription and can thus heavily influence the rate of transcription [4, 14]. Transcription factor binding sites (TFBSs) are therefore often found near the promoter. However, distal regulatory elements can be far away from the gene in terms of the genomic distance, but can still contribute to the modulation of the target gene by inducing spatial proximity through DNA looping. TFBSs are often clustered together in cis-regulatory modules, sequence regions with a typical length of 100 to 1000bp in which a high density of binding sites is found. Such functional modules are responsible for the signal integration in the cell in the sense that they translate the inputs of many TFs in interplay, but also of cofactors that they recruit to the DNA. This yields an overall signal that a gene should exert [14]. More on TFs and complexes of TFs will follow in Section 2.1.4. There are also other important means of context-dependent transcriptional regulation in the cell, like the packaging of the chromatin in general or specific epigenetic mechanisms, which in this regulatory step include posttranslational modifications of the DNA or histone tails [10, 43]. Since they play no important roles in the projects primarily discussed in this thesis, these factors are not introduced in detail.

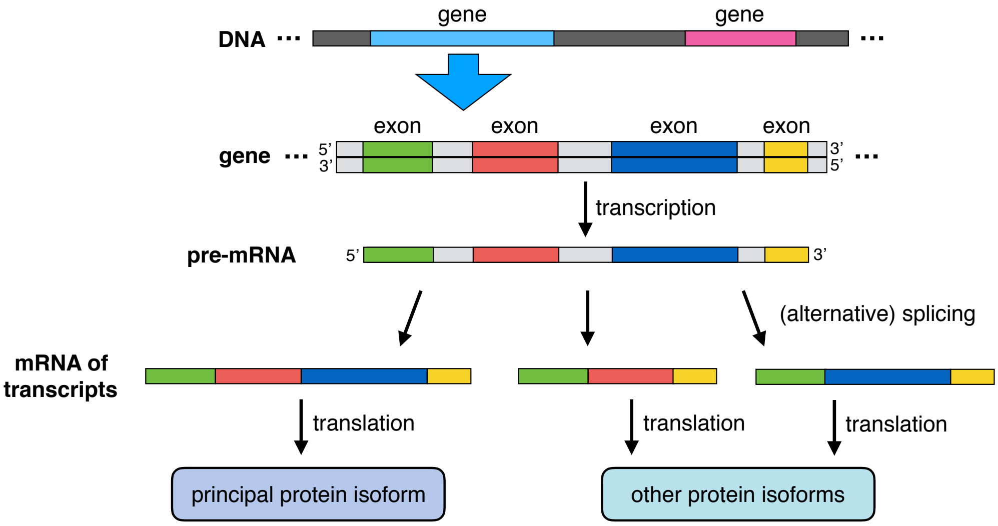


Figure 2.2: *From gene to protein variants. Genes are segments of the DNA that are partitioned into intros and exons. After the transcription to pre-mRNA, the single-stranded RNA can be processed into various mature mRNAs that differ in their exon composition. Their translation can thus yield a diverse set of protein isoforms although they were defined by the very same gene.*

Most genes consist of exon regions that code for protein sequences as well as of non-coding regions called introns (see Figures 2.1 and 2.2). As the first intermediate product of transcription, the pre-mRNA of a gene comprises all its introns and exons. In the next step, this RNA is processed to the mature mRNA by splicing out the introns and conducting other modifications that are not of further interest here, such as the polyadenylation of the 3′-ends of genes, or RNA editing. Figure 2.2 shows this posttranscriptional process in the overall context of transcription and translation.

Astonishingly, eukaryotes are able to synthesize alternative versions of transcripts from the full-length gene by changing the composition of exons that are included in the mRNA and, subsequently, are able to generate various protein isoforms from the very same definition of the gene [2, 42]. In human, 90-95% of multi-exon genes are subject to such alternative splicing (AS) events [42, 44]. The choice of the transcript composition, and therefore the protein isoforms that are translated, has considerable impact on the protein interactome [7, 45–47], important implications in development [48–50] and, when something goes wrong, adverse effects on health [51–54].

However, not all transcripts that are transcribed will become viable protein products. Control mechanisms of the cell in that regard are, for example, the modulation of the cellular localization of the product, targeted degradation by microRNA-associated processes, or surveillance mechanisms such as nonsense-mediated mRNA decay and non-stop decay [2, 42].

Measuring the expression of genes and transcripts

Whereas the genome defines the repertoire of the cell in terms of which protein-coding genes or variants thereof it could possibly produce, gene expression profiling tells us which genes or transcripts are currently expressed in a cellular sample and in which amounts. This snapshot of expressed mRNA is called the transcriptome.

The history of methods to determine which genes are expressed in a sample goes back to the 1970s when mRNA was separated by gel electrophoresis, transferred to a membrane and finally visualized using labeled probes by Northern blotting [55]. The appearance of reverse transcription and the polymerase chain reaction paved the way for new methods in the beginning of the 1990s. By reverse transcription of mRNA into complementary DNA (cDNA), the cDNA corresponding to transcripts could be utilized as a template for the exponential amplification using polymerase chain reaction. The amount of specific transcripts could then be measured by adding labeled sequence-specific DNA probes, for example [56].

In the mid 90s, the multiplexity of expression profiling, e.g. the ability to measure the expression of many genes or transcripts with the same experiment, increased significantly with the introduction and widespread adoption of new techniques which for the first time allowed to assess the abundances of hundreds and even thousands of target mRNAs in parallel. Besides serial analysis of gene expression [57], whereby the occurrences of sufficiently descriptive sequence tags are recognized, DNA-microarrays, or short microarrays, have been the most prominent experimental approach for gene expression profiling for quite a time and are still frequently used today [58]. In microarray analyses, the mRNA is reverse transcribed to cDNA whereby the nucleotides that are incorporated are labeled with a fluorescent dye. The labeled cDNA is then placed on the array to enable the hybridization to a defined set of complementary probes of nucleotide oligomers that were attached to the array by the manufacturer. The abundance of a particular cDNA in the probe can then be measured optically by its fluorescence intensity. By using dyes of different colors, classically red and green, it is even possible to gather differential information on two states in one experiment by cohybridization of the distinctly labeled samples to the same microarray [58].

The latest leap took place when next-generation sequencing (NGS) appeared and gained momentum in the mid 2000s. Previously, there were established lower throughput protocols such as Sanger sequencing [59], which worked by introducing labeled dideoxynucleotides that terminated the chain extension of DNA because the hydroxyl group at the $3'$ position is absent. Due to the significantly increased speed and cost effectiveness that NGS methods achieved by their individual highly parallel implementations of base determination, they gradually replaced earlier methods when the new NGS machines became commercially available [60, 61]. The first setup that was released to the market was the 454 technology by Roche Applied Sciences [62], followed by SOLiD [63] and Illumina (formerly Solexa) [64]. While millions of reads are determined per NGS run, a downside of the new methods were the rather short read lengths. Newer approaches negate this handycap [60] and, more recently, even applications of NGS with tiny amounts of input material, e.g. as obtained in single cells, became feasible [65].

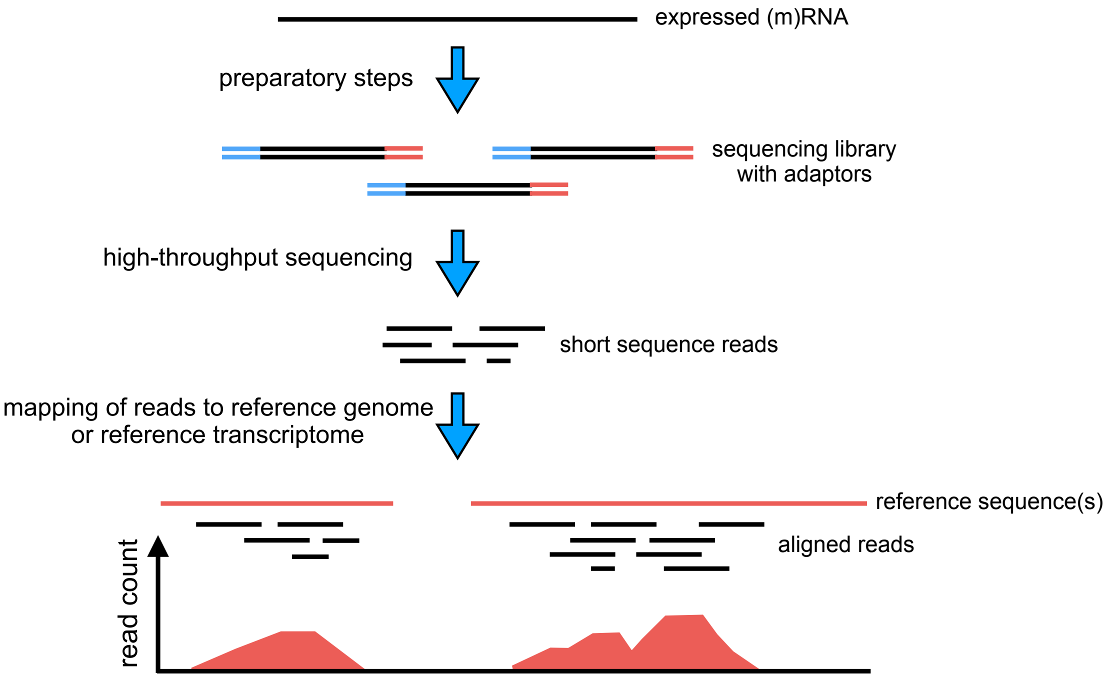


Figure 2.3: *RNA-sequencing in a nutshell. Suitable sequence libraries of cDNA fragments with adaptors are constructed from the isolated (m)RNA and their sequences are subsequently determined as read data by applying next-generation sequencing techniques. The reads are finally mapped to either a reference genome or a collection of reference gene or transcript sequences to obtain counts that can be associated with corresponding genes or transcripts, respectively.*

RNA-sequencing (RNA-seq) [66] utilizes NGS and is currently the best practice to determine transcriptomes because it alleviates shortcomings of microarrays by allowing quantification with much higher sensitivity and dynamic range. Furthermore, unlike any method that uses hybridization of the cDNA, RNA-seq is not limited to a defined set of complementary probes of interest but can be conducted without any prior knowledge on the exact transcript sequences that should be measured. Thus, RNA-seq is also able to report novel splice variants and mutations in the sequence [67].

Figure 2.3 sketches the basic approach of RNA-seq. After the (m)RNA of the sample is isolated, sequencing libraries of fragmented cDNA with machine-specific adaptors are constructed that, among other parameters, differ in terms of the transcript enrichment that is performed. Depending on the focus of the experiment, classes of RNA can be enriched or depleted deliberately. If one is only interested in protein-coding transcripts, the original RNA can be enriched with mRNA by filtering for polyadenylated tails, for example. If microRNAs are the target of choice, filtering by size is conducted [67]. NGS is then used to generate reads of the thus prepared fragments. The protocols allow to sequence transcripts either from one direction, yielding single-end reads, or from both sides, which corresponds to paired-end reads. The latter is advantageous for tasks like the detection of unknown splice variants and generally leads to a more robust alignment, especially in genomic regions that are harder to map [67].

To associate NGS reads with the expression of a particular transcript, classically all reads are mapped to an appropriate reference genome of the organism [68]. Then, reference transcript annotation data can be employed to relate mapped reads to actual transcripts and to count the occurrences of mapping events per transcript. Those counts can then be used to quantify the abundances of each transcript, whereas gene abundances are derived by summarization of the counts of all associated transcripts. Popular sources for reference transcriptomes, which are often not limited to protein-coding genes but can in principle also cover other functional RNAs, are Ensembl [69], RefSeq [70] or GENCODE [71]. The choice of the annotation data influences the result to a certain degree [72] and, as mentioned earlier in the discussion on the number of human genes and transcripts, one should be aware that we certainly still lack a complete coverage of the transcriptome [37]. A general obstacle when reads are aligned to the DNA sequence is that their mapping is often ambiguous in the sense that reads may often be aligned to more than one transcript. Sophisticated quantification tools such as RSEM [73] model this uncertainty of the read mapping statistically and resolve it by, for example, employing expectation maximization to adjust the numerical values accordingly. A newer generation of RNA-seq quantification methods, kallisto [74] and Salmon [75], follow a significantly faster approach by using pseudoalignments. Instead of aligning the reads to the complete genome, which is usually the most time-consuming step in a quantification pipeline, the reads are directly mapped to the defined transcript sequences from reference annotations. They also sometimes allow for bootstrapping of the expectation maximization approach mentioned previously and thus even the technical variance in the optimization procedure that untangles the uncertainty of read mapping can be assessed [74].

Depending on the intended usage, the raw read counts are often converted into better suitable expression measures to normalize the values within samples with respect to, for example, large differences in the amount of reads generated by sequencing runs and other technical variances. One approach would be to divide the counts per transcript by the number of mapped reads. Because longer transcripts are more likely to produce more reads, the value is additionally corrected by the length of the annotated region of interest. This measurement unit is known as reads per kilobase (of exon) per million mapped reads (RPKM) [66]. The equivalent for paired-end read data is called fragments per kilobase (of exon) per million mapped reads (FPKM) [76]. A disadvantage of RPKM and FPKM is that the expression values of especially lowly expressed transcripts are often highly influenced by a small share of transcripts that account for a large fraction of the mapped reads. Thus, the RPKM/FPKM values of a transcript may differ significantly between samples even if the transcript had the same frequency in the samples' respective pools of RNA. By making each expression value dependent on the values of all other transcripts, the unit transcripts per million (TPM) represents a more robust alternative to gauge the relative expression in a sample [77, 78]. It can be derived by rescaling the values with respect to the total expression units in the measurement. Given the FPKMs of all transcripts, the TPM of a transcript $\mathfrak{i}$ is then

$$\text{TPM}_i = \frac{\text{FPKM}_i}{\sum\limits_{\forall \text{ transcripts } j} \text{FPKM}_j} * 10^6$$

whereby a scaling factor is added to relax numerical problems.

A prime application of expression data is to investigate the changes in the transcriptome between two cellular states by differential expression (DE) analysis. To allow for such a statistical assessment of sufficient power with regard to expected sample sizes, it is generally worthwhile or even necessary to model the problem appropriately based on assumptions of the distributions of relevant entities. Basic knowledge on such statistics will be introduced in Section 2.2.1. If reads are considered to be independently sampled from a fixed set of genes in RNA-seq, read counts can be modelled by a Poisson distribution. The Poisson distribution, however, only has a single parameter that determines its mean and variance. Since the variance is generally larger than the mean expression values, the related negative binomial distribution is often used instead. This distribution has the appealing property that mean and variance can be related by a dispersion factor which is then approximated instead of both individual parameters [79, 80]. Estimating both mean and variance reliably for each gene would require much larger sample sizes.

Although plenty of well-established solutions are already available, there is still much new work done on the topic of DE methodology [81, 82]. As an example, sleuth [83] is a recent DE tool that is able to make use of the faster nature of pseudoalignments and bootstrapping capabilities of modern quantification tools to additionally introduce technical variance of the quantification confidence into the assessment.

Popular public datasets on expression data

All my method development efforts used RNA-seq data that was available to the public. When it comes to gene expression data in general, and RNA-seq data in particular, there are plenty of services that provide experimental data in preprocessed or raw formats.

Public storage services that allow the upload or download of RNA-seq results are, for example, hosted by the National Center for Biotechnology Information (NCBI) which allows to store and gather expression data at the NCBI Gene Expression Omnibus [84] and hosts another database for NGS-based data with the NCBI Sequence Read Archive [85]. Another popular service in that regard is ArrayExpress [86], which is operated by the European Bioinformatics Institute. We gathered the datasets that we utilized in the evaluation and application of CompleXChange [87] from such storage services (see also Chapter 6).

Other important sources of public RNA-seq data that I frequently used in my work were the efforts by huge consortia. There, mostly a clearly defined main topic is considered for which many different experimental measurements besides expression profiling were conducted by the laboratories involved. The

ENCyclopedia Of DNA Elements (ENCODE) project [28] is historically the follow-up to the Human Genome Project and focused on the deciphering of regulatory elements in the DNA and their effect on the transcriptome. The Cancer Genome Atlas (TCGA) [88] is a recently discontinued and very popular knowledgebase with a plethora of data on many cancer types. BLUEPRINT [89] is a European project focused on hematopoietic cells. Furthermore, the Genotype-Tissue Expression (GTEx) project [27] provides gene expression data for ten thousands of samples, often even together with the genotype of the donor.

TCGA expression data from breast cancer patients was used in the evaluation of PPIXpress [90] (see also Chapter 3) and various data on melanoma, namely expression, mutation and survival data, were useful in one of our collaborative projects [91]. The broad data on blood cell types from BLUEPRINT allowed us to study developmental transitions in PPICompare [92] (see also Chapter 4). While I did not publish a project that used human data by ENCODE, I regularly benefited from the huge dataset, e.g. in my attempt to define TF complexes that are important in pluripotency that I briefly describe in Section 6.6.3. Also, human GTEx and ENCODE mouse data were used to define tissue-specific expression in TFmiR2 [93].

2.1.2 *On proteins*

Besides DNA and RNA, proteins are the most important biological molecules. Alongside many other capabilities of proteins, they can act as enzymes and thus catalyze biochemical reactions, they give the cell its structure, manage its energy storage, or enable signal transduction and the integration of such signals for regulatory purposes. Prime examples of tasks from the previous paragraphs that are completed by proteins are the transcription of DNA, the regulation of this process and all further processing steps towards mature mRNA. The entirety of proteins in a cell during a certain condition is called the proteome [2].

Proteins are synthesized by translation of mRNA into a chain of amino acids (AAs) that subsequently folds into a specific three-dimensional structure and may also contain disordered portions. In this fundamental biochemical process that takes place at the ribosomes, the four different nucleotides given in an mRNA template are read out in triplets and converted into the 20 natural AA residues according to rules that are known as the genetic code [2].

AAs consist of a common backbone which is used to connect the individual units and additionally comprise variable side chains that are specific to each AA. This diverse set of potential building blocks enables to include residues that feature differing physicochemical properties with regard to their charge, hydrophobicity, size and the functional groups that they include. The distribution of AAs therefore has strong implications for the structure of the protein. Globular proteins that are soluble in water generally have hydrophilic AAs on their surface and hydrophobic residues buried within their core. The membrane-spanning portions of integral membrane proteins, on the other hand, are fairly hydrophobic on the outside of the protein that is facing the lipid acyl chains of bilayer membranes. This manifold of possibilities is certainly a driver that facilitates proteins to come in so many forms and functions to ultimately be the cell's jacks of all trades [94].

The AA sequence is considered the primary structure of proteins. By folding of the protein in its natural condition, structural motifs such as α-helices or β-sheets are formed, which are primarily stabilized by hydrogen bonds. These folding patterns describe local substructures that are attributed to the secondary structure of the protein. In an example on higher-order structures in Figure 2.4, the coil-like α-helices are shown magenta colored and the antiparallel strands of β-sheets are highlighted in yellow. The ternary structure then concerns the overall shape of the protein that follows from specific combinations of secondary structures in the specific context. Important functional units of several structurally conserved secondary structure elements therein are called protein domains and are introduced in the next subsection. Finally, the quaternary structure of a protein means the stable topology that is formed by the permanent aggregation of two protein subunits that are encoded by distinct amino acid chains [94]. Some proteins, however, completely or partially lack a well-defined three-dimensional structure. This flexibility of disordered protein regions can, for example, allow for a much broader spectrum of interaction partners by dynamic adjustments of binding interfaces [95, 96].

Although the amino acid sequence of a protein is fixed after translation, the cell still has a certain amount of control over each protein by covalent modification of amino acid residues. This biochemical mechanism allows to dynamically alter properties of amino acids to posttranslationally adapt the activity or structure of a protein by, for example, phosphorylation of a specific residue [2, 97].

In the regulation of gene expression, the posttranslational modification of histone tails is a crucial control circuit. As mentioned earlier, histones are important for the packaging of the DNA. When certain lysine residues in the N-terminal tail of an histone are acetylated, the lysines' positive charges are neutralized which weakens the strength of the interaction with the negatively charged DNA. This simple covalent switch is consequently able to directly influence the packaging of the DNA [10].

On protein domains

Although proteins come in many sizes and in an elusive number of shapes, a comparably small amount of regular folding patterns is constantly recurring in very different proteins and across species. Especially functionally important parts of proteins are often found as such evolutionary conserved modules. These so-called protein domains are units of protein organization that describe independently folding stable substructures of typically 40 to 350 amino acids in length [2, 98]. Figure 2.4 shows an example of a domain that is part of two different proteins from different organisms.

Small proteins with a particular function frequently contain exactly one domain that facilitates this activity. In larger proteins several protein domains may work together to achieve the intended task. Due to this very defined

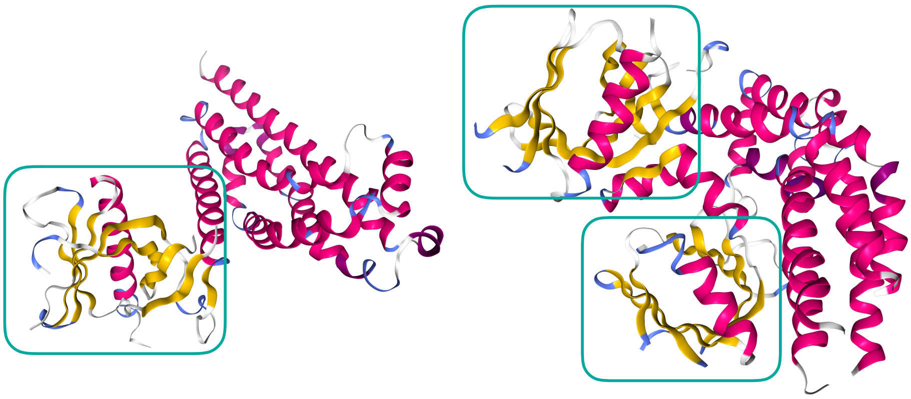


Figure 2.4: *Example of a protein domain. The Pleckstrin homology domain or PH domain (Pfam accession PF00169) is frequently found in proteins with signaling purposes and structurally characterized by two perpendicular antiparallel β-sheets which are followed by an α-helix. The length of the connecting loops in between is variable [104]. A fragment of the mouse protein Dbs (PDB entry 1RJ2, version 1.2) is shown on the left side and the human FARP1 (PDB entry 4H6Y, version 1.3) is presented on the right side. Both structures exhibit the PH domain (occurrences highlighted by turquoise rectangles), once in 1RJ2 and twice in 4H6Y. PDB structures were visualized using [21] and colored by their secondary structures. α-helices are shown in magenta color and β-sheets are shown in yellow.*

functionality, it is not surprising that most evolutionary gene duplication and recombination events cover DNA segments that approximately match protein domains. Consequently, protein domains are also important building blocks in protein evolution [98].

While this is not a generally applicable truth, protein domains also show an overall tendency to align with exon boundaries [99]. Splicing out an exon can therefore often control if a protein domain is included in the final protein isoform or not. This premise motivated the fundamental idea behind the method PPIXpress [90] (see also Chapter 3).

Although there is a clear consensus that protein domains are the basic structural and evolutionary building blocks of proteins, there are many ways to exactly define and categorize domains as well as to judge their presence in proteins. Interestingly, the classical approaches that originated in the 90s are still the most important resources to date [100]. The methods can be broadly classified into structure- and sequence-based approaches and by their level of manual intervention: SCOP (Structural Classification Of Proteins) [101], for example, is completely based on structural data and highly curated while the equally established structure-based service CATH (standing for its internal classification system into Class, Architecture, Topology, Homology) [102] involves a much higher degree of automation. In contrast, other popular approaches like Pfam [103] solely apply sequence profiling and do not directly employ structural knowledge in their models.

Basically all projects in this thesis at some point integrate data on protein domains. The framework that is the basis for my work uses the Pfam annotation in that regard, because it is well-established and the prevailing standard with respect to naming domain types in domain-domain interaction data [105–108]. It thus makes sense to internally label and describe domains solely on the basis of their Pfam accession. More specifically, the Pfam-A database is used because it is of high confidence and has permanent accessions.

Pfam-A domain entries are made from manually curated representative seed protein sequences from which a multiple sequence alignment is created. This initial alignment serves as the template to train a hidden Markov model that is subsequently used to find hit candidates in, for example, UniProt [109]. Sequence regions that scored sufficiently well in this screening step according to conservative family-specific detection thresholds are then added to the set of relevant sequences from which the final alignment and the probabilistic model are derived [103, 110].

With version 27.0, already 90% of human proteins included at least one domain annotated by Pfam [111]. Pfam received its latest update to version 32.0 in Sept. 2018 and currently contains data on 17, 929 domain families with corresponding multiple sequence alignments and hidden Markov models [112].

Measuring the abundance of proteins

Proteome-wide quantitative measurements of protein abundances are a much more challenging experimental effort than the large-scale assessment of gene or transcript expression. In proteomics, which stands for the analysis of proteomes, a detection method requires a very high sensitivity because the input material cannot be amplified such as DNA or RNA. Furthermore, amino acids have much more diverse physicochemical properties compared to oligonucleotides [113, 114]. Still, proteins are often the relevant biomolecules of interest and not their corresponding mRNA. Thus in the best case, since a whole mesh of dependencies in the regulatory sense but also degradation processes are missed in between an mRNA and its final protein product, the protein itself should be measured directly if possible. Furthermore, only proteomics allows to determine facets such as the posttranslational modifications of proteins.

As in the case of early approaches to gene expression, the history of analytical techniques to detect proteins also started in the 1970s with gel electrophoresis and Western blotting [115, 116]. Specific proteins could then be selected by, for example, specific antibodies.

A notable breakthrough in the field took place when mass spectrometry (MS) became applicable in proteomics. The most notable technical advantages towards this goal were certainly the development of proper protein ionization techniques in the 80s and 90s that made proteins amendable to MS. Interestingly, the advent of the genomic era and the implicated storage of known sequences was also a necessity for the application in practice. Without such databases it would not have been possible to match detected peptide fragments in MS spectra to data of known protein sequences [117].

Because the general workflow is important for plenty of protein-centric analyses, the procedure of a generic MS-based proteomics experiment will be outlined briefly following the best practices according to popular overview articles [114, 117, 118]. After the extraction of the protein material from the sample, a typical proteomics experiment starts with the digestion of the proteins into short peptides. Depending on the exact purpose of the study, this pool of peptides (or already the undigested full proteins) can be prefractionated or enriched regarding attributes or features, e.g. by their mass, posttranslational modifications, specific antibodies or interaction partners (see, for example, affinity purification in the following Section 2.1.3). The remaining peptides are then separated by one or more steps of liquid chromatography such that they elude one by one from the columns. Each peptide is then ionized and sprayed into the mass spectrometer and one or two MS measurements take place. The latter protocol is called tandem MS or MS/MS. A mass spectrometer measures the mass-to-charge ratio $\frac{m}{z}$ of peptide ions. In tandem MS, these ions are additionally fragmented and all parts measured again by a second run of MS. The overall $\frac{m}{z}$ values and measured patterns of fragments finally allow to confidently identify the peptides that are found in a sample. On the basis of widely available databases on protein sequences, peptides can then be related to proteins.

Label-free absolute protein quantification is then possible by summarizing the ion counts measured for each protein and subsequent scaling the values on the basis of, for example, spiked-in reference peptides for which the concentration was known [118]. Label-based quantification techniques, like SILAC (stable isotope labeling by amino acids in cell culture) [119], are alternative approaches that are more robust to technical variability and sample handling than absolute quantification. Here, stable isotopes are used to distinguish between two samples that are analyzed together whereby the origin of their peptide ions can then be discerned due to their distinctive masses. By doing this, exact abundance ratios can be determined for proteins [114].

Since measuring the complete transcriptome quantitatively is much simpler than to measure the proteome of a sample, mRNA levels are commonly used to approximate protein abundances in computational and systems biology. While mRNA expression was found to certainly have a crucial role in the majority of dynamics found for protein levels and the correlation between the measures was often strong, the experimental insights so far also showed that the transcriptome alone was not sufficient to explain all changes in protein abundance [120–124]. Furthermore, the strength of the relation seems to be highly dependent on the system and cell types investigated [125, 126].

Popular public datasets on protein abundance data

As just mentioned, due to the comparably high effort involved in the experimental determination of sample-specific protein abundances, there is currently much less data available on proteomes than on transcriptomes and sample sizes of datasets are generally also considerably smaller.

Popular web services and consortia often only host the results of a single study with one sample per tissue or cell type. The Human Proteome Map [127], for example, provides the results on 30 healthy human tissues and cell type samples evaluated using MS. The Human Protein Atlas [128] profiled the proteins found in 44 human cell types using protein-specific antibodies whereby transcriptomics experiments were additionally conducted for most of the samples. ProteomicsDB [129], on the other hand, is a service that stores the results of 16, 857 MS-based proteomics experiments compiled from public data and in-house experiments in a centralized but protein-centric way.

We used the proteomics data on two blood cell types provided by the Human Proteome Map in our case study for PPICompare [92] (see also Chapter 4). The protein abundance results of the database were utilized to calibrate a sound and non-arbitrary transcript expression threshold for the protein interaction network construction step in the pipeline.

2.1.3 *On physical interactions between proteins*

For a long time, biochemical research put its focus on studying individual biomolecules in isolation rather than on their interplay. The fundamental functional units in the cellular environment, however, are aggregates of proteins that work together in an orchestrated fashion. Multiprotein complexes are responsible for the vast majority of processes in the cell and their duties comprise tasks such as the realization of a manifold of biochemical modifications, enabling cellular communication through signaling pathways or even providing molecular motors (see also the previous Section 2.1.2 on proteins) [2]. Bruce Alberts once even described the cell as "a factory that contains an elaborate network of interlocking assembly lines, each of which is composed of a set of large protein machines" [12].

A physical interaction between two proteins is termed protein-protein interaction (PPI) and a binary complex of two proteins is called a dimer. If more than two proteins assemble, we speak of (multiprotein) complexes or oligomers. Complexes comprising identical proteins are commonly specified as homodimers/-oligomers whereas complexes of distinct members are termed heterodimers/-oligomers.

Furthermore, PPIs can be classified into obligate interactions, which is the case if the interactors do not feature stable structures on their own in vivo, and non-obligate interactions, if the interacting proteins are functional on their own. The latter can be further divided into stable interactions that are permanent and those interactions that are transient, which means they are able to assemble and disassemble spontaneously in a dynamic and context-dependent manner [130].

Binding interfaces of PPIs generally exhibit complementarity in both shape and chemical properties. Overall, the fraction of hydrophobic residues on such interfaces is smaller than in the protein core but larger than found for non-interface surfaces. For hydrophilic residues the opposite trend can be observed [130–132].

Owing to the crucial importance of PPIs and protein complexes in general, the last years brought forth many large-scale studies that strove towards the ultimate goal of acquiring complete knowledge on interactomes by mapping the entirety of PPIs within an organism. The development of the necessary experimental and computational tools in that regard started with the model organism yeast [133, 134] before the field progressed to the investigation of the human interactome [135–138].

Based on these pioneering endeavors, the size of the human protein interactome has been estimated to comprise somewhere between 150, 000 and around 650, 000 interactions [139, 140]. Although the amount of PPIs that could be captured by individual studies increased tremendously over the last decades, the yield of each individual effort is still by far not in this approximated range [141]. When the results of such studies are merged, however, our state of knowledge is approaching such orders of magnitude. The popular database BioGRID [142] in the current release (version 3.5.179 of Dec. 2019), for example, lists 396, 398 non-redundant physical interactions for humans.

It should be noted that such collections only represent the total repertoire of PPIs that could be found in a cell. Actual snapshots of the physical interactome of proteins in various cellular states will differ wildly because the actual interplay is highly dynamic in time [143, 144] and space [145]. The interaction partners of an active PPI must be expressed together at the same time, potentially even in the correct isoforms or featuring/missing specific posttranslational modifications, they must be located in the same cellular compartment in sufficient spatial proximity, and their binding topology must be devoid of interference by, for example, other proteins competing for the binding interface [146]. These issues will be of fundamental importance in Chapters 3 and 5 and are therefore discussed in more detail there.

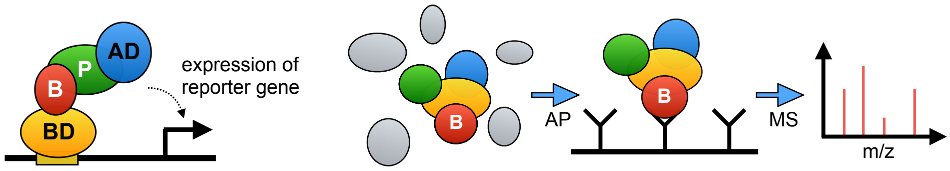
Measuring interactions between proteins

PPIs that facilitate either binary interactions between two proteins or those of multiprotein complexes can be investigated at different levels and in various ways. The perfect specification of a protein complex would comprise the composition of the complex, e.g. the identification of all its member proteins, the stoichiometry of the constituents and the topology of the assembly.

The exact topology of protein complexes can only be determined in a direct way by methods that procure spatial information, such as X-ray crystallography, nuclear magnetic resonance or electron microscopy [147]. For large multiprotein complexes such methods can involve years of work to analyze even subunits of the structure [148] which are then connected by computational efforts [146, 149]. Stoichiometries and copy numbers of complex components can be obtained by quantitative MS [138, 147].

However, in most cases only the composition of complexes is examined. Which experimental techniques should be preferred for this task heavily depends on the scale of the study that is envisaged, e.g. an unbiased sampling of a whole proteome versus investigating only a clearly defined subset of proteins, and the nature of the interactions that should be determined, e.g. binary PPIs or complexes, stable or transient PPIs, PPIs dependent on posttranslational modifications or other complex partners.

Yeast two-hybrid (Y2H), which can detect direct pairwise interactions between two proteins, and affinity purification coupled to mass spectrometry (AP-MS), which finds stable copurifying protein complexes, are the most prevalent experimental approaches to measure protein interactions in the sense



(a) *Y2H*

(b) *AP-MS*

Figure 2.5: *Measuring protein interactomes. Yeast two-hybrid (a) and affinity purification–mass spectrometry (b) are two popular experimental techniques to determine protein interactions between a certain bait protein (abbreviated as B) and a specific prey protein (abbreviated as P) or a pool of target proteins of interest. In the illustration of the Y2H method in (a), BD represents the binding domain and AD the activator domain of a transcription factor. Grey proteins in (b) are proteins in the background that are not captured by affinity purification. The figures were adapted from [150].*

of protein complex composition and are therefore introduced briefly. For an overview and comparison of less common techniques, please refer to other sources, for example the comprehensive review by [150]. In the context of protein interaction and protein complex detection, the reference protein of interest that is used to "fish" for interaction partners is commonly called the bait protein and interaction partners that are investigated as potential targets are often referred to as the prey. We will also stick to this established notation in the following. Figure 2.5 additionally supports all explanations graphically.

Y2H was initially developed around 30 years ago [151] and its capabilities were steadily improved in quality and throughput [137, 150, 152]. The protocol is based on a TF that is fragmented into its DNA-binding domain and its transcriptional activation domain (see also Section 2.1.4 on the modularity of TFs). Since the two domains are still able to activate the expression of a target gene when they are in sufficient proximity to each other, it is possible to fuse the binding domain to the bait protein and the activation domain to a prey protein. If the constructed bait and prey fusion proteins directly interact, the reporter gene is expressed in such a system (see also Figure 2.5a). In the classical approach, the yeast GAL4 protein is separated into its two domains and LacZ is used as a reporter gene to identify interactions by simple galactose selection [151].

A Y2H setup is able to detect weak binding, is relatively simple with a low cost, runs in vivo and scales extremely well from small efforts to the whole proteome. Still, there are common problems that may lead to artifacts in the data. Fusing proteins may introduce problems because the conformation of the proteins or their binding interfaces may be altered compared to their native states. More so, if a protein that itself includes an activation domain is fused to the binding domain, the reporter gene might be activated without any physical interaction at all. Also, the choice of the host, generally yeast, might influence the results, both proteins need to access the nucleus in this approach and false positives may be reported due to overexpression of the candidates. At last, the indirect readout prevents to capture the dynamics of binding processes in time or space [150, 152].

A careful experimental protocol can still ensure results of relatively high confidence [152]. Sophisticated Y2H implementations were recently used to determine interactomes that even consider protein mutations [153] or the differences induced by individual isoforms of proteins [7, 51].

With the emergence of MS as a technique to identify the protein composition of biological samples (see also previous Section 2.1.2), the library-independent detection of protein complexes became possible in a high-throughput manner by employing affinity purification upstream of MS [150, 154]. In AP-MS, the bait protein is immobilized on a solid support, such as a gel or magnetic beads, and serves as a fixed anchor to capture prey proteins from a soluble phase. Interaction partners can then be enriched by washing out proteins that are not bound to the bait and the remaining putative complex members can finally be characterized by a subsequent MS screening (see also Figure 2.5b). The fixation of the bait can be achieved by either immunopurification/immunoprecipitation, e.g. by employing specific antibodies that bind to the target, or by fusing a standardized tag to the protein which accomplishes the immobilization [154].

When suitable antibodies are available, proteins can be used in their native form while tags need to be fused and may therefore pose problems. Still, utilizing tags can benefit the experimental designs in several ways. They can allow to determine the interactomes of several bait proteins in one run or to perform successive affinity purification steps on the basis of individual tags, for example. The latter protocol is called tandem affinity purification and very common to alleviate typical problems of AP-MS such as non-specific binding events by copurification of random background proteins. Furthermore, AP-MS barely detects weak PPIs, direct and indirect binding cannot be distinguished, and the approach is, due to cell lysis and purification steps, also not able to determine interactions with spatial or temporal resolution [150, 154].

Since AP-MS detects stable aggregations of proteins, it is predestined to be employed for the experimental discovery of multiprotein complexes. To ensure that only robustly copurified complexes are reported by the analysis, usually several tandem AP-MS steps are conducted in which all members of the complex candidate are used as baits individually. Computational methods are then applied to assess if the number of observed copurification events is statistically significant [155]. This workflow brought the community several classical datasets on protein complexes [133, 134]. More recent proteomics studies are even able to add stoichiometries and protein abundances to such analyses [138].

Each experimental method to determine PPIs has its strengths and weaknesses. All methods will report a certain share of false positive interactions and, at the same time, miss other interactions. The strict implementation of control experiments is therefore a necessity to obtain reliable data. When possible, studies are sometimes complemented by the results of additional experimental approaches [150, 152].

From interaction experiments to (weighted) interactome data

As introduced in the previous section, there are many approaches that yield information on PPIs or protein assemblies in general. Yet, the scale, scope and reliability of the results from individual studies are often vastly variable. Some researchers tried to capture entire proteomes [137, 138], not necessarily using the same experimental techniques, other projects only considered very concise subsets of the proteome in isoform-resolution [51] or even considering the effects of mutations [153].

Protein-protein interaction networks (PPINs) are an effort to collect and integrate the heterogeneous knowledge on interactions that accumulated over the years. Basically, they are static networks with proteins as the nodes and the pairwise protein interactions that were derived from many independent experiments as the edges. Representing and modeling the biological interplay as such mathematical graphs allows to approach the data with the established framework and algorithms of graph theory.

For experimental data from methods that inarguably measure direct physical interactions of proteins, such as Y2H, the transformation of the results into the notion of pairs and their integration into networks is straightforward because bait and prey proteins clearly specify the interaction partners of the PPI. For data from AP-MS, on the other hand, this step is less clear because several proteins may be copurified together. Such one-to-many relationships of bait to prey proteins need to be somehow interpreted as pairwise PPIs. Commonly, this is done by either applying the spokes model, in which only the bait protein is connected with the other proteins, or the matrix model, in which all copurified proteins are thought to interact [156]. In an early study comparing results in yeast, the spoke model, which minimized the amount of false positive interactions, was shown to be three times more accurate than the matrix model, which overestimates the number of real PPIs [157].

As briefly mentioned earlier, BioGRID [142] is one of the largest databases integrating data on protein interaction experiments from the literature. In its recent release for human, the currently almost $400,000$ PPIs were derived from $30,601$ unique publications (version 3.5.179 of Dec. 2019). Overall, this BioGRID release contains data on physical interactions between proteins, and also genetic and chemical associations or posttranslational modifications of proteins, from $71,178$ publications on all major model organism species. Other popular databases of similar size are IntAct [158] (that merged with the resource MINT [159] in the course of the MIntAct project) or the metadatabases iRefIndex [160], mentha [161], HIPPIE [162], which specialized on integrating data on human, or PrePPI [163] that also adds predicted interactions to the incorporated curated data. Metadatabases merge the PPIs that are collected in several primary databases by their own criteria. For a weekly update of mentha, for example, only experimental data on direct interactions is considered and annotations from curated databases are used to assign a reliability score (more on such scores follows below). Although STRING [164] is also often mentioned in the context of protein interactomes, it is, strictly speaking, a gene-gene interaction network describing functional associations. This notion only happens to include physical interactions between the proteins coded from the genes among other pairwise relations.

With the exception of HIPPIE, all data sources mentioned were either used in a project of this thesis or were at some point part of the feature set of a software tool that is described in this thesis.

A noteworthy remark at this point is that, as stated before, these interactome maps, irrespectively of their reliability, should never be considered as accurate representations of the interactions that are active in a living cell. The best interpretation is that PPINs are a static scaffold of what could be seen somewhen and somewhere in a cell given the combined results found for many state-, time- and space-dependent specific protein interactome studies. Data integration, especially by the utilization of gene expression profiling, is a common means to put this unspecific knowledge on PPIs into a meaningful cellular context [165–169]. Chapter 3 introduces our method PPIXpress [90] that even allows to construct isoform-specific interactomes with expression data in transcript-resolution.

Another issue of PPINs is the large number of false positive interactions and the unclear amount of missing interactions. Indirect binding events, as reported by experimental data from AP-MS for example, are often falsely interpreted as direct physical interactions. Since all assays and protocols for the experimental detection of PPIs have individual strengths and weaknesses, even the overlap between the results of different studies that in principle assessed the same proteomes is often surprisingly low [170–172]. Naturally, in an environment such as the protein interactome, which is prone to a manifold of detection errors and additionally suffers from incomplete experimental coverage, one can never expect to deal with a perfect representation of the actual biological processes. It therefore makes sense to devise a way of rating the confidence in a PPIs and thus gauge the likeliness of an interaction.

Unweighted networks that only represent their pairwise relations qualitatively were declared a "dead end" in a popular review article on ecological networks [173] (networks such as food chains, which depict a "who eats whom"-relationships, for example). In a perfect world, measuring binding affinities between all pairs of proteins would likely present an optimal way to add a quantitative dimension to networks of protein interactions. Realistically, we currently lack suitable experimental data on interaction strengths and an improvement of the situation is not in sight [174]. Databases such as MINT or mentha instead compute heuristical evidence scores for each PPI by combining the annotated data of experimental methods that detected its presence (see [159] for details). The score associated with an interaction then depends on the number of experiments that confirmed the existence of the PPI and also considers the reliability of the protocols that were used in the experimental detection. Others, such as STRING or PrePPI, additionally integrate heterogeneous data on, for example, mRNA coexpression, protein colocalization, various genomic features, literature mining, or even structural assessment using matches to template complexes. In most methods, each feature describes a certain subscore which is computed independently. Subscores are then combined to a final score by some mathematical definition or statistical approach and the resulting quantitative measure is usually scaled between 0 and 1. Still, although the range suggests an interpretation of the scores as probabilities, the scoring systems are often not particularly suitable for comparisons across databases [175].

PrePPI [176, 177] is a PPIN that is used in multiple projects in this thesis and represents a metadatabase that also includes predicted interactions and an elaborate PPI weighting. In total, six public databases on experimentally-determined protein interactions are integrated in PrePPI and expanded by predicted PPIs based on structural data and non-structural evidence [176]. The structure-based score is based on a selection of sufficiently representative structural neighbors that are determined for all proteins taken into consideration. Non-structural evidence that is integrated in the PrePPI model comprises the essentiality of protein pairs (if both proteins are essential for the survival of the cell), coexpression of the associated protein-coding genes, the functional similarity of the proteins regarding GO and MIPS and their evolutionary similarity. Protein pairs that are located in different cellular compartments serve to construct a negative reference set [176]. All individual evidence is finally integrated to a measure of reliability by using a Bayesian approach and likelihood ratios as introduced in [178].

For a new version of the database [163], the modeling was expanded by several new features such as the existence of interactions among orthologs and the coexpression of orthologs in model organisms, the increased likeliness of interaction when a protein interacts with many structurally similar proteins and finally introducing a protein-peptide interaction scoring to the predictive step. In total, around 1.5 million high-confidence interactions (probability> 0.5) are listed and rated in the human PPIN of this improved version of PrePPI.

On domain-domain interactions

The stable formation of a protein interaction can, on a more detailed level, often be related to a distinct interaction between specific domains of the binding partners [146]. Since we lack structural data on most proteins and the majority of PPIs, our knowledge on protein domain annotations and domain-domain interactions (DDIs) provides a universally applicable and well-suited alternative that at least allows to model binding events between proteins to a certain extent [146].

Applications of practical relevance are, for example, found in protein complex prediction or in the analysis of PPINs in general [17, 90, 179–181]. In their role as mediators of PPIs, DDIs are pivotal in our methods PPIXpress [90] (see also Chapter 3) and (J)DACO [17] (see also Chapter 5). Interactions between protein domains are therefore an important part of all research projects that are presented in this thesis.

DDIs are usually identified on the basis of structurally resolved pairs of interacting protein domains. The common principle is then to first find Pfam domains in those experimentally determined three-dimensional conformations and to assess if neighboring domains are sufficiently close to each other to interact. This information is processed and provided for many domain types by

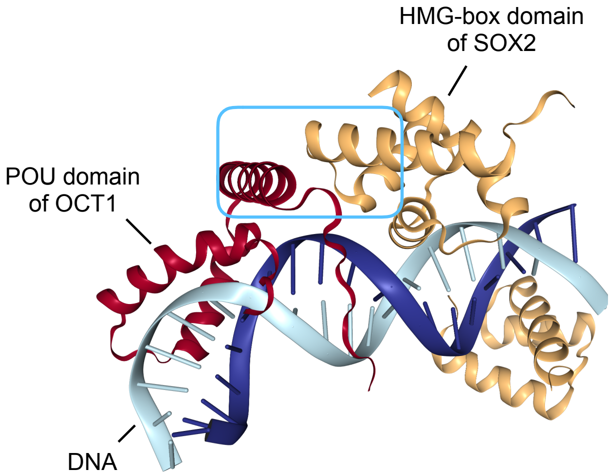


Figure 2.6: *Example of a DDI between a POU and a HMG-box domain. The example shows the DNA-binding parts of OCT1 and SOX2 in PDB entry 1O4X (version 1.2) visualized using [21]. The interaction interface is highlighted by the blue rectangle.*

web services such as iPfam [108] and 3did [107]. Figure 2.6 shows an example of a DDI found in 3did.

This structurally derived domain-level interactome is utilized in and additionally enriched by computational approaches that predict DDIs between domain families which are less well covered by structural data. Among the many statistical methods that were employed in that regard, popular prediction pipelines applied, for example, maximum likelihood estimation or linear programming to rate the likeliness of DDIs given known PPIs [182–184]. Often, additional data sources were integrated, such as data on domain fusion events and functional annotations [185].

Structure-based and inferred data on DDIs were also collected in user-friendly metadatabases. DOMINE [105] integrated two databases of PDB-derived DDIs and 7 predicted data sources which were classified into high, medium and low confidence interaction according to a simple scheme. Also, IDDI [106] integrated data by three structure-based DDI sources and 20 computational datasets. Here, a numerical scoring scheme was applied to gauge the reliability of each putative interaction between domain types.

Unfortunately iPfam, IDDI and DOMINE are not only not updated anymore, even their web services went offline during the time of my doctoral studies. For this reason, their latest data was either already supplied with the initial release of PPIXpress or included in later versions in the case of iPfam (see also the Addendum Section 3.6.1 on software updates). 3did is still updated regularly and comprises 13,499 DDIs for 9,185 domain families in its current release (Pfam version 32.0, PDB version 2019_01).

From interaction networks to multiprotein complexes

As mentioned before, multiprotein complexes are the primary workforce of the cell. The faithful experimental or computational determination of the entirety of possible protein complexes, also called the complexome, is therefore an essential step towards the greater understanding of the interwoven processes in biological systems.

Some experimental approaches, such as AP-MS, allow to gain insights on which proteins aggregate in complexes. As in the case of PPIs, this knowledge on determined protein complexes is also collected, processed and made available as web services. Relevant manually curated databases in that regard are CORUM [186], which covers 4274 mammalian protein complexes in its most recent release CORUM 3.0, and CYC2008 for yeast [187], which comprises 408 heteromeric complexes in *S. cerevisiae*. A very different approach to the problem was taken by the creators of hu.MAP [188]. Instead of aggregating the final results from independent studies in their resource on human complexes, the authors integrated over $9,000$ AP-MS experiments and evaluated the combined data anew on the basis of this much broader experimental sampling and modern machine-learning approaches.

We utilized such public datasets of human reference protein complexes in our evaluation of rewired interactions in PPICompare [92] (see also Chapter 4) and as an alternative input in the case study of CompleXChange [87] (see also Chapter 6).

Since the direct experimental mapping of protein complexomes is error-prone and tedious, the prediction of complexes by computational methods emerged as an alternative way to improve our understanding of the interplay of proteins and the modular organization of cells. The broad knowledge on the matter as stored in PPINs thereby serves as the main source of information. Consequently, even the information that was gained by experiments that only report pairwise interactions but not their actual assemblies, such as Y2H, can aid in determining the range of protein complexes as holistically as possible.

Computational methods for the prediction of protein complexes vary wildly in their overall strategy. They may construct complexes in divisive (“top-down”) or agglomerative (”bottom-up”) ways, may consider overlap of complexes or assume that complex candidates are always disjoint, may take additional biological data into account to infer who assembles with whom, and not all methods may make use of weighted interactome data [189, 190].

What unites all approaches, however, is the utilization of the topological information in PPINs because it is presumed that protein complexes represent modular units in such networks that are densely connected [191, 192]. Protein complex prediction methods are therefore algorithms that resemble the general clustering problem on graphs (see also Section 4.1.1 on clustering data) in a more specific context. It is therefore not very surprising that in the early 2000s the history of protein complex prediction began with the application of a general mathematical clustering algorithm, namely Markov clustering, on protein interaction data [193]. This clustering algorithm detects dense regions in networks by simulating random walks. Shortly after, the first dedicated approach to predict clusters of proteins from PPI data followed with MCODE [194]. MCODE uses unweighted network data and is based on the edge clustering coefficient, a measure of how tightly knit a neighborhood is [195]. A plethora of approaches that took their take on the problem followed [189, 190].

Yet in my view, general clustering approaches with non-overlapping results are not ideal to find protein complexes. Rather than representing fixed stable complexes, modules in PPINs often delineate regions in which submodules of proteins assemble combinatorially. Therefore many member proteins may be shared between biological complexes and a prediction method should account for that possibility [144, 145].

ClusterONE (clustering with overlapping neighborhood expansion) is a more recent complex prediction method that is able to consider overlapping complexes and can make use of weighted interactome data [196]. Since its core principle is a pillar on which the DACO algorithm was designed, we will go a little bit more into the details here. Given a weighted protein interaction network, ClusterONE conducts a local greedy optimization of a metric called cohesiveness around a set of seed proteins. Seed proteins are either defined by the user or automatically selected from all proteins in decreasing order of their degree (number of neighbors) in the network if they have not been included in a complex, yet. The cohesiveness $f(V)$ of a set of proteins V quantifies the potentially worthwhile property of complexes candidates to be densely connected among each other but at the same time to be well-secluded from non-members of the set:

$$f(V) = \frac{w^{\text{in}}}{w^{\text{in}} + w^{\text{bound}} + p|V|}$$

where w^{in} denotes the summarized weight of all internal interactions between members of V and w^{bound} is the weight of all interactions on the boundary between members of V and the remaining network. With $p > 0$, $p|V|$ serves as a penalty term that additionally offsets the boundary weight to model yet undiscovered interactions missing in the data. The cohesiveness $f(V)$ is optimized locally by starting from the individual seed proteins that are declared as single protein clusters. In each iteration, it is tested if adding a protein that is adjacent to a member of the complex or if removing a complex member on the border of the current complex candidate leads to the largest increase in cohesiveness. This most beneficial step is then conducted or the locally optimal result returned if no further increase can be achieved. Complex candidates that overlap more than a certain threshold according to an overlap score are then merged in subsequent steps. Finally, complexes with less than three members and complexes of insufficient density are filtered out before the candidates are returned.

All approaches that were mentioned so far only take into account the structure of the interactome network. The integration and appropriate modeling of additional data has also proven beneficial in the context of protein complex prediction. Valuable contributions were made by including functional annotation data [197, 198] or using literature mining of biomedical articles [199].

Still, crucial biological factors such as the actual expression of protein-coding genes of interaction partners at the same time (temporal information) or structural limitations of binding interfaces (spatial information) are lacking when it comes to the context-sensitive deciphering of the dynamic complexome and only static PPINs are considered. While the temporal state of the interactome is commonly approximated by including gene expression profiles in the analyses [200–203], considering structural data to account for binding site competition and to enumerate combinatorial binding possibilities is also feasible, but only possible for a small share of the proteome [145, 204, 205].

Modeling binding interfaces on the basis of DDIs (see previous subsection) has proven to be a worthwhile practical alternative to the limited amount of structural data [17, 179, 180, 206–208]. In such domain-domain interaction network (DDIN) models, binding interfaces are approximated by protein domains and DDIs are thought to facilitate PPIs. Binding site competition can then be described if each domain is constrained to only support one DDI simultaneously. In the context of protein complex prediction, DDINs were used to filter false positive complex candidates that, when assessed with the model, did not allow for a stable binding topology [179, 180], or to define connectivity in stochastic simulations [206, 208].

With the development of the domain-aware cohesiveness optimization algorithm DACO [17] we also added our share of contribution to the topic of predicting combinatorial protein complexes that consider mutual exclusivity of binding interfaces from data on DDIs. It is the main topic of Chapter 5 and introduced in detail in Section 5.3.1. By using contextualized input PPINs constructed by our tool PPIXpress [90] (see also Chapter 3) and transcript expression data, temporal information on the cellular state that even accounts for alternative isoforms of the proteins can be incorporated in the complex prediction by DACO.

2.1.4 *On transcription factors and transcription factor complexes*

TFs and TF complexes (TFCs) are crucial components of cellular control that basically entail all biological topics that were covered before. As mentioned earlier in Section 2.1.1, TFs are DNA-binding proteins that have the ability to modulate the rate of transcription. Furthermore, they are highly modular proteins with clearly defined functional domains and employ physical interactions to perform their biological functions (see Figure 2.6 above).

Protein complexes that involve TFs are the determinants of eukaryotic life and control all essential processes from the cell cycle in yeast [209] to mammalian cell fate decisions [210–212]. Owing to their importance for cellular regulation and their worthwhile information content, TFCs are the focus of all case studies in this thesis that are related to protein complexes.

Most TFs comprise two essential parts: a DNA-binding domain, which specifies which sites in the genome are targeted by the factor, and an effector (or regulatory) domain, which may modulate transcription in many ways, e.g. by directly affecting core processes of transcription, by mediating relevant PPIs, or by facilitating the activity of the TF in a ligand-dependent manner (see also principle of Y2H in Section 2.1.3). These functional modules are usually conserved, in the case of DNA-binding domains often even structurally, and the evolutionary shuffling of this universe of separable components yielded the manifold of known TFs [16, 213, 214]. Given this high degree of modularity, it is not surprising that TFs are more likely to be alternatively spliced than most genes and that there is a tendency towards splice events that encompass complete protein domains [215].

TF families are commonly defined on the basis of their DNA-binding domains since those are well-characterized for most TFs. Only 12 to 15 structural folds describe the families of all DNA-binding domains in eukaryotes. The major TF families of zinc finger, Homeodomain, basic helix-loop-helix or basic leucine zipper proteins were already known in the 1980s [14, 214].

TFs generally bind the major groove of the DNA because the exposed hydrogen bonds therein allow for a specific detection of the sequence context. The actual sequence specificity of TFBSs is then defined by a core motif of only 6-12bp [2, 13]. Since a typical human gene contains compatible binding sites for very many or even most TFs, individual regulatory proteins clearly lack specificity. Furthermore, almost all binding sites predicted from motifs are non-functional in most cellular contexts [216]. Only the defined cooperative interplay of TFs as complexes and their environmental dependencies such as cofactors or chromatin states are able to explain cellular control with the necessary level of detail [8, 214]. The deciphering of gene regulation from TF binding to gene expression remains a grand challenge of systems biology and the computational prediction of relevant binding events and their effect on target genes usually involves the integration of a broad assortment of experimental data, such as data on histone modifications or chromatin accessibility, that supplement the knowledge on binding motifs. A recent approach towards that issue is, for example, TEPIC [217].

Binding motifs of TFs are studied experimentally by chromatin immunoprecipitation with DNA-sequencing (ChIP-seq). In ChIP-seq, proteins that are bound to the DNA are first crosslinked to the DNA and the DNA is subsequently sheared. In the following, the protein-DNA fragments are enriched for fragments containing the target TF by using a suitable antibody and immunoprecipitation (compare to affinity purification of AP-MS in Section 2.1.3). The respective remaining DNA to which the target protein was crosslinked is then released and sequenced to determine a representative collection of sequence stretches that contain the binding regions of the TF of interest [218]. When appropriate antibodies are used, the same protocol can be used to detect histone modifications. Since binding regions determined by ChIP-seq are larger than the relevant TFBS themselves, motif discovery algorithms such as DREME [219] are finally applied to isolate the conserved core binding sequence motif within all sequence patches that were determined from the fragments.

TF binding motifs are usually encoded as position-specific weight matrices in which each of the four nucleotides has a score assigned at each motif position. The binding strength or preference to bind can then be evaluated for an arbitrary sequence segment by multiplication of the corresponding scores at each position [214]. While alternative models exist, e.g. based on nucleotide pairs to model position-dependence in the motif [220], they are far less common than the straightforward mononucleotide weight matrices. Popular services that provide TF motifs are, for example, JASPAR [221] and HOCOMOCO [220]. In its most recent version 11, HOCOMOCO is a compilation of 680 human and 453 mouse TF binding motifs that were derived by the integration and reanalysis of many experimental datasets. HOCOMOCO data of versions 9 to 11 were utilized in various projects presented in this thesis.

A second major component of TFs is the effector or regulatory domain that provides mechanisms to actively modulate specific phases of transcription. Furthermore, effector domains may provide binding pockets for ligands, that enable to sense and react to external stimuli by causing structural changes, or may control nuclear trafficking of the protein as an alternative way of regulating the biological activity of the TF [214].

To impact the expression of genes, TFs employ a diverse range of mechanisms. Some are able to directly interact with RNA polymerase or the general transcriptional machinery through PPIs that are facilitated by their effector domain. In doing so, they can modulate the rate of the assembly of the transcription initiation complex, improve the recruitment of important factors such as RNA Polymerase II or affect the elongation phase of transcription. Since all steps of this pathway can be adjusted independently, the contributions of several TFs or TFCs that concern individual phases of transcription can act synergistically. A TF that comprises a very similar binding domain but lacks the effector function can act as a transcriptional repressor by simply blocking other TFs or the transcription machinery sterically from binding to the target region [14, 16, 214].

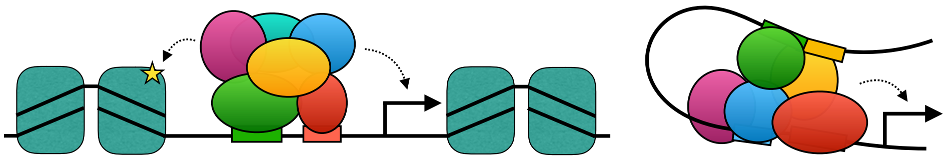
Another mode of action is the recruitment of additional regulatory proteins that are called cofactors. Cofactors can cooperate in the regulation of expression in various ways to activate (coactivators) or repress the transcription of target genes (corepressors) although they do not have the capability to bind to the DNA themselves. In functional complexes with TFs, different classes of cofactors act as mediators of TF activity by contacting other important regulatory proteins, allow for the remodeling of the chromatin to alter the accessibility of binding sites or biochemically modify histones and other proteins [14, 16, 214].

TFs and their regulatory complexes can also operate from distal regulatory elements and still influence the transcription of target genes by bending the DNA into loops. The alteration of the three dimensional structure facilitates spatial proximity between gene promoters and regulatory regions, e.g. enhancers, that are far away in the sequence [8, 13, 14].

Typical modes of action are also presented graphically in Figure 2.7.

Cooperativity between transcription factors

The binding of individual TFs is sufficient to induce or repress the expression of a target gene in prokaryotes. In higher organisms, however, the signal of single factors is not specific enough to govern the expression of the larger number of genes and the exponentially larger number of expression patterns [4, 216, 222]. The necessary level of regulatory detail found in higher eukaryotes is only possible through the sophisticated integration of many regulatory inputs such as the TFs that are bound to a regulatory region, the cofactors and ligands that are present or also specific histone marks. Their cooperative interplay defines the context-dependent regulatory outcome [8, 13].



(a) *TFC at the promoter* **(b)** *TFC at enhancers*

Figure 2.7: *Examples of TFC function and their modes of action. TFCs can bind proximal (a) or distal to the affected gene promoter (b) if the necessary binding site constraints are satisfied. In both cases, they may also recruit accessory factors. Such multiprotein complexes can often directly influence the rate of transcription (see arrows to the transcriptional start) or may adjust other important control layers of eukaryotic gene regulation, such as the posttranslational state of histones (see arrow to starred histone tail in (a)).*

One way how biology implements this ingenious cellular control circuit is the cooperative binding of several TFs in cis-regulatory regions. The implications are best explained with an example. In Figure 2.7a, for example, the TFBSs of the green and the red TF (respective binding sites marked by the same color) need to satisfy certain distance requirements to allow the binding of the complex that is shown. The target genes can therefore be specified very precisely compared to the regulation by individual factors. Also, all TFs and cofactors of the specific regulatory assembly need to be present to induce the intended regulatory effect (implementation of AND-logic) and the final outcome depends on the final protein assembly. This operating principle allows for an amazing degree of context-specific regulation. Since many TFs are able to recruit various cofactors with opposite effects, it is possible to inverse the regulatory signal that is exerted by changing the availability of cofactors, for example [8, 13, 14]. Thus the knowledge on such assemblies is very informative: if we know about the exact protein composition of such a complex, we can potentially infer its function.

However, there are more layers that define the relevant context. As mentioned earlier, TFs and their cofactors can be able to adapt the structure of the chromatin and to modify its constituents (see also Figure 2.7a). Besides cofactors that set or reset biochemical marks of histone tails, proteins can be recruited that methylate the cytosines in DNA. While this targeted methylation of DNA positions impedes the binding capability of many TFs, the affinity of reader proteins of methylation marks, such as MeCP2 or proteins of the MBD family, is increased [11]. Thus, another layer of dependency is added to the regulatory interplay. Similar mechanisms of readers and writers are described for histone marks [223].

2.2 COMPUTATIONAL TOOLS AND SERVICES

All computational topics that are needed for the projects of this thesis but were not addressed previously are introduced hereafter.

2.2.1 *On statistical hypothesis testing*

Besides the development and formulation of hypotheses, a primary task of science is to verify or falsify untested hypotheses by appropriate experiments and subsequent statistical evaluation. The latter is usually conducted by defining a null hypothesis H_0, which is the commonly accepted belief considered to be true, and an alternative hypothesis H_1 that may explain a new theory. If there is sufficient evidence against H_0, which mostly means to show that its realization is very unlikely according to a threshold probability, H_0 is rejected in favor of H_1 [224]. Statistical hypothesis testing thus works like a legal trial: *in dubio pro reo*. We default to the drop of allegations if there is no strong competing evidence otherwise.

The common framework for statistical hypothesis testing is a hybrid merging the concept of the p-value introduced by Fisher [225] and the concept by Neyman and Pearson whereby a null and an alternative hypothesis are selected based on an admissible significance level α [226]. The null hypothesis is then rejected if the p-value $p < \alpha$. Despite longstanding criticism of mixing the Fisherian- and frequentist approach as well as applying arbitrary significance thresholds, this hybrid approach and the significance level $\alpha = 0.05$ are the omnipresent standard in the literature [227].

Formally this means there is a parameter space Θ which can be partitioned into two disjoint subsets Θ_0 and Θ_1 and the data or sample drawn from a distribution that involves parameter θ either satisfies $H_0 : \theta \in \Theta_0$ or $H_1 : \theta \in \Theta_1$. In practice, decisions and p-values are derived from the distribution of a test statistic T that is applied to the data. Test statistics are functions that transform the not necessarily numeric input data into scalars and thus enable the sampling of the data. The distribution of the resulting random variables under such a function T is then used to define a critical region for which H_0 is rejected. How a suitable T and the critical region are selected depends on what is tested by the hypothesis and the distributional pattern that is assumed for or encountered in the data. Figure 2.8 visualizes this basic principle on the example of a one-sided test, which means it is tested if a sample or population is larger or smaller than expected. Accordingly, a two-sided test assumes equality of the populations in H_0 and has critical regions in both tails of the distribution. Then, the inequality of populations can be tested. The p-value is the probability under H_0 of observing a value of the test statistic that is at least as extreme than what was observed and can aid to rate the value of a test statistic quantitatively [224].

In the following subsections I will first introduce some popular statistical tests that have been used in this thesis and then cover the problem of testing many independent statistical hypotheses simultaneously.

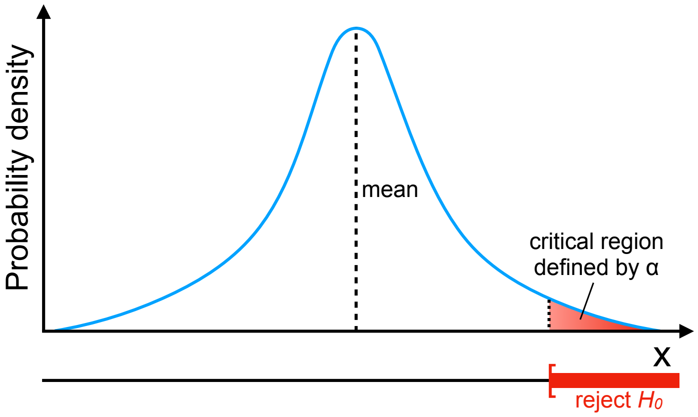


Figure 2.8: *Example of a one-sided hypothesis test. We reject the null hypothesis* H_0 *if the value* x *of the test statistic is within the critical region defined by significance level* α*.*

name	assumptions on data distribution	origin
Wilcoxon rank-sum test	none	[229, 230]
Student's t-test	populations have normal distributions with equal variances	[231]
Welch's t-test	populations have normal distributions	[232]
Wilcoxon signed-rank test	none	[229]
Paired t-test	differences between populations have normal distributions	[231]

Table 2.1: *Popular two-sample tests. Unpaired tests are shown in the upper part, the two tests below are intended to check paired samples of data. The Wilcoxon rank-sum test is also known as the Mann-Whitney U test or Wilcoxon-Mann-Whitney test.*

Common statistical tests and their applicability

An essential class of statistical tests concerns the comparison of two populations (such tests are also called two-sample tests) regarding their measures of central tendency, e.g. if their distributions share the same mean or median. A manifold of such tests are readily available in statistical software packages.

Because their application may require certain assumptions it is very important to correctly differentiate the exact use-cases of individual test, though. Some tests expect a certain distribution of the data points, for example. Such test are called parametric tests. Additionally, there are special tests for paired data, which means the tests examine if there is a difference between the related pairs rather than a shift in tendencies between populations [228]. Table 2.1 gives a brief overview on tests that have been used in the projects discussed in this thesis.

Furthermore, it can often be of interest to gauge the statistical likeliness of a single event (these are called one-sample tests) given a certain discrete distribution model for which we have the parameters.

For the outcome of independent binary events that occur with probability p, the binomial distribution describes the probability to have k successes after repeating the experiment n times:

$$P_b(k|n,p) = \binom{n}{k} p^k (1-p)^{n-k}.$$

For statistical tests we often want to pose the slightly different question of how likely it was to observe at least k successes after n repetitions. This can be obtained by simple summarization of the event probabilities:

$$P(X \geqslant k|n,p) = 1 - \sum_{i=0}^{k-1} P_b(i|n,p).$$

We used the binomial distribution to model and assess rewiring probabilities of protein interactomes in PPICompare [92] (see also Chapter 4), for example. The multinomial distribution describes the issue for the case of more than two classes.

Another discrete distribution that is omnipresent in computational biology is the hypergeometric distribution. Suppose an urn with N marbles of which K have a certain feature, e.g. a specific color. Then the hypergeometric distribution defines the probability to gather k marbles with this feature in n draws without replacement [224]. The popular application of the distribution in enrichment analyses is introduced in Section 2.2.3.

Permutation testing

In the evaluation of computational predictions and hypotheses a common issue is to ascertain if a result is actually significant. Or, more precisely in the language of statistics, how likely it was to achieve a result at least that good by chance. In practice such questions can often not be quantified using pre-implemented tests and distributions. Permutation testing (also called randomization testing or exact testing) can aid in such situations. Instead of assuming a distribution, it constructs the sample distribution of the test statistic under the null hypothesis by resampling of the data or shuffling outcome values between observations. If the null hypothesis is true, the distribution of the test statistic for the randomized data should not differ from that of the real data.

Given a test statistic $T(x)$, which can be the mean difference between populations but also a very specific quality metric for the evaluation at hand, and N datapoints that can be randomized somehow (for example regarding their labeling), a p-value can be computed by permutation testing as

$$p_{\text{perm}} = P(T > t) = \frac{1}{N!} \sum_{j=1}^{N!} I(T_j > t)$$

whereby t is the observed test statistic, the T_j are the test statistics of the permutated data and $I(x)$ is the indicator function that returns 1 if x is true and 0 otherwise. For practical considerations one mostly samples only a part of the permutation space. Then N! becomes the number of randomization iterations conducted [224].

Permutation sampling is, for example, used in the GSEA method [233] (see also Section 2.2.3) or in our analyses of deregulated protein complexes [87] (see also Chapter 6).

The multiple testing problem

As introduced previously in Section 2.2.1, significance level α describes the likelihood of observations to be true under the null hypothesis H_0. With the common rejection threshold $\alpha = 0.05$ an event that has a very low probability given H_0 may still lead to the false rejection of a true null hypothesis. This is called the type 1 error [224].

Imagine 20 hypotheses are tested simultaneously with significance threshold $\alpha = 0.05$. Then the probability to find at least one significant hit by chance can be calculated as

$$\begin{aligned} P(\text{at least one sign. hit}) &= 1 - P(\text{no sign. hit}) \\ &= 1 - (1 - 0.05)^{20} \approx 64\%. \end{aligned}$$

Even with only 20 separate tests, which is a miniscule order of magnitude compared to, for example, testing all genes of an organism for differential expression, we are already likely to falsely reject at least one null hypothesis. We may thus observe "discoveries" that are just coincidences.

A simple and conservative way to account for that consists of adjusting the rejection threshold based on the number of simultaneous tests. For m hypothesis tests H_{0i} vs H_{1i} (with $i \in \{1, \dots, m\}$) and the p-values of corresponding statistical tests $p_1, \dots, p_m$, the Bonferroni method only rejects H_{0i} if $p_i < \frac{\alpha}{m}$ [234]. This is an adjustment of the familywise error because it ensures that the probability of at least one false positive rejection is equal to at most α [235].

Since this multiple hypothesis testing correction is very stringent, often a more reasonable approach is to control the false discovery rate (FDR) instead which is defined as the expected number of false rejections divided by the total number of rejections. A popular approach that was frequently used in this thesis is the Benjamini-Hochberg method for FDR adjustment: given a fixed significance level α, the p-values $p_1, \dots, p_m$ of all m hypothesis tests are sorted and the largest k is determined that satisfies $\forall k \in \{1, \dots, m\} : p_k \leqslant \alpha \frac{k}{m}$. Then all null hypothesis H_{0i} are rejected for which $i \leqslant k$ [236].

2.2.2 *Annotation of genes, proteins and pathways*

Since the advent of modern whole-genome high-throughput experiments, life science researchers are confronted with results of sizes that are hardly comprehensible by just manual investigation. This entailed the necessity of computationally accessible resources that catalog the current knowledge on biological entities in a way that best assists the analysis, description and interpretation of such large datasets. The most popular databases for systematically annotating genes, proteins and their interplay in pathways with useful information will be introduced briefly together with the powerful concept of enrichment analysis that makes use of these descriptions to highlight biological tendencies in any experimental outcome.

The Gene Ontology

The Gene Ontology (GO) annotation system originated from the three model organism databases FlyBase [237], Mouse Genome Informatics [238] and the Saccharomyces Genome Database [239] that aimed to produce a common vocabulary to depict the roles of genes and gene products for any organism. Such a vocabulary would not only help to describe biological entities clearly but should also simplify to infer information from orthologs. The resulting ontology system is structured as a directed acyclic graph which specifies a hierarchical organization of annotation terms that are connected by predefined relationships such as "is a", "is part of", "regulates" and others. Three distinct categories were set as the roots of GO tree structures to describe certain attributes of genes or gene products:

- **biological process:** the biological objective to which the gene or gene product contributes,
- **molecular function:** the biochemical activity of a gene's product,
- **cellular compartment:** the localization in the cell where the gene product is active.

Descending from those most general annotation terms each child-term then becomes more and more specific with each level further down in the hierarchy [240]. Due to this structure, each term also implies an association with all its more general parent terms. This entails the attractive property that each gene or gene product can be simply labeled with those annotation terms that match the specificity of the respective evidence best. Figure 2.9 shows a small portion of the GO around the biological process term "histone H3 acetylation" (identifier GO:0043966). The hierarchy guides us that each protein that works in "histone H3 acetylation" is also one that matches the broader description terms "histone acetylation" or even "histone modification" in general. Roles for more exact terms, though, like a participation in the process of "histone H3-K27 acetylation" cannot be inferred from the ontology since that would require more specific evidence.

The complete ontology comprised around $45,000$ GO terms in the release of Sept. 2018 [242]. Condensed versions with a smaller representative subset of the annotation data like GO SLIM are also available. Recently the idea of GO-Causal Activity Modeling was introduced to expand the GO by a more expressive standard. In the future, literature-based subject-relation-object triplets describing the semantics of existing GO annotations should be combined into larger models that, for example, may represent whole pathways [242].

While GO data is very popular in enrichment analyses (see Section 2.2.3), and allows to infer similarity measures between genes and proteins [243] or

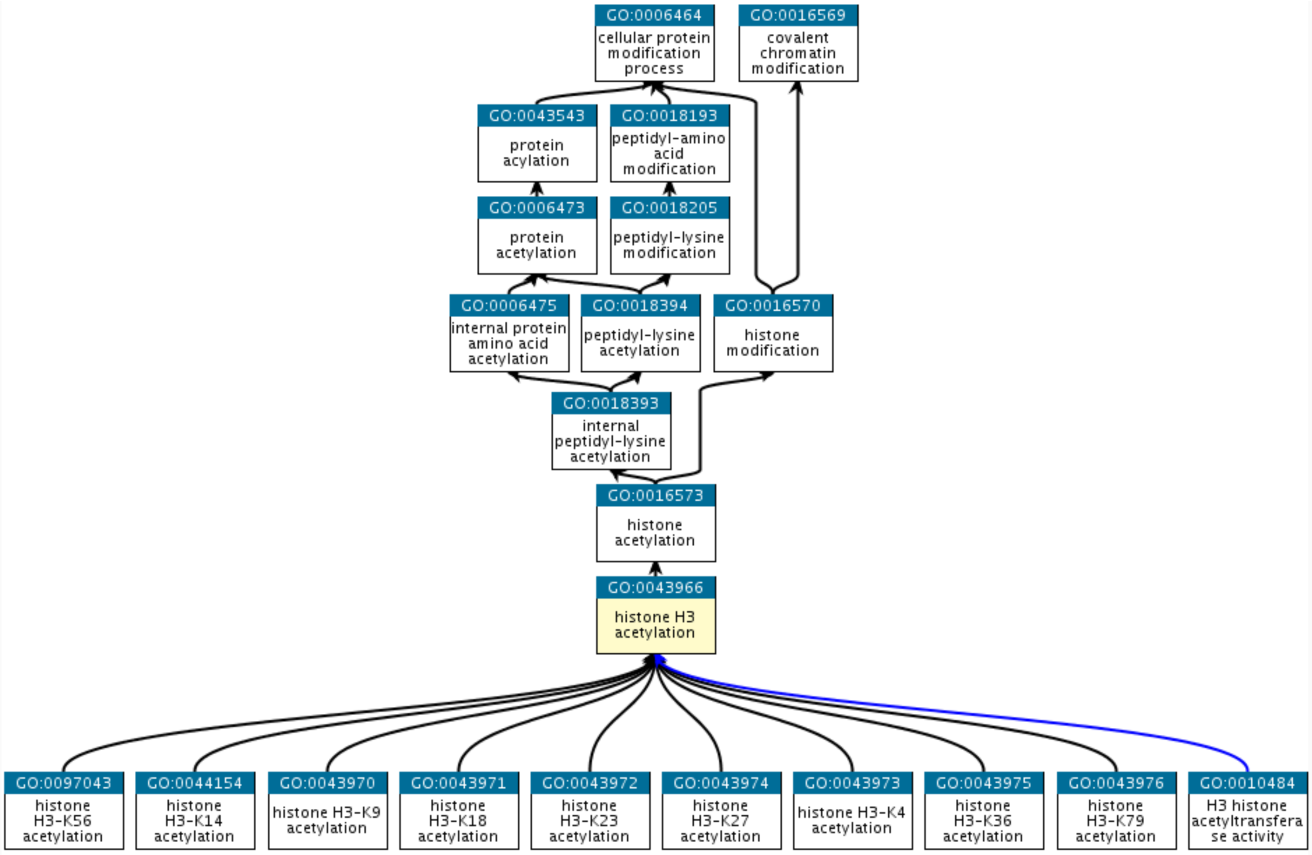

GO:0006464 cellular protein modification process
GO:0016569 covalent chromatin modification
GO:0043543 protein acylation
GO:0018193 peptidyl-amino acid modification
GO:0006473 protein acetylation
GO:0018205 peptidyl-lysine modification
GO:0006475 internal protein amino acid acetylation
GO:0018394 peptidyl-lysine acetylation
GO:0016570 histone modification
GO:0018393 internal peptidyl-lysine acetylation
GO:0016573 histone acetylation
GO:0043966 histone H3 acetylation
GO:0097043 histone H3-K56 acetylation
GO:0044154 histone H3-K14 acetylation
GO:0043970 histone H3-K9 acetylation
GO:0043971 histone H3-K18 acetylation
GO:0043972 histone H3-K23 acetylation
GO:0043974 histone H3-K27 acetylation
GO:0043973 histone H3-K4 acetylation
GO:0043975 histone H3-K36 acetylation
GO:0043976 histone H3-K79 acetylation
GO:0010484 H3 histone acetyltransferase activity

Figure 2.9: *Example of the GO structure around the biological process term "histone H3 acetylation". Black arrows depict "is a"-relationships and the blue arrow denotes a "is part of"-relationship. The figure was adapted from a graphical representation that was generated using the QuickGO web service [241].*

can aid to correct batch effects [244], its usage is not devoid of any pitfalls. Like all large-scale efforts regarding biological data, GO annotations are incomplete and suffer from study bias which can especially effect whole-genome analyses such as enrichment calculations for differential gene expression results [245]. Furthermore, the ontology itself lacks a heterogeneous level of detail which is problematic when measuring semantic similarity between genes or proteins using the GO [246].

Special application "Hallmarks of cancer": In the early 2000s the famous cancer researchers Douglas Hanahan and Robert A. Weinberg attempted to organize common traits of cancer biology into a system of hallmarks [247, 248].

In their initial paper they proposed six hallmarks as the general alterations found in cancerous cells: self-sufficiency in growth signals, insensitivity to anti-growth signals, evading apoptosis, limitless replicative potential, sustained angiogenesis, and tissue invasion and metastasis [247]. While they anticipated a simplification of the systematic understanding of this disease class in the coming years of research, Weinberg himself admitted that the matter became even more complex [249]. In their update paper on cancer hallmarks they added two more hallmarks, reprogramming energy metabolism and evading immune response, and introduced two traits that are understood to enable cancer: genome instability and mutation, and tumor-promoting inflammation [248].

While individual hallmark choices in their model or its translational benefit in clinics have drawn some criticism, their concept of "Hallmarks of cancer" still serves as the blueprint for our current understanding of cancer in the literature [250]. Both individual papers were cited more than $30,000$ times according to Google Scholar in April 2019.

The GO presents an excellent framework to actually compile all relevant genes or proteins belonging to a hallmark in an automatized way. Suzuki et al. manually curated sets of GO identifiers that should properly cover each individual cancer hallmark [251]. With such a mapping of GO term collections to hallmarks and the property of the GO hierarchy that specific terms also imply more general terms, one can easily retrieve and compile lists of all genes or proteins associated with each cancer hallmark using the web services of QuickGO [241] or AmiGO [252], for example. We did exactly this and used the notion of cancer hallmarks in our manuscript on PPIXpress [90] (see also Chapter 3) and in our collaboration paper on the roles of NFAT1 and TMX1/3 in melanoma [91].

Annotation of pathways

Similar to what is done to annotate genes and proteins there are resources that collect all genes and proteins associated with biological pathways in a granularity that even covers their participation in certain steps of the processes as directed edges in reaction graphs. Two popular pathway databases that have also been used in my projects are introduced in the following.

The Kyoto Encyclopedia of Genes and Genomes (KEGG) is a huge knowledgebase effort that dates back to 1995 as part of the Human Genome Program of Japan [253]. The web service aims to enable a systematic view on the current information we have on the biochemical networks of genes, proteins and other biomolecules in interplay for many organisms. KEGG is divided into the three databases KEGG PATHWAY, KEGG GENE and KEGG LIGAND. While KEGG GENE and LIGAND store the collected annotational knowledge on genes and ligands as we know it from other resources, KEGG PATHWAY introduced the new feature of genes and gene sets which are linked here into networks describing pathways [253]. Starting out from most known metabolic pathways and many regulatory pathways in 1999 [254], KEGG PATHWAY for human comprised $7,838$ pathways covering $20,109$ proteins in the most recent release (Release 90.0, April 2019) [255].

The Reactome Pathway Database is, compared to KEGG, a newer curated pathway service that is available under an open-source license (data as well as framework) and has a stronger focus on biochemical reactions [256]. In Reactome, for example, even posttranslational modifications of proteins are covered and described if necessary. Furthermore, Reactome's graphical representation features a global view on all reactions covered in an organism while allowing to zoom into individual pathways seamlessly whereas pathways in KEGG are separated entities that need to be displayed individually. In the current version 2,255 pathways in human are listed which involve 10,825 proteins in 12,416 reactions (March 2019 release) [257].

Pathway data were used extensively in our manuscript on PPICompare [92] to map protein interactions to pathway edges, see also Chapter 4, but also in most other projects to specify enrichment of genes participating in certain pathways.

2.2.3 *On enrichment analyses*

Enrichment analyses are a very common first computational step in the analysis and interpretation of large gene and protein sets of interest such as those obtained from differential expression analyses or hits from proteomics. The approaches deal with the question if certain categories of genes or proteins, like those that belong to a certain GO term or pathway, are statistically enriched in the hits or up-/downregulated compared to the total population of genes or proteins. Categories that could be tested may equally be the genomic position, coregulation, coexpression or any other attribute that can be determined for the biological molecules assessed. This information can then help to dissect the results, offer guidance for further investigation and serve to understand the biology involved [258]. Similar computational approaches are also found in metabolomics [259].

Popular web services for enrichment analyses of gene and proteins sets that were employed in the projects spanning my thesis are DAVID [260], PANTHER [261] and the locally developed Genetrail2 [262].

The two classical methods for enrichment analysis in bioinformatics will be introduced in the following.

Overrepresentation analysis

Overrepresentation analysis in this context deals with the question if a defined subset of genes or proteins is enriched in some annotation category compared to the total set.

A classical statistical formulation can be directly derived from this problem statement. For simplicity I will follow the original notion of the approach in [263] and just refer to genes and gene sets as well as categories of genes instead of more exact descriptions that could be covered such as annotation terms, pathways or anything more specific.

Given that M of the total population of N genes belong to a category C, the probability that genes in C also appear x times by chance in a subset of K genes drawn from the population is described by the probability density function $P_H(x, N, M, K)$ of the hypergeometric distribution:

$$P_h(x|N, M, K) = \frac{\binom{M}{x}\binom{N-M}{K-x}}{\binom{N}{K}}.$$

Subsequently, the p-value p_C for a one-sided test for overrepresentation of category C can be determined as

$$p_C = 1 - \sum_{i=0}^{x} P_h(i, N, M, K).$$

Because usually many categories are tested, multiple hypothesis correction should be applied afterwards.

Other statistical models like the binomial distribution (equal to the hypergeometric distribution for large N), Fisher's exact test or the χ^2 test are also used in practice to simplify the computational effort [264].

Gene Set Enrichment Analysis

The original implementation of Gene Set Enrichment Analysis (GSEA) [233] aims at determining if genes associated with a biological pathway have a statistically relevant tendency to rather appear towards the up- or downregulated spectrum of differential expression results. I will stick to this notion for the explanation of the method, but, of course, other biological entities and categories can be tested as well using this framework and arbitrary relevance measurements can be used.

Contrary to overrepresentation analyses that only need a list of genes of interest devoid of any numerical rating, the input of GSEA is the total universe of genes assessed together with their associated deregulation scores. GSEA ranks all genes by their score in a list we will call L and computes a running-sum for each set of genes S representing the members of a biological pathway. This is done by iterating over the gene list L in sorted order whereas the score of each gene $g \in S$ is added to the sum while the score of a $g \notin S$ is subtracted. The enrichment score $ES(S)$ is then defined as the maximum deviation from zero during the summarization process and corresponds to a weighted Kolmogorov-Smirnov statistic [233]. Figure 2.10 visualizes this running-sum approach.

For randomly distributed gene sets S, $ES(S)$ will likely be rather small compared to sets that are overrepresented at the top or bottom of the ranked list L. An empirical p-value can then be gathered by estimating a null distribution $ES_{NULL}(S)$ on the data by permutating the phenotype labels and counting how often the enrichment scores determined for the set S and randomized data were equal to or exceeded $ES(S)$ [233]. Other permutation approaches are equally possible in this framework depending on the exact application at hand [87, 265, 266].

Edge Set Enrichment Analysis (ESEA) uses the principle of GSEA to infer significantly deregulated pathways from differential correlation between genes that share edges in pathways [266]. We also used the core ideas of GSEA in CompleXChange to detect seed proteins that are enriched within deregulated protein complexes [87], see also Chapter 6.

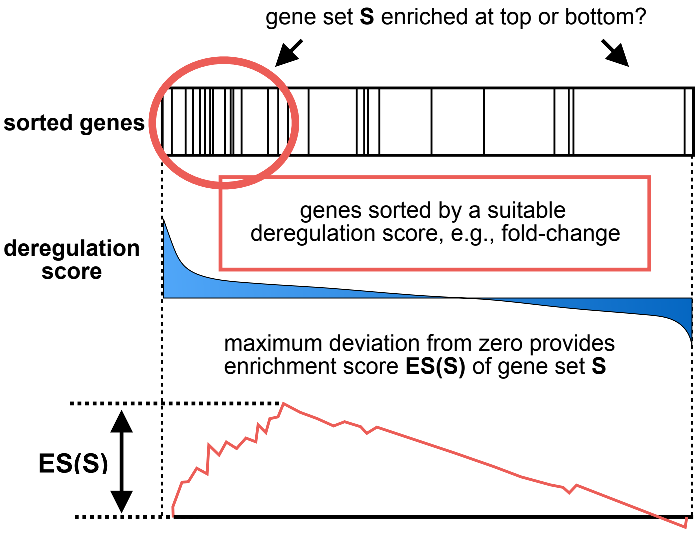

gene set **S** enriched at top or bottom?
sorted genes
genes sorted by a suitable deregulation score, e.g., fold-change
deregulation score
maximum deviation from zero provides enrichment score **ES(S)** of gene set **S**
ES(S)

Figure 2.10: *The idea behind the GSEA approach. To estimate the significance of the distribution of members of a gene set* S *in a list of genes* L*, an enrichment score* ES(S) *is computed by summarization and its likeliness by chance assessed by permutation testing. Figure adapted from [233].*

3 CONSTRUCTION OF CONDITION-SPECIFIC PROTEIN INTERACTION NETWORKS WITH TRANSCRIPT RESOLUTION

This chapter introduces an approach to construct protein-protein interaction networks that are tailored using transcript expression data and the application of this contextualization method to breast cancer data. Sections 3.2 to 3.5 were adapted and expanded from Will, T. and Helms, V., "PPIXpress: construction of condition-specific protein interaction networks based on transcript expression", *Bioinformatics*, 2016 [90]. I initiated this project and the study, designed and implemented the software, performed data analysis, conceived the figures and wrote the original manuscript. Volkhard Helms aided in designing the study, interpreting the data as well as editing of the manuscript. Supplementary materials that are published were omitted here, please refer to the online materials `https://doi.org/10.1093/bioinformatics/btv620`. The approach is provided as the tool PPIXpress which is available at `https://sourceforge.net/projects/ppixpress/`.

3.1 PREREQUISITES

3.1.1 *On retrieving data*

Besides the actual data integration methodology introduced in Section 3.3.1, a major challenge of this project was the amount of data that should be retrieved on the fly. The two essential approaches that were used in PPIXpress are introduced briefly in the following subsections.

Direct HTTP(S) / FTP downloads

Hypertext Transfer Protocol (HTTP) [267] and File Transfer Protocol (FTP) [268] are omnipresent protocols to transfer data over networks. Both protocols have individual strengths and weaknesses that originate from their intended purposes. FTP, the older protocol, can be understood as a protocol on the file level that even allows listing and browsing directories on the remote server. Until an actual file transfer is established, quite a number of client-server communication steps - each one taking its share of time - can be needed by design. HTTP, the protocol of modern web browsing, is faster in that sense, but provides a steady overhead of partially unnecessary meta-data in each transmission that is used by HTTP clients to interpret and treat the data stream appropriately in terms of encoding and content type, for example.

As a very simply example use case of HTTP downloads in PPIXpress, imagine that the user input consists of a reference protein interaction network with proteins given as UniProt accessions and expression data annotated with Ensembl identifiers. To then be able to retrieve all relevant annotation data from the correct Ensembl database, PPIXpress at first needs to infer the organism from the protein identifiers. This is achieved by querying at least one of the protein accessions in UniProt using a programmatically accessible and well-parsable version of the site and retrieving a few lines of text. Here, information on the human NANOG protein can be enquired, for example:

```
https://www.uniprot.org/uniprot/Q9H9S0.txt
```

Note that the service that is called here uses HTTPS (Hypertext Transfer Protocol Secure), an extension of HTTP that encrypts each connection, and not HTTP. During the life cycle of PPIXpress the adoption of HTTPS as the default protocol of websites increased tremendously due to pressure of the major web browsers [269, 270]. UniProt's switch to secured connections was included in PPIXpress version 1.20 by adapting the corresponding calls. Since a simple FTP call looks the same to the end user on this high level of abstraction, its exemplification is omitted here.

A more interesting additional example involves the usage of HTTP-GET, which is a simple way to request a resource from a web server that is generated based on parameters specified by the call. When PPIXpress detects HGNC identifiers (HUGO Gene Nomenclature Committee) in protein interaction data, all gene names are converted into proper UniProt protein accessions on the basis of the most-current curated naming data downloaded by a customized request to the HGNC web service [271]:

```
https://www.genenames.org/cgi-bin/download/custom?col=gd_app_sym&col=md_
    prot_id&status=Approved&format=text&limit=0
```

Here *https://www.genenames.org/cgi-bin/download/custom* is called and the remaining part of the address, separated by the question mark, specifies parameter pairs that the server uses and - dynamically or statically, depending on the exact implementation - answers to in the desired way. The individual pairs are separated by *'&'* and provided in the format *parameter name = parameter value*. Here, the two *col* parameters request that columns for HGNC gene symbols (*col=gd_app_sym*) and UniProt accessions (*col=md_prot_id*) are returned. The output is restricted to approved genes only (*status=Approved*) and the complete list of results (*limit=0*) is returned in a textual format (*format=text*).

Besides UniProt and the HGNC web service, 3did, iRefIndex and mentha data are retrieved using HTTP. IntAct data is retrieved using FTP.

(My)SQL queries

MySQL is a relational database management system distributed by Oracle that implements the Structured Query Language (SQL) standard [272]. Relational databases are structured in tables of rows and columns whereby entries in distinct tables can be related to each other and tables can be merged by *join* operations [273]. Queries in SQL are used to retrieve data from MySQL databases.

A large part of the data that is integrated in PPIXpress is retrieved from Ensembl using MySQL. In release 95, the Ensembl *Core* database on human data in the GRCh38 reference assembly, *homo_sapiens_core_95_38*, contains 74 interconnected tables. Figure 3.1 shows a subset of the tables that are relevant for PPIXpress.

PPIXpress needs to be able to associate all transcripts of an organism with Pfam domain annotations, for example. To obtain this information it is necessary to join the 4 data tables *transcript*, *translation*, *protein_feature* and *analysis* while restricting analyses to Pfam annotations. The following SQL query lists all Pfam domains found in each protein coded by a transcript:

```
SELECT transcript.stable_id, protein_feature.hit_name, protein_feature.hit
    _start, protein_feature.hit_end
FROM transcript, translation, protein_feature, analysis
WHERE transcript.canonical_translation_id = translation.translation_id AND
     translation.translation_id = protein_feature.translation_id
AND protein_feature.analysis_id = analysis.analysis_id AND analysis.logic_
    name = 'Pfam'
```

Start and end positions are included to enable multiplicity of domain family occurrences.

MySQL is also used to query the UCSC Genome Browser database [274] to allow a conversion of UCSC transcript identifiers to Ensembl transcripts. This is necessary when loading transcript expression data by TCGA [88] into PPIXpress.

3.2 INTRODUCTION

Protein-protein interaction networks (PPINs) are an important pillar of data integration in computational biology and have been used in a large number of studies and approaches. Generally, such networks are collections of physical interactions between pairs of proteins compiled from different experiments [275].

Full PPINs provide a convenient overview of the interactome of an organism. Yet, they do not reflect the true wiring exhibited by the cell in a specific state, because an interaction can only be realized if both partners are available. Pruning the full network to the set of proteins whose genes are expressed in the same condition has proven to be a straightforward solution for this. This allowed investigating the interaction landscape across tissues [165–167] as well as the origin of tissue-specific diseases [168]. Furthermore, it improved the prediction of disease genes [169].

An estimated 95% of human multi-exon genes undergo alternative splicing (AS) [44] and the specific isoform of a protein was shown to have a considerable impact on its ability to bind interaction partners [45–47]. Thanks to the ability of quantifying individual transcripts nowadays, it thus appears worthwhile to also increase the granularity of condition-specific networks to this resolution.

Domain-domain interaction networks (DDIN) depict interactions between individual protein domains and provide a convenient framework to relate interaction sites with sequence information. In contrast to models based on atomistic structural data, DDINs allow for universal applicability [17, 179, 180]. So far, the only methodical effort regarding the effect of AS on interaction networks is found in the Cytoscape 2.x plugin DomainGraph. When linked to

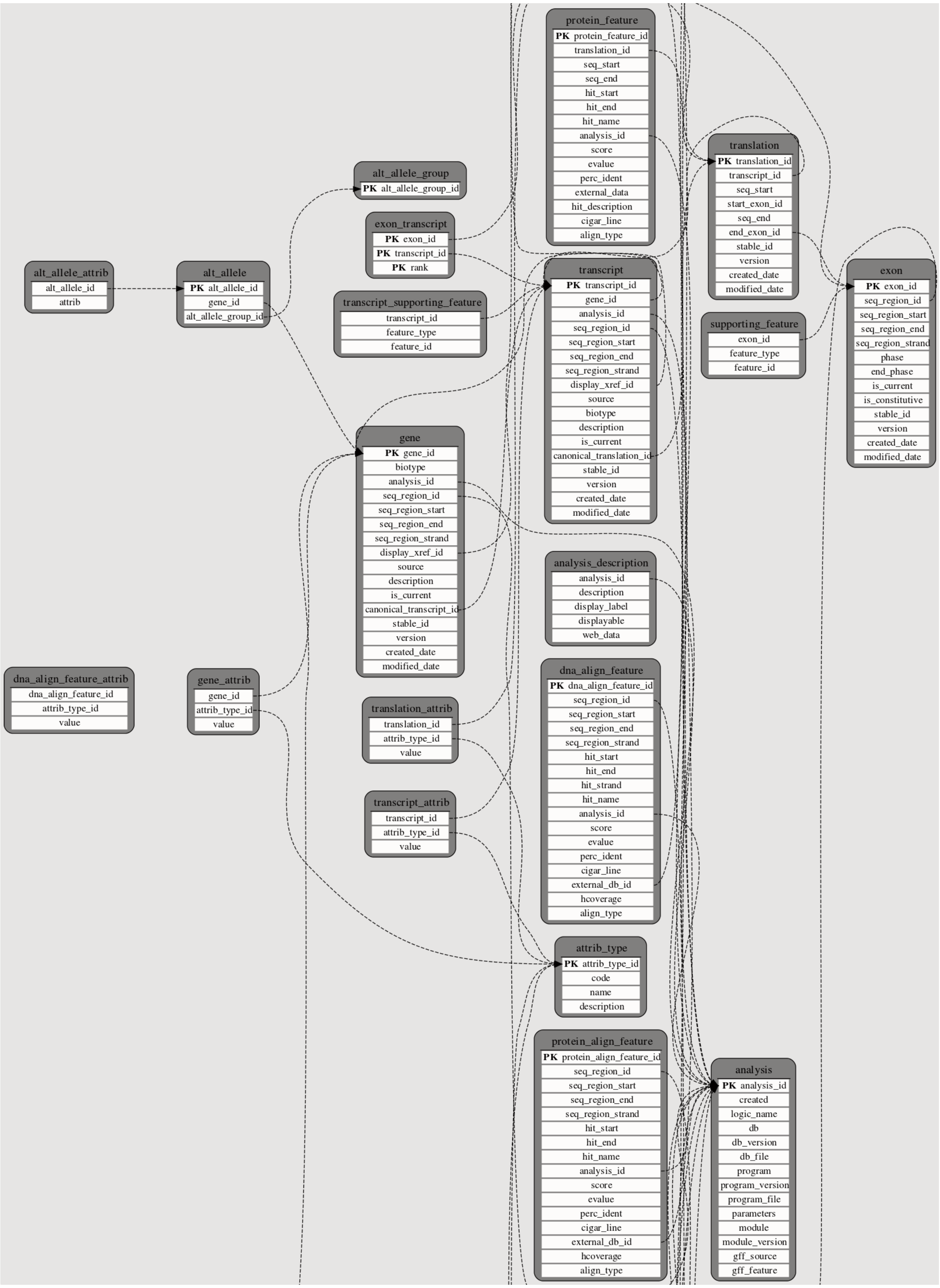


Figure 3.1: *Subset of the Fundamental Tables group in the Ensembl Core database (release 95) and their relations among each other. Figure adapted from* https://www.ensembl.org/info/docs/api/core/diagrams/Core.svg.

the AS analysis tool AltAnalyze, DomainGraph can highlight protein domains in DDINs that are affected by differential exon usage [181]. However, this tool is intended for visual exploration. While the user can manually estimate the implications of respective changes as PPIN and DDIN are visualized together, the tool does not allow to automatically infer conclusions for the PPIN on a whole-proteome scale.

With PPIXpress, we aim here at providing a simple standalone solution for the automatic construction of condition-specific protein interaction networks based on domain information and transcript expression. In addition to this core functionality, the tool is able to retrieve current protein interaction data, to add functional association scores from the STRING database [164] to unweighted networks, and it can output the underlying condition-specific DDIN for each sample. It allows the usage of compressed input files in the gzip-format and is well-suited for batch-processing.

3.3 MATERIALS AND METHODS

3.3.1 *PPIXpress*

The input data for PPIXpress consists of a reference PPIN with condition-unspecific interactions and at least one sample of transcript- or gene-level expression data. From that, the tool constructs the condition-specific subnetworks for each transcriptome. Thus each network only comprises those interactions from the reference that are considered active in the sample.

Networks can be provided in the simple input format (either interacting UniProt, HGNC or Ensembl gene pairs line-by-line, optionally with a weight) or can alternatively be retrieved from IntAct [158] for a certain organism. The current version supports expression data in the following formats: Cufflinks FPKM files [76], GENCODE or comparable Ensembl-annotated GTF files [276], TCGA RNASeq data or textfiles with expression-levels per line as commonly exported by popular R-based tools. All other data sources that are used internally are automatically retrieved in their most current versions. Furthermore, the user may optionally change the expression threshold (absolute or percentile-based) that is applied, limit the analysis to the gene-level, or inquire specific versions of retrieved data. PPIXpress is freely available at `https://sourceforge.net/projects/ppixpress/`. A user guide and example data are provided together with a precompiled executable and the complete source code.

The basic principle of PPIXpress is outlined in Figure 3.2 and will be explained in the following paragraphs. Details regarding the annotation with domain data and datasets are covered in Section 3.3.2.

Relating protein and domain interactions

In the initial mapping stage, a one-to-at-least-one relationship between interactions in the given PPIN and the corresponding DDIN is established such that all PPIs found in the reference PPIN should be supported by at least one

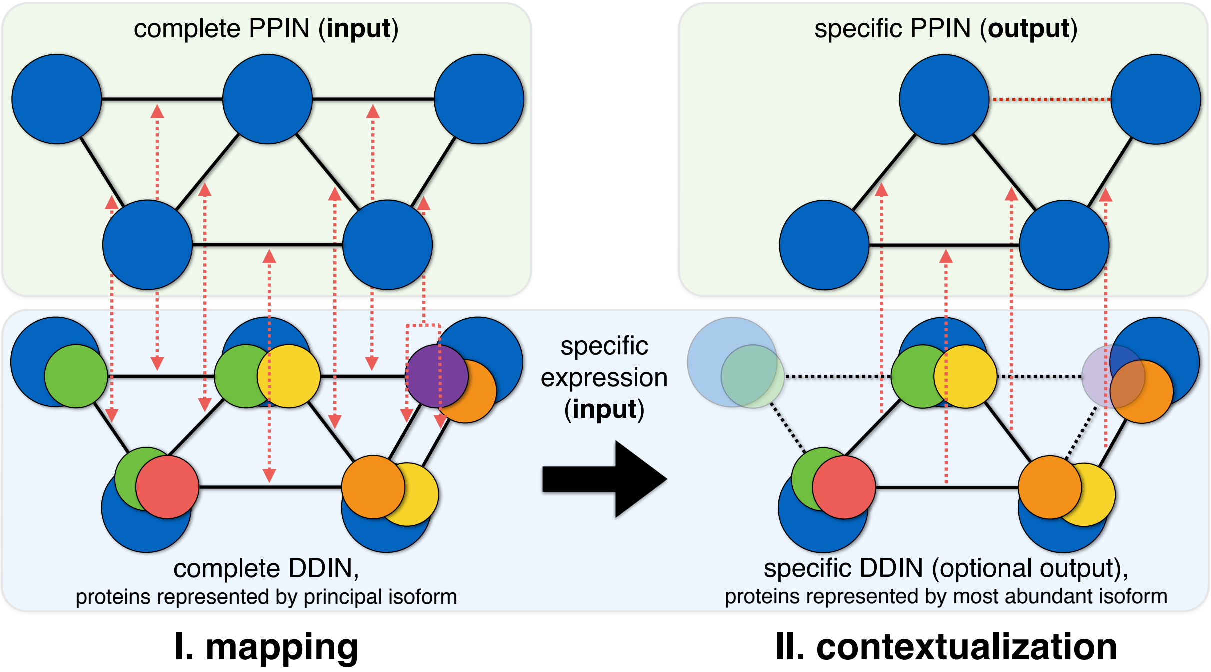

complete PPIN (**input**)
specific PPIN (**output**)
specific expression (**input**)
complete DDIN,
proteins represented by principal isoform
specific DDIN (optional output),
proteins represented by most abundant isoform
I. mapping
II. contextualization

Figure 3.2: *The PPIXpress approach can be divided into two stages. Initially, complete PPIN and DDIN are related to each other, whereby artificial domains (here shown in green) may be introduced to ensure a complete connectivity on the domain-level. This correspondence is then used to filter sample-specific domain-domain interactions (DDI) derived from transcript expression data and to map these back to the supported protein-protein interactions (PPI). The details are covered in the main text. Proteins without any expressed transcripts, as well as domains that are not found in the most abundant transcript are shown translucent. In this example, a method that only uses gene expression data would miss the disappearance of the protein interaction shown as a red dashed horizontal line in the top right picture.*

underlying DDI. In this step, PPIXpress considers the longest isoform of each protein as its representative in the DDIN, because large-scale experimental analyses and most databases usually declare it as the principal variant [277, 278]. Hence, the annotated domain compositions of the longest isoforms are used to construct a network on the domain-level that is then related to the reference PPIN. According to a dataset of feasible interactions between domain types (see Section 3.3.2), edges between interacting domains of distinct proteins are established in the DDIN if the protein pair is also connected in the PPIN. Thereby, it is noted which DDI or which DDIs support each individual interaction between proteins in the network because different DDIs may ratify the same PPI on this level.

If a protein interaction cannot be assigned to any domain interaction at this stage, we add artificial domains to the affected proteins. Those domains are utilized to also establish links in the DDIN between those interaction partners in the reference PPIN whose binding cannot be explained otherwise by available domain interaction data. Introduction of such artificial domains allows our approach to sustain a complete correspondence between the two network-layers. Adding fictitious protein domains to overcome the sparsity of domain-level data was introduced before to improve the performance of protein complex prediction approaches that make use of such data [17, 180]. While [180] introduced this idea in a non-deterministic way, PPIXpress uses a deterministic approach as described in [17]. These non-physical domains are thought to be present in every transcript coding for the protein and thus implement the behavior of gene-based methods where domain-level annotation fails to explain the protein-level outcome. This way, the methodology guarantees a seamless and safe transition to the performance of the gene-level approach whenever available data coverage on DDIs cannot explain the macroscopic observation.

Condition-specific construction

After the sample-independent mapping step, the expression data is incorporated to contextualize the network. Initially, all transcripts above a user-defined threshold are determined for the specific expression sample. From those, only the most abundant transcript of each protein is chosen to build a sample-specific DDIN, whereas all others are neglected. Based on this specific DDIN derived from the expression data and the previously determined mapping of DDIs to PPIs, a specific PPIN is constructed that only contains interactions that are supported by domain-level evidence. Here, it is not important if an individual PPI is backed by one or several DDIs; the existence of a single support is sufficient.

Viewed differently, PPIXpress used with transcript data first prunes the reference PPIN in a node-specific manner such as the established methods that are based on gene expression, but additionally trims the network in an edge-specific way guided by the domain data. The resulting network is therefore always a subnetwork of one obtained from a construction method based one gene expression. Figure 3.2 shows an example for this (red dashed interaction). These additionally considered 'edgetic' changes, as they are called in recent literature, are increasingly thought to be of crucial importance for phenotypic traits [153, 279, 280]. If PPIXpress is switched to the gene-level mode all genes with expressed transcripts (or all above the threshold if only gene expression data is given) are taken into account. The longest coding transcript, the same reference as in the initial mapping, is selected as the representative of the protein. Thus the gene-level behavior is replicated while the specific DDINs are also reported.

Although data and methodology would in principle allow to process the contribution of a weighted ensemble of transcripts at this stage, we decided to introduce the strong assumption to discard all but the most abundant transcripts per protein. On the one hand, there is increasing biological evidence that generally only one dominant transcript per gene acts as the main contributor in a cellular condition [281–283]. On the other hand, quantifying the distribution would require several additional parameters that may render the model unnecessarily complex and consequently the tool less appealing to the user. The discretization thus equally satisfies biological as well as practical considerations.

3.3.2 Datasets and protocols in PPIXpress

Protein interaction networks

Self-interactions between proteins are not considered by PPIXpress because they interfere with classical types of network analysis such as complex prediction or disease gene prioritization. Furthermore, if the input PPIN is annotated with one of the non-UniProt accessions, they are converted using the HGNC web service or Biomart [284], depending on the identifiers at hand.

Domain annotations

Internally, the tool queries UniProt [285] to infer the organism that is dealt with from the network data. With this knowledge, all required annotation data is retrieved from the appropriate and most recent Ensembl database [286] by MySQL queries. The data comprise the relations between proteins, transcripts and genes, but also the assignment of resulting protein domains to transcripts. Only transcripts that can be directly associated to Swiss-Prot proteins are considered.

PPIXpress uses domain annotations derived from the manually curated Pfam-A database [287]. Pfam domains in Ensembl are detected for each transcript individually using InterProScan [288] and are automatically updated with every new release. As Pfam-A domains are non-overlapping and have predetermined family-specific detection thresholds that are used by InterProScan to filter for matches, neither additional parameters nor any postprocessing are needed within PPIXpress for this step [110]. Moreover, queries and internal data structures are designed to reflect the repeated occurrence of the same domain type within a protein in the optionally returned sample-specific DDINs.

Domain interaction data

To provide a comprehensive knowledgebase of physical interactions between protein domain types with PPIXpress, we precompiled high-confidence domain interaction data from DOMINE [105] and IDDI [106]. Both are integrated databases that assign reliability estimations to their available datasets. In DOMINE (version 2.0) interactions were classified into disjoint categories according to their estimated confidence. In IDDI (release May 2011) numerical confidence values were assigned to each interaction. As those primary resources appear not to be updated anymore, we additionally integrated automatic retrieval of current data from the 3did [107] and iPfam [108] databases whose interaction data is exclusively inferred and automatically updated from the RCSB Protein Data Bank [19].

By default, PPIXpress uses a high-confidence subset of DOMINE and IDDI (see definition of PRE_{HC} in Section 3.3.3) expanded by the most recent 3did/iPfam data. Interactions between domains of the same type are taken into account if they are annotated.

3.3.3 *Datasets and protocols in evaluation*

Protein interaction networks

Data of experimentally determined physical interactions between proteins in human (*H. sapiens*, taxon 9606), mouse (*M. musculus*, taxon 10090), fruit fly (*D. melanogaster*, taxon 7227), and yeast (*S.cerevisiae S288c*, taxon 559292) were retrieved from IntAct (release 189) [158] using PPIXpress. For human we additionally compiled a second PPIN from physical interactions between human proteins in BioGRID (release 3.3.124) [289]. Here, a conversion to UniProt accessions was carried out using mapping data from HGNC [290] that was downloaded on May 5., 2015.

Expression data

For the case study, transcript expression data for breast cancer (BRCA) was retrieved from TCGA [291] as level 3 Illumina HiSeq-RNASeq V2 data based on RSEM quantification [73] and filtered to the portion of 112 matched normal/tumor samples (last updated Jan. 14., 2015).

Since it is a common threshold across popular RNA-seq quantification methods [167, 168, 292], by default all transcripts (or genes if only gene expression data is given) with an abundance value above 1.0 are considered as expressed in PPIXpress. For the case study we also used this standard threshold.

Domain annotations

For all conducted analyses we used data from Ensembl release 79.

Domain interaction data

To evaluate the potential influence of the DDI dataset on the results, we compiled different subsets of data from the aforementioned sources (see Section 3.3.2): PRE_{HC} only contained those interactions from DOMINE that were inferred from structure or within the category of highest-confidence predictions and those interactions from IDDI whose confidence values exceed a threshold associated with an accuracy of 90% in the benchmarks of their original publication. PRE_{VHC} is a subset of PRE_{HC} that was restricted to the experimentally known interactions in DOMINE and the portion of IDDI that achieved the highest accuracy of 98% in [106]. 3did/iPfam contains the retrieved data from these two structure-based databases and ALL-DDI denotes the merged dataset $\text{PRE}_{\text{HC}} \cup$ 3did/iPfam. All conducted analyses were based on data from iPfam version 1.0 and 3did version 2015_02. Table 3.1 outlines the respective sizes of the four DDI datasets used.

Whole-genome rewiring of protein interaction networks

For all 112 cases in TCGA with matched BRCA data from both normal and tumor tissue from the same patient, we constructed condition-specific protein interaction networks for both states and counted the changes in every comparison across all matched samples. To keep track of the changes within

	ALL-DDI	PRE_{HC}	3did/iPfam	PRE_{VHC}
domain types	7,449	6,193	5,920	4,354
domain interactions	30,551	26,377	10,953	6,285

Table 3.1: *Amount of domain-domain interaction data in the different datasets used.*

the interactome we marked each interaction that was only observable in the network of the disease sample as positive count and those that appeared only in the network of the healthy sample as negative. Figure 3.3 illustrates the overall approach for the network construction and comparison. For the evaluation the networks were constructed with PPIXpress using different methodologies and data. All steps were assessed for all settings individually. To obtain comparable abundance thresholds for protein precursors in all methods, a gene was considered to be abundant if at least one of its transcripts was abundant in a given dataset.

Besides noting the changes across all samples, a rewiring probability P_{rew} per interaction was individually computed for each matched sample pair and then averaged over all samples. P_{rew} was approximated as the number of rewiring events (interactions added + interactions removed between samples) divided by the number of interactions in normal or tumor, whichever was smaller. Since such differential networks summarized over all matched samples will inevitably contain many changes that occur only in few patients, we added a filtering step. On the basis of P_{rew} a one-tailed binomial test was applied to check how likely it was to observe a certain number of rewiring events of an individual interaction over all samples by chance. P-values were adjusted using Benjamini-Hochberg [236] and only the interactions with adjusted p-values below 0.05 were retained.

Randomized implementation of PPIXpress

To assess the assumption that only the most abundant transcript of each protein contributes to the specific DDIN (see Section 3.3.1), we modified PPIXpress to randomly select any of the transcripts above the expression threshold for each protein instead. For this randomized implementation we repeated the evaluation of the case study 100 times. As the construction method was applied to 112 * 2 samples (all matched pairs) per iteration during that process and the variance among the results was quite low, we think 100 iterations were sufficient for this comparison. Since the ALL-DDI dataset was used with the randomized method it is referred to as RANDOM(ALL-DDI) in the following tables.

"Hallmarks of cancer" data and analysis

We associated proteins with 10 currently established hallmarks of cancer on the basis of a handcrafted list of relevant GO terms by [251]. We retrieved all proteins in human with such an annotation using QuickGO [241] on May 5., 2015. Associations inferred from automatic annotation (GO evidence IEA)

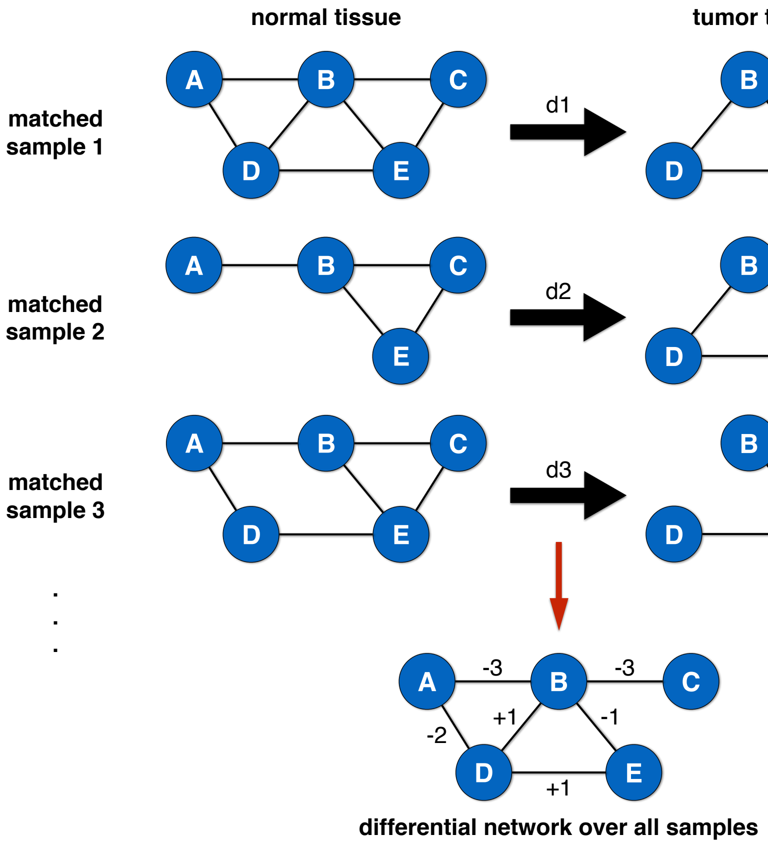


Figure 3.3: *For all matched BRCA samples from TCGA we built protein interaction networks using different methodologies. d1 to d112 denote changes in topology between normal and tumor interaction network in each matched pair. These differences were determined in every single patient and summed up in a differential network shown at the bottom. Here, the interaction between proteins A and B, for example, disappeared for all three shown matched sample pairs. Thus the edge between A and B is annotated with −3 in the differential network shown at the bottom.*

		size of network		fraction of contributing proteins / fraction of matched PPIs			
organism	data source	proteins / interactions	avg. degree	ALL-DDI	PRE_{HC}	3did/iPfam	PRE_{VHC}
human	BioGRID	15,086 / 156,271	10.4	0.517 / 0.264	0.506 / 0.256	0.364 / 0.099	0.334 / 0.093
human	IntAct	13,665 / 81,460	6.0	0.437 / 0.246	0.428 / 0.241	0.287 / 0.103	0.259 / 0.097
mouse	IntAct	7,149 / 15,742	2.2	0.264 / 0.173	0.259 / 0.169	0.144 / 0.068	0.135 / 0.068
fruit fly	IntAct	10,178 / 38,592	3.8	0.102 / 0.039	0.099 / 0.037	0.051 / 0.014	0.041 / 0.012
yeast	IntAct	5,993 / 76,003	12.7	0.530 / 0.186	0.513 / 0.181	0.272 / 0.038	0.230 / 0.036

Table 3.2: *The annotation coverage of DDINs for different reference PPINs and different DDI datasets.*

were discarded as those are often inferred from protein interactions. A protein interaction was associated with a hallmark term if at least one of its involved proteins was part of the corresponding set of hallmark proteins.

Enrichment analysis

Enrichment analysis was performed using DAVID 6.7 [260]. We specifically checked for enriched KEGG pathways [293] and GO biological processes [294], set the proteins included in the respective input network as the background and kept the default settings of DAVID otherwise.

3.4 Results and Discussion

3.4.1 *Coverage of DDI datasets in practice*

We first examined how many protein interactions are typically supported by at least one non-artificial domain interaction in the mapping stage of PPIXpress (interaction coverage) and how many proteins have domain annotations that contribute to that (protein coverage). We did this across various reference PPINs of several organisms and on the basis of different high-confidence DDI datasets as described in Section 3.3.3.

The results are shown in Table 3.2. In all cases, a larger dataset allowed to relate a larger part of the reference protein interactions to known domain interactions. PRE_{HC}, for example, contained around 2.4 times as many DDIs as 3did/iPfam (compare respective columns in Table 3.1) and could relate 2.3-4.8 times more PPIs to DDIs depending on the reference network examined (compare PRE_{HC} and 3did/iPfam in Table 3.2). The addition of recent structural data from 3did/iPfam to the precompiled integrated dataset only led to a small improvement in interaction coverage (for all networks consistently below 1%, compare ALL-DDI and PRE_{HC} in Table 3.2). The human interactomes had the best coverage of interactions for all DDI data examined. However, even in the best case, still only about half of the proteins and roughly a fourth of the interactions could be associated with supporting domain information at all. Since the density of the PPINs was very heterogeneous (avg. degrees ranged from 2.2 to 12.7, see Table 3.2), the ratio of proteome and interactome coverage is not meaningful across networks. Interestingly, the coverage of proteins associated with hallmarks of cancer was higher than those of non-hallmark proteins (Tables S1/S2).

Whereas this analysis suggests that the partial coverage of domain annotation may reduce the value of the proposed approach, it emphasizes the importance of a flexible approach that is able to integrate both well and poorly annotated proteins seamlessly. What are the reasons for that tenuous coverage? While the vast majority of proteins in human and yeast are annotated with at least one Pfam domain [111], even in our largest dataset only 7,449 of the 14,831 domain types in Pfam 27.0 [287] have known domain interactions. Aside from the fact that some domain types may not be meant to facilitate protein interactions anyhow, experimental coverage of domain interactions is still sparse and not expected to near completion in the near future [105, 295]. Even if data on actual interactions was more comprehensive, respective binding interfaces have to be related to conserved protein building blocks of any kind to make the data universally applicable. However, interactions can also be mediated by disordered regions between domains [95] which are difficult to account for by a general annotation scheme as Pfam and are underrepresented there [111]. This is, for example, the case for the pluripotency transcription factor Oct4 [296].

3.4.2 *Rewiring of protein interactions in breast cancer*

Since deregulation of splicing factors and accompanying alterations in protein products are known to contribute to tumorigenesis [52–54], transcript-based network construction may benefit an analysis in that context. Thus we present as a case study a comparison of the changes in the interactome between matched healthy and tumor samples from 112 breast cancer patients as explained in Section 3.3.3 that we conducted with different network construction approaches.

In cancer, changes in the interaction network can be expected to include proteins that are associated with certain hallmarks of cancer and that are frequently found in biological processes and pathways related to cancerogenesis. Based on this assumption we assessed whether the transcript-based methodology of PPIXpress was advantageous to the established gene-based network adjustment and to what extent selecting particular DDI datasets influenced the results.

Moreover, we examined the effect of our decision to exclusively rely on the domain annotation of the most abundant transcript above the threshold for each protein (see Section 3.3.1). To evaluate this, we randomized the transcript selection in PPIXpress as explained in Section 3.3.3. On average only 1.55 ± 0.038 transcripts were expressed per protein of the BioGRID network in each sample and 1.58 ± 0.040 in IntAct. Unsurprisingly, the number of discrete domain assemblies among those expressed transcripts was even smaller (Tables S3/S4) and they mostly resembled the domain composition of the longest-coding transcript (Tables S5/S6), since the principal domain composition often remains consistent among different isoforms [297].

An overview of the network sizes during the construction phase and some statistics are provided in Supplementary Tables S7-S9. Using either gene- or transcript-based filtering of the input PPIN, the normal to tumor conversion was accompanied with a net loss of around 130-150 proteins and 900-1,200 interactions, depending on the reference PPIN (compare respective rows in Tables S7/S8). In line with expectations, the networks constructed by the gene-based approach were always the largest ones and had on average around 20 proteins and 800 interactions more than networks built from transcript data using the ALL-DDI dataset, for example. Furthermore, the sizes of networks built using transcript-based approaches slightly decreased with the amount of DDI data involved (compare columns in Tables S7/S8 and see Table S9 for a statistical evaluation). Networks constructed with transcript resolution and the largest dataset ALL-DDI possessed on average around 10 proteins and 450 interactions less than those built using the smallest dataset $\mathrm{PRE}_{\mathrm{VHC}}$. Both observations can be accounted for by the subnetwork property of our transcript-based construction that we explained in Section 3.3.1. Since the sizes of the constructed networks were similar to results by [168] who considered 12,669 protein-coding genes to be expressed in healthy breast tissue, we deem our expression threshold to be suitable. Independently of DDI data considered, all transcript-based methods detected more rewiring events across the individual matched sample pairs and overall a higher number of significant changes in interactions compared to the gene-based approach (see first two rows per network in Table 3.3 and respective rows in Tables S7/S8). In the case of the ALL-DDI dataset, including transcript data into PPIXpress allowed to detect 357 additional significant rewiring events in BioGRID and 120 additional events in the IntAct network (compare respective columns in Table 3.3).

Hallmarks of cancer

We first checked how many of the significantly rewired interactions per construction method affected proteins that can be related to hallmarks of cancer (see Section 3.3.3 for definition). Table 3.3 shows aggregated results for this analysis. Details for the individual hallmark terms are given in Supplementary Tables S10/S11. A statistical assessment of the differences between the methods on the basis of Wilcoxon signed-rank test is made in Supplementary Tables S12/S13. Overall, a construction based on transcript expression was, independently of the reference PPIN and DDI dataset used, able to find a significantly larger number of differential interactions that could be associated with hallmarks of cancer than the gene-level approach ($p < 0.001$ in all cases, see first column in Table S12). For example, of the statistically relevant rewiring events revealed by the transcript-based construction on the basis of the ALL-DDI dataset 315 more interactions indeed affected at least one protein associated with any hallmark term compared to the interactions detected by gene expression and 103 more in IntAct, respectively (compare third row per network in Table 3.3 and Tables S10/S11). The fraction of such interactions among all differential interactions was not significantly higher (see first column in Table S13). Thus only the amount but not the density of relevant information was higher compared to a gene-based construction. With few exceptions this also held for individual hallmark terms (see fifth row per network in Table 3.3 for mean values and Tables S10/S11 for details). Regarding the absolute number of interactions affecting certain protein sets, only in 'Genome Instability and Mutation' and 'Avoiding Immune Destruction' in IntAct some of the transcript-based runs

	GENE	ALL-DDI	PRE_{VHC}	RANDOM(ALL-DDI)
BioGRID				
P_{rew}	0.067 ± 0.016	0.069 ± 0.017	0.068 ± 0.017	0.078 ± 0.017
sign. rewired interactions	9,754	10,111	10,022	$8,661 \pm 55$
part. in any hallmark term	7,028	7,343	7,273	$6,265 \pm 42$
fraction in any hallmark term	0.721	0.726	0.726	0.723 ± 0.002
avg. part. per hallmark term	1,749	1,841	1,820	$1,529 \pm 15$
avg. fraction per hallmark term	0.179	0.182	0.182	0.177 ± 0.001
IntAct				
P_{rew}	0.077 ± 0.019	0.079 ± 0.020	0.078 ± 0.019	0.086 ± 0.018
sign. rewired interactions	5,184	5,304	5,280	$4,783 \pm 25$
part. in any hallmark term	3,484	3,587	3,571	$3,168 \pm 25$
fraction in any hallmark term	0.672	0.676	0.676	0.662 ± 0.002
avg. part. per hallmark term	808	835	834	704 ± 9
avg. fraction per hallmark term	0.156	0.157	0.158	0.147 ± 0.001

Table 3.3: *Results for the rewiring analysis of the breast cancer vs. normal interaction networks in terms of rewired interactions that affect proteins associated with hallmarks of cancer as defined by [251]. The rewiring of an interaction was defined as significant according to the statistical protocol described in Section 3.3.3. An interaction was said to participate in a hallmark term if one of its associated proteins belonged to the corresponding set of hallmark proteins. The rows labelled "fraction" depict the relative proportion of hallmark-associated interactions among all detected interactions. Comprehensive results for the individual hallmark terms and all DDI datasets are provided in Supplementary Table S10 for the BioGRID network and in Table S11 for IntAct, respectively.*

performed slightly worse than the gene-based method (see respective rows in Table S11). The runs based on the highest-confidence DDI dataset PRE_{VHC} never reported fewer interactions in any term category, though.

When the transcript per protein was randomized as explained in Section 3.3.3, the transcript-based analysis gave worse results than the gene-based and all non-randomized transcript-based approaches (Tables S10-S13). In particular, significantly less hallmark-relevant interactions were detected ($p < 0.001$ in all cases, last row in Table S12). The difference in relevant fractions per hallmark term was not significant, though (see last row Table S13). Interestingly, it still found more interactions related to 'Enabling Replicative Immortality' in both reference networks than the gene-based methodology (see respective row in Tables S10/S11). As this was the only example in all analyses where the randomized method was superior to the gene-based one, we examined in what regard the proteins in that set differed from all others, and how the gene-based method could lose that much predictive power there. There was no noteworthy difference in the coverage of the interactions among this subset of proteins but an increase in protein coverage compared to all other hallmark sets with 82% in BioGRID (avg. hallmark proteins: 68%, details in Table S1) and 76% in IntAct (avg. hallmark proteins: 63%, details in Table S2).

	GENE / ALL-DDI	GENE / PRE_{VHC}	GENE / RANDOM(ALL-DDI)
BioGRID			
common rew. interactions	9,665	9,716	$8,308 \pm 7$
exclusive rew. interactions	89 / 446	38 / 306	$1,445 \pm 7$ / 352 ± 54
affected proteins	117 / 424	58 / 326	$1,401 \pm 5$ / 344 ± 47
KEGG PWC	1.0 / $1 * 10^{-17}$	1.0 / $5 * 10^{-15}$	$(2 \pm 1) * 10^{-10}$ / $(1 \pm 8) * 10^{-7}$
enriched terms (KEGG)	2 / 34	1 / 19	16 ± 1 / 18 ± 5
highest enrichment (KEGG)	0.0013 / $2 * 10^{-17}$	0.0037 / $9 * 10^{-16}$	$(2 \pm 0.1) * 10^{-9}$ / $(1 \pm 8) * 10^{-10}$
enriched terms (GO BP)	6 / 116	0 / 87	108 ± 4 / 97 ± 15
highest enrichment (GO BP)	$7 * 10^{-5}$ / $4 * 10^{-16}$	1.0 / $6 * 10^{-16}$	$(3 \pm 1) * 10^{-12}$ / $(5 \pm 0.4) * 10^{-12}$
IntAct			
common rew. interactions	5,118	5,155	$4,653 \pm 16$
exclusive rew. interactions	66 / 186	29 / 125	531 ± 16 / 130 ± 17
affected proteins	87 / 213	46 / 145	571 ± 8 / 150 ± 17
KEGG PWC	1.0 / $9 * 10^{-11}$	1.0 / $4 * 10^{-6}$	$(9 \pm 9) * 10^{-6}$ / 0.12 ± 0.32
enriched terms (KEGG)	0 / 15	0 / 11	21 ± 1 / 7 ± 4
highest enrichment (KEGG)	1.0 / $4 * 10^{-13}$	1.0 / $4 * 10^{-8}$	$(4 \pm 5) * 10^{-13}$ / $(3 \pm 0.1) * 10^{-6}$
enriched terms (GO BP)	0 / 35	1 / 20	26 ± 2 / 13 ± 8
highest enrichment (GO BP)	1.0 / $3 * 10^{-6}$	0.0367 / $3 * 10^{-5}$	$(6 \pm 0.1) * 10^{-12}$ / $(4 \pm 8) * 10^{-4}$

Table 3.4: *Shown are significantly rewired interactions that were exclusively found either by the gene- or transcript-based methods and the proteins that are affected by them. In all but the first lines per network, left values denote the outcome regarding interactions and proteins exclusively found in the gene-based approach and right values the same for the transcript-based construction. Affected proteins are proteins linked to significantly rewired interactions. Enrichment was determined according to Section 3.3.3 and defined as* $p < 0.05$ *(Bonferroni-adjusted). KEGG PWC abbreviates the enrichment of KEGG pathway 'hsa05200:Pathways in cancer' and GO BP abbreviates the GO category 'biological process'. Comprehensive results for all DDI datasets are provided in Supplementary Table S14 for the BioGRID network and in Table S15 for IntAct, respectively.*

Thus comparatively many proteins had at least one annotated domain that contributed to the DDI/PPI mapping, but the majority of the networks was still covered by artificial domains. Furthermore, the proteins associated with 'Enabling Replicative Immortality' had the most variable domain compositions among the expressed transcripts per protein in both BioGRID (8% more than any other protein subset, see Table S3) and IntAct (6% more than any other protein subset, see Table S4). Intriguingly, they also had the smallest fraction of expressed transcripts per protein that had the same domain composition as the principal protein isoform (2.5-3.4% smaller than any other protein subset, see Tables S5/S6). Consequently, the proteins in 'Enabling Replicative Immortality' had the largest divergence from the principal domain composition among all protein subsets that we examined and thus behaved most different compared to the gene-based construction.

Enrichment of exclusively found interactions

Next we examined for all transcript-based variants one-by-one if significant changes were missed compared to a gene-based construction and which alterations were found in addition. To quantify the relevance of rewiring events exclusively reported by either method, enrichment analysis was performed on the affected proteins (see Section 3.3.3). An outline of the results is presented in Table 3.4, details are listed in Supplementary Tables S14/S15.

Generally, the results originating from gene- and transcript-based construction methods diverged more strongly the more DDI data was incorporated (see first rows in Tables S14/S15). While a larger DDI dataset enabled transcript-based approaches to detect more interactions that were not considered by the gene-based adjustment, also more interactions that were detected by the gene-based approach were not detected (see second row per network in Table 3.4 and respective rows in Tables S14/S15). In all cases, the exclusively found significant changes revealed by network construction based on transcripts featured many more enriched pathways and GO processes and also a much higher enrichment of individual terms compared to the portion of interactions that were only found by the gene-based approach. KEGG term 'hsa05200:Pathways in cancer', for example, was not enriched in the exclusive results of the gene-based approach but strongly in those of the transcript-based method, independent of the DDI dataset used (adjusted $p < 10^{-5}$ in all cases, see fourth row per network in Table 3.4 and respective rows in Tables S14/S15). It is worth pointing out that the identified enriched terms are closely linked to carcinogenetic processes suggesting that the rewired interactions are not simply random alterations overall (Tables S16/S17). The most prevalent changes exclusively found by the transcript-based method using the largest DDI dataset, for example, were found across 66 matched samples in both networks. Four of the five exclusively found rewiring events across both networks that occurred 66 times were related to the loss of an interaction of FLT1 (P17948, 'Vascular endothelial growth factor receptor 1'), a tyrosine-protein kinase that acts as a cell-surface receptor for several cancer-relevant signaling cascades [285].

When the transcripts used to construct the specific DDIN were randomized, the positive impact of the transcript-level data vanished in comparison to the established gene-based methodology (see last column in Table 3.4).

3.5 CONCLUSION

PPIXpress exploits domain interaction data to adapt protein interaction networks to specific cellular conditions at transcript-level detail. For the example of protein interactions in breast cancer we showed how this increase in granularity positively affected the performance of the network construction compared to a method that only makes use of gene expression data. A platform-independent and dependency- as well as installation-free implementation is provided that only requires little manual effort by the user.

3.6 ADDENDUM

3.6.1 *Updates of PPIXpress*

As of the end of 2019, around three years past the initial release of PPIXpress, 22 updated versions of the software were made available. The graphical user interface of PPIXpress 1.20 is shown in Figure 3.4. This chapter describes the feature set of version 1.01 that already included several helpful suggestions made by reviewers of the original manuscript. Their remarks on the software included adding the possibility to set expression thresholds according to percentiles, automatic retrieval of 3did and iPfam interaction data to be on par with current knowledge in that regard, or the option to input protein interaction networks using Ensembl or HGNC gene identifiers. Later updates then added support for direct usage of transcript- and gene-level quantification outputs by the popular tools RSEM [73] and kallisto [74], as well as expanded automatic retrieval of interaction data by including the databases IRefIndex [298] and mentha [161]. These sources are especially valuable in practice since they are updated regularly (mentha even weekly) by integrating the most current data of many established databases. Since the service is discontinued, the latest public data of iPfam [108] is now included in each release of PPIXpress. Furthermore, UniProt accessions can be automatically updated to their current primary accession[1]. This is important because input data may not be using the most recent identifiers of a protein.

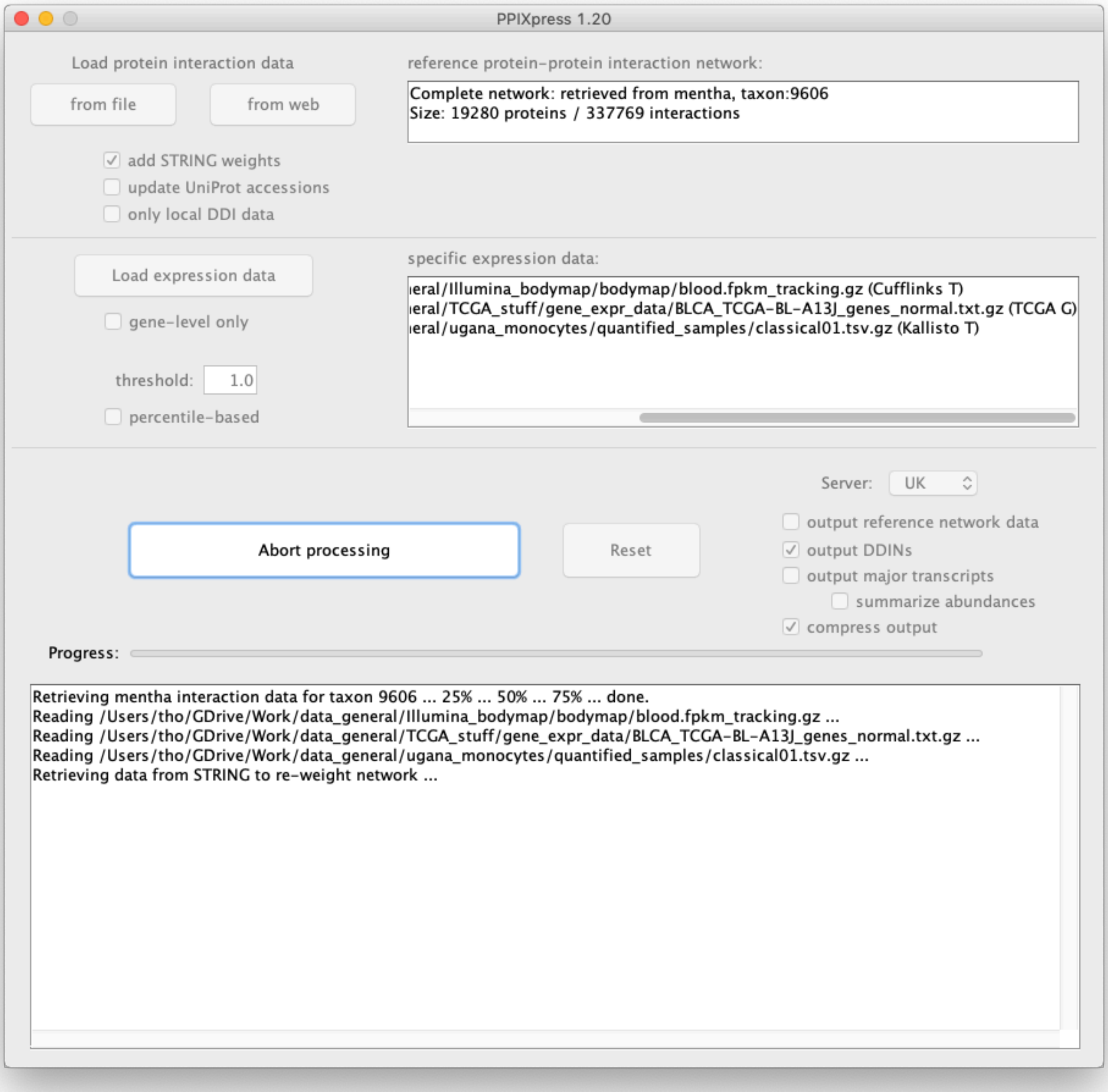
With version 1.12 we added the retrieval of Ensembl biotype definitions for each individual transcript to account for mRNA surveillance mechanisms. Because they will not yield viable protein products [2], proteins represented by transcripts that are tagged with biotypes ″nonsense-mediated decay″ or ″non-stop decay″ are by default withdrawn from calculations and thus not included in the constructed network.

As a suitable basis for follow-up tools, such as PPICompare (see Chapter 4) and CompleXChange (see Chapter 6), the abundance of the most abundant transcript (or the sum of all expressed transcripts coding for the protein) is reported in the output of PPIXpress since version 1.15. Optionally, these abundance values can be normalized by transcript lengths (since version 1.18).

3.6.2 *On reweighting interactions rather than applying discrete cutoffs*

PPIXpress uses a discrete cutoff to discern proteins that are expressed from those that are not and only chooses the most abundant one out of all transcripts coding for a protein as its representative in terms of domain composition. A more natural approach would probably be to integrate the numerical values that we have on the abundances of each isoform (or more correctly the corresponding transcript, depending on the exact input data) to realize a continuous approach which weights each contribution in a biologically sound way. We decided against such a weighting for two reasons that were briefly discussed in Section 3.3.1. First of all, the most relevant protein isoform seems to be dominant anyhow in real biological samples [281–283]. Mixing this with minor

[1] See https://www.uniprot.org/help/accession_numbers.



reference protein–protein interaction network:

Complete network: retrieved from mentha, taxon:9606
Size: 19280 proteins / 337769 interactions

☑ add STRING weights
□ update UniProt accessions
□ only local DDI data

specific expression data:

ıeral/Illumina_bodymap/bodymap/blood.fpkm_tracking.gz (Cufflinks T)
ıeral/TCGA_stuff/gene_expr_data/BLCA_TCGA-BL-A13J_genes_normal.txt.gz (TCGA G)
ıeral/ugana_monocytes/quantified_samples/classical01.tsv.gz (Kallisto T)

□ gene-level only

threshold: 1.0

□ percentile-based

Server: UK

□ output reference network data
☑ output DDINs
□ output major transcripts
□ summarize abundances
☑ compress output

```
Retrieving mentha interaction data for taxon 9606 ... 25% ... 50% ... 75% ... done.
Reading /Users/tho/GDrive/Work/data_general/Illumina_bodymap/bodymap/blood.fpkm_tracking.gz ...
Reading /Users/tho/GDrive/Work/data_general/TCGA_stuff/gene_expr_data/BLCA_TCGA-BL-A13J_genes_normal.txt.gz ...
Reading /Users/tho/GDrive/Work/data_general/ugana_monocytes/quantified_samples/classical01.tsv.gz ...
Retrieving data from STRING to re-weight network ...
```

Figure 3.4: *The graphical user interface of PPIXpress* 1.20.

contributions could consequently dilute an actually pertinent biological signal. Second, given that the abundance of each individual isoform would also have an effect on their binding probabilities, the importance of each transcript's contribution to the domain composition would likely be best described statistically by some adjusted Boltzmann distribution which may require additional parameters that should be tuned [299]. Since there is no adequate data on something like that and the straightforward approach that we implemented is actually well compatible with biological observations, we decided against such a more complicated methodology in the original version of the software and manuscript.

Still, PPIXpress includes a function that actually uses the information on all transcripts and infers a reweighted sample-specific network. This function is not documented in the user guide and only accessible from the code base by the function *constructAssociatedWeightedNetworksFromTranscriptAbundance()* in the class *framework.NetworkBuilder*. The approach is simple and devoid of additional parameters that would require tuning: for each protein all expressed isoforms (or corresponding transcripts) contribute to a weighted domain composition rather than just taking the domain composition of the most abundant isoform. Domains are thus not only present or not, instead they have an empirical probability associated which we call the *domain prefactor*. For each domain found in the isoforms of a certain protein it is computed as

$$\frac{\sum \text{abundances of all isoforms of the protein containing the domain}}{\sum \text{abundances of all isoforms of the protein}}.$$

The domain prefactors of two interacting domains can then be multiplied to obtain the factor for reweighting the according interaction of the PPIN. If more than one DDI of the sample-specific data can support the reference PPI, the one with the largest weight and therefore the most likely option is chosen to annotate the output protein interaction. Figure 3.5 clarifies the idea with an example. Here, the interaction of A and B could be supported either by the DDI of the green and orange domains or between the red and purple domains.

As a final note, the notion of a weighted domain composition is only a statistically averaged representation of this ensemble and not linked to any biological entity. The latter would be indeed the case for the default behavior of using the domain composition of the most abundant isoform.

3.6.3 *Updates regarding related research*

The first experimental study on a larger scale on isoform-specific protein interactions [7] was published shortly after the PPIXpress manuscript was accepted. For the paper [7], the interactomes of 366 protein isoforms encoded by 161 genes were profiled and assessed against a library of 13,000 genes which the group established in earlier work [137]. The results showed that the inclusion of isoforms in the search for protein interaction partners led to a remarkable 3.2-fold increase in the number of interactions, less than a third of all interactions found in the screening were exerted by reference isoforms. Strikingly, different isoforms of the same protein could even have completely different interaction

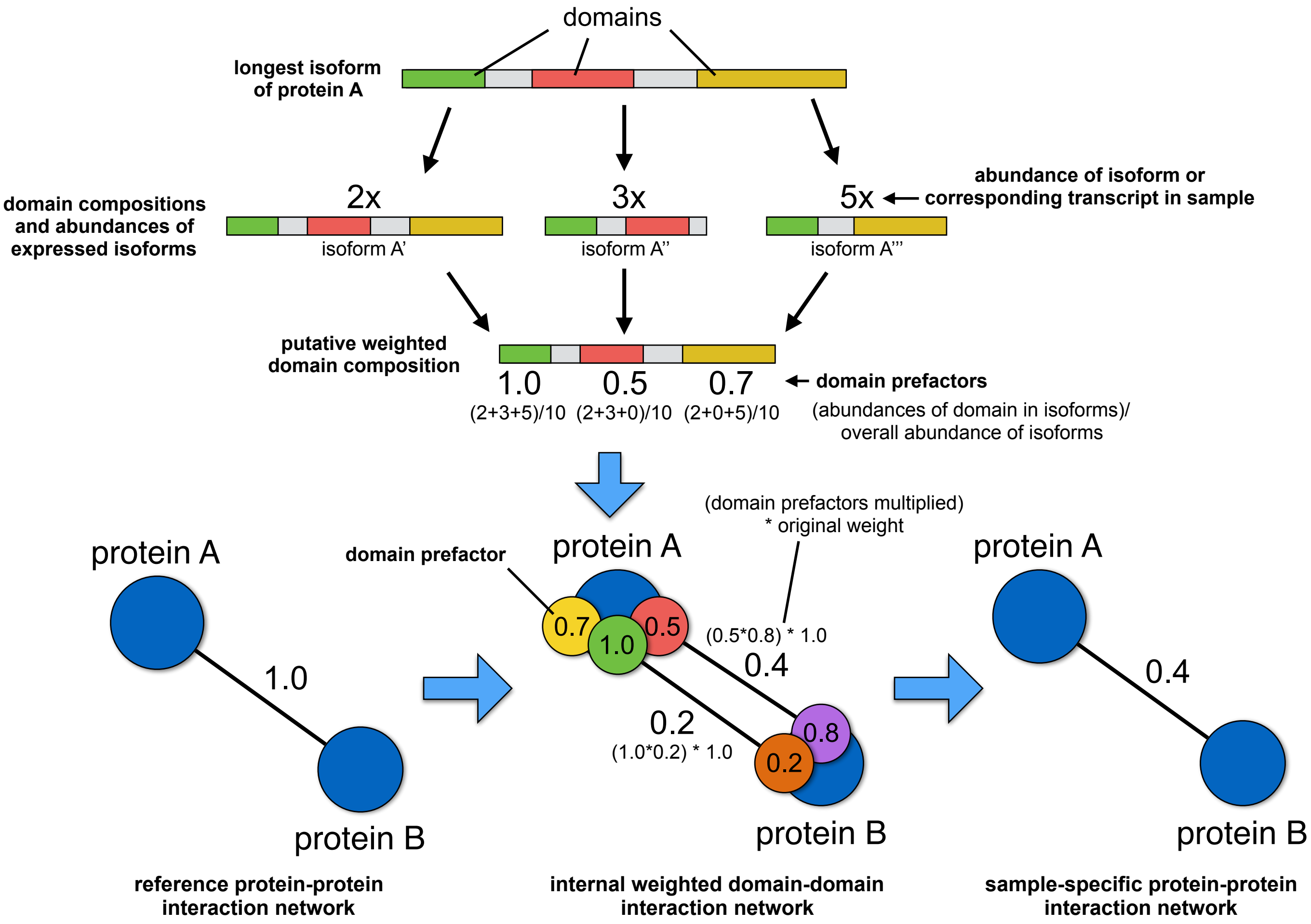

domains
longest isoform of protein A
domain compositions and abundances of expressed isoforms
2x
isoform A'
3x
isoform A"
5x
isoform A'"
abundance of isoform or corresponding transcript in sample
putative weighted domain composition
1.0
(2+3+5)/10
0.5
(2+3+0)/10
0.7
(2+0+5)/10
domain prefactors
(abundances of domain in isoforms)/ overall abundance of isoforms
protein A
1.0
protein B
reference protein-protein interaction network
domain prefactor
protein A
0.7
1.0
0.5
(domain prefactors multiplied) * original weight
(0.5*0.8) * 1.0
0.4
0.2
(1.0*0.2) * 1.0
0.8
0.2
protein B
internal weighted domain-domain interaction network
protein A
0.4
protein B
sample-specific protein-protein interaction network

Figure 3.5: *Example for the simple interactome reweighting implementation in PPIXpress. Here, a protein A has three expressed isoforms in a sample which all comprise a different set of domains (marked by different colors) and have individual abundances. Isoform A′ is found 2 times, A″ is found 3 times and A‴ is found 5 times. From this information a weighted domain composition is calculated that is then used to determine prefactors for reweighting the original weight annotations of the input reference PPIN.*

partners. In such cases, there was a strong tendency towards differences within disordered regions of the isoforms rather than changes of conserved sequence regions like domains, though. The loss of annotated protein domains, like those used by PPIXpress, could indeed often explain the directed loss of specific interactions which supports the core idea behind PPIXpress. A similar behavior was observed for the loss of short linear motifs in sequence regions which accumulated such otherwise relatively unspecific motifs.

On the basis of this first experimental dataset [7], it was later additionally shown by another group that mapping domain interactions in the fashion of PPIXpress allows for a reasonable approximation of the isoform-specific interactome [207].

3.6.4 *Outlook*

PPIXpress in its current design solely uses information on Pfam domains to infer changes to the protein interactome from transcripts or protein isoforms expressed in a sample. As we already discussed broadly in Section 3.4.1, using the established methodology and wealth of data on Pfam domain families as well as their interaction preferences was a conceptually safe decision that, however, still left room for improvement in terms of relating as much of the input protein interactome to domain interactions as possible. While this coverage is dependent on the protein interactome assessed, see Table 3.2 but also the more recent data in Table 5.1, by their very nature Pfam domain annotations inherently lack coverage of less conserved or even disordered regions of the proteome [111]. Extending PPIXpress to include another type of sequence-based descriptive motif could improve this coverage considerably. First of all, this motif type needs to be known to facilitate protein interactions and there should be a valuable amount of data available, and second, it should at best be rather complementary to Pfam in the sense that it is less likely to be biased towards rather conserved sequence regions.

Given these requirements and the recent indication of their relevance in interaction rewiring by splicing [7], short linear motifs, which are contiguous amino acid modules in proteins that can be specified and identified comparably simple by using regular expressions [300–302], should be very promising candidates for such an extension. There are plenty of rich resources on short motifs like especially the ELM (eukaryotic linear motif) database [303] but also other databases [304–306] and more general approaches working on short peptides exist [307]. As intended, they often cover less conserved or even disordered regions [308] and may thus enable PPIXpress to get a grasp on interactions mediated by such regions as they are often of importance in crucial cellular control mechanisms [45, 95, 96, 296]. Because linear motifs are very short they are, unlike Pfam motifs, innately prone to false positive predictions [300]. Since the input PPIN serves as the template for potential motif interactions that are allowed by the PPIXpress methodology, which means we already filter knowledge-based for valid protein pairs, then the chance of false information should be decreased considerably. This needs to be assessed and confirmed, though.

Apart from the rewiring due to changes in the transcriptome, the edge-specific adaption of the interactome is also a hot topic in the context of mutations [153, 279, 280]. Because PPIXpress' essential merit is to relate the change in genomic regions to individual protein interactions, it appears obvious to then transfer and apply this core idea to the mapping of mutations and interactions. Classical tools like SIFT [309] or PolyPhen2 [310] aim to predict if a mutation affects the general function of a protein, so they only present a node-centric view of the issue rather than an outcome specific for individual interactions. Only newer tools that rely on structural data, like dSysMap [311] and StructMAn [312], are currently inferring effects on specific protein interactions. This is a gap that could be filled by the data integration approach of PPIXpress.

My then student and now office colleague Andreas Denger applied the PPIXpress model in his bachelor thesis "The effects of genetic mutations on protein interaction networks" to study if interaction-specific effects of non-synonymous single nucleotide polymorphisms can be predicted via relating protein domains with individual protein interactions. Hereby he used the mapping scheme of PPIXpress (see Figure 3.2) to decide which domains are relevant for protein interactions. Then, the change in amino acid properties (physicochemically, BLOSUM100 score [313]) as well as SIFT and PolyPhen2 predictions were used to declare a mutation as deleterious or neutral for the interaction(s) affected. He could use a dataset by Sahni et al. [153] that was relatively recent at that time to test if this method would be beneficial and found that the added information could indeed increase the accuracy of predicting alterations to individual interactions. When PolyPhen2 predictions alone were used to decide if a mutation has an effect on the protein affected, for example, the percentage of successfully classified edge deletions increased by 23% when the information on the specific edge by the mapping of PPIxpress was added compared to just applying the alterations to all interactions of the affected protein.

4 REWIRING OF THE PROTEIN INTERACTOME DURING BLOOD DEVELOPMENT

This chapter describes the tool PPICompare that enables differential analysis of protein-protein interaction networks in a way that also describes transcriptomic alterations causing rewiring events. We used the tool to analyze developmental transitions in hematopoiesis. Sections 4.2 to 4.5 were adapted and expanded from Will, T. and Helms, V., "Rewiring of the inferred protein interactome during blood development studied with the tool PPICompare", *BMC Systems Biology*, 2017 [92]. I initiated this project and the study, designed and implemented the software, performed data analysis, conceived the figures and wrote the original manuscript. Volkhard Helms aided in designing the study, interpreting the data as well as editing of the manuscript. Supplementary materials that are published were omitted here, please refer to the online materials `https://doi.org/10.1186/s12918-017-0400-x`. A platform-independent implementation of the tool PPICompare is available at `https://sourceforge.net/projects/ppicompare/`.

4.1 PREREQUISITES

4.1.1 *On clustering data*

Clustering or cluster analysis is the task to group together samples of data points so that samples within the same group are more similar to each other in terms of a certain distance function than to samples in other groups. The term and first applications date back to the 1930s with origins in anthropology [314] and psychology [315, 316]. Since no labeling of the samples is needed in clustering, the procedure belongs to the class of unsupervised learning methods. For an example of a supervised method, see Section 6.1.2 on the classification of data.

There are diverse approaches to clustering data that, due to their very different views on the matter, basically solve alternative definitions of the problem [317]. Classical types of methods are, for example, based on centroids where clusters are modelled by central points to which its members are nearest, like in the popular k-means algorithm [318, 319]. Clustering can be done distribution-based where statistical distributions define the likeliness of a cluster assignment [320] or density-based where tools such as DBSCAN [321] connect dense regions of the data into clusters. Furthermore, there are graph-based approaches, like spectral clustering [322], in which the samples are modelled as the vertices of a graph whose edges describe the distances between samples and graph theory is used to assign clusters. Hierarchical clustering approaches are probably the most popular class of methods. Since we applied such a clustering in the following project, I will go into more detail with this concept.

On hierarchical clustering

Hierarchical clustering (HC) is a very flexible framework for clustering data that can be adjusted to the task at hand by plugging in diverse functions: a distance function (also called metric) $d(a, b)$ that depicts how (dis)similar two samples a and b are and a linkage criterion $L(A, B)$ that uses $d(a, b)$ to determine a linkage distance between two sets A and B of samples.

Compared to other clustering approaches, HC is very appealing due to its flexibility of distance and linkage functions, its conceptual simplicity and because it captures modularity in the data well, e. g. it reports subclusters in clusters and not only clusters that are disjoint. The latter feature can also be seen as a disadvantage because a minimal amount of manual inspection and interpretation is required because various cluster assignments are reported rather than a single automatically derived partitioning of the data that tells the user where an individual cluster starts and ends [320, 323]. The output of HC is called a dendrogram and depicts the hierarchy of groupings as a tree. See Figure 4.4 for a real-world example of a dendrogram derived by such a method.

HC can be performed using two very distinct strategies. Clustering can be conducted “bottom-up" by successively merging pairs of clusters whereby each element is initiated as its own cluster when starting (*agglomerative*), or “top-down” by placing all elements in one cluster and performing successive splits (*divisive*). Divisive clustering is usually only applied when one is interested in a small number of clusters in huge datasets, agglomerative clustering is generally the more common approach [320]. Pairwise distances $d(a, b)$ for all samples are either already the input for the approaches or are precomputed once. Merging and splitting are then decided on the basis of the distance between clusters $L(A, B)$. In agglomerative clustering the most similar cluster pair is merged in each step until only one cluster is left, whereas divisive clustering works reversely and thus always splits the most distant candidate pair.

Table 4.1 lists commonly used metrics applied in HC. In principle every distance function on the feature vector of the data can be used for clustering provided that it complies with the type of data in the description of the features. This is important in this respect because descriptors are not limited to numerical data. Furthermore, it should be noted that different functions can have different requirements regarding their preprocessing. There may be a need for normalization of the data, for example [323].

The three classical linkage criteria are shown in Table 4.2. There are more sophisticated choices, like Ward’s method which tries to minimize the within-cluster variance in each merging step [324], and while all linkage functions have specific strengths and weaknesses depending on the structure of the data they are applied to, the elementary functions listed here are still the most relevant ones in practice [323].

In our project we used the UPGMA (unweighted pair group method with arithmetic mean) approach [327] with either the Correlation or the Hamming distance as the distance function, depending on the type of input data (continuous numeric / Boolean).

name	definition
Euclidean distance (L_2-norm)	$d_{euc}(a,b) = \sqrt{\sum_i^n (a_i - b_i)^2}$
Manhattan distance (L_1-norm)	$d_{man}(a,b) = \sum_i^n \lvert a_i - b_i \rvert$
Correlation distance	$d_{cor}(a,b) = 1 - \rho(a,b)$
Hamming distance	$d_{ham}(a,b) = \lvert\{i \in \{1,\ldots,n\} \lvert a_i \neq b_i\}\rvert$

Table 4.1: *Popular metrics used in HC. The upper three functions are defined for numerical data while the Hamming distance as shown here is intended to be applied to Boolean vectors. The Pearson correlation coefficient* ρ *is applied in the Correlation distance. Notably, there are various ways to define distance functions based on correlation. Often metrics are additionally normalized by the number of features* n.

name	definition
minimum / single linkage [325]	$L_{min}(A,B) = \min_{a \in A, b \in B} d(a,b)$
maximum / complete linkage [326]	$L_{max}(A,B) = \max_{a \in A, b \in B} d(a,b)$
average linkage (also called UPGMA) [327]	$L_{avg}(A,B) = \frac{1}{\lvert A \rvert \lvert B \rvert} \sum_{a \in A, b \in B} d(a,b)$

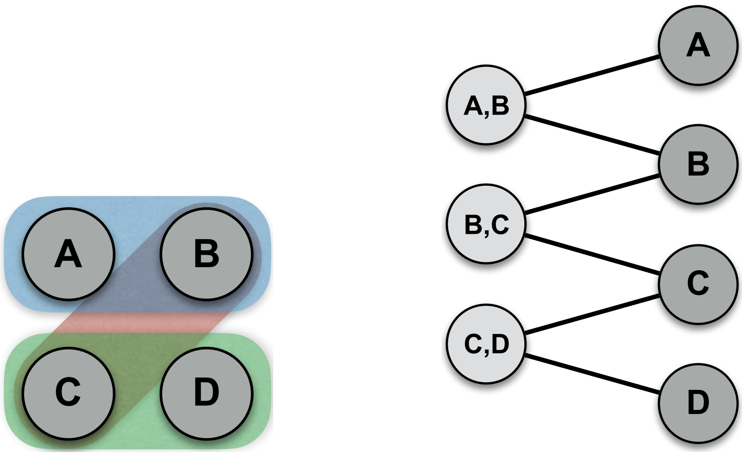
Table 4.2: *Popular linkage criteria in HC. Every valid distance function* $d(a,b)$ *can be used to compare the linkage between two clusters* A *and* B.

4.1.2 *On set-cover problems*

Given a finite set of elements X and a collection F of subsets of X such that $\forall x \in X, \exists f \in F : x \in f$, the set-cover problem is to find a minimum-size subset of F such that the union equals X [328]. The hitting-set problem is an equivalent reformulation of the set-cover problem in which a bipartite graph represents the subsets F as vertices on the left, all elements of X as vertices on the right and edges depict the memberships of elements in subsets. The objective is then to find the smallest subset of left-vertices such that all right-vertices are covered [329]. Figure 4.1 exemplifies an instance of the problem in the two formulations.

Optionally a weight $w : F \to \mathbb{R}^+$ can be introduced on the subsets F to define a weighted set-cover problem. Then the goal is to find the minimum cost subset of F that suffices to cover X [330].

Finding the optimal set-cover is one of Karp's 21 classical NP-complete problems [329]. While there is thus no algorithm that yields the optimal solution in polynomial time, greedy algorithms can provide good approximations for such tasks in polynomial time [328, 330]. Typically set-cover problems are solved using integer linear programming, a special class of linear programming (see Chapter 6.1.1), or by applying a fast heuristic approach.



(a) *set-cover formulation*   **(b)** *hitting-set formulation*

Figure 4.1: *Set-cover problem example. The subsets* $f \in F$ *are differentiated by their coloring in the set-cover formulation (a) while they are represented by the vertices on the left side in the hitting-set formulation (b).*

The classical greedy methods for set-cover problems will be introduced briefly here because they are quite straightforward and the weighted version is implemented in our software PPICompare. Both methods consecutively select one subset from F in each iteration until the union of selected subsets satisfies the set-cover criterion. The algorithm for unweighted problems repeatedly chooses the $f \in F$ that adds the largest number of yet uncovered members of X until all elements are covered [328]. For weighted set-cover problems, in each selection step the $f \in F$ is chosen that minimizes the cost $\frac{w(f)}{s}$ of adding an element to the cover instead whereby s here denotes the number of elements that are appended to the cover by the selection of f [331].

4.2 INTRODUCTION

Generally, every apparatus is better specified by the connection of its parts than by the sole list of parts. In the same way, the state of a cell is better described by the cooperative action of its active molecular machinery than by a simple list of its genome-encoded building blocks. Consequently, decades of research have gone into detecting physical interactions between proteins. Aggregating all this effort into comprehensive protein-protein interaction networks (PPINs) that represent the known protein interactome of an organism has been an important achievement [150, 275].

However, a static representation of the full interactome does not reflect its wiring in different tissues, cell types, diseases or any other arbitrary cellular state. Experimental data on protein-protein interactions (PPIs) in particular contexts is very limited and it is unclear whether its amount will increase substantially in the near future [150, 332]. Previous experimental studies typically focused on very specific issues, such as the perturbation of individual interactions by disease mutations [153, 279] or posttranslational modifications [333], and covered only small subsets of the proteome. The general lack of context-sensitive interactome data is commonly overcome by computational methods that integrate condition-specific gene expression data with the known PPIN of that organism so that at least the influence of that factor is considered on a genome-wide scale. A straightforward approach is to filter the PPIs to the protein-coding genes that are expressed in a certain condition. This strategy was applied before in the contexts of tissues and cell types [165–167, 334] and of diseases [168, 169, 335].

The aforementioned limitation concerning condition-specific experimental evidence on PPIs as well as its solution of integrating additional data equally apply to the study of alterations in molecular networks [336, 337]. Most biologically-motivated differential network methods, regardless whether they depict physical interactions between proteins or another kind of pairwise relation, utilize a data-type dependent correlation measure to assess rewiring [266, 338–340]. Other methods put a stronger focus on the topology of the networks [341] or additionally make use of heterogeneous ontology information [342]. Conceptionally, correlation of gene expression is a reasonable measure of pairwise association in the context of biological interactions between genes or corresponding proteins. In the very case of protein interactions, however, the notion does neither unveil which transcriptomic alteration caused a rewiring nor provide sufficient information to assess the implications of alternative splicing (AS) events. Although AS has a substantial effect on the wiring of the interactome [7, 45, 46, 51], it is not yet accounted for by any current computational approach. Appropriate consideration would require the integration of expression data with transcript resolution and a general model that is able to relate protein isoforms to specific interactome phenotypes.

We recently introduced PPIXpress [90] (see also Chapter 3), a PPIN contextualization method that enables users to account for the effect of AS events on the interactome based on transcript-level expression data. Using knowledge on the viable interactions between protein domains and the domain composition of protein isoforms, the method first relates each protein interaction in the full PPIN to an underlying domain interaction. Then it uses this correspondence to infer the condition-specific presence of PPIs given the protein isoforms indicated by the expression data. Non-transcriptomic effects on protein interactions are not covered by this approach. As an extension of this work, we propose here the differential PPIN tool PPICompare that compares the inferred interactomes between samples of two groups and tracks the cause of each alteration. The tool determines statistically significant between-group rewiring events and annotates each rewiring process with the underlying cause (one or both corresponding genes missing, or interacting domains missing due to differential transcript usage). Also, PPICompare constructs a small set of the most relevant alterations to the transcriptome that explain all systematic differences in the networks. A first application of the novel software is presented on the example of hematopoiesis [343] using data generated by the BLUEPRINT epigenome project [89, 344, 345]. To our best knowledge this work represents the first study of rewiring processes of the protein interactome during development with similar scope and granularity.

4.3 MATERIALS AND METHODS

4.3.1 *PPICompare*

PPICompare is currently designed to be used as an extension to our tool PPIXpress for constructing condition-specific protein interaction networks [90] but can also be applied to suitable input data generated in alternative ways. As basis for the subsequent analysis, contextualized PPINs are constructed with PPIXpress for each transcript expression sample. This is explained in detail in the subsection "Constructing blood cell interactomes" below.

Given two groups of condition-specific PPINs built from the same reference PPIN, PPICompare detects all interactions that are significantly rewired between samples of the groups. In [90] we presented the underlying principle of the statistical model and applied it to the special type of matched datasets in a case study on breast cancer. Here, we extended the methodology to arbitrary groups of networks and provide a stand-alone software tool for this type of analysis. In particular, it reports descriptive statistics of the actual reasons for each rewiring event and determines a small set of the most relevant alterations to the transcriptome that explain all systematic differences in the networks. All output is written to files in the format of node- and edge-attribute tables that can be imported into other tools like, for example, Cytoscape [346]. A platform-independent Java 8 implementation of PPICompare that is able to efficiently utilize current multi-core CPUs is freely available at `https://sourceforge.net/projects/ppicompare/`. A user guide and example data are provided together with a precompiled executable and the complete source code.

For both practical as well as biological reasons discussed in [90], PPIXpress only adjusts the presence of interactions according to the expression data but does not alter their weight annotations. Consequently, a differential analysis of the derived networks is done based on discretized information. While discretization always implies a loss of information, it also simplifies the state space of the problem considerably and it can deflate noisy data. Advantages and disadvantages of using discretized expression data are discussed in [347], for example.

Figure 4.2 outlines the individual steps in the workflow of PPICompare. The details of panels A) to C) are described in the following three paragraphs.

A) Examining the interactome differences between all inter-group pairs of samples

In the first stage of the differential analysis (Figure 4.2A), each sample in the first group is compared to each sample in the second group in terms of their PPIs. Ideally, a group of samples stands here for a representative distribution of interactomes for a condition under study. For every pairwise comparison i a differential network Δ_i monitors whether a particular interaction (u, v) between proteins u and v is only found in one of the two groups. PPICompare considers the first group as the reference system and the second group is compared to it. An interaction (u, v) that is exclusively found in the sample of the second group is thus noted as $\Delta_i(u, v) = +1$. Likewise, an interaction (u, v) lost in the second group is noted as $\Delta_i(u, v) = -1$. All N individual pairwise observations

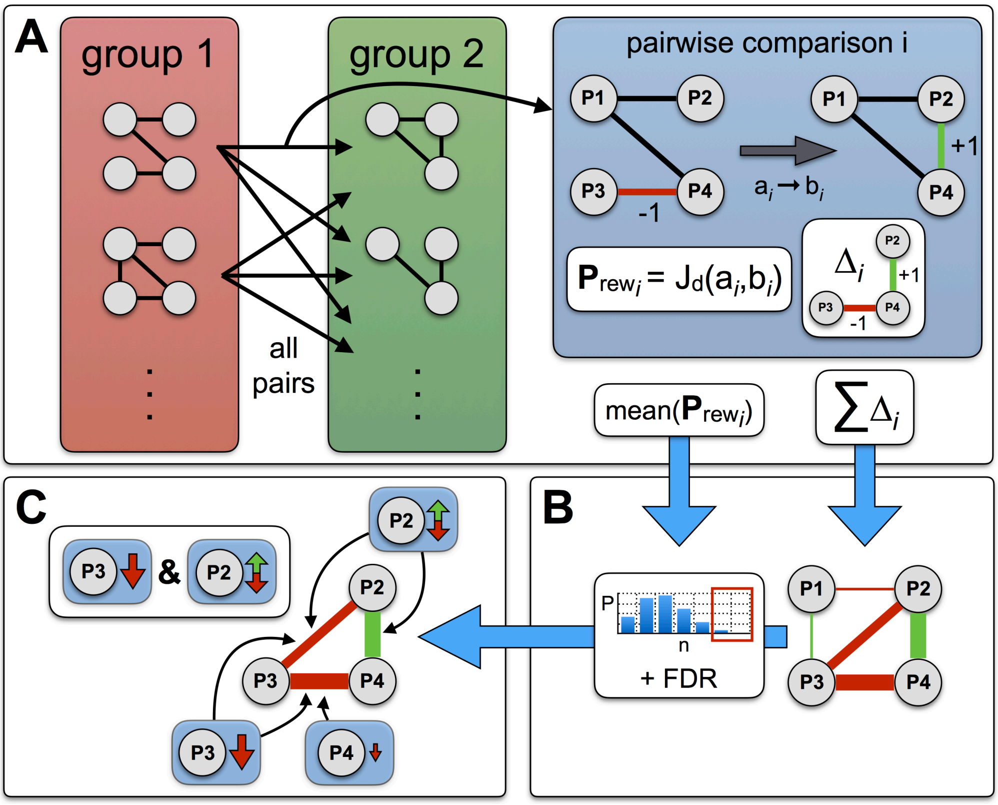


Figure 4.2: *Workflow of PPICompare. A) Examine the interactome differences between all inter-group pairs of samples. B) Assess the significance of and the reasons for each rewiring event. C) Discern a small set of likely changes in the transcriptome that explain the rewiring. Details are described in the main text.*

are weighted equally and summed up to obtain a global differential network Δ whereby each edge (interaction) is annotated with the signed number of changes affecting it in the inter-group comparisons: $\Delta(u,v) = \sum_i^N \Delta_i(u,v)$. As a result of this, rewiring events with opposing observations, where both addition and removal events were detected for the same interaction, are downweighted in a natural way. The unchanged portion of the interactome does not appear at all in the differential network. Potentially emerging null-sum annotated edges in the cumulative network Δ are removed after the summarization.

Besides tracking the amount of rewiring per edge, PPICompare quantifies the fraction of interactions that are changed in each pairwise comparison i by a rewiring probability P_{rew_i}. We defined P_{rew_i} as the number of rewired interactions normalized by the size of the union of interactions in both samples. This is basically the Jaccard distance [348] of the edge set. Thus $P_{rew_i} = 1 - \frac{|a_i \cap b_i|}{|a_i \cup b_i|}$, where a_i and b_i are the respective sets of interactions in the samples compared in comparison i. In the matched comparison scheme of [90] we used the number of interactions of the smaller one of both PPINs as a stringent normalization factor. Taking here the union of the corresponding interaction sets for normalization in the Jaccard distance allows application of the method to more variable non-matched data, because a value in $[0, 1]$ is ensured. Note that all pairwise comparisons are independent from each other. The final inter-group rewiring probability P_{rew} is then obtained as the average of all individual pairwise probabilities P_{rew_i}: $P_{\text{rew}} = \frac{1}{N}\sum_i^N P_{\text{rew}_i}$.

B) Assessing the significance of and the reasons for each rewiring event

P_{rew} can be interpreted as the probability of each interaction to be rewired. A one-tailed binomial test is then used to assess the statistical significance of candidate rewiring events (u, v) in the differential network Δ against this background (Figure 4.2B). For each candidate $(u, v) \in \Delta$ and a given P_{rew}, PPICompare computes the likeliness of observing at least the annotated number of rewiring events $|\Delta(u, v)|$ over all N pairwise comparisons by chance:

$$p_{(u,v)} = 1 - \sum_{i=0}^{|\Delta(u,v)|-1} \binom{N}{i} (P_{\text{rew}})^i (1 - P_{\text{rew}})^{N-i}.$$

The p-values are subsequently adjusted using the Benjamini-Hochberg procedure [236]. Only rewiring events below a user-defined false discovery rate (FDR) threshold are processed further and reported. Although a significance filter based on sufficient deviation from the background rewiring could be considered as naïve and very conservative, it represents a straightforward statistical model that ensures only reliable results are reported to the user.

Since version 1.05, PPIXpress can optionally report the major isoform that was associated with each individual protein during the construction of the condition-specific interaction network. As a consequence, PPICompare can use the output of PPIXpress to exactly reproduce and annotate which change or which changes in the transcriptome altered an interaction between samples of the two groups. Since each interaction depends on the presence as well as the compatibility of both interacting proteins, the two essential causes of rewiring events are either a major shift in the abundance of at least one interaction partner between groups (differential expression, DE), or a switch of the major isoform of at least one of the proteins that alters the domain composition in a way that affects the interactome (alternative splicing, AS). Whereas alterations to both proteins are in principle not necessary to explain changes to an interaction, even redundant pairs of causes are explicitly monitored by PPICompare because they could point to a different mode of regulation, such as the purposeful coexpression of complex partners. PPICompare determines and reports the individual distributions of all causal reasons for each significantly rewired interaction.

C) Discerning a small set of likely changes in the transcriptome that explain the rewiring

To identify the events that caused the systematic rewiring of the PPINs between the groups under study, it is reasonable to look for a set of transcriptomic changes that is both very likely given the data and of small cardinality.

The association of causes and affected interactions can be thought of as a bipartite graph, where one class of nodes are the significantly rewired interactions and the second class are individual causal reasons (change in expression or splice form of a single protein). In such a graph, the alterations point to the interactions they affect (see Figure 4.2C). Here, we tracked how often a transcriptomic cause i is relevant for each rewiring event. Thus, we know the number of pairwise comparisons pw_i in which the alteration happened and the number of significantly rewired interactions rw_i that were affected by it. Since the importance of a rewiring reason i should be related to its frequency across all comparisons and rewired interactions, we score each one with $s_i = pw_i \times rw_i$. Determining then a small set of those reasons that explain all rewiring events and consists of preferably important members is a weighted set-cover problem [330].

As this problem is classically defined as a minimization problem, we converted the scores s_i into weights w_i by setting $w_i = s_{max} - s_i$, where $s_{max} = \max(s_i) + 1$. The addition of one prevents the possibility of numerical equality and subsequent loss of information in the ratio that is then optimized. To efficiently solve this weighted set-cover problem for large instances, PPICompare implements a greedy algorithm with provable performance guarantees [331]. The algorithm repeatedly selects the rewiring reason i with the minimum ratio of w_i divided by the number of rewiring events that it additionally explains. This is done until all significant rewiring events are covered. The resulting solution set is part of the standard output of PPICompare.

Note that the notion of a reduced set refers here to the relevance in the interaction networks only. At a higher level, some crucial alteration which is not necessarily of transcriptomic origin and is simply not reflected in the differential interactome may, of course, reside upstream in the hierarchy of causal regulatory effects and thus be of more importance.

4.3.2 *Constructing blood cell interactomes*

Specific PPINs for samples of 11 hematopoietic cell types were constructed on the basis of transcript expression data from the 7th data release (Sept. 2015) of the BLUEPRINT epigenome project [89, 344, 345]. From the provided preprocessed data of the consortium we considered all samples of blood stem cells and precursors derived from cord blood and all samples of common mature cell types derived from venous blood that had at least 3 samples for this tissue of origin. The downloaded data included RNA-seq data on hematopoetic stem cells (HSCs, 6 samples), multipotent progenitors (MPPs, 3 samples), common myeloid progenitors (CMPs, 3 samples), common lymphoid progenitors (CLPs, 5 samples), megakaryocyte erythrocyte progenitors (MEPs, 4 samples), granulocyte monocyte progenitors (GMPs, 3 samples), erythroblasts (EBs, 7 samples), and megakaryocytes (MKs, 5 samples). Regarding common mature cell types that met those criteria we obtained data for neutrophils (Ns, 10 samples), monocytes (Ms, 5 samples), and naïve CD4 T cells (CD4s, 8 samples).

For consistency, we followed the strategy used in [89] from which we took our input data and of others who investigated blood cell types during development [349–351]. Thus, we based our analyses on the ontological relationships defined by the classical dichotomy model of hematopoiesis [343, 352]. Although recent insights based on data from single-cell sequencing challenge this established

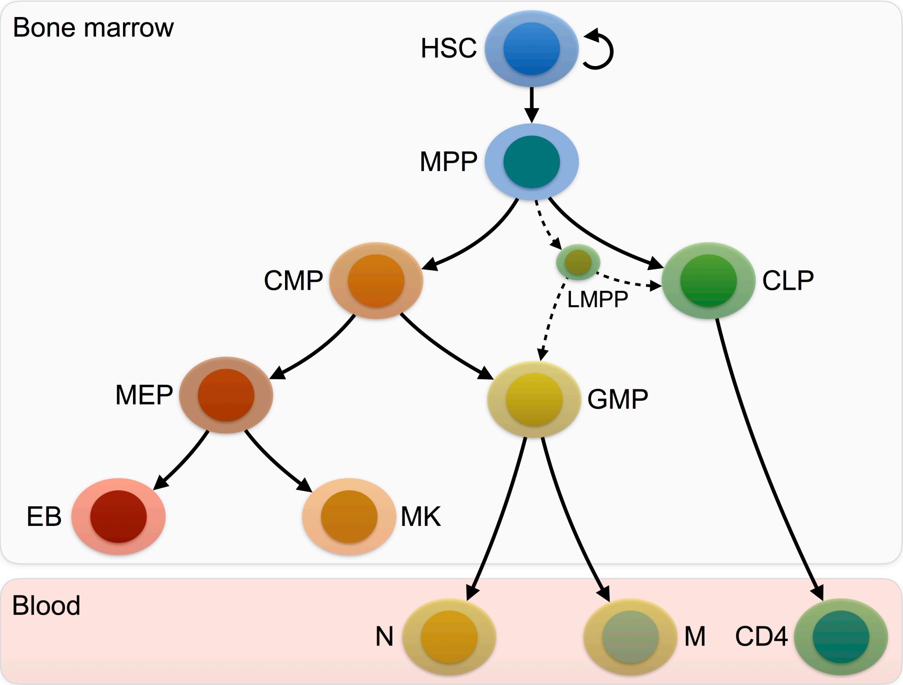


Figure 4.3: *Hierarchy of hematopoietic differentiation stages used as basis for our study. For reasons discussed in the main text, we only considered classical ontological relationships for all analyses (solid lines) and did not include more recent models and their accompanying novel entities. Lymphomyeloid-restricted progenitors (LMPPs, first proposed by [356]) are shown as an example for emerging relationships that are not covered by our data (dotted lines). In this model, MPP, CMP, MEP and GMP are developmental branching points and will be investigated in detail later.*

model of hematopoiesis, the model characterized by the BLUEPRINT data was not analyzed with respect to protein interactions so far and there appears to be no clear consensus on a revised model yet [353–357]. Figure 4.3 shows a schematic representation of the developmental relationships among the cell types we examined.

The preprocessed RNA-seq data of the 7th BLUEPRINT release was quantified with RSEM [73]. For better comparability between samples [358, 359], PPIXpress uses transcripts per million (TPM) as the relevant expression measure for RSEM output files. For all transcript expression samples we built protein interaction networks with PPIXpress (version 1.08) for a range of TPM thresholds from 0.0 TPM to 1.0 TPM in steps of 0.01. This means that only proteins with an associated transcript that was expressed above this cutoff were considered in the respective network contextualizations. Using PPIXpress, we retrieved the full protein interaction network for human (taxon 9606) from mentha [161] (data of 18. Jan. 2016). Outdated UniProt accession numbers (release 2015_12) [285] were updated automatically by PPIXpress. The resulting human reference protein interaction network contained information on 221, 158 physical interactions between 17, 292 proteins. Furthermore, PPIXpress retrieved annotation data from Ensembl (release 83) [360] and domain interaction data from 3did (release July 2015) [107] and iPfam (version 1.0) [108] for the mapping of protein interactions to domain interactions. With this data, 49.1% of the proteins in the reference PPIN were annotated with at least one domain that contributed to the PPI association. 20.3% of the PPIs were covered by domain interactions and thus may be potentially altered by AS events that our model can capture. Note that this partial coverage is in an expected range for domain annotations and domain-domain interaction data [90]. Interations that are mediated by disordered regions between such conserved domains are currently not considered by PPIXpress because they hardly comply with universally applicable annotation schemes. These practical limitations certainly confine the ability of the pipeline to detect the contribution of AS on the in-vivo rewiring of the proteome in its entirety. See [90] for more details concerning the methodology.

To establish a good TPM threshold, we utilized additional independent data on proteome abundance from the Human Proteome Map (HPM) [127] on individual hematopoietic cell types. We used their mass-spectrometry data on the abundance of proteins mapped to HGNC protein-coding genes [290] and considered each protein as present if its corresponding abundance value was larger than zero.

4.3.3 *Datasets and protocols in evaluation*

Participation in complexes, annotational homogeneity, and betweenness of interactions

To determine whether an interaction within a known complex is rewired, we downloaded the data on human protein complexes from CORUM (release Feb. 2012) [361] and checked whether interacting protein pairs belong to a known complex.

Furthermore, we annotated all interactions in our reference PPIN with the semantic similarity of the interactors concerning the three GO ontologies biological process (BP), molecular function (MF), and cellular compartment (CC) [294]. Semantic similarities were obtained using GOSemSim (version 1.28.2) [362] with default options and annotation data from org.Hs.eg.db (version 3.2.3) [363]. Also, we determined the betweenness of the interactions, which is the normalized sum of the fraction of all-pairs shortest paths that include this interaction. Betweenness values were computed with NetworkX (version 1.10) [364] on the basis of the reference PPIN.

Association and enrichment of rewiring events within pathways

We mapped deregulated interactions to the biological pathways they might affect. A related approach based on the coexpression between adjacent genes in pathways was proposed by [266] and termed Edge Set Enrichment Analysis (ESEA).

We retrieved preprocessed KEGG [365] and Reactome [366] pathway data as undirected graphs from the ESEA R package (version 1.0) [266] and converted the HGNC gene names to UniProt accessions using mapping data from the HGNC web service (accessed on March 26th, 2016) [290]. We followed the example of [266] and only considered pathways with at least 15 and at most 1,000 connections in the original pathway data. The remaining pathway-annotated links were then related to the exact interactions in our reference interactome data. 3, 394 PPIs (1.5% of our reference PPIN) among 1, 624 proteins (9.4% of our reference PPIN) could be exactly mapped to 106 KEGG pathways. 7, 318 PPIs (3.3% of our reference PPIN) among 2, 617 proteins (15.1% of our reference PPIN) corresponded to 495 Reactome pathways. Enrichment of pathways was calculated on the basis of a hypergeometric test as is often done for gene sets [367]. P-values were subsequently adjusted for KEGG and Reactome pathways independently using the Benjamini-Hochberg procedure [236]. Since PPICompare only distinguishes between rewiring events that are statistically significant and those that are not, the GSEA-based approach [233] of ESEA to identify pathway enrichment is not applicable for our task.

Unspecific enrichment analysis of deregulated proteins using DAVID

Unspecific protein-set enrichment analysis was conducted with the DAVID web service (version 6.7) [260] using default settings. We set all proteins in the reference PPIN as the background for the analysis. The reported significances of term enrichments refer to the p-values adjusted using the Benjamini-Hochberg correction [236].

Proteins relevant to hematopoiesis and their regulatory targets

As proteins relevant to blood development, we considered all human proteins annotated with GO term GO:0030097 (“hemopoiesis”) using QuickGO [241] on May 30th, 2016. In our reference PPIN this was the case for 480 proteins. We refer to these as “hematopoiesis proteins” in the remaining text. Additionally, we downloaded literature-curated annotations of experimentally validated gene regulatory relationships in human from the TRRUST database (version 12/08/2014) [368]. Due to the importance of distal regulatory interactions in eukaryotic development [13] and the confidence of the data, we preferred this database to more extensive data derived from proximal binding sites in gene promoters. The regulatory network contained data on 727 transcription factors (TFs) and 7, 906 interactions between proteins in the reference interactome. Among these TFs were 101 hematopoiesis proteins. Combining both data sources, 1, 274 proteins were either hematopoiesis proteins or proteins directly regulated by a hematopoietic TF.

Enrichment of a query regarding a specific protein set defined by this data was then determined using a hypergeometric test. As protein sets we analysed the combined set of hematopoiesis proteins and targets of hematopoietic TFs, the set of hematopoiesis proteins, and its subset of hematopoietic TFs.

Furthermore, we determined enrichment of targets associated with TFs covered by our regulatory data. Since the sets of targets of each TF were tested individually, the p-values for each TF were subsequently adjusted using the Benjamini-Hochberg procedure [236].

4.4 RESULTS AND DISCUSSION

Using PPIXpress and transcript expression data from BLUEPRINT we constructed the protein interactomes of 59 samples representing 11 different types of blood cells for different expression thresholds (see Methods section). To ensure that the biological analyses regarding developmental transitions were based on a single expression discretization parameter that best reflects the actual protein concentrations in the cell, we used mass spectrometry-based proteome abundance data from HPM [127] for guidance (see Supplementary Results S1.1). All further analyses presented were performed on the protein interaction networks constructed with the HPM-derived threshold of 0.31 TPM. Furthermore, we checked by a subsampling approach how robust the rewiring detection methodology was if only a small number of samples was available for comparison (see Supplementary Results S1.2). Apparently, groups with at least 3 samples provide meaningful results. As there is no computational pipeline with comparable features and scope, we did not contrast PPICompare with other tools.

4.4.1 *The rewiring of the blood interactome during development*

For a biological interpretation of the derived protein interaction networks, we compared all cell types that are adjacent in developmental progression according to the classical model of hematopoiesis as depicted in Figure 4.3. PPICompare (version 1.0) was applied to the corresponding PPINs generated with HPM-optimal threshold and the default FDR of 0.05. Table 4.3 summarizes the differences in the interactome sizes detected at developmental transitions.

Developmental branching points associated with lineage commitment are most distinct in terms of quantitative rewiring

Without a tool such as PPICompare, the average net difference in the number of interactions between proteins $\Delta n_i \to n_j$ (third column) is the only differential measure that can be analyzed. On its own, it provides little information on how many and no information on which interactions actually emerged or vanished during a conversion from i to j. For two of the four developmental branching points that were considered in our model of blood development (see Figure 4.3), the net difference even had a different sign depending on the direction of the transition in the branch. Interestingly, this was exactly the case when a bifurcation is passed that determines a lineage choice, namely, when descendant cells of MPPs either evolve toward the erythro-myeloid (MPP→CMP) or toward the lymphoid lineage (MPP→CLP) and, later in the developmental tree, when descendants of MEPs belong either to the erythroid (MEP→EB) or to the myeloid lineage (MEP→MK).

As a consequence of the high variance among the network sizes of most cell types, the standard deviation $\sigma(\Delta n_i \to n_j)$ was larger than its mean change for most developmental steps. We analyzed whether this within-group variance is an artifact from the network discretization. Yet, the interactome sizes showed a similar variability when all transcripts with non-zero expression

	protein interaction networks		PPICompare results			
transition	interactome sizes $n_i \rightarrow n_j$	$\Delta n_i \rightarrow n_j$	P_{rew}	obs_s/obs_{all}	rew_+/rew_-	$rew_+ - rew_-$
HSC→MPP	$101,235 \pm 30,315 \rightarrow 111,556 \pm 10,069$	$10,321 \pm 31,944$	0.372	15/18 (0.83)	311/123	188 (0.32σ)
MPP→CMP	$111,556 \pm 10,069 \rightarrow 117,254 \pm 3,176$	$5,698 \pm 10,558$	0.278	9/9 (1.00)	856/423	433 (0.50σ)
MPP→CLP	$111,556 \pm 10,069 \rightarrow 79,383 \pm 31,849$	$-32,173 \pm 33,402$	0.455	15/15 (1.00)	1/705	-704 (0.94σ)
CMP→MEP	$117,254 \pm 3,176 \rightarrow 117,768 \pm 8,692$	$513 \pm 9,254$	0.261	8/12 (0.67)	3,955/2,532	1,423 (0.10σ)
CMP→GMP	$117,254 \pm 3,176 \rightarrow 121,051 \pm 6,427$	$3,796 \pm 7,169$	0.256	6/9 (0.67)	8,468/5,556	2,912 (0.12σ)
MEP→EB	$117,768 \pm 8,692 \rightarrow 111,326 \pm 28,549$	$-6,441 \pm 29,842$	0.348	20/28 (0.71)	3,021/4,146	$-1,125$ (0.18σ)
MEP→MK	$117,768 \pm 8,692 \rightarrow 132,598 \pm 8,456$	$14,831 \pm 12,126$	0.293	12/20 (0.60)	10,574/3,848	6,726 (0.67σ)
GMP→N	$121,051 \pm 6,427 \rightarrow 67,007 \pm 9,203$	$-54,044 \pm 11,225$	0.585	24/30 (0.80)	3,895/41,599	$-37,704$ (1.46σ)
GMP→M	$121,051 \pm 6,427 \rightarrow 113,534 \pm 2,762$	$-7,517 \pm 6,995$	0.337	10/15 (0.67)	15,763/21,407	$-5,644$ (0.27σ)
CLP→CD4	$79,383 \pm 31,849 \rightarrow 120,282 \pm 19,498$	$40,898 \pm 37,343$	0.512	30/40 (0.75)	17,181/1,919	15,262 (0.69σ)

Table 4.3: *Quantitative changes of blood interactomes during developmental transitions. The net change in number of interactions $\Delta n_i \rightarrow n_j$ is reported as the mean difference between all samples per cell type and its standard deviation. obs_s is the minimum number of rewired observations out of all pairwise comparisons obs_{all} that were necessary for a rewiring event to be called significant in PPICompare applied to that transition. For increased comparability, the fraction as a floating-point number is given in brackets. The number of rewiring events deemed significant by PPICompare is depicted as rew_+ for emerging interactions and rew_- for vanishing interactions. In addition to the net change among significant rewiring events, its absolute deviation to $\Delta n_i \rightarrow n_j$ in terms of standard deviations $\sigma(\Delta n_i \rightarrow n_j)$ is shown in brackets.*

(equivalent to a TPM threshold of 0.0) were presumed abundant for each cell type instead of the stricter threshold used in the analyses (see Supplementary Table S1). Furthermore, hierarchical clustering of the original expression data was not able to distinguish the progenitor cell types properly (see Figure 4.4A). Thus, the high variability seems inherent to the data. Besides, clustering on the basis of the inferred interactomes had problems to properly separate some other cell types (see Figure 4.4B) which were also grouped suboptimally when clustered by discretized expression data (see Supplementary Figure S1). Heterogeneity is common in this context because cell populations that were separated by specific surface markers often still contain hidden diversity in the form of subpopulations. Sample variability, but also the dilution of it, is therefore a general issue for averaged snapshots made in bulk measurements of such samples [369, 370]. A high degree of transcriptomic heterogeneity within grouped cell types of the hematopoietic system is well-described for early developmental stages [354, 355, 357, 371] and also for various terminal cell types [372–374].

PPICompare reports a reasonable amount of rewiring events

With PPICompare we identified for all developmental steps the statistically significant subsets of emerging (rew_+) and vanishing (rew_-) interactions. From this, the net change $rew_+ - rew_-$ was computed. The direction of this net change of detected interactions was always the same as that of the observable mean net difference although this must not necessarily be the case. With the exception of the transition CMP→MEP, the absolute change according to $rew_+ - rew_-$

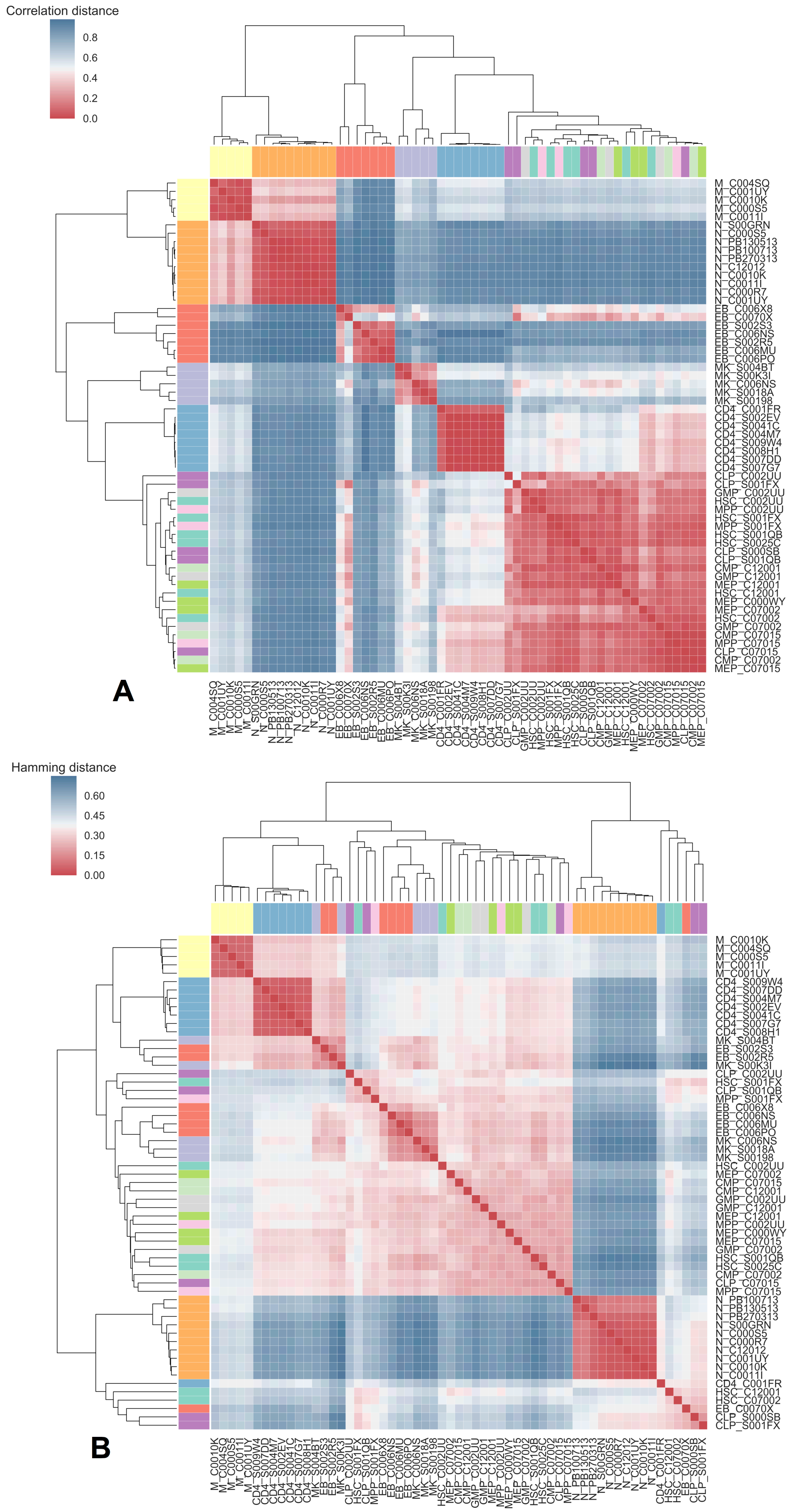


Figure 4.4: *Hierarchical clustering of hematopoiesis cell types. Results of average linkage clustering (UPGMA) applied to all samples based on A) the correlation of the transcript expression data (vector of expression values for transcripts associated with a UniProt accession in Ensembl 83) and B) the normalized Hamming distance between inferred protein interactomes (Boolean vector of abundance concerning all significantly rewired interactions). Cell types are additionally distinguished by colored labels.*

was always smaller than $\Delta n_i \rightarrow n_j$. Considering that the tool requires relevant rewiring events to occur sufficiently more often than expected from the rewiring background, it is not surprising that it provided smaller absolute estimates. Still, the deviation of the PPIXpress estimate from the mean net rewired interactions was within 0.5σ for most transitions and well below 1.5σ for all transitions we examined. Furthermore, P_{rew} and $\sigma(\Delta n_i \rightarrow n_j)$ were positively correlated (Pearson corr. coeff. 0.82). The statistical criterion used to filter out the significant portion of the differential interactome ensures to withdraw all rewiring events of questionable relevance. If one aims at also uncovering slight alterations, PPICompare is best applied to grouped samples that deviate as little as possible between groups.

Adding to that, the magnitude of the absolute net change hides the actual amount of rewiring. In the developmental transition GMP→M, for example, the 37, 170 rewired interactions (17% of the complete interactome known in human) considered significant by PPICompare only entailed an absolute net change of 5, 644 interactions. As a side note, neither $\text{obs}_s/\text{obs}_{\text{all}}$ and P_{rew} (Pearson corr. coeff. 0.3) nor $\text{obs}_s/\text{obs}_{\text{all}}$ and obs_{all} (Pearson corr. coeff. -0.15) were correlated and PPICompare determined a wide range of significance thresholds from 60% of all observations up to all comparisons for individual transitions. This shows that the statistical model adapted to the individual set of between-group rewired interactions independent of the rewiring probability and the number of samples.

Unfortunately there is neither a gold-standard nor a representative set of qualitative statements for comparison. For the non-terminal developmental stages in human bone marrow (all but the lower 3 rows) the very first transition HSC→MPP was reported to be mostly driven by the deregulation of non-protein-coding transcripts, whereas protein-coding transcripts were more important in later stages [89]. Furthermore, quantitative proteome and transcriptome analyses of mouse HSC and MPP populations likewise showed that protein abundance and transcript levels were correlated positively and few proteins were differentially expressed (47 of 4, 037 assessed proteins) [375]. If those findings are transferred to the interactome, fewer changes should be expected in the transition at the apex of the hierarchy than in later transitions. This was indeed the case for the results of PPICompare but less so for the mean net difference.

4.4.2 *A causal view on the rewiring of the blood interactome during development*

Next we examined which changes to the transcriptome caused interactions to emerge or vanish when direct developmental descendants were compared. For each significantly rewired interaction, PPICompare automatically tracks how often transcriptomic alterations of the interactors occur during the pairwise comparison between groups. The causal deregulation events that are covered by the method can be classified either as differential expression of one of the two genes coding for the interaction partners (DE), alternative splicing of one partner (AS), or corresponding transcriptomic changes to both partners (DE/DE, DE/AS and AS/AS). We analyzed in two different ways how these

type	cause proportional [%]	cause exclusive [%]	absolute loss [%]	relative loss [%]
DE	84.73	69.14	15.59	18.40
AS	1.03	0.53	0.49	48.06
DE/DE	13.81	4.67	9.14	66.19
DE/AS	0.41	0.06	0.36	85.81
AS/AS	0.02	0.00	0.01	87.23
mixed	0.00	25.60	/	/

Table 4.4: *Distribution of the transcriptomic alterations that entailed significant rewiring events. Shown is the impact of conceivable types of expression changes on interaction partners regarding all individual rewiring events per transition. The six types of expression changes were weighted by their proportional contribution to each event during the pairwise comparison step (cause proportional) or as the sole contributing cause (cause exclusive). In the latter case, rewiring events that had more than one explanatory transcriptomic cause in a transition were annotated as "mixed". Additionally, the amount of causal relevance lost due to this stricter notion is given as in absolute and relative terms.*

modes of PPI-regulation contributed to the differential interactome during hematopoiesis. First, since more than one type of transcriptomic alteration may have been detected, we weighted the contribution of each type proportionally to its occurrence in each rewired interaction (cause proportional). Secondly, we only allowed a single type per rewired interaction and else classified its causing type as ″mixed″ (cause exclusive). Table 4.4 lists aggregated results over all state transitions. Figure 4.5 provides details for individual transitions.

Differential gene expression of a single protein is the prevalent cause of rewiring for developmentally sequential adjacent blood cell types

Overall and for both types of analyses, most statistically significant changes to the interactome during hematopoiesis were driven by differential expression of a single protein, followed by differential expression of both partners, and by AS of a single one. The combinations of differential expression and AS of one partner each and AS of both interacting proteins were only relevant in few cases (see Supplementary Table S2). Imbalance concerning the direction of changes for individual modes of deregulation (see upper panels of Figure 4.5) was mostly caused by the considerable share of individual transitions to all rewiring events. More than half of the "mixed" events describing emerging interactions can be attributed to the strongly net positive change of the transition CLP→CD4. An even larger fraction of the vanishing "mixed" events and more than three quarters of the vanishing DE/DE events stem from GMP→N (see Supplementary Table S2). Rewiring events solely driven by AS ocurred more frequently in emerging interactions. This directional bias was independent of the net change of all contributing transitions (see Supplementary Table S2). We noted no preference of rewiring events driven by AS of one interaction partner towards either early or late developmental stages (see lower panels of Figure 4.5).

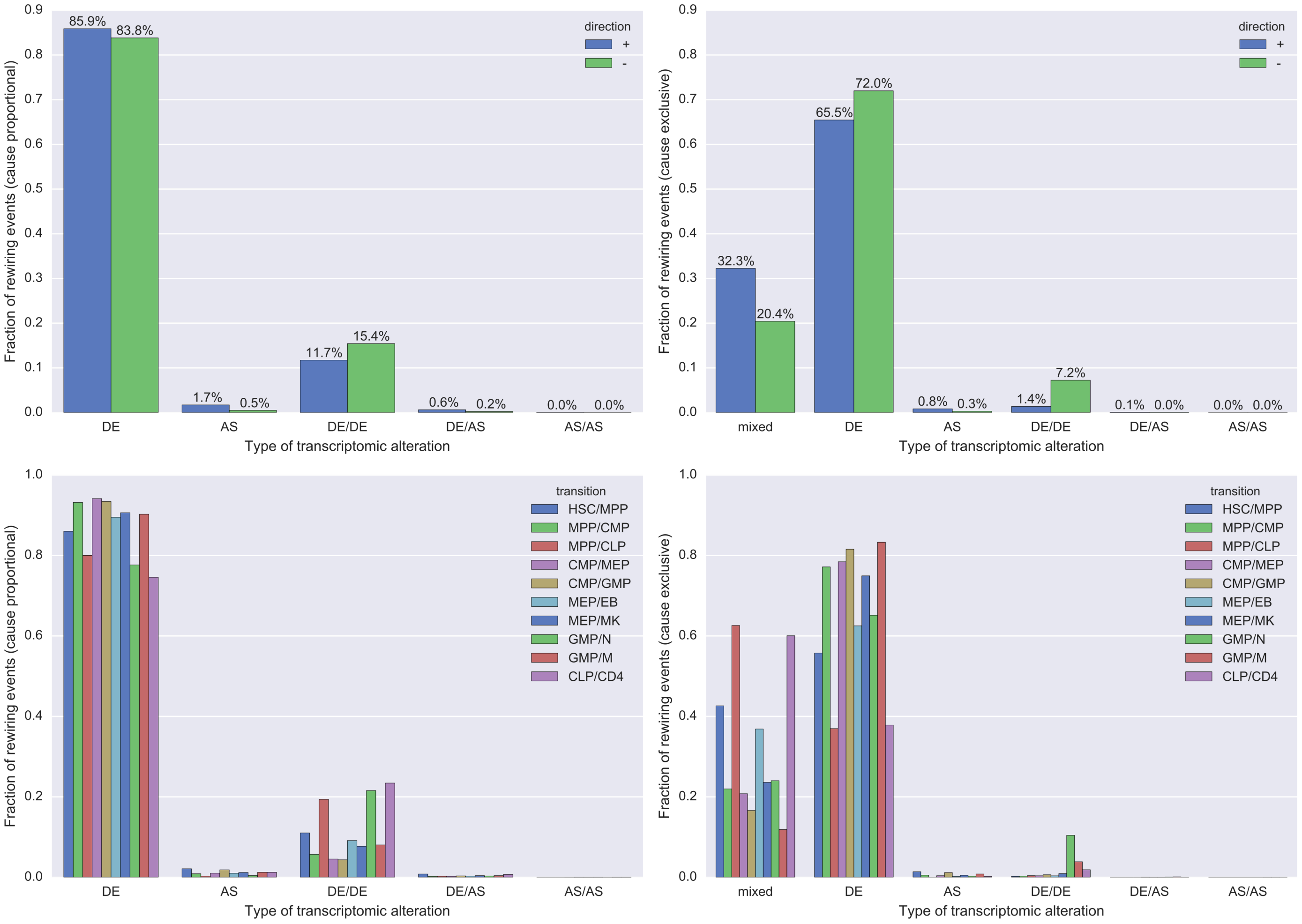


Figure 4.5: *Distribution of the transcriptomic alterations that entailed significant rewiring events. Shown is the impact of the considered types of expression changes on interaction partners regarding all individual rewiring events per transition. The types were weighted by their proportional contribution to each event during the pairwise comparison step (left plots) or as the sole contributing cause (right plots). In the latter case, rewiring events that had more than one explanatory transcriptomic cause in a transition were annotated as "mixed". The types of causes were either normalized by the direction of the rewiring events (upper plots) or by their contributions to individual transitions (lower plots). In the top plots, "+" (blue) means emerging interactions and "-" (green) means vanishing interactions. The lower three developmental transitions are those towards terminally differentiated cell types found in blood.*

This general order of importance that we observed for the different modes of deregulation, in particular DE being more prevalent than AS, seems plausible. We already mentioned possibly confounding factors such as the incomplete coverage of the interactome with domain annotation data that PPIXpress uses to detect AS events of influence (only about half of the proteins and a fifth of the interactions in the reference interactome are covered, see Section 4.3.2). Despite of this missing information, regulation of gene expression is generally considered to be the main determinant of cellular specificity [281, 297] whereas splicing is more relevant between individuals [283]. The contribution of AS, however, certainly depends on the developmental system under study and is likely to be higher in the human brain [6, 51], for example.

Alternative splicing is necessary to explain many significant rewiring events in hematopoiesis

Although the contribution of AS seems minor in comparison to differential gene expression (below 1% in exclusive causes), 871 rewiring events across all developmental transitions considered here could only be fully explained by including AS (see AS, DE/AS and AS/AS in Supplementary Table S2). These cases would have been missed by methods that only rely on gene expression. Rewiring events that were exclusively regulated by AS across all comparisons in a transition were enriched (adj.p < 0.05) in pathway annotations concerned with the post-elongation processing of mRNA (affecting genes associated with splicing and polyadenylation), the cell cycle (G2-M checkpoint and control of the pre-replication complex by the activator of S phase kinase DBF4), transcription initiation, the transport of mRNA, as well as the regulation of phagocytosis (see Supplementary Table S3 for details on interactions, databases and pathway terms). Our approach to determine interaction-centric enrichment of pathway annotations is outlined in the Methods section.

For example, we found that three genes which code for components of the spliceosome complex (PRPF4B, SNRNP70, SRSF3) switched their major isoform to a variant that undergoes nonsense-mediated decay (NMD) at specific points during blood development and therefore did not produce functional protein products anymore. This regulatory mechanism has been described for several splicing factors such as SRSF3 (Serine/arginine-rich splicing factor 3) [376, 377], which we found to be turned off during the transitions GMP→N and CLP→CD4. We found that this was also the case for SNRNP70 (U1 small nuclear ribonucleoprotein 70 kDa) in the transition CMP→GMP. The protein was then activated again in the GMP→M transition but not in the branching to neutrophils (where SRSF3 was also deactivated). In [297], spliced protein isoforms detectable in mass spectrometry were also enriched with nuclear ribonucleoproteins. Furthermore, PRPF4B (Serine/threonine-protein kinase PRP4 homolog) switched to an active isoform in GMP→M. Since PPIXpress (version 1.08) only uses domain annotations of protein-coding transcripts, protein interactions that were associated with a domain interaction were correctly predicted to vanish if the corresponding transcript was classified to undergo NMD. Supplementary Table S4 provides a detailed listing of rewiring events associated with known protein complexes across all stages of hematopoiesis.

Different types of alterations can cause the same rewiring event

When we required each rewiring event to be consistently deregulated in the same way in all between-group comparisons for the respective transition, the contributions of most alteration types decreased severely by up to 87% compared to their proportional contribution. The reason for this is that they mostly occurred together with other transcriptomic changes (see last two columns of Table 4.4). To associate rewiring events with modes of deregulation in a definitive manner, we will use this strict interpretation of regulatory changes in the remaining text. Still, considering that individuals can show a varying composition of major protein isoforms in the same cell type [283], it is plausible that different alterations to the transcriptome may drive the same net change to their interactomes.

With the exception of the transition GMP→N, events caused by a mixture of alteration types were more prevalent in transitions with higher P_{rew} (see lower right distribution in Figure 4.5, Pearson corr. coeff. 0.90 when GMP→N was left out). The relative loss in that regard was largest for what we will call "co-deregulatory" types of regulation in the following (rewiring events caused by DE/DE, DE/AS and AS/AS events, see last two columns of Table 4.4). This raises the question if simultaneous deregulation of interaction partners is actually a meaningful mode of control or if the observations where this was noted were the result of concealed individual deregulation events across different intermediate stages of development.

Simultaneous deregulation of interaction partners shows tendency towards rewiring within functional modules

Protein interaction networks are thought to be organized in a modular fashion. Several studies, mainly concerned with highly connected (hub) proteins in yeast, showed that there are two basic types of such proteins in interaction networks. Hub proteins either operate intramodular and are coregulated with their interaction partners to work together on the same task as a cohesive unit, or they act as intermodular connectors of different functional modules and are expressed independently of their neighbors [144, 145, 378, 379]. Whereas those essential implications of the modular structure also apply to the human interactome, the complexity there is beyond dichotomous classification [379]. Yet, interaction partners that are specifically regulated together should more likely belong to the same functional module in the PPIN. Therefore they should also be more likely involved in the same protein complexes, work in the same biological process, have similar function, and be colocalized [144, 145]. Furthermore, the betweenness, a measure from graph theory to delineate modules, should be lower for intramodular interactions than for intermodular interactions [380, 381].

We compared rewired interactions caused by deregulation of only one interaction partner with those where the expression of both interaction partners was altered and to those with mixed causes in this respect. To test their tendency to reside in functional modules, we considered the involvement of the affected interaction partners in known CORUM protein complexes [361]. Also, we analyzed the similarity of their interaction partners regarding all GO ontologies (biological process, molecular function and cellular compartment) [294], and the betweenness of the affected interaction in the reference PPIN. The results are visualized in Supplementary Figure S2.

We found that rewiring events caused by simultaneous deregulation affected indeed more often known protein complexes (fraction of interactions associated with reference complexes increased from 3.8% to 5.3%) and had significantly lower betweenness (median betweenness decreased by 14%, two-sided Wilcoxon rank-sum test $p < 10^{-30}$). Also, co-deregulated interaction partners were more likely to work on similar processes (median GO biol. process similarity increased by 2%, two-sided Wilcoxon rank-sum test $p < 0.03$) and had comparable similarities of GO molecular functions and GO cellular compartments. Taken together, these soft factors support the interpretation that co-deregulated partners in the PPIN are more likely part of the same functional module.

Interestingly, rewiring events caused by DE/DE and DE/AS were predominantly (relative and absolute) found in transitions towards the terminal developmental stages (see lower right panel of Figure 4.5 and Supplementary Table S2). Among those, vanishing interactions during the progression of GMPs to Ns and Ms were highly enriched with annotations concerning cell cycle progression (see Supplementary Table S5). More specifically, interactions disappeared that are important for the G2-M checkpoint and for the activation of the pre-replication complex. For the transition CLP→CD4, this was not the case for any mode of regulation. Since T cells are proliferating [382] and Ns and Ms are cell types that are generally non-proliferating [383, 384], some of these alterations of protein interactions are likely associated with cell cycle exit.

Furthermore, GMP→N was of special interest in that regard, because it showed by far the highest amount of co-deregulation (4,786 rewiring events caused by DE/DE, DE/AS or AS/AS, see Supplementary Table S2) and also the largest overall amount of rewiring as indicated by P_{rew} (see Table 4.3). When analyzed in detail, the transition to terminal neutrophils is a stepwise process with five intermediate stages that are, unfortunately, not resolved by the BLUEPRINT data. Within those finer-grained steps, proliferation, in fact modulated by the expression of cell-cycle proteins, steeply decreases during an early stage and is completely absent after the very next [383]. Whereas this regulatory process is thus not completely synchronous, the net effect is still correctly described by our analysis.

Besides the deactivation of the cell cycle, a surprisingly large number of co-deregulated changes to the interactome were associated with the depletion of interactions of other coherent molecular machineries, namely RNA polymerase III (Pol III) and tRNA processing (see Supplementary Table S5) as well as mitochondrial ribosomes (see also Supplementary Table S4). This latter finding matches the fact that mitochondria are very rare in Ns and not used for energy metabolism [385, 386]. In contrast to this, the (partial) depletion of Pol III has, to our best knowledge, not been explicity described in the literature. Pol III is responsible for entirely different functions in immune cells. Its inhibition restrains phagocytosis and cytokine secretion in macrophages due to its role in tRNA production [387], but it can also act as a sensor to detect foreign DNA [388]. However, its inhibition does not alter the response of Ns in that regard [389]. Owing to the short lifespan of Ns, it may simply be an economical decision of budgeting cellular resources.

Small set of likely transcriptomic alterations

PPICompare provides an optimization approach that suggests a small set of likely changes in the transcriptome that explain all significant rewiring events. In every transition each of these alterations to a single protein yielded between 6.6 and 17.4 rewiring events on average (average of 11.4 over all transitions). The number of all proteins affected by any significant rewiring event was on average 5.3 times larger than the number of proteins in the respective small set of changes (see Supplementary Table S6). From now on, we will refer to this smaller set of proteins as the "reduced set" of proteins affected by rewiring.

The optimization approach tends to select hub proteins in the differential network (see Figure 4.6 (left) for an example and Supplementary Figure S3 (upper half) for complete results). This is not very surprising given that the score s_i increases if such a protein was transcriptionally deregulated. Also, it is biologically reasonable because an appropriately deregulated protein will cause rewiring around itself. Interestingly, selected proteins were not necessarily highly connected proteins in the reference interactome whereas those rewired proteins that were not in the reduced set tended to have above average degrees in the complete network (see Figure 4.6 (right) for an example and Supplementary Figure S3 (lower half) for the complete results). The latter observation likely increased their chance of acting as interaction partner of a deregulated protein and thus be part of the differential network.

Figure 4.7 outlines the contributions of the two elementary modes of transcriptomic alterations per protein, DE and AS, to the individual sets and altogether. Also in the reduced set, most of the deregulation events were driven by DE. Yet, the overall proportion of AS was about twice as large as in the comparisons shown previously (in each transition at least 1.3%). Also, the fraction of AS was larger among emerging interactions. The usage of alternative protein isoforms was equally important in all transitions we analyzed.

Important alternative splicing events are found in proteins broadly associated with transcriptional control

To assess the functional scope of alternative transcript usage, we submitted the set of all 134 proteins which underwent AS in the sets of most relevant events in any transition to enrichment analysis using the DAVID web service [260] (see Methods section).

DAVID characterized the gene set to be preferentially located in the nucleus (e.g., "nucleoplasm" 2.6 fold enriched), and preferentially concerned with the organization and regulation of chromatin (e.g., "chromatin organization" 3.7 fold enriched and "chromatin modification" 3.2 fold enriched) and with transcriptional regulation (e.g., "DNA binding" 1.9 fold enriched and "transcriptional regulation" 1.9 fold enriched). The family of Basic-Leucine zipper TFs

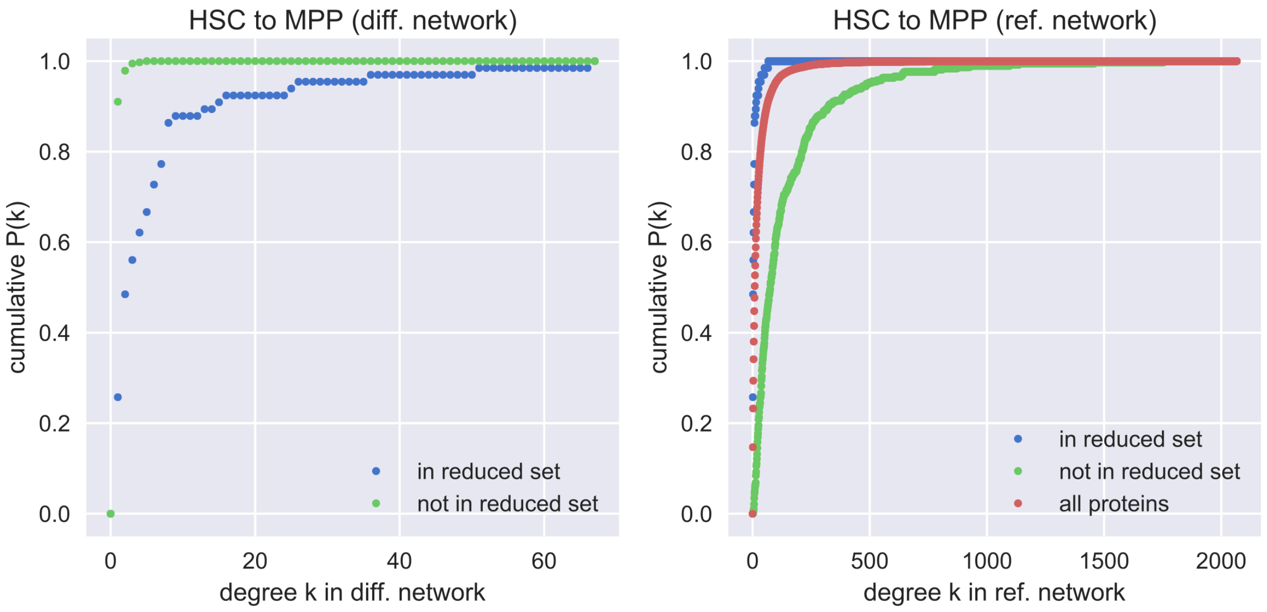


Figure 4.6: *Cumulative degree distributions of rewired proteins. Cumulative degree distributions of the rewired proteins of the transition HSC→MPP in the corresponding differential network (left) and the distributions of the rewired proteins in the reference protein interaction network (right). The rewired proteins are additionally split up into those in the reduced set and the remaining ones, "all proteins" depicts all proteins in the reference network.*

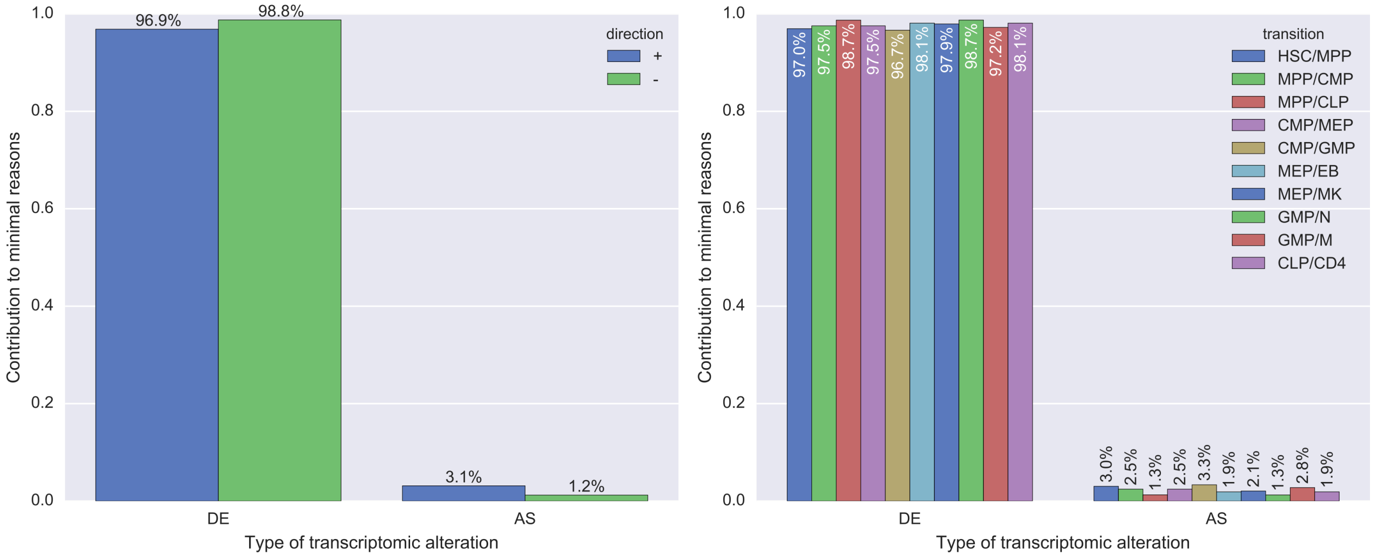


Figure 4.7: *Distributions of alteration types for the minimum amount of explanatory reasons for rewiring events. Shown is the contribution of the two elementary types of conceivable protein alterations in PPICompare, DE and AS, to the solutions of the optimization regarding the small sets of likely changes that explain all rewiring. The contributions are normalized by their direction (left plot) or by their proportion in individual transitions (right plot). In the left plot, "+" (blue) means emerging interactions and "-" (green) means vanishing interactions.*

seemed to be especially relevant (e.g., “Basic-leucine zipper (bZIP) transcription factor” 9.8 fold enriched, but not significant after adjustment). Further enriched clusters involved post-translational regulatory mechanisms like ubiquitination and related processes (e.g., “Ubl conjugation pathway" 3.8 fold enriched). A detailed listing of all results is provided in Supplementary Table S7.

The accumulation of such terms in the altered interaction partners points to the combinatorial and synergistic control of transcription, which is of central importance in all critical developmental circuits in eukaryotes [8, 13, 211, 212]. This specificity is of special interest because individual interactions between different TFs or TFs and cofactors seem to be deliberately switched in a targeted way by AS although both factors are expressed in the cell.

Interactions between proteins in the reduced set are likely connectors of functional modules

Co-deregulation of proteins in the small set of changes could hint at important coregulated processes. We started to inspect this possibility by evaluating significantly rewired interactions among proteins of the reduced set in the same fashion as for the general case of simultaneous deregulation. The results are also visualized in Supplementary Figure S2.

Interactions altered by those events are associated with more reference protein complexes than the network average but with fewer than the co-deregulated events. They did not differ from co-deregulation events concerning the similarity of processes and colocalization. Whereas the functional similarity was only slightly increased (median GO functional similarity increased by 6%, two-sided Wilcoxon rank-sum test $p < 0.02$), there was a striking increase in the betweenness values compared to simultaneous deregulation (median betweenness increased by 31%, two-sided Wilcoxon rank-sum test $p < 10^{-78}$). The betweenness values were even significantly higher than those of rewiring events for which consistently only one protein was deregulated (median betweenness increased by 13%, two-sided Wilcoxon rank-sum test $p < 10^{-69}$). This speaks against a possible intramodular role of such interactions in the interactome, but rather hints at a function as intermodular connectors between functional modules. Such connections are very important in signaling, for example, and their dysregulation can be crucial [390]. In fact, the interactions between proteins in the reduced set were enriched in signaling pathways for all developmental transitions (see Supplementary Table S8). The apoptosis-relevant interaction of Bcl-2 (BCL2) with the Bcl-2 modifying factor (BMF) [391], for example, emerges in the transition HSC→MPP and is an interaction between proteins of the reduced set (first tab Supplementary Table S8).

Complementing this, we used the respective sets of emerging and vanishing interactions individually to determine direction-consistent connected components (CCs) among the reduced protein sets in each transition. The results are listed in Supplementary Table S9. Although there existed very large CCs among those interactions (including up to 2,005 proteins in GMP→N, for example), even the large CCs contained comparably few (at most 27) and rather small known CORUM complexes (the largest complex overlapping a CC contained 5 proteins). Within functional modules, one would rather expect that deregulated CCs would preferentially coincide with complexes, though.

The reduced set of affected proteins is representative to blood development

In our study, significant rewiring events can be expected to affect proteins that are related to hematopoiesis. We examined this hypothesis by testing how likely it was to sample at least a certain amount of proteins deemed relevant in this context from the reference PPIN by chance. The importance of proteins in that regard was classified according to protein sets that we compiled from GO annotation data and regulatory data from TRRUST [368] (see Methods section for details).

We first checked for overrepresentation of hematopoiesis proteins and the regulatory targets of hematopoietic TFs. The latter ones were included to also account for proteins that are not obviously associated with hematopoiesis, but that are equally probable to be deregulated due to their direct dependency on regulators of blood development. The set of all proteins affected by rewiring events was highly enriched for those proteins across all transitions (for all transitions $p < 10^{-5}$, see left half of first sheet in Supplementary Table S10). Except for the transition MPP→CMP, the reduced set of deregulated proteins always contained in all other transitions significantly more of those relevant proteins than expected by chance (for all other transitions $p < 0.022$, see right half of first sheet in Supplementary Table S10).

Similar results were obtained for the set of hematopoiesis proteins without the targets (see second sheet in Supplementary Table S10 for details).

Known hematopoietic transcription factors are among the drivers of rewiring events

Then, we investigated if known hematopoietic TFs were rewired more often than expected by chance and if targets of certain TFs were overrepresented in the two protein sets determined (see Methods section for details).

Whereas the complete set of proteins involved in rewiring events was highly enriched in hematopoietic TFs (for all transitions $p < 3 * 10^{-4}$), this was mostly not the case for the reduced set of proteins (see third sheet in Supplementary Table S10). Examples of such rewiring events are discussed below.

Likewise, we found an enrichment of TF targets in the complete set for all transitions. In all but one case this even included known hematopoiesis regulators (see left half of fourth sheet in Supplementary Table S9). Again, enrichment was only reported in four transitions for the reduced protein sets (see right half of fourth sheet in Supplementary Table S10). Thus, while the optimization procedure can help to effectively decrease the number of proteins of interest, depending on the task at hand the reduction may come along with a loss of information.

TFs for which targets were overrepresented in different developmental transitions are listed in Supplementary Table S10. We omitted a discussion on potentially enriched hematopoiesis regulators therein since we believe that much more data should be integrated to appropriately account for important details. The regulatory data that we used is confident but comparably sparse and only considering direct regulatory targets is an oversimplification in that regard. Also, combinatorial regulation should be considered in the context of hematopoiesis [212] and the chromatin state of each cellular condition is relevant [349, 350].

4.4.3 *Consequences of rewiring during blood development*

At last, we took a brief look into which interactions were changed. The output files of PPICompare are formatted as node- and edge-attribute tables to enable seamless support of network visualization tools such as Cytoscape. Figure 4.8 shows an illustration of the resulting differential network for the transition HSC→MPP whereby the dense central region is enlarged. Remarkably, this highly connected part of the network is characterized by changes to the interatome between different TFs and between TFs and cofactors. Such assemblies of transcriptional regulators indeed often have a pivotal role in the context of developmental control [8, 13, 211, 212]. Thus, we will focus our attention on this subset of proteins and discuss some of the rewiring events involving proteins considered most relevant by the internal optimization of PPICompare (blue nodes in the visualization).

The TF Fos-related antigen 1 (FOSL1) is a prime example for alternative transcript usage. Upon transition from HSCs to MPPs, its most abundant transcript was switched from ENST00000448083 to ENST00000312562 in every between-group comparison. This shift resulted in the inclusion of a basic-leucine zipper domain (PF00170) which is needed for any dimerization of the protein and thus enabled formation of several new interactions to other regulatory proteins. Among those were coactivator proteins such as the (histone) acetyltransferases p300 (EP300) and CREB-binding protein (CREBBP) which are both important integrators of regulatory signals in the hematopietic and other developmental systems [392]. Since such proteins are ubiquitously expressed in all cells, a sole analysis of differential expression would not have been able to detect a difference in that regard between HSCs and MPPs. Interactions of FOSL1 with other TFs that were viable after splicing involved c-Jun (JUN), Jun-D (JUND), c-Maf (MAF) or Activating Transcription Factor 4 (ATF4). Together with factors from the Fos-family, these are exchangeable constituents of the TF complex AP-1 and as such control processes including proliferation, differentiation and apoptosis [393, 394]. Further emerging interactions to TFs included binding to DNA damage-inducible transcript 3 (DDIT3), that is involved in response to cellular stress, and c-Myc (MYC). Besides its general implication in processes such as cell division, apoptosis, cellular growth, angiogenesis and differentiation, c-Myc is specifically concerned with the balance of self-renewal and differentiation of HSCs [395].

Lymphoid enhancer-binding factor 1 (LEF1) is another protein that changed its expression state in each single comparison and possesses regulatory capabilities in developmental processes beyond the lymphoid lineage [396]. The PPICompare results help in explaining how LEF1-binding may affect its targeted sequence regions mechanistically in MPPs compared to HSCs. Facilitated by the detected differential recruitment of various histone modifying proteins

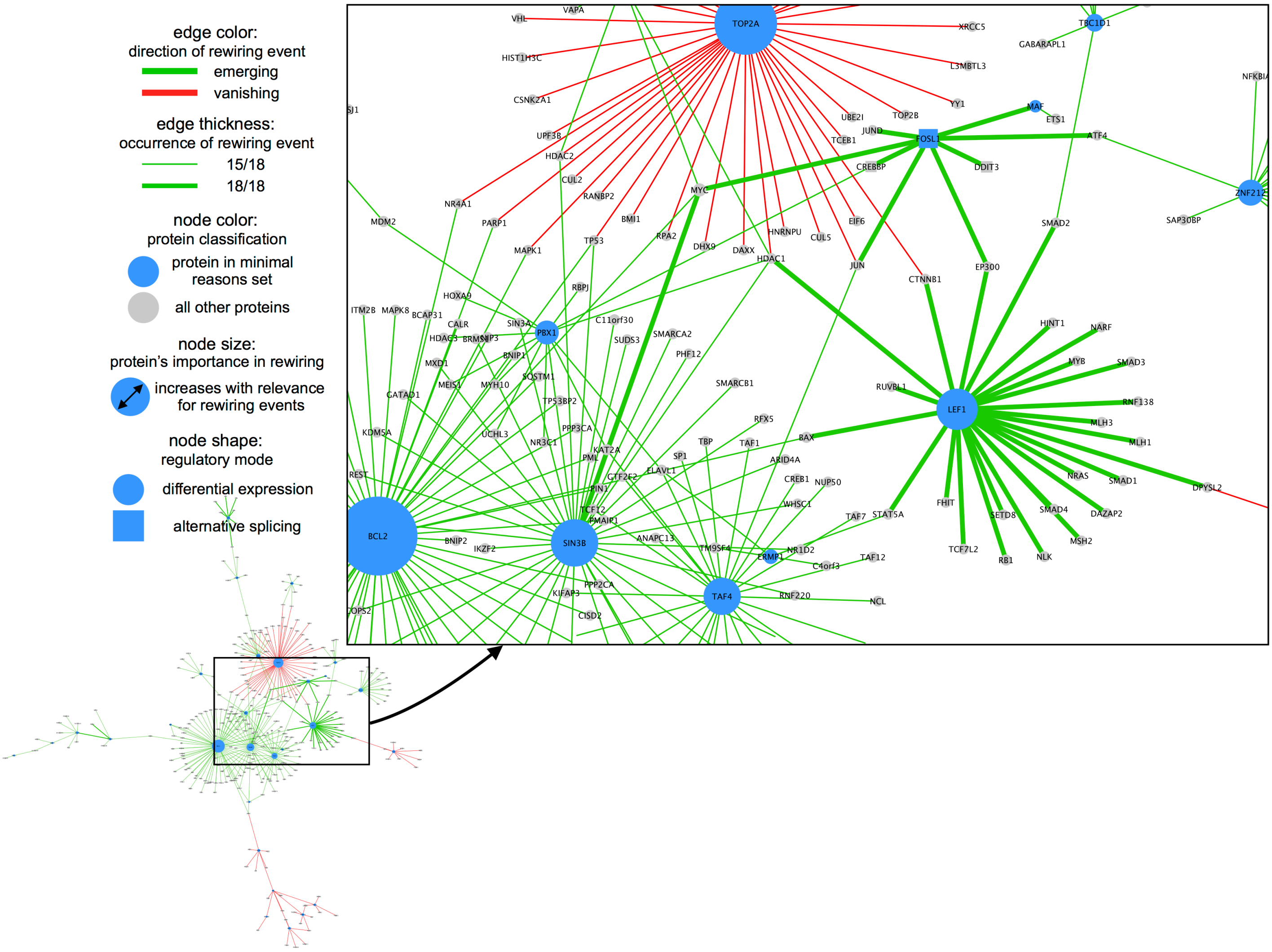


Figure 4.8: *HSC→MPP rewiring events in Cytoscape. We visualized the differential network of the transition HSC→MPP in Cytoscape 3.3 [346] using the default output files of PPICompare. The nodes depict all proteins affected by significant rewiring events. All proteins (internally UniProt accessions) are displayed with their associated gene's name. Proteins that belong to the "small set of likely changes" are colored blue. The size of nodes increases with their importance score as described in the Methods section. Furthermore, protein nodes with a rectangular shape were solely deregulated by AS (here: FOSL1). Green edges depict emerging interactions and red edges the vanishing ones. The edge thickness indicates how often the event was observed throughout the pairwise comparisons (here either in 15 or in 18 of 18 comparisons). Here, only the largest connected component of the differential network is shown (lower left).*

(EP300, HDAC1, SETD8), it could act as the DNA-binding factor for chromatin remodeling events in MPPs, for example. Also, LEF1 may form complexes with β-catenin (CTNNB1), T-Cell Factor 4 (TCF7L2), and other proteins (HINT1, RUVBL1) implicated in Wnt/β-catenin signaling, a crucial developmental pathway [397–399]. It may also bind to c-Myb (MYB), a TF controlling regulation of hematopoietic progenitors [400]. Moreover, its abundance in MPPs enabled interactions with the Bcl-2 associated X protein (BAX), which also binds to the important apoptosis regulator Bcl-2 (BCL2). The expression of the latter was upregulated here when hematopoietic progenitor cells become more commited. As correctly determined by PPICompare, Bcl-2 has plenty of new interaction partners in MPPs and thereby ensures a balance of complexes with pro- and antiapoptotic influence (besides BAX: BCL2L1, BAK1, both not visible in figure) [401].

Another upregulated protein deemed important by us was the adaptor protein Sin3b (SIN3B) which facilitates the association of other proteins to epigenetic silencers (REST, HCDA2). Although it apparently did not exert this function in HSCs, it provided c-Myc (MYC) with this capability after the progression to a progenitor cell and furthermore enabled a repressive function of the important hematopoietic TF Helios (IKZF2) of the Ikaros-family [402].

Besides those examples for transcriptional control in HSC→MPP, Supplementary Table S11 lists all pathways that are affected by rewiring events. We grouped the events into changes to interactions that were shared between transitions or those exclusive to a certain transition at a developmental branching point.

4.5 CONCLUSION

Combining PPIXpress and PPICompare enabled us to investigate the dynamics of cause and consequence within the human protein interactome during developmental branching and progression to the extent that this is reflected by transcript expression data. In principle, one can easily detect alterations to any pathway or changes to functional protein complexes, like those concerned with transcriptional regulation. Furthermore, the provided software can aid in suggesting promising targets for the development of new PPI inhibitors, an emerging class of molecules in drug discovery [403, 404]. Beside the general genome-wide trends studied here, the presented pipeline is equally powerful to address very specific questions about rewiring of protein interactions.

4.6 ADDENDUM

4.6.1 *Retrospective*

Although it is often inevitable, I have the personal opinion that the usage of arbitrary cutoffs in analyses as well as in tools is utterly unsatisfying. Contrary to that I chose a discretized approach as the basis of PPICompare with its merits and perils. Discretization often simplifies the usage and interpretation of data, in the best case it diminishes technical as well as biological noise when applied well. But, of course, discretizing any input data always comes with a loss of detail [347].

A PPICompare tool without a discretized view on the interactions would have been entirely possible. Around the timespan of the project I already had a version of PPIXpress available that would have allowed me to create weighted sample-specific interactomes (see Section 3.6.2). Depending on the exact input data those networks would enable to associate a weight or probability with each interaction in each sample. A weighted interactome could then be assessed by employing standard statistical tests comparing distributions of numerical values by grouped samples (see Section 2.2.1). Furthermore, "transcriptomic reasons for rewiring", as I called them, could have been inferred by integrating results of common differential expression analysis workflows of which there are plenty to chose from [405]. The basis for a continuous data version of PPICompare would thus already have been set.

Still, PPICompare uses the discretized model that we also employed in the study of PPIXpress (see Section 3.4.2). By simplifying the data, each sample in PPICompare is characterized by a set of protein interactions and protein isoforms coded by the respective representative transcripts that are present. There is no grading of any kind associated with the interactions and there is no ensemble of transcripts that are linked to each protein, just one single transcript. All notions and relations are thus intuitive and unambiguous. This direct relatedness of interactome and transcriptome ultimately enables the straightforward inference of transcriptomic drivers causing each rewiring event and thus allows for a very clear analysis of causality in network dynamics. The non-discretized ensemble view in the continuous variant outlined above would lack this direct correspondence. In fact, as long as changes in interactome and transcriptome are evaluated independently, disregarding the exact statistical approach at hand, there is no guarantee that each rewiring event is backed by an event of differential transcription of any kind.

This conceptual simplicity should benefit the potential target audience, namely experimentalists wanting to get the most out of the data they already produced. PPICompare was, for example, applied in [406] to assess the differential interactome between ICAM1-positive and -negative neutrophils in the context of experimental autoimmune encephalomyelitis, which is a model system to study multiple sclerosis.

5 SCALING UP DOMAIN-AWARE COHESIVENESS OPTIMIZATION

This chapter describes JDACO, the multithreaded Java implementation of the domain-aware cohesiveness optimization algorithm DACO [17] that I originally prototyped in Python. Alongside some technical details, a comparison of the performance of both implementations is made in a benchmark study. All code files and binaries for each version of (J)DACO are available for download at `https://sourceforge.net/projects/dacoalgorithm/`.

5.1 PREREQUISITES

5.1.1 *Python vs. Java: data structures and parallelization*

Python and Java are exceptionally popular general-purpose programming languages (March 2019, see for example: TIOBE index[1] or PYPL[2]). Both languages are independent of operating systems and hardware because they are either run in an interpreter (Python) or compiled to bytecode, an intermediate representation of special instruction sets which is interpreted and executed in a runtime environment (Java). Because it abolishes the need to satisfy any dependencies or tedious compilation and installation procedures, this is a very appealing property for developers and users because it simplifies the distribution and usage of software implemented in such languages. Since the actual choice of the interpreter can change internal details in some languages, Python in the following always refers to CPython, the open-source reference implementation of Python and also the most widespread Python environment.

Without going into the details, although the languages are very different in terms of programming paradigms and internals, Python as well as Java are reasonable choices for the implementation of any kind of software, including scientific algorithms. Still, sometimes one tool is the better option for a certain task at hand. With the development of PPIXpress, the data retrieval as done by DACO could be completely outsourced because a more powerful, much faster and more convenient software tool was available that solves the task of constructing input networks for the algorithm. With the omission of this considerable part of the necessary functionality, the focus of the new implementation could be completely shifted towards the algorithmic effort. In that regard, Java is without doubts the favourable choice of the two languages. I will justify this reasoning on the important benefits that are gained in terms of data structure flexibility and parallelization capabilities.

[1] `https://www.tiobe.com/tiobe-index/`
[2] `http://pypl.github.io`

Built-in data structures

In common modern programming languages "the batteries are already included", as one says. This means that usually a selection of important standard data structures are readily available in highly optimized implementations without the need of installing or packaging additional libraries. A high level of convenience by minimization of additional dependencies is a desirable situation especially when less versed users may be part of the target audience. Python and Java both have well-laced standard libraries that enable a manifold of duties with comparably few lines of code. But in terms of data structures, Java has a clear edge because it allows a very specific adjustment of the exact implementations. I will illustrate my point on the example of list data structures.

All Python lists are implemented as a dynamic array[3], which is a managed abstraction of an array that resizes itself as more elements are added. The same data structure can be used with Java's *ArrayList* implementation. Also, a similar but synchronized and thus concurrency capable variant is offered with *Vector*. Furthermore, Java also allows to use a double-linked list with the *LinkedList* class[4]. Here, no resizing is necessary when an unknown amount of elements is added to the list during execution. Additionally, Java has traditional unmanaged arrays for "primitive types", which are elementary data types such as Booleans, bytes, chars, as well as typical ranges of integer and floating point numbers. Such classical arrays have much less internal programmatic overhead when iterating and accessing elements but require manual memory management by the developer.

While different list implementations all allow to somehow store, retrieve and iterate over elements, their individual performance characteristics may vary significantly. The broad choice of implementations offered in Java can allow to select the most suitable data structure for a specific scenario. The same examples can be made for other elementary storage concepts such as sets and dictionaries.

In addition to that, Java already offers special data structures for multithreaded software. Using such data structures is not per se beneficial when many threads are utilized, but they are generally worthwhile when their content is modified and shared between threads. A general principle to enable concurrent usage of a standard data structure is by locking or synchronizing the access, e.g. when a thread t wants to modify components of the data structure, all other threads need to wait until t is finished if they want to use any of the thus stored data. Consequently, the threads will finish their assigned labor slower than without the limitation of access. With sophisticated data structures circumventing the need for such a global locking mechanism such pitfalls can be minimized. A Java *HashMap* is an implementation of a dictionary that allows for very fast read-, write- and membership-queries (on average $O(1)$,

[3] list implementation of Python 3.8: https://github.com/python/cpython/blob/3.8/Objects/listobject.c

[4] list implementations of Java 8: https://docs.oracle.com/javase/8/docs/api/java/util/List.html

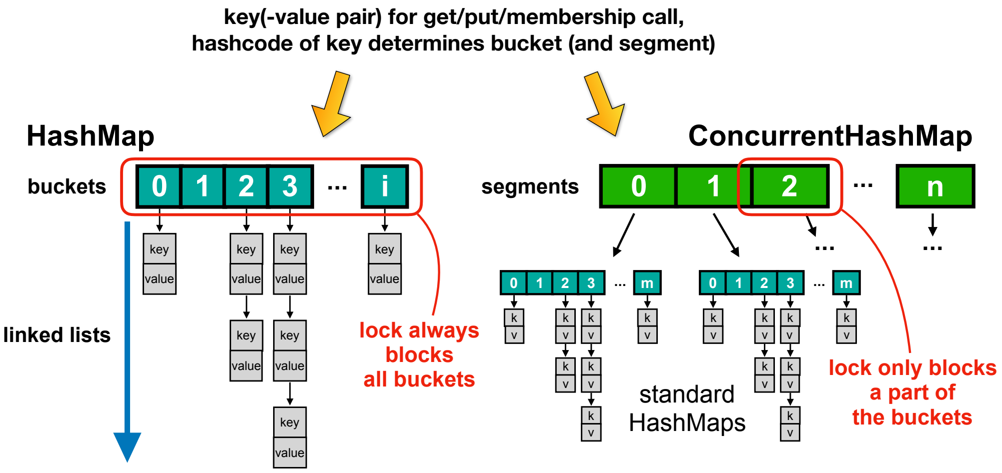

key(-value pair) for get/put/membership call,
hashcode of key determines bucket (and segment)
HashMap
buckets
0 1 2 3 ··· i
key
value
linked lists
lock always blocks all buckets
ConcurrentHashMap
segments
0 1 2 ··· n
0 1 2 3 ··· m
k
v
standard HashMaps
lock only blocks a part of the buckets

Figure 5.1: *Standard and concurrent HashMap implementations. Segmentation of the internal data structure allows to only lock those parts of the hashtable that are modified. Here, the classical textbook versions of such data structures are shown. In Java 8, linked lists have been replaced by binary search trees as the key-value pair storage data structures.*

given a well-spreading hash function on the keys and that the hashtable has a reasonable size) by maintaining an internal hashtable. To add a key-value pair, to retrieve the value of a key or to check the presence of a key in the dictionary, a hashcode is computed for the key object and used to assign the key to a certain bucket of the hashtable. Then only a short linked list (or a small binary search tree in recent Java 8 implementations[5]) needs to be modified or iterated to perform the operation [328]. If a compute thread adds a key, the whole hashtable is locked and all other threads that may want to access the data need to wait. The concurrent implementation *ConcurrentHashMap*[6] circumvents this performance issue by partitioning the hashtable into segments which can be blocked individually. Forced idling times for other threads are thus far less likely in practice. Figure 5.1 illustrates the core buildup of the two data structures and highlights how they differ in terms of synchronization for multithreaded usage.

Parallelization capabilities

The revolution of microelectronics that made our modern age of information possible started rapidly. As postulated by Moore in 1965, the number of transistors per chip doubled every two years for quite some time [407]. Thus, every new generation of silicon chips became more powerful and cheaper at the same time in an exponential pace. After decades of successfully shrinking chip structures and thus increasing circuit densities critical physical limits were finally hit in the early 2000s. Power usage and heat constraints set hard boundaries on the operating frequencies and thus also on the processing speed of conventional semiconductor technology. Still, manufacturers managed to

[5] Java 8 source code: http://hg.openjdk.java.net/jdk8/jdk8/jdk/file/tip/src/share/classes/java/util/HashMap.java

[6] Java 8 source code: http://hg.openjdk.java.net/jdk8/jdk8/jdk/file/tip/src/share/classes/java/util/concurrent/ConcurrentHashMap.java

hold Moores's predicted rate of advancement by developing chips with many power-efficient cores instead of fast single-core architectures [408]. From the 2010s on the exponential growth of processing power could ultimately not be met anymore. More so, due to quantum effects in the nanometer scale, the end of the unprecedented progress made with silicon transistors is inevitable in the near future [409].

For developers this means that parallel programming is a necessity to fully utilize modern hardware and to maximize scalability. To achieve this, runtime-critical algorithm segments obviously need to allow a partitioning into simultaneously solvable subproblems. But even if that is the case the programming language used can still obstruct an optimal result.

In principle, both Python and Java can make use of many-core processors. In practice, Python suffers from a design decision that is called the global interpreter lock. The lock ensures that only one thread can execute bytecode at once in a Python interpreter which simplifies the memory management of objects and allows the convenient usage of C libraries that are not necessarily thread-safe. Without this safety measure, simultaneous access and manipulation of shared data may lead to inconsistent data. This design decision was made early in the development of the language and over time many components depended on the global lock eventually making it unchangeable [410]. If the tasks are completely independent, one can work around it by spawning multiple interpreters and processes that communicate through inter-process communication. This solution, however, comes with a high computational and memory overhead.

As already teased above, Java, on the other hand, has powerful features aiding the efficient programming of multithreaded software. First of all, Java threads can communicate with each other with low overhead in the same virtual machine. Second, modern Java comprises convenience features that allow for a very productive approach towards parallelization like parallel stream operations[7] or the transformation of recursive algorithms into parallel implementations by a Fork/Join framework[8]. The latter was not used in JDACO in favor of a custom algorithm-specific thread pool which, adding to the previous section on data structures, Java made possible by also including a plethora of thread-safe data structures which were an important corner point for the improved implementation of the DACO algorithm. A thread pool is basically a group of worker threads of fixed size, typically the number of cores the user wants to utilize, that execute jobs from a work queue one after another whereby the threads are reused as often as necessary.

5.2 INTRODUCTION

Modern life science progressed into an age of high-throughput whole-genome analyses. Determining the complete transcriptomes [22, 283, 291, 411] or proteomes [127–129] in a sample-specific manner has become a fairly standard task

[7] https://docs.oracle.com/javase/8/docs/api/java/util/stream/package-summary.html

[8] https://docs.oracle.com/javase/tutorial/essential/concurrency/forkjoin.html

nowadays. As a consequence, an enormous amount of data in that regard is publicly available.

However, when it comes to the many-faceted interplay of proteins in interactions and complexes, we are not on this level yet. On a whole-organism scale, protein interactions as well as complexes are still treated as static entities that are contextualized by data integration efforts to infer specific interactomes [165, 168, 332] or specific complexomes [87, 124, 201]. While this was predominantly realized using gene expression data in former efforts, we recently showed with our tool PPIXpress [90] (see also Chapter 3) how transcript-level expression data can be utilized to construct protein interactomes that even account for alternative splicing in an input sample.

Classical computational methods that delineate protein complexes in binary protein-protein interaction networks (PPINs) typically aim at finding densely connected regions in such networks [193, 194, 196]. Such approaches are conceptionally very appealing since they resemble mathematical clustering and work well to detect all types of large self-contained protein complexes. Apart from those, the highly interconnected subgraphs that are reported by such methods are more likely functional modules comprising overlapping protein complexes that transiently interact with each other dynamically in time and space. Without additional data and modeling, protein complex prediction methods that only rely on PPINs are, even if the member proteins are expressed together at the same time and in the same cellular compartment, not able to distinguish if the involved protein interaction sites are mutually exclusive and the competition for those binding sites may thus encode a combinatorial manifold of potential complexes rather than one actual complex [143, 145, 146, 191].

Cases of mutual exclusive binding site competition in PPINs can be marked on the basis of spatial clashes found in structurally resolved parts of the interactome. With this additional information, all simultaneously viable subnetworks within predetermined dense regions in PPINs can be enumerated and feasible complexes then be identified by a subsequent prediction step [204]. Due to the exponential grow of the number of subnetworks that need to be processed, the computational cost of this approach is very high. Protein domains and the interactions between them were shown to present a practical alternative to the comparably sparse coverage of structural knowledge on interactomes. Then, proteins are dissected into their conserved domains and known domain-domain interactions (DDIs) can serve as the scaffold that explain protein interactions mechanistically. If each individual protein domain is only allowed to support one interaction, binding constraints can be inferred from this domain-domain interaction network (DDIN) model [17, 179, 180, 206–208]. Such DDIN-based approaches were used to filter predicted complexes to those subsets of members that are devoid of conflicting binding site utilization without enumerating all possibilities [179, 180], or the domains were used to define connectivity in stochastic simulations [206, 208].

With our domain-aware cohesiveness optimization algorithm DACO we filled the gap of a comparably fast complex prediction tool that also unravels the combinatorial diversity of complexes within dense regions of the interactome by combining the DDI model with local cluster optimization and branching [17].

Since we are now able to make use of the current wealth of RNA-seq data to contextualize the interactome input data for such a prediction with PPIXpress and therefore all data retrieval could be outsourced nicely, an improved implementation of DACO with a refined feature set and improved runtime appeared a worthwhile endeavor. We here present JDACO, the modern Java implementation of the original DACO Python prototype implementation, and show how it performs in comparison to its predecessor.

5.3 MATERIALS AND METHODS

5.3.1 *Domain-aware cohesiveness optimization*

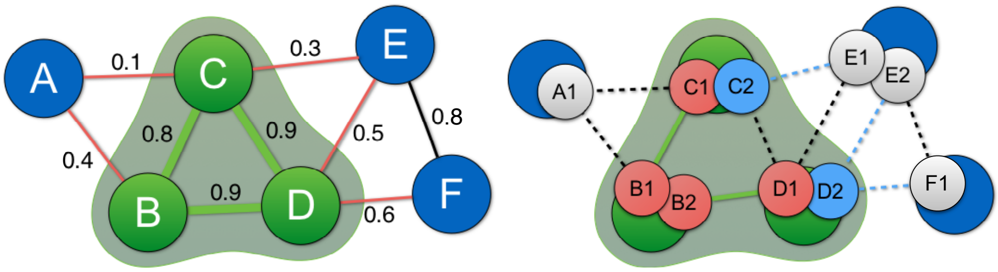
DACO fuses local greedy optimization of the cohesiveness as introduced by ClusterONE [196] (see also Section 2.1.3 for an introduction to the method) with the idea of modeling mutual exclusivity of interactions by also considering the connectivity among individual protein domains and constraining each domain to support at most one active interaction [179, 180]. Requiring connectedness on the finer-grained level of domains and this constraint, that serves to approximate the occupancy of shared binding sites, introduce a ruleset for branching into equally probable states during the optimization which ultimately enables a combinatorial enumeration of complex candidates.

Like ClusterONE, DACO uses a PPIN and a set of seed proteins as the main input. Additionally it uses the information of a DDIN that corresponds to the given protein interactome in the sense that each interaction on the protein level is backed by a domain interaction (see also Chapter 3, especially Figure 3.2). In the original Python implementation the DACO implementation took care of the retrieval and construction of a matching DDIN. For JDACO, the matching domain-based interactome is a mandatory user input that can be constructed with PPIXpress, for example. Figure 5.2 graphically outlines the information content that is provided by the two related interactome networks and aids to introduce some definitions.

The weighted PPIN is used to calculate the cohesiveness $f(V)$ of selected proteins V according to

$$f(V) = \frac{w^{\text{in}}(V)}{w^{\text{in}(V)} + w^{\text{bound}}(V)}.$$

Here, $w^{\text{in}}(V)$ is the sum of all inner interactions, namely all protein interactions that are among the putative complex members V (marked green in Figure 5.2a), and $w^{\text{bound}}(V)$ is the overall weight of interactions on the boundary between members and adjacent proteins that are not part of the candidate (shown red in Figure 5.2a). For simplicity and because modern protein interactomes, like PrePPI [163], are already densely populated, we omitted the penalty term that is found in the cohesiveness calculations of ClusterONE. Notably, only the weights of interactions on the protein-level and the protein set V to be evaluated are



(a) *protein-protein interaction network* **(b)** *domain-domain interaction network*

Figure 5.2: *The information represented in the two DACO network layers. A weighted PPIN (a) and its corresponding DDIN (b) are shown whereby the proteins that are part of a complex candidate consisting of proteins* $V = \{B, C, D\}$ *are marked in green. Protein names are omitted in the representation of the DDIN to make room for domain labeling. All inner interactions, protein interactions between members of the complex candidate or active DDIs between domains of such members, are also drawn in green in both networks. In the PPIN specifically, boundary interactions between complex members and other proteins are colored red because they define the boundary weights relevant for the cohesiveness calculations and are the potentially beneficial incident proteins that are not yet members of the complex. In the DDIN, domains that are occupied by active interactions are marked in red while the domains C2 and D2, the only unused domains in the complex candidate and therefore relevant to facilitate further expansion of the complex, as well as their respective potentially relevant domain interactions are highlighted in blue. Such model-compliant possibilities for expansion of the complex define the incident proteins, which are E and F here in the example. Also, proteins C and D are boundary proteins which are defined as those proteins that only have one domain occupied. Because a removal of such a protein does not disrupt the spanning-tree underlying the complex on the level of domain interactions they are the only valid options to shrink the complex in the next step.*

relevant in the cohesiveness calculations whereas no knowledge of any kind on the domain-level is utilized for this task. By combining the holistic information of many weighted interactions it is expected that unreliable individual values are averaged out reasonably when ranking complex candidates [196].

Just as the PPIN serves as the base for the optimization metric, the DDIN acts as the main determinant of which further protein connections are valid in the model and thus permissible expansions of the current complex candidate. Following previous approaches [179, 180], the DACO method requires that a protein complex that fulfills the model assumptions is connected by a spanning-tree of active domain interactions in the DDIN (green interactions in Figure 5.2b). Since each protein domain is limited to enable only one interaction, a complex candidate can only be expanded by including proteins which possess at least one domain interaction connected to an unoccupied domain of a current complex member (blue interactions in Figure 5.2b). Such potential candidates for enlarging the complex are called incident proteins (see also Figure 5.2b). Likewise, only those proteins in V which have only one domain engaged are suitable candidates to reduce the size of the complex. These are called boundary proteins in the following explanations of the algorithm (see also Figure 5.2b). In our example in Figure 5.2b, protein B does not comply with this definition, for example. Its removal from complex candidate V would disconnect proteins C and D on the domain-level and thus violate the model assumptions by breaking the loop invariant of connectedness.

Contrary to ClusterONE, DACO starts the iterative search for protein complexes from pairs of proteins interacting with high confidence rather than clusters of single proteins. In practice this means that starting states for the algorithm are determined from all interactions of seed proteins with their direct neighbors that have a weight exceeding a threshold value P_{pb}. We decided on that strategy because interactome data is quite noisy [170] and, because the least amount of network data and weights are integrated at this point, this very first step of the expansion process is most prone to even negligible perturbations. Then, although several almost equally well-rated alternative branches may exist at this starting point, one would invariably bias the whole optimization towards one single local minimum. Broadening the ensemble of starting points, on the other hand, assists to better grasp the combinatorial manifold of complexes and ensures that no reasonable complex is lost.

At each step, the current state of the protein complex candidate V and its utilized domain interactions D_V are used to determine incident and boundary proteins on the domain-level (recall Figure 5.2b). All options to modify V to V' by the addition of an incident protein or by the removal of a boundary protein are then evaluated in terms of the resulting cohesiveness $f(V')$ and the choice that maximizes the measure is selected. If neither adding nor removing a protein can further increase the cohesiveness compared to $f(V)$, V is already locally optimal and thus returned. If the removal of a protein leads to the highest cohesiveness, the current state is adapted by removing this protein from V and by deactivating the distinct domain interaction that was mediated by its single occupied domain. Then the next iteration of the algorithm is conducted for the modified state. Most of the times during the execution, a protein p will be added to the complex candidate. Naturally, V' is then becoming $V \cup \{p\}$ and, additionally, any expansion requires added connectivity on the domain-level in our model. Due to the definition of incident proteins there is then at least one domain interaction that is qualified to accomplish this. If we only need to consider one possibility, this exact domain interaction is stored as active in $D_{V'}$, rendering the newly interacting domains unusable in subsequent iterations, and we continue to iterate. Often, however, more than one domain interaction may serve as the spanning-edges to include p and the option selected will affect the occupancy of domains (see Figure 5.3) and thus heavily influence subsequent steps. Because the metric optimized is identical for all choices, the algorithm then simply evaluates all possibilities. This case is also exemplified in Figure 5.3. For practical considerations a maximal search depth parameter that concludes the calculations with a certain complex size is obligatory in DACO. In the project presented hereafter in Chapter 6, we showed that 5 proteins are a sufficient size limit for DACO when human transcription factor complexes are predicted, for example (see last part of Section 6.4.1).

Pseudocode that includes some of the optimization details for the procedure in each DACO iteration (see Algorithm 5.1) and the management of the search-tree exploration (see Algorithm 5.2) are presented after a short overview on implementation enhancements that already improved the runtime tremendously (as shown in my Master thesis [18]).

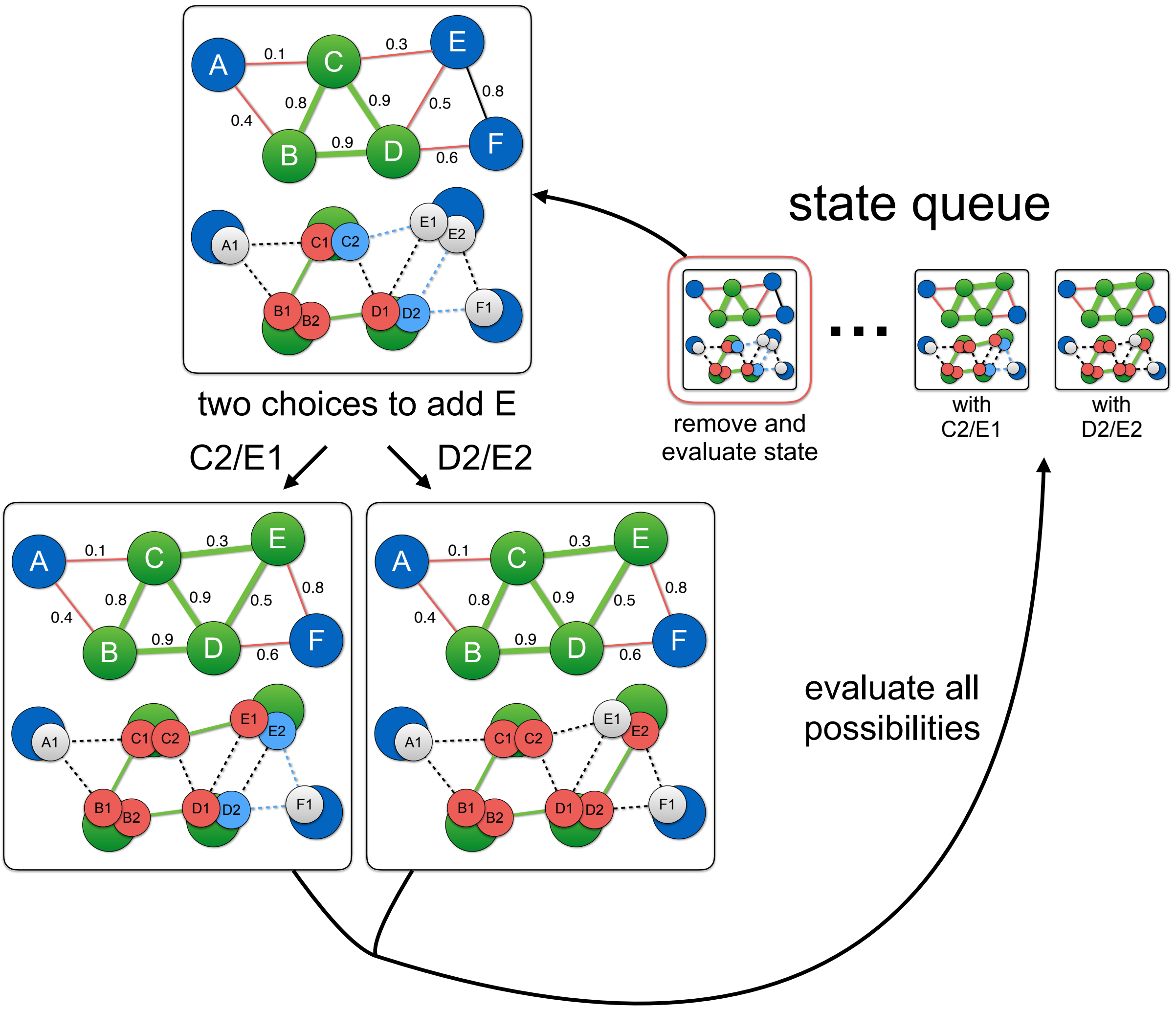


Figure 5.3: *Branching in DACO exemplified. All states that are left to be investigated are organized in a queue that is processed one by one. In the current step, merging protein E into the complex candidate is the most beneficial choice in terms of the cohesiveness. Since two domain interactions, C2/E1 as well as D2/E2, are suited to guarantee connectivity in this example, both realizations of the resulting candidate complex are appended to the queue of pending states and evaluated independently. The exploration of the search space is done in a breadth-first manner. This branching and the processing of all options are important because the exact domain choices predetermine the capability for expansion of the protein complex in later steps. Here, for example, selecting C2/E1 still allows to include protein F in a further step whereby selecting D2/E2 does not allow for further additions.*

Algorithmic optimization in the original DACO implementation

When total outer weights are precomputed for each protein in the network, the naïve computation of the cohesiveness $f(V)$ requires a computational effort that is quadratic in the size of the complex candidate, thus $O(|V|^2)$. With a little bit of bookkeeping it is not necessary to compute it from scratch in each iteration of the algorithm, though. As before, $w^{in}(V)$ and $w^{bound}(V)$ are the internal and boundary weights of V. Those have already been computed in a previous step. Now w^{in}_p additionally denotes the summarized weight of all interactions connecting protein p with members of V. In the same fashion, w^{bound}_p denotes the total weight of interactions between p and proteins that are not members of V. For the case of an addition $V' = V \cup \{p\}$, the inner weight $w^{in}(V \cup \{p\}) = w^{in}(V) + w^{in}_p$ and the boundary weight $w^{bound}(V \cup \{p\}) = w^{bound}(V) - w^{in}_p + w^{bound}_p$ can be defined from the old weights and the changes induced by the inner and outer weight contributions of the new protein p. Determining w^{in}_p and w^{bound}_p is only a linear effort in each step. Analogously this can be applied for the removal of a protein p. Then the inner weight is $w^{in}(V' = V \setminus \{p\}) = w^{in}(V) - w^{in}_p$ and the boundary weight is $w^{bound}(V' = V \setminus \{p\}) = w^{bound}(V) + w^{in}_p - w^{bound}_p$. Updated cohesiveness values can thus be simply derived by saving and recalling the previous inner and boundary weights as well as monitoring the changes induced by adding/removing p. Please refer to the Supplementary Section 1.2 of the original DACO publication [17] for full cohesiveness equations of such a stepwise cohesiveness calculation. This optimized implementation of the cohesiveness calculation was omitted in the pseudocode algorithm for sake of a better readability.

The most expensive outcome of each DACO iteration is the branching of the algorithm into many different realizations on the domain-level (see Algorithm 5.1, line 33). Often proteins include the very same domain family annotation multiple times. Then, every domain of the same type is connected to the very same domain(s) in each incident protein. For the algorithm, however, only one realization of each combination of protein types is relevant for a neighboring protein pair because they will all have the same connections to the outside and therefore the consequences in subsequent steps will be identical. To avoid unnecessary branching, DACO only considers one variant for such cases when domain interaction choices are determined.

Also, even though the cohesiveness of the resulting complex candidate is always the same when the branching into the specific domain interaction choices is performed, the respective spanning-tree that connects the members differs. Whereas the domain interactome itself does not have any qualitative rating of its interactions that could be exploited at this point, current integrative protein interaction data [161, 162, 164, 177] are often weighted in a range that can be interpreted as probabilities (see also the respective part of Section 2.1.3). Thus an overall likeliness of the tree connecting all members of V can be assigned by multiplying the corresponding protein interaction weights associated with the domain interactions that are active in the state (see Algorithm 5.1, line 27 and 34). This probability can then serve as a pruning criterion that filters unlikely realizations at an early stage (see Algorithm 5.1, line 35).

Another technique that was used to minimize the calculation time was to include memoization and thus to avoid visiting the same states and their outcomes more than once. In the DACO prototype this was implemented by storing the actively selected domain interactions D_V of each state that was processed (see Algorithm 5.2, line 13). If this exact state is then detected another time, the processing of this subtree of the search is simply skipped (see Algorithm 5.2, line 7).

5.3.2 *Functional alterations made to original algorithm and its implementation*

When growing starting pairs for the cohesiveness optimization only the pairs above the pair-building threshold P_{pb} are taken into account. In the original implementation two pairs per seed protein were always included even if they did not pass this threshold.

Also, we added a user-adjustable complex probability threshold parameter P_c into the algorithm instead of applying the fixed cutoff of 0.5 (compare Algorithm 5.1, line 35). When P_c is not explicitly specified by the user, it is automatically set to $(P_{pb})^{max\_depth-1}$. This means that we basically want a spanning-tree on the domain-level that is on average at least as likely as the starting interactions that are selected.

Furthermore, resulting complexes that include no seed proteins are removed in the final postprocessing. Although this rarely happens in practice it cannot be ruled out due to the removal step.

Adaption made to Python prototype

The original Python implementation of DACO retrieved all annotational data that the tool needed to construct the input networks protein by protein from UniProt [109]. While it was capable to do so by itself, the construction of the input data by PPIXpress is much faster, more robust and generally more convenient. More so, PPIXpress allows to contextualize the input data in a sample-specific manner. Therefore we completely removed the data retrieval part in the adapted Python DACO and added new code for DDIN/PPIN format input as given by PPIXpress. The core algorithm was then adjusted as described above. The old as well as this new adapted implementation of the DACO prototype are available at `https://sourceforge.net/projects/dacoalgorithm/`.

5.3.3 *Optimized software design, new technologies and features*

The JDACO implementation was completely rewritten from scratch in Java 8 with the goals of exploiting multithreaded computations for the benefit of improved scalability as well as following a clear modular design that separates the data retrieval functionality for an optimal usage together with PPIXpress.

In the original DACO implementation a single loop basically controls the complete algorithm (see Algorithm 5.2). The loop manages the one-by-one evaluation of all states encountered by executing the step-function (see Algorithm 5.1) and also decides what is done with its output, e.g. it takes care

Algorithm 5.1 Domain-aware cohesiveness optimization step function: step(V, D_V, P) with current proteins V, active domain interactions D_V and current probability P

```
determine $V_{inc}$ and $V_{bound}$ from the DDIN and $D_V$
max $\leftarrow f(V)$
action $\leftarrow$ terminate

for $\forall p \in V_{inc}$ do
    $V' = V \cup \{p\}$
    if $f(V') >$ max then
        max $\leftarrow f(V')$
        action $\leftarrow$ add $p$
    end if
end for
for $\forall p \in V_{bound}$ do
    $V' \leftarrow V \setminus \{p\}$
    if $f(V') >$ max then
        max $\leftarrow f(V')$
        action $\leftarrow$ remove $p$
    end if
end for

if $action$ = terminate then
    return complex candidate $V$

else if action = remove $p$ then
    $V' \leftarrow V \setminus \{p\}$
    determine domain interaction $d \in D_V$ that connected $p$ to $V'$ in DDIN
    $D_{V'} \leftarrow D_V \setminus \{d\}$
    $P' \leftarrow P/$ (weight of $d$'s equivalent interaction in PPIN)
    return compute job with parameters ($V'$, $D_{V'}$, $P'$)

else if action = add $p$ then
    $V' \leftarrow V \cup \{p\}$
    $l \leftarrow$ empty list
    for $\forall d \in$ DDIN so that $D_V \cup d$ connects $V'$ in the DDIN do
        $P' \leftarrow P*$ (weight of $d$'s equivalent interaction in PPIN)
        if $P' \geqslant 0.5$ then
            append compute job with parameters ($V'$, $D_V \cup d$, $P'$) to $l$
        end if
    end for

    if $|l| = 0$ then
        return complex candidate $V$
    end if
    return list of compute jobs $l$
end if
```

Algorithm 5.2 GrowthManager(starting_pairs, max_depth) with starting pair states *starting_pairs* and complex size limit *max_depth*

```
results $r \leftarrow$ empty set
memoized states $m \leftarrow$ empty set
state or compute job queue $q \leftarrow$ initialize with *starting_pairs*

**while** $|q| > 0$ **do**
    state description $(V, D_V, P) \leftarrow q.pop()$
    **if** $D_V \in m$ **then**
        continue
    **else if** $|V| = max\_depth$ **then**
        $r.add(V)$
    **else**
        result $j \leftarrow step(V, D_V, P)$
        $m.add(D_V)$
        **if** result $j$ is a set **then**
            $r.add(j)$
        **else if** result $j$ is a list of compute jobs **then**
            extend queue $q$ by jobs in result $j$
        **else if** result $j$ is a single compute job **then**
            $q.append(j)$
        **end if**
    **end if**
**end while**
**return** results $r$
```

of complex candidates returned or new branches that need to be considered. JDACO follows a more decentralized software design to embrace the independence of worker threads and thus to increase the potential efficiency of parallel operations. Figure 5.4 outlines how the individual parts of the algorithm are working together. The new architecture, some relevant technical details as well as novel features are introduced in the following.

Starting calculations in JDACO

For reasons discussed earlier, the DACO algorithm starts its iterative optimization of candidate complexes from starting pairs rather than single seed proteins. Pairs are constructed from the user-defined seed proteins and their adjacent interaction partners for which the confidence in the interaction exceeds a given weight threshold. In the new implementation we added the optional function to set and determine a percentile of the distribution of all the interaction weights as the respective cutoff.

The default behavior of JDACO follows the previous DACO implementation and runs all distinct starting pair states determined for each seed protein one after another in independent search processes analogously to the procedure described in Algorithm 5.2. This approach has the practical advantage that users can follow the progress of the calculations in the sense that there is a

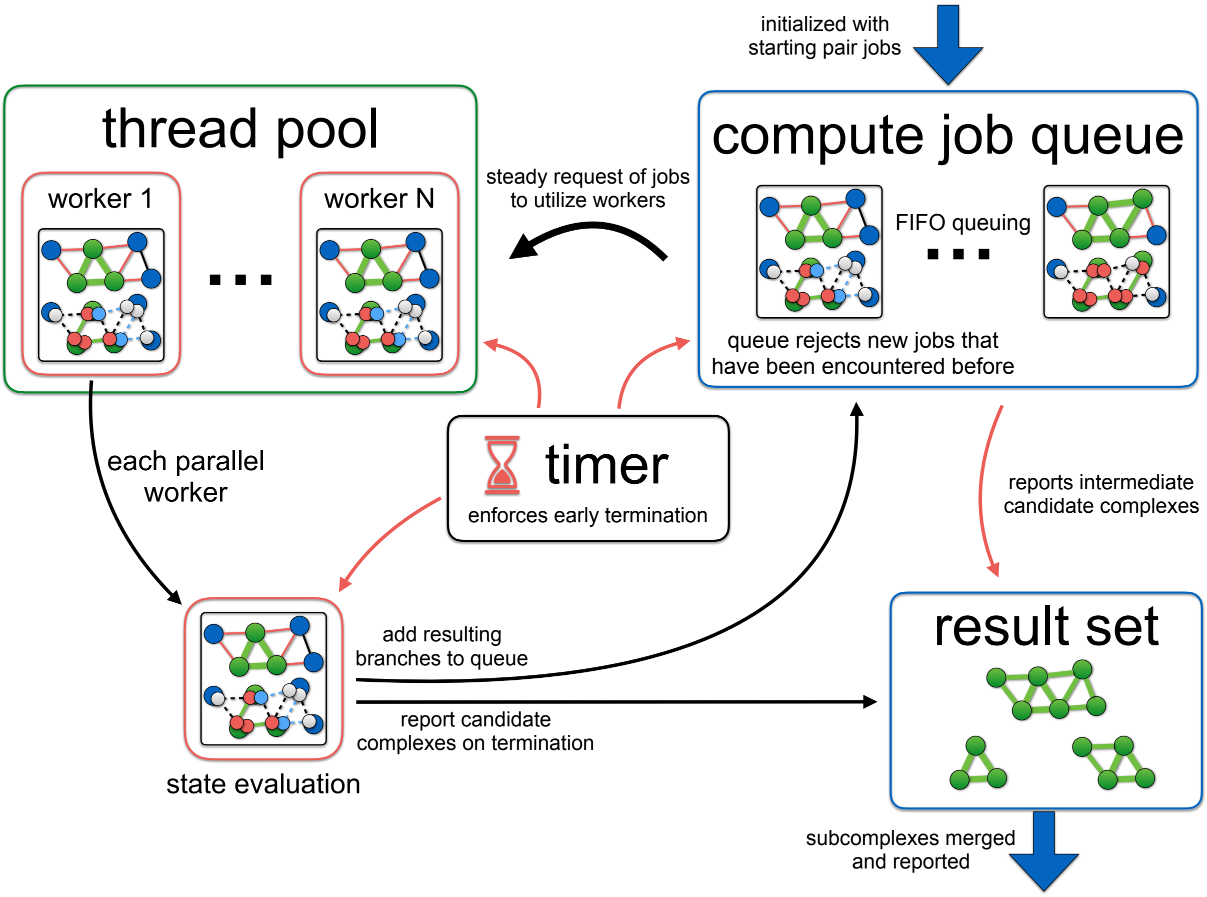


Figure 5.4: *Algorithmic responsibilities in JDACO. Black arrows in the overview depict the default processes that are performed by the specific part of the algorithm whereas red arrows are only active when early termination is enforced. Please refer to the main text for a comprehensive introduction.*

direct feedback of the currently processed seed protein, its number of starting states (but only those that have not been encountered before), the protein complexes that were predicted from those starting states and also how many seed proteins are left. Organizing the compute jobs in such a manner poses no disadvantages in singlethreaded operation. For optimal utilization of many threads, on the other hand, it is beneficial to gather all starting pairs in the work queue independent of the seed protein they stem from. In JDACO the user can enable a "high-performance mode" which does exactly that and sacrifices the output on the progress for a better utilization of the hardware.

Worker management and responsibilities

At the core, the basic procedure of DACO as explained in Section 5.3.1 was retained. All designated starting states are initially put into a queue that is processed and also filled during the execution of the DACO algorithm. Since all statements here refer to an actual implementation rather than an explanation of the algorithmic core principle and because the overall processing is a different one, I will also use the term *compute job queue* here. This queue is part of a custom *thread pool* executor implementation in JDACO which allows to process the queued jobs, the algorithmic states that are not yet evaluated, in a multithreaded manner. Since the *LinkedBlockingQueue* implementation that was used follows the first in - first out (FIFO) principle and because most of the evaluations will lead to an extension of the complex candidate and should have comparable execution times, the search tree is approximately traversed in a breadth-first manner.

In contrast to the old implementation in which the function controlling the state queue basically governed the scheduling of the calculations but also the management of the results, the thread pool's area of responsibilities in JDACO is condensed to utilizing its worker threads to full extent. By enabling encapsulated access to the queue in this custom implementation, the worker threads cannot only carry out state evaluations (see Algorithm 5.1) concurrently, but, at the same time, they are empowered to extend the compute job queue themselves on the fly if branching occurred and they are directly reporting final candidate complexes to the *result set*. Thus not only the algorithmic evaluation but also the subsequent management overhead was shifted into the range of tasks that is run in parallel.

When a worker processes a job that yields new branches that need to be evaluated and thus wants to append the jobs to the compute job queue, the custom scheduler ensures to only accept jobs that have not been in the queue before. It does that by maintaining a set data structure derived from *ConcurrentHashMap*. This enables concurrent modification of the set with minimal waiting times and a fast membership check given the hashing is done in a good way. Analogously to the memoization approach of the old implementation, the active domain interactions are used to define each algorithm state and therefore are able to ensure the uniqueness of computations. Each domain interaction in JDACO is stored as a data structure holding an ordered string pair (s_l, s_r). Because collisions, unequal objects that are mapped to the same bin in such hashing data structures, penalize the runtime of such set-membership checks, the implementation aims to nicely spread the elements stored by employing a hashcode computation that is hand-tuned according to best practices [328]. In this case the hashcode is computed as $p * h(s_l) + h(s_r))$ where p is a prime number and $h(x)$ means the hashcode of x.

Another case for which the high degree of control over the internal data structures in Java was exploited beneficially by tailoring them to the task at hand was the storage of performance relevant data accessed by the worker threads. As an example, the mapping of proteins to their domains and the mapping between domains, so basically the DDIs, are stored as simple arrays in the maps rather than as managed data structures like an *ArrayList*, the dynamic equivalent which uses an array internally. Because the network object is initialized once and never changed again, convenience and abstraction features of the storage data structure are irrelevant. Therefore one rather benefits from the constant factor that is gained in terms of iteration and data retrieval speed simply because the inherent internal overhead, e.g. by general memory allocation and access management as well as sizing checks, is considerably smaller for a bare-bones array.

Default handling of complex candidates and early termination

If the algorithm terminates, the executing worker thread itself will directly report the resulting complex candidate to the *result set* which is internally stored as a *ConcurrentHashMap* used as a set. Using a concurrent data structure for this task allows the workers to simultaneously report candidate complexes in a way that minimizes potential waiting times by locking and should thus serve best for the overall performance.

At last, JDACO allows to optionally specify a runtime limit for each starting pair state (or all starting pair states in high-performance mode). This feature was made possible by the new concurrent design. When enabled, a timer is started with the algorithm execution and when the specified duration has passed, all busy calculation threads and workers are stopped, the timeout is reported to the user, and intermediate results are collected from workers and job queue and finally transferred to the result set (see also Figure 5.4). In practice this means a higher *max_depth* setting can be used for a run and the overall time can still be kept within a manageable time even when individual pathological cases are appearing in the course of the execution.

5.3.4 *Evaluation data and methodology*

Interactomes and seed proteins

All input networks, protein-protein and domain-domain interaction networks, were constructed using PPIXpress (version 1.20) [90]. We used the option to construct sample-unspecific networks which means the complete networks were used and there was no contextualization step that tailored the networks according to an expression data input. Furthermore, UniProt accessions were updated automatically by the tool. In the process, data from Ensembl (release 94) [69], 3did (release July 2018) [107] and UniProt (release 2018_10) [109] were retrieved and utilized by PPIXpress.

For yeast, we used the same data as in the original DACO publication [17]. The yeast PrePPI network [177] was taken as the input interactome and the 148 transcription factors of the Yeast Promoter Atlas [412] were used as seed proteins. Please refer to [17] for details on this data.

For benchmarks on human data we used the latest version of the human PrePPI network as in the publication on CompleXChange [163] (see also Chapter 6) and additionally retrieved a current release of the mentha PPIN (version of 26.11.2018) [161] with PPIXpress. For both human protein interactomes we used the full set of 678 transcription factors in HOCOMOCO (version 11) [220] as seed proteins.

Relevant sizes of the respective input datasets contrasted are shown in Table 5.1.

Benchmark setup

To benchmark and compare the two implementations we used the JDACO 1.03 binary (in high-performance mode) and the adapted DACO prototype version 1.02 with caching enabled. Both implementations in respective versions

	PrePPI (yeast)	mentha (human)	PrePPI (human)
proteins (with domain annotations [%])	6,191 (61.7)	19,215 (51.4)	18,449 (70.9)
protein interactions (with associated domains [%])	232,554 (20.9)	337,525 (17.8)	1,527,283 (37.7)
domain interactions	299,364	641,075	10,665,429

Table 5.1: *Sizes of the three DACO implementation benchmark datasets.*

	PrePPI (yeast)	mentha (human)	PrePPI (human)
starting-pair threshold P_{pb}	0.75	0.454	0.988
complex probability cutoff P_c	0.5	0.019	0.952
max_depth	10	6	5

Table 5.2: *Parameters used for each benchmark dataset. The parameter set applied to the yeast data was taken from the original DACO publication [17]. For the human datasets, starting-pair thresholds P_{pb} were set to the upper 10% (or 90th percentile) of the corresponding network data weights using the percentile function. The respective domain spanning-tree probability cutoffs P_c were set automatically by JDACO according to $(P_{pb})^{max\_depth-1}$ and max_depth parameters were set to yield a computational effort of comparable order of magnitude across the benchmark inputs.*

are made available on the SourceForge page of the DACO project: `https://sourceforge.net/projects/dacoalgorithm/`. The applications were ran with Oracle's Java Runtime Environment 8.192 and Python 2.7.15 as supplied by Ubuntu Server 18.04.2 on a server with two Intel Xeon Gold 6138 processors (2 GHz, together they have 40 cores/80 threads by simultaneous multithreading which is a technique to increase the utilization of processor architectures by allowing the parallel execution of multiple independent threads in a physical core).

We applied JDACO with the same core parameters as in the original DACO publication [17] to the yeast data. For the input data on human, we used the new percentile function with the 90th percentile (upper 10%) of all interaction weights in respective PPINs as a guidance to determine a weight threshold P_{pb} for the construction of seed pairs. Maximal search depth parameters *max_depth* were set to result in runtimes of comparable scale for all inputs and complex probability cutoffs P_c were set automatically by JDACO. Table 5.2 lists all parameters in detail.

All computations were run 5 times to ensure appropriate sampling of runtimes.

5.4 RESULTS AND DISCUSSION

Since all measurements were repeated, the average speedup factor was derived as the mean runtime of PDACO, termed P_{avg}, divided by mean runtime of JDACO, abbreviated by J_{avg}. Deviations of the speedup factor were based on the extreme distances of the respective standard deviations P_{std} and J_{std} of the replicated experiments. Thus the upper limit of the speedup factor was derived

	PrePPI (yeast)		mentha (human)		PrePPI (human)	
method	runtime [s]	speedup factor	runtime [s]	speedup factor	runtime [s]	speedup factor
PDACO	31154.0 ± 215.7	/	5379.0 ± 35.0	/	31397.0 ± 571.7	/
JDACO (1 thr)	7652.4 ± 123.1	4.1 (4.0 – 4.2)	952.4 ± 12.9	5.6 (5.5 – 5.8)	1267.4 ± 16.4	24.8 (24.0 – 25.6)
JDACO (2 thr)	3860.8 ± 23.2	8.1 (8.0 – 8.2)	496.2 ± 7.7	10.8 (10.6 – 11.1)	648.2 ± 5.3	48.4 (47.2 – 49.7)
JDACO (4 thr)	2049.6 ± 8.8	15.2 (15.0 – 15.4)	252.0 ± 1.9	21.3 (21.0 – 21.6)	349.6 ± 3.7	89.8 (87.2 – 92.4)
JDACO (8 thr)	1107.4 ± 10.0	28.1 (27.7 – 28.6)	137.0 ± 1.4	39.3 (38.6 – 39.9)	200.8 ± 3.2	156.4 (151.1 – 161.8)
JDACO (16 thr)	632.4 ± 29.3	49.3 (46.8 – 52.0)	74.4 ± 2.1	72.3 (69.9 – 74.8)	124.6 ± 4.8	252.0 (238.3 – 266.8)
JDACO (32 thr)	399.4 ± 23.0	78.0 (73.2 – 83.3)	44.0 ± 0.9	122.2 (119.0 – 125.6)	83.4 ± 5.5	376.5 (346.7 – 410.4)
JDACO (64 thr)	305.0 ± 6.5	102.1 (99.3 – 105.1)	34.0 ± 0.6	158.2 (154.3 – 162.3)	62.8 ± 1.2	500.0 (481.9 – 518.7)

Table 5.3: *Benchmark results for PDACO and JDACO. Deviations of the speedup factor are presented as the extreme distances of the respective standard deviations* P_{std} *and* J_{std} *of the replicated experiments. The upper limit of the speedup factor is therefore shown as* $\frac{P_{avg}+P_{std}}{J_{avg}-J_{std}}$ *and the lower limit as* $\frac{P_{avg}-P_{std}}{J_{avg}+J_{std}}$.

by $\frac{P_{avg}+P_{std}}{J_{avg}-J_{std}}$ and the lower limit given as $\frac{P_{avg}-P_{std}}{J_{avg}+J_{std}}$. The results of the runs are listed in Table 5.3 and visualized in a log-log plot in Figure 5.5.

Independent of the exact test dataset or number of threads allowed to be used by the implementation, JDACO determined protein complexes significantly faster than PDACO in every single test case. Even without multithreading PDACO took at least 4 times longer in our benchmark examples. In practically relevant core counts for modern laptops and workstations, 4 to 16 threads, a speedup of at least one and up to two orders of magnitude was measured. The factor by which JDACO is sped up relative to its predecessor implementation also seemed to relate to the size of the input data because the runtime advantage was larger for more demanding datasets (see the respective number of domain interactions in Table 5.1).

As can be seen in the relatively straight lines of the graphical depiction of the speedup factor in Figure 5.5, the performance of JDACO also scales very well with the amount of available cores until 32 threads. After that, although the performance still increases, the slope shows a slight decline across all datasets. Because the hardware setup that was used for all computations had only 40 physical cores this relative slowdown can likely be attributed to the loss of efficiency by the utilization of virtual cores in simultaneous multithreading, i.e. two threads sharing pipelines and caches within a single CPU core, rather than actual physical units.

5.5 CONCLUSION

The original Python prototype implementation of the DACO algorithm was able to automatically retrieve the necessary annotation data to construct a DDIN and relate it to the given input PPIN. Those interdependent networks were then used to determine the manifold of combinatorial protein complexes around input seed proteins. Splitting data preparation and complex prediction into PPIXpress and JDACO has several practical advantages that allow approaching vast new areas of application. Besides retrieving the necessary data for integration

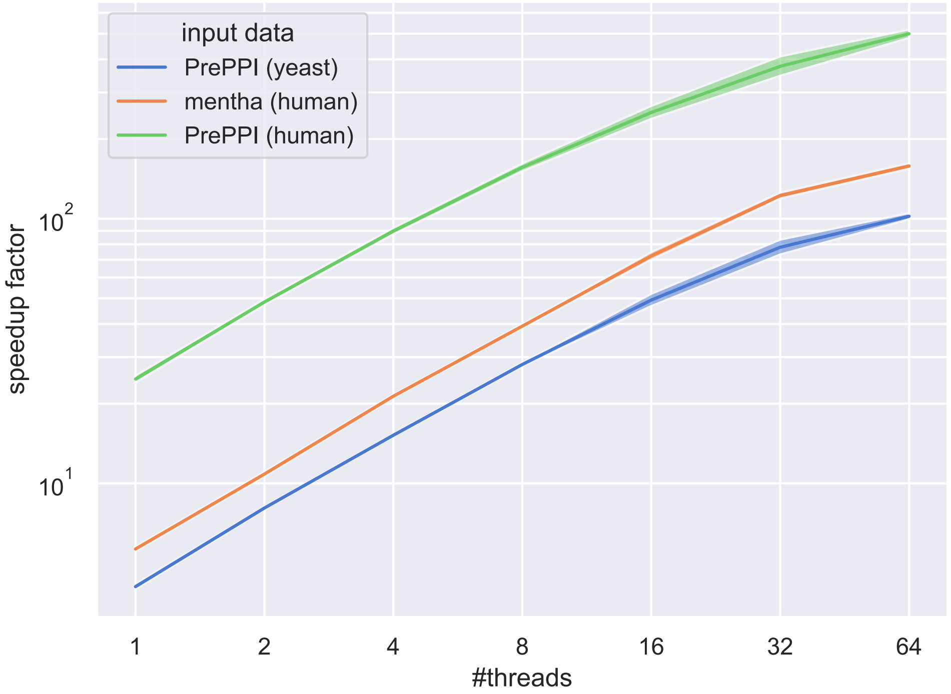


Figure 5.5: *Speedup and scalability of JDACO compared to PDACO. The speedup factors of calculations conducted using JDACO with a specific number of threads being utilized in relation to those calculations done by PDACO are shown in a log-log plot. The speedup factor was defined as the ratio of mean runtimes of the measurements made for the implementations and the shaded regions around the averaged speedup factors depict the deviations as defined in the main text.*

much faster, PPIXpress allows to construct the input networks in a sample-specific manner and unprecedented transcript resolution, or unspecific, as before. JDACO, the new Java implementation of the DACO algorithm, offers convenient new features and, more importantly, has a substantially better performance than its Python predecessor. While DACO was absolutely sufficient to determine general complexomes in an organism-wide way, the usage of JDACO with PPIXpress effortlessly allows to scale up the granularity to sample-dependent ensembles of protein complexes by harvesting from the wealth of available transcriptome studies. A possible application would, for example, be to outline differential complexomes with CompleXChange [87] (see also the next Chapter 6).

5.6 ADDENDUM

5.6.1 *Retrospective*

Originating from my master thesis, the DACO algorithm and all concepts around its application to find combinatorial protein complexes from seed proteins are the chronologically oldest projects in this thesis. I put the idea of a faster implementation into practice right after starting the core classes of the

Java framework underlying all my software developmental efforts. Only after that I started with the development of PPIXpress and the other tools.

Of course, implementations in other programming languages like C/C++ would have been possible and may have been even faster than the current Java implementation JDACO. For example, I thought about the possibility of speeding up the application even more by exploiting the general compute capabilities of modern graphics cards (or graphics processing units (GPUs)). For architectural reasons that I could not work out here, though, GPUs only work optimal for data parallelism, thus when a large amount of data needs to be processed in the very same way using the very same progression of instructions on a fixed set of input data. Even simple flow-control instructions like if-statements would enforce high performance penalties. More so, elementary data structures, like dictionaries are extremely difficult to realize and topics of research [413, 414]. Simple block based parallelism as in matrix computations, e.g. just partitioning clear badges of compute jobs, is also not possible because the search space of this optimization problem folds up at runtime.

5.6.2 *Outlook*

Contextualization by PPIXpress and the huge reduction of compute time by JDACO render the investigation of dynamic complexomes practically feasible even for large sample sizes. Besides the splicing machinery [415] or even the Mediator complex [416] in general, potentially interesting study targets in the area of gene regulation that are even less studied than transcription factor complexes, could be complexes involving proteins that conduct posttranslational modifications of histones and DNA [223] or RNA-binding protein complexes [417], for example.

More generally, application to other types of complexes that also include defined classes of proteins and are likely combinatorial within their modules in nature are certainly worthwhile. Potentially interesting candidates could be complexes relevant for signaling pathways or in cell-cycle control [191].

6 DIFFERENTIAL ANALYSIS OF PROTEIN COMPLEXES WITH COMPLEXCHANGE

This chapter is concerned with the estimation of protein complex abundances and their differential analysis using the tool CompleXChange. Besides an extensive assessment of its methodology, the results of a differential analysis concerning human monocyte subtypes are shown. Sections 6.2 to 6.5 were adapted and expanded from Will, T. and Helms, V., "Differential analysis of protein complexes with CompleXChange", *BMC Bioinformatics*, 2019 [87]. I initiated this project and the study, designed and implemented the software, performed data analysis, conceived the figures and wrote the original manuscript. Volkhard Helms aided in designing the study, interpreting the data as well as editing of the manuscript. Supplementary materials that are published were omitted here, please refer to the online materials `https://doi.org/10.1186/s12859-019-2852-z`. A platform-independent Java binary, a user guide with example data and the source code are freely available at `https://sourceforge.net/projects/complexchange/`.

6.1 PREREQUISITES

6.1.1 *Linear programming*

Linear programming (LP) is an approach to optimize the outcome of a linear objective function that is subject to a set of linear constraints and dates back until the 1940s [418].

An instance of a LP task is called a linear program. In the so-called standard form of linear programs, the problem is defined by n numbers $c_1, c_2, \dots, c_n \in \mathbb{R}$ that define the objective function that should be maximized, as well as m numbers $b_1, b_2, \dots, b_m \in \mathbb{R}$ and mn numbers $a_{ij} \in \mathbb{R}$ (with $i \in \{1, \dots, m\}$, $j \in \{1, \dots, n\}$) that shape the $m + n$ constraints. Then LP determines those $x_1, x_2, \dots, x_n \in \mathbb{R}$ that

$$\begin{aligned} &\text{maximize} && \sum_{j=1}^{n} c_j x_j \\ &\text{subject to} && \sum_{j=1}^{n} a_{ij} x_j \leqslant b_i, \quad \forall i : 1, \dots m \\ & && x_j \geqslant 0, \quad \forall j : 1, \dots n. \end{aligned}$$

Arbitrary linear programs, which do not necessarily include non-negativity constraints ($\forall x_j \geqslant 0$) and may use linear equality constraints or constraints using greater-than-or-equal-to relations, can always be converted into standard form [328].

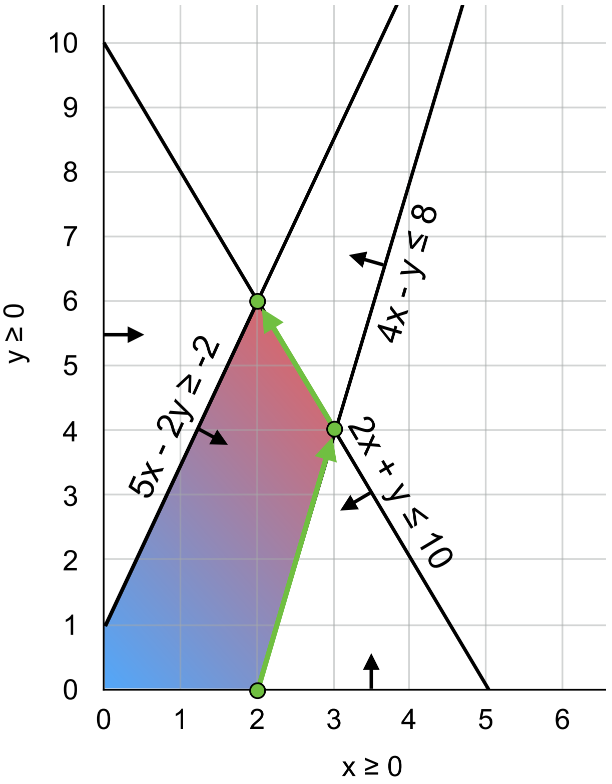


Figure 6.1: *Example of the linear program outlined in Equation 6.1. Each constraint is shown as a hyperplane (here in 2D: a line) that partitions the solution space and a direction which side of the separating plane is allowed by the constraint. The feasible region that is delimited by the constraints is shaded whereby the color gradient accentuates the increase of the objective value. The green arrows and points depict a possible path of a simplex algorithm run that finally finds the optimal solution with $x = 2, y = 6$.*

More of the basic terminology on LP can be best clarified with the aid of an example. Imagine the following optimization problem:

$$\begin{aligned} \text{maximize} \quad & x + y \\ \text{subject to} \quad & 4x - y \leqslant 8 \\ & 2x + y \leqslant 10 \\ & 5x - 2y \geqslant -2 \\ & x, y \geqslant 0. \end{aligned} \tag{6.1}$$

Small linear programs of only two variables can often be illustrated nicely. Figure 6.1 shows a visualization of the example stated in Equation 6.1. Here, the constraints outline the feasible region in which every choice of x and y satisfies all constraints. It can be shown that if the objective function has a maximum value within the feasible region, then the optimal solution is an extreme point on the boundary of the feasible region [328]. Thus to solve the optimization task one only needs to consider such extreme points. Unfortunately, depending on the problem size the number of points to be considered can still be very large.

The very popular simplex method [419] uses this property and operates on the vertices of the polytope defined by the feasible region which is termed the

data class	classified *positive*	classified *negative*
reference *positive*	true positive (tp)	false negative (fn)
reference *negative*	false positive (fp)	true negative (tn)

Table 6.1: *Confusion matrix in binary classification. For simplicity, Boolean class descriptions are assumed.*

simplex. Starting on any vertex, the algorithm moves along the edges of the simplex and iteratively optimizes the objective value by following the local optimum, e. g. by proceeding to the most promising neighboring vertex. If no further increase of the objective value is possible, a local optimum is found and reported as the final result. This is possible because the feasible region is convex and therefore a local optimum is also a global optimum [328]. Figure 6.1 shows the principle for our small example problem. The open-source solver lpsolve [420] that we applied in the following project implements a revised version of the simplex method.

Another successful class of methods are interior-point methods that, as the name suggests, approach the optimal solution from the interior of the feasible region [421].

6.1.2 *Classification of data*

Classification is the task of assigning a label or category to a sample which is inferred from features of the sample and prior knowledge on training samples for which the labels are known [320]. The first mathematical description of an approach grouping a new sample into one of two classes was Fisher's linear discriminant analysis [422]. Since the samples of the training data are necessarily labeled, classification belongs to the class of supervised learning methods. For an example of a unsupervised learning approach, see Section 4.1.1 on the clustering of data.

A crucial step in the workflow of classification is the selection of an appropriate machine learning model. The model in that sense is defined by the features that are used to infer the predictions, the actual method that implements the classification as well as its specific tuning parameters that need to be set [235]. Suitable performance measures are needed to decide on which features, classifier and parameters are suited best for a task at hand and to conduct a final assessment of the predictions. For simplicity, Boolean class descriptions are assumed for the following explanations. Standard measures of classification performance are based on the number of correctly classified samples (true positives and true negatives) and incorrectly classified ones (false positives and false negatives). The definitions are clarified by the confusion matrix shown in Table 6.1. Based on the numbers of those four counts, common performance metrics such as the accuracy, precision, recall (also called sensitivity) or specificity can be calculated as given in Table 6.2 [423].

Especially when a classifier is trained and tested on the same dataset, every well-adaptable machine learning method that can make use of a sufficient

measure	definition	short description
accuracy	$\frac{tp+tn}{tp+fn+fp+tn}$	overall correctness
precision	$\frac{tp}{tp+fp}$	correctness of all cases labeled positive by the classifier
recall / sensitivity	$\frac{tp}{tp+fn}$	effectiveness to identify positive labels
specificity	$\frac{tn}{fp+tn}$	effectiveness to identify negative labels

Table 6.2: *Common performance measures in binary classification. For simplicity, Boolean class descriptions are assumed.*

number of input features will show an overly optimistic prediction performance. This effect is called overtraining. The only metric of practical importance, however, is the generalization performance which quantifies the ability of a model to classify independent data that it has not seen yet [235]. A simple and widely used approach to assess the generalization performance of a classifier is cross-validation (CV) . In CV the overall dataset is somehow partitioned into two subsets of samples whereby the classifier is then trained on one subset, the training set, and tested on the yet unseen subset, the test test. Often several iterations of this general procedure are performed to randomize the initial partitioning and reduce the variability of the results. Then the averaged performance metrics are reported as more realistic estimates of the classifier's performance on unseen data [235, 320].

One implementation of this principle is k-fold CV. Here, the n samples of data are partitioned into k subsets of roughly equal size. The classifier is then trained and tested k times whereby each time another subset is held out in the training phase and only taken for assessment. The mean performance of the classifier over all k runs is then reported as the final result. The special case when each sample is in its own subset is called leave-one-out CV [235, 320]. In stratified k-fold CV, the method that we used in the following project, the partitioning of the data tries to maintain the distribution of the labels in all subsets.

On classification with random forests

Random forests are a class of machine learning methods that are based on the integration of an ensemble of decision trees. The core ideas and theoretical foundations arose during the 90s and early 2000s [424, 425]. The principle and its application in classification will be introduced briefly.

Decision trees are data structures that aid in guiding decision processes. Starting from the root node of the decision tree, in each node an attribute of the sample is queried that defines to which next node the walk is proceeded. This strategy is repeated until a leaf of the tree is hit which then returns a classification label [235]. Figure 6.2 shows an example of the principle for the decision if taking an umbrella would be advised today given knowledge on current weather conditions and a forecast.

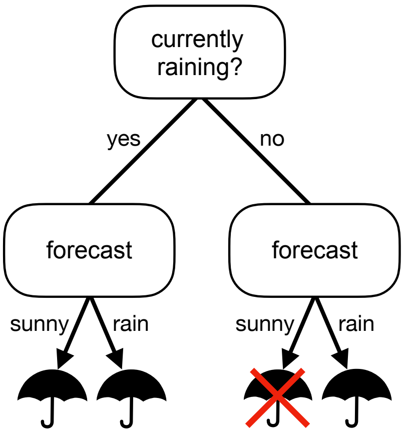


Figure 6.2: *Example of a decision tree. Only if it is neither raining at the moment nor there is a forecast of rain it is recommended to leave the umbrella at home.*

Since especially decision trees that are grown very deep tend to overfit their training data and aggregation of such "weak learners" is an efficient approach to reclaim generalization performance [235], a random forest classifier uses a collection of B such trees, the random forest. When learning the model, in each iteration b one tree T_b is constructed and added to the forest. The basic procedure for this is straightforward. Starting from the root node, a tree is grown by recursively calling the following steps on each expanded node until a termination criterion is achieved: randomly select $k << m$ of all m features, pick the best splitting feature among the selected ones according to some metric, then split the node into two daughter nodes and recurse. Classification can then be conducted by either having a majority vote of the trees in the forest or by averaging their contributions [235]. Details are depending on the exact implementation of the method and the parameter set that was used. The documentation of the scikit-learn library [426], which was used in the following project, provides a good overview on the plethora of tunable parameters found in a common random forest classifier implementation[1]. Among other possibilities, the termination of the tree-building process can be made dependent on the maximum depth allowed for the decision trees or the minimal number of samples required to allow a split of the node, for example. The feature that is suited best as a splitting criterion can be decided by various metrics like the entropy, the gain of information, or the Gini impurity, a special measurement for decision trees introduced in the CART method [427]. Furthermore, by default each tree is built from a bootstrapped dataset that is generated by drawing samples from the original training set with replacement. By "spicing up" the training sets of the individual trees in this fashion a potential correlation of the trees in the forest is minimized. Other important parameters are the number of trees in the forest B and the number of features compared in each split k. Best practices on ranges and heuristics suggesting good starting points for those parameters are well established [428, 429].

[1] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

6.2 INTRODUCTION

Cellular function is a team effort because proteins rarely perform their biochemical tasks all alone. Instead, proteins frequently collide with other gene products in the crowded environment of the cell, they may selectively bind to other proteins driven by physical interactions, they may dynamically assemble into complexes in a well-coordinated manner and accomplish their tasks cooperatively [12, 430]. Such multiprotein complexes may be either clearly defined modules of interaction partners that represent permanently assembled molecular machines or combinatorial formations of transient interaction partners in a dynamic interplay [144, 145, 201].

Whereas the experimental detection of protein complexes is generally speaking a mature field, it is still time-consuming and subject to high false-discovery rates. Quantitative profiling of the complete complexome in a condition-specific way is currently not feasible in a high-throughput fashion [147, 155, 431, 432]. More so, direct quantitative measures are limited to a definite protein space and only cover pairwise complexation [433–436].

Nowadays a plethora of data on gene expression and an increasing amount of data on proteome abundances enable to also approach the dynamics of the condition-specific complexome by computational methods. Guided by static compilations of protein interactions, the correlation of gene expression or protein abundance between putative interaction partners was used as a proxy to study their collective behavior [124, 144, 437]. Besides, the topic was examined by integrating expression data with known protein complexes [201] and annotated pathways [438]. However, such simplified models lack a ruleset addressing how proteins that are expressed in low amounts and that are shared between different binding partners may limit complex formation. Approaches dealing with such interdependencies and the limitedness of gene products have been attempted by stochastic simulations with according computational effort [206] and by linear optimization on fixed sets of reference complexes [15, 439]. The latter studies only considered a very limited complexome and took a simplistic view at differential abundances across cellular states.

Whereas databases of experimentally detected protein complexes continue to serve the community well, they are inherently incomplete - especially when it comes to dynamic combinatorial complexes - and thus can only partially explain all the relevant interplay [188, 440]. Proteins concerned with the regulation of transcription and the chromatin state, for example, are highly interwoven subsets of physically interacting proteins and form complexes in a time-, context- and condition-specific manner. In particular transcription factor complexes are master regulators of all levels of eukaryotic life ranging from the yeast cell cycle [209] to key determinants of cellular fate in mammals [210–212]. We showed with our combinatorial complex prediction algorithm DACO [17] that by integrating connectivity constraints inferred from interactions between protein domains, one is able to unravel the ensemble of biologically feasible protein complexes even for challenging modules of the interactome. With our more recent development PPIXpress [90] (see also Chapter 3) and transcript expression data, the input data for DACO can be contextualized to a level of detail that even takes into account potential effects of alternative splicing when inferring sample-specific interactomes.

Here, we present the differential analysis software CompleXChange as a terminal step of a pipeline consisting of PPIXpress-contextualized and DACO-derived protein complexes, or arbitrary alternative input protein complexomes. The tool quantifies protein complexes, includes several statistical testing procedures, is open-source and can easily scale up to 10,000s of interdependent complexes on a standard computer.

6.3 MATERIALS AND METHODS

CompleXChange facilitates differential analyses of the protein complexome. It is intended to be used with input data on two groups of samples for which protein complexes and protein abundances are predicted by the tools JDACO [17] (version 1.0+) and PPIXpress [90] (version 1.15+). The software can also be applied to suitable input data from alternative origin, of course. An alternative workflow is provided below on the example of reference complexes taken either from CORUM [361] or from hu.MAP [188]. A ready to use platform-independent Java 8 binary, a user guide with example data and the source code of the program are freely available for download at `https://sourceforge.net/projects/complexchange/`https://sourceforge.net/projects/complexchange/. The general workflow is outlined in Figure 6.3.

6.3.1 *Approximating protein complex abundances*

In the first computational step, CompleXChange infers complex abundances from the input data, namely total protein abundances and protein complexes, for each individual sample. To speed up the calculations, CompleXChange automatically utilizes multicore systems in this step by exploiting the independence of the samples.

Binding affinities between proteins are neglected as suitable data is lacking currently and in the foreseeable future [174]. Instead, we assume that the formation of complexes in a cellular sample is governed by two basic rules: the total amount $p_{i,tot}$ of each protein $i \in P$ in the sample is fixed, see Equation 6.2, and the abundance c_m of a complex $m \in C$ is limited by its least abundant member protein, see Equation 6.3. Thus

$$\forall i \in P : p_{i,tot} = \sum_{m \in C} p_{i,m} + p_{i,res}, \tag{6.2}$$

$$\forall m \in C : c_m = \min_{i \in C_m} p_{i,m} \tag{6.3}$$

where C_m denotes the set of proteins that make up complex m, $p_{i,m}$ is the amount of protein i that is assigned to complex m, and $p_{i,res}$ is the residual quantity of protein i that is unbound in the sense of the input proteome P and complexome C. We do not consider the case that single proteins may occur as multiple copies in a protein complex because there are few data available and the information is absent in the notion of complexomes derived from interaction networks. The methodology can in principle be extended to cover stoichiometries of the important class of homo-oligomeric protein complexes if that information should become widely available at genomic scale in the future. At the moment, our concept of neglecting such complexes leads to an over-representation of the other complexes that these proteins are involved in. Figure 6.3B visualizes an application of the algorithm to an artificial example where "Iter: $x.y$" means iteration x and step y.

Step 0: Initial distribution of proteins

The algorithm starts by distributing equal portions of the total abundance of each protein $p_{i,tot}$ to the complexes it is participating in. Thus $\forall i \in P : p_{i,res} = 0$ and

$$\forall i \in P, m \in P_i : p_{i,m} = \frac{p_{i,tot}}{|P_i|} \tag{6.4}$$

where P_i is the set of all complexes that include protein i. This step is only executed once.

Step 1: Tracking surplus capacities

After the initial fill-up in Step 0 and subsequent redistribution steps in later iterations, the limiting proteins in each complex are determined and all c_m are set according to Equation 6.3. Thereby, all complexes limited by a protein i in this iteration are kept track of in L_i. Residual capacities are subsequently updated by the surplus protein amount, thus

$$\forall i \in P : p_{i,res} = \sum_{m \in P_i} (p_{i,m} - c_m). \tag{6.5}$$

The respective quantities per complex are then adjusted accordingly, $\forall i \in P, m \in P_i : p_{i,m} = c_m$.

When its limiting proteins have zero residual capacity at this point, their share in the complex will remain fixed in further iterations and thus the complex is saturated. Saturated complexes and proteins solely found in saturated complexes are therefore set aside and not considered in future iterations (see change to red text color for complex annotations in Figure 6.3B).

Step 2: Detecting convergence

The sum of residual capacities $\sum_{\forall i \in P} p_{i,res}$ after Step 1 is monotonically decreasing and the optimal state is found when no further meaningful decrease is possible. The iterative optimization stops and the complex abundances c_m are returned when either $\Delta \sum_{\forall i \in P} p_{i,res} < \epsilon$, the preset maximum number of iterations is reached or all complexes are saturated. Details regarding default termination parameters are given in Supplementary Section S1.1.

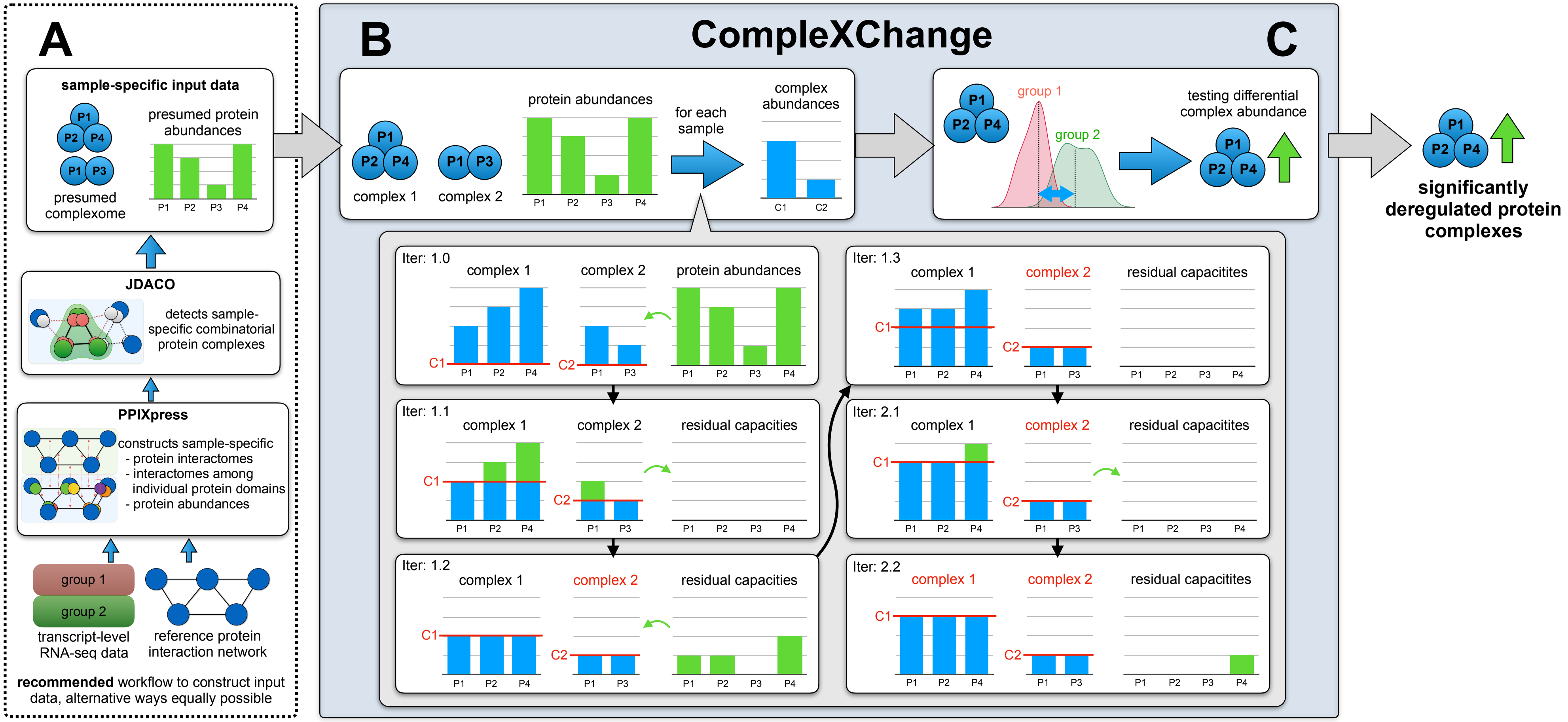

A
sample-specific input data
P1 P2 P4
P1 P3
presumed complexome
presumed protein abundances
P1 P2 P3 P4
JDACO
detects sample-specific combinatorial protein complexes
PPIXpress
constructs sample-specific
- protein interactomes
- interactomes among individual protein domains
- protein abundances
group 1
group 2
transcript-level RNA-seq data
reference protein interaction network
recommended workflow to construct input data, alternative ways equally possible
B
CompleXChange
C
complex 1
complex 2
protein abundances
for each sample
complex abundances
C1 C2
Iter: 1.0
Iter: 1.1
Iter: 1.2
Iter: 1.3
Iter: 2.1
Iter: 2.2
residual capacitites
residual capacities
group 1
group 2
testing differential complex abundance
significantly deregulated protein complexes

Figure 6.3: *Workflow example for CompleXChange. A) Suitable input data can be constructed easily with either PPIXpress and JDACO or in suitable alternative ways. CompleXChange then performs B) the approximation of complex abundances, and C) the detection of differential complexes. Details are described in the main text.*

Step 3: Redistributing residual capacities

To counteract optimization confinement (pathological examples can be artificially constructed where protein amount is swapped back and forth without any meaningful improvement) and accelerate convergence, a logistic saturation function that is decreasing rapidly with each iteration sets a distribution prefactor $\lambda \in [0.99, \ldots, 0.09)$ (see Supplementary Section S1.1 for details) by which the residual amounts of limiting proteins ($\{i \in P \mid |L_i| > 0\}$) are preferentially distributed to complexes they limit:

$$\forall \{i \in P \mid |L_i| > 0\}, m \in L_i : p_{i,m} = p_{i,m} + \frac{\lambda p_{i,res}}{|L_i|} \tag{6.6}$$

with thus remaining capacitites $\forall \{i \in P \mid |L_i| > 0\} : p_{i,res} = (1 - \lambda) p_{i,res}$. Finally, the complete residual capacities of all proteins are distributed equally as $\forall i \in P, m \in P_i : p_{i,m} = p_{i,m} + \frac{p_{i,res}}{|P_i|}$ and therefore $\forall i \in P : p_{i,res} = 0$. From here, the algorithm proceeds with Step 1 in a new iteration.

6.3.2 *Detection of differential complexes*

After annotating each complex detected by JDACO with an abundance value per sample, we statistically evaluate the numerical difference of the abundance of individual complexes between groups (see Figure 6.3C). To limit unnecessary testing, complexes that should undergo testing have to be detected in at least a sizeable fraction of samples of either group (default: 0.75). The group-specific distributions of each complex that passed this filtering step are then subjected to two-sided statistical tests. Implemented statistical tests are the Wilcoxon rank-sum test (default test; unpaired, non-parametric), Welch's unequal variances t-test (unpaired, parametric), Wilcoxon signed-rank test (paired, non-parametric), and the paired t-test (paired, parametric). Multiple testing adjustment is subsequently performed using the Benjamini-Hochberg procedure [236] and significantly deregulated complexes are reported. Additional options that are implemented in the code but not discussed here are (a) to base the differential analysis on subsets of complexes detected to help the detection of alterations in robust core complexes, or (b) to solely use combinations of user-specified seed proteins as the reference of interest. Please refer to the user guide for details.

Furthermore, CompleXChange includes an optional analysis that determines seed proteins that occur more often than expected by chance in up- or down-regulated complexes. If this option is selected, a ranked list of protein complexes is constructed by assigning a score to each evaluated complex. This score is set as the negative logarithm of their raw p-value and the sign of their direction of deregulation. In doing so, the task resembles the established approach Gene Set Enrichment Analysis (GSEA) but is applied to proteins in scored protein complexes in an analogous way. The implementation is done according to the original GSEA paper [233] and the same FDR as in the differential analysis is applied. By default $10,000$ iterations are made in the randomization step and only seed proteins are considered that belong to at least 10 complexes.

6.4 results and discussion

The results will be divided into three major parts. First, we introduce the datasets that were used in our evaluation, then we assess the performance of CompleXChange. Finally, we analyze the results of an application of CompleXChange to derive the differential transcription factor complexome of classical and non-classical monocytes.

6.4.1 *Datasets and processing of data*

Preparing sample-specific transcript expression data

Raw RNA-seq data for 17 samples of classical monocytes (CMs) and for 17 samples of non-classical human monocytes (NCMs) [405] (16 sample pairs matched by donor among them) were retrieved from the SRA (accession SRP082682) [85]. A subset of 58 RNA-seq samples of finnish women among the human lymphoblastoid cell line samples (LCLs) of the GEUVADIS data [411] was downloaded from EBI ArrayExpress (accession E-GEUV-1) [86] analogously to [83]. The raw sequencing data was quantified using kallisto 0.43.1 [74] and the annotation data on human protein-coding transcripts of GENCODE release 27 (GRCh38.p10, Ensembl release 90). Kallisto was applied with bias-correction enabled and default options otherwise. Fragment length estimates for the single-end sequenced monocytes data were set according to the original publication [405]. One hundred iterations of bootstrapping were carried out to account for technical variation in a subsequent differential analysis using sleuth [83].

From interaction networks to transcription factor complexomes

From the weighted human protein-protein interaction network PrePPI [163, 177] we downloaded its most recent high-confidence release (defined by a probability of interaction above 0.5) on 17. Jan. 2017. On the basis of this reference interactome we constructed sample-specific protein-protein interaction networks as well as corresponding domain-domain interaction networks for all quantified transcript expression samples with PPIXpress 1.18 [90]. For this, the most recent updates were automatically retrieved from Ensembl (release 90) [360], UniProt (release 2017_09) [441] and 3did (release Sept. 2017) [107]. The reference network contained information on 18,451 proteins and 1,527,335 interactions. 70% of the proteins and 37% of the protein interactions were mapped to domain interactions and thus can benefit from the transcript granularity of the data and the methodology of PPIXpress that adapts the interactome in an isoform-specific manner. The usefulness of this model based on conserved domains was recently confirmed experimentally [7, 207]. All transcripts with a non-zero TPM value were deemed expressed. Approximate protein abundances were taken as the sum of TPM values for all expressed transcripts coding for the protein. Notably, when assigning abundance values, PPIXpress (since version 1.12) by default excludes transcripts with Ensembl biotype annotations 'nonsense-mediated decay' or 'non-stop decay'. Although protein abundances are still approximated by mRNA expression, the pipeline already accounts for well-understood post-translational surveillance mechanisms and

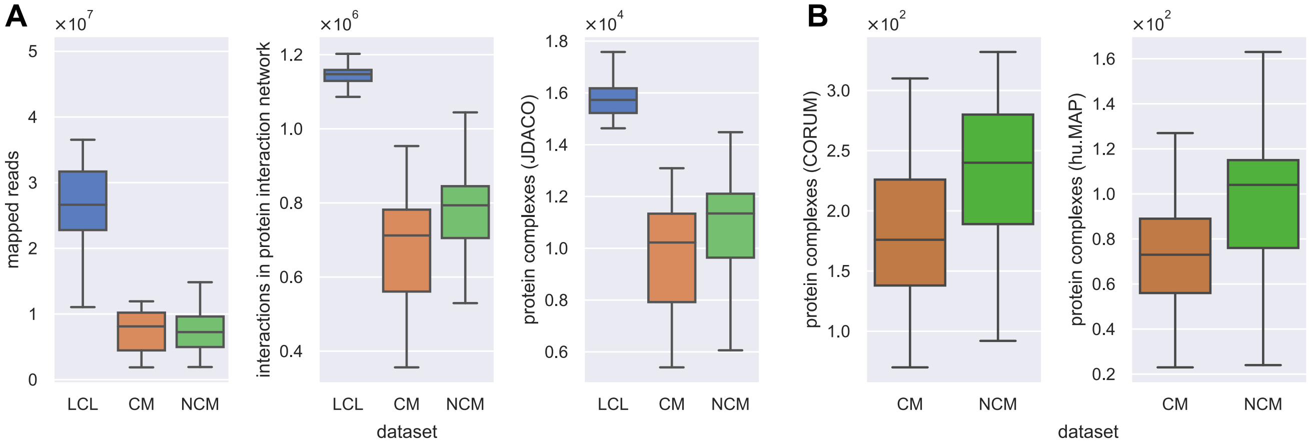


Figure 6.4: *Size distributions of input data. A) Number of mapped reads (left), size distributions of sample-specific protein interaction networks (middle) and JDACO predicted complexomes (right) for individual samples of the GEUVADIS lymphoblast cell line (LCL) data and classical (CM), as well as non-classical monocytes (NCMs). B) Sizes of sample-specific monocyte complexomes derived from the data of CORUM (left) and hu.MAP (right).*

should thus provide more reasonable estimates than mere gene expression data until equally rich genome-wide proteome abundance data are available in appropriate sample sizes.

Finally, transcription factor (TF) complexes were predicted for each sample with JDACO 1.0 [17] by employing the 601 TFs annotated in HOCOMOCO v10 [442] as seed proteins in the respective protein and domain interactomes. The seed pair threshold was set to 0.95 (PrePPI weights are probabilities), the maximal complex size to 5 proteins (optimized tradeoff between allowed complex size and runtime) and default parameters were used otherwise. The thus derived complexome will serve as the default input for our analyses. Figure 6.4A visualizes the distributions of mapped reads (left) as well as interactome (middle) and predicted complexome sizes (right) for the three groups of samples used in the study.

To illustrate how CompleXChange can also be used in alternative workflows, we downloaded the manually curated human protein complexome of 2916 complexes in CORUM (3.0) [361] and the precompiled dataset of 4526 hu.MAP complexes [188] which was derived by data integration efforts. After filtering for complexes with at least one TF, the 454 remaining CORUM transcription factor complexes (TFCs) comprised complexes involving 159 TFs. The 277 remaining hu.MAP TFCs covered 183 TFs. The thus derived TFCs of each data source where then used as reference complexomes to construct sample-specific subsets for which all member proteins have a non-zero protein abundance (as given by PPIXpress, see above) in the particular monocyte samples considered here. Figure 6.4B shows the respective complexome sizes for all monocytes samples. When assessing the sizes of TFCs in the CORUM and hu.MAP data, the vast majority of complexes was within the threshold of 5 proteins per TFC that we used in our predictions (see Figure S1).

6.4.2 *Assessing the methodology*

We first evaluated the performance of the algorithm that approximates protein complex abundances implemented in CompleXChange. For this, we compared the ComplexChange results to an approach where the problem was formulated as a linear program [15, 439]. Simulated data with known ground truth was used to benchmark the two methods. Furthermore, we checked if CompleXChange was susceptible to reporting deregulated complexes erroneously and how it behaved using limited data. To emulate rather complete complexomes, all method evaluation was conducted using the extensive predicted complexomes of each sample.

Comparing abundances computed by CompleXChange and linear programming

Using both the CompleXChange algorithm and an existing approach based on linear programming (LP) we computed abundance values of predicted protein complexes for all 92 samples on monocytes and lymphoblastoids. The LP approach was implemented according to the equations in [439] using the established open-source solver lpsolve (v5.5) [420]. Figure 6.5 visualizes the correlation of complex abundance estimation results between both methods (left), runtimes for each method (middle) and the fraction of complexes per sample that were assigned with an abundance of zero by the LP-based approach (right).

The predicted protein complex abundances were overall very similar to each other with an average correlation of 0.90 ± 0.06 (Figure 6.5, left). Computing the LP results took on average 2.8 ± 0.9 times longer (Figure 6.5, middle) than using CompleXChange on identical input data ($p < 10^{-16}$, two-sided Wilcoxon signed-rank test paired by sample). Notably, the LP formulation resulted in many zero solutions. On average, $85\% \pm 1\%$ of all complexes in a sample were assigned an abundance of zero (Figure 6.5, right) although all member proteins in input complexes have non-zero abundance by definition. Whereas the LP result is numerically optimal given its formulation and constraints, non-sparse abundance results as returned by CompleXChange - even if they are very small - appear biologically more reasonable solutions. Furthermore, zero-inflated complex abundance distributions would require an adjusted statistical treatment [443, 444].

Benchmarking complex abundance estimation on simulated data

As pointed out before, there exists so far no adequate experimental reference data to test the complex abundance estimation against. In lieu of this, we generated input data of known ground truth by randomized construction on the basis of realistic complex compositions and expression values from our prepared samples. The construction reverses the simple idea that a protein which is exhaustively incorporated into complexes and has no unbound portion ($p_{i,res} = 0$) consequently sets the maximum abundance of all complexes it is part of. For the construction of this synthetic dataset, the total abundance $p_{i,tot}$ of each limiting protein is randomly drawn from sample data. To ensure that such a sampling does not suffer from biological bias, we assessed if protein

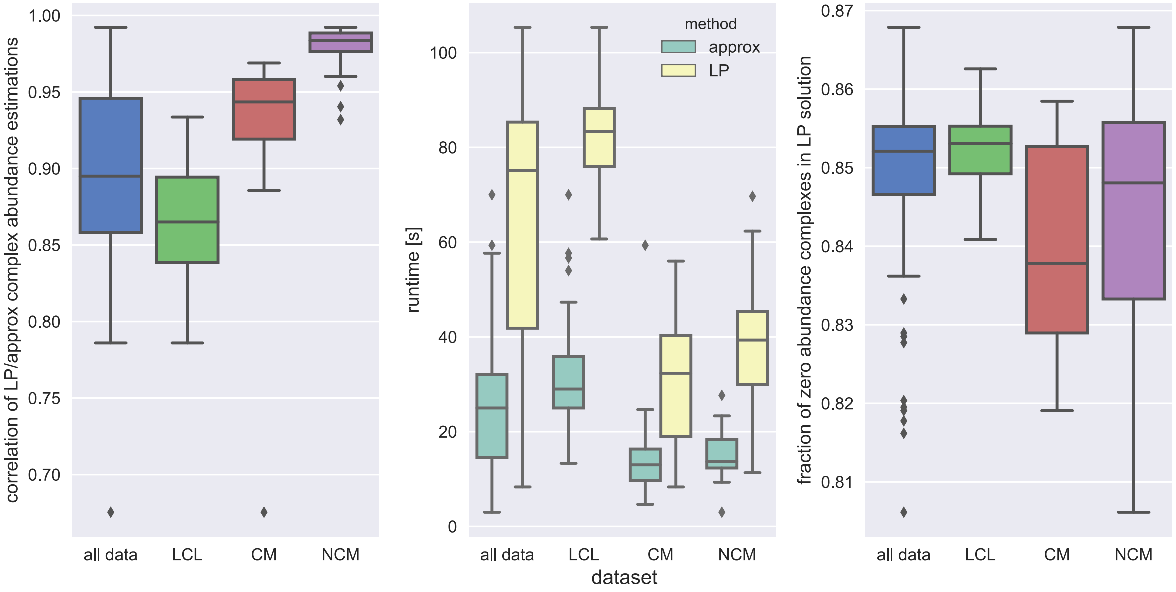


Figure 6.5: *Comparison of complex abundance estimations by the iterative approximation in CompleXChange (approx) and by linear programming (LP). Shown are the correlation of their results (left), the necessary runtimes for each method (middle) and the fraction of zero abundance complexes reported by the LP approach (right) per dataset as well as accumulated for all data. Runtimes were calculated as the average of 3 repetitions for each method and sample.*

abundances correlate with the number of complexes a protein participates in. In the data on (N)CMs and LCLs this was clearly not the case (average correlation -0.005 ± 0.003). The arbitrary association of proteins with abundance values should therefore be unproblematic.

Distributing a respective share of all limiting proteins to the complexes in which they take part can then be modeled in various ways (model parameter I). All $p_{i,m}$ are determined by definition (see Equations 6.2-6.3) and only residual capacities $p_{i,res}$ of non-limiting proteins remain to be set (model parameter II). These in turn specify the $p_{i,tot}$ of the artificial input data. Model parameter I, the distribution of limiting protein abundances among their associated complexes, was realized using three independent modules: equal distribution with modeled noise (abbreviated as eqd-[noise parameter]), sampled from an empirical distribution (ed) from ComplexChange approximation results and an assumption-free random distribution (rndd). Model parameter II is the unbound ratio parameter that models the extent of residual capacities of non-limiting proteins. The detailed construction schemes as well as our estimates on reasonable noise parameter ranges are documented in Supplementary Section S1.2.

To judge the relative performance of the CompleXChange approximation algorithm we also applied the LP approach and two randomized modifications of the CompleXChange algorithm to the artificial reference data. In the first randomized variant of the algorithm, input abundance values of proteins were shuffled before applying the abundance estimation method (abbreviated rnd (in)), i.e. input protein abundances did not match the abundance of the proteins associated in the ground truth. In the second variant, the complex abundance results derived from the correct input data were shuffled (abbreviated rnd (out)). We tested 12 combinations of parameters over all 92 samples for 20 iterations each for all methods (see Supplementary Section S1.2 for details on parameter sets). To assess the smoothness of the CompleXChange approximation performance, a broader set of 42 combinations including some intermediate values was used for benchmarking. We compared the artificial data for which we knew the ground truth with the results of the individual methods in terms of the correlation of known/predicted complex abundances. The results are shown in Figure 6.6 in dependency of the distribution parameter (left) and the unbound ratio parameter (right). More detailed results for all individual parameter sets are shown in Figure S5.

Both CompleXChange approximation and the LP approach performed far better than the randomized methods whose results were generally not correlated with the reference complex abundances (see Figure 6.6 and Figure S5 for details). The correlation of the CompleXChange results with the reference was significantly higher than those from LP across all modeling parameter sets ($p < 10^{-14}$ for all parameter sets, see Table S1 and Figure S5 for details). Interestingly, the performance of both methods was more strongly affected by the unbound ratio (model parameter II) than by the modeling of the distribution of protein product (model parameter I). This is even more apparent when a broader choice of modeling parameters is applied, as was done for the CompleXChange abundance estimation (see Figure S6). Consequently, a good coverage of the complexome sets the ruleset of interdependency and also limits excess protein product. The typical complexome size in our study (see Figure 6.4, rightmost) was about a magnitude larger than, for example, that used in the study describing the application of the LP-based approach to human [15] (1,338 human protein complexes taken into account).

Detection of false positives in negative control data

The subset of Finnish women in the GEUVADIS data that we prepared is assumed to be rather homogeneous. Hence, random sampling of groups therein was used before as a negative control in the assessment of differential expression methods [83]. When we analogously applied the same testing approach to find deregulated complexes in the GEUVADIS data, CompleXChange showed a high robustness against false positive reports when group sizes were reasonably balanced. For details, we refer to Supplementary Section S2.1.

Sample size dependency of results

We also checked by subsampling on a reference dataset (see [80, 83, 445]) how CompleXChange behaved when only a small number of samples is available for differential analysis. The results indicated that at least 10 samples per group should be used, if possible. For details, see Supplementary Section S2.2.

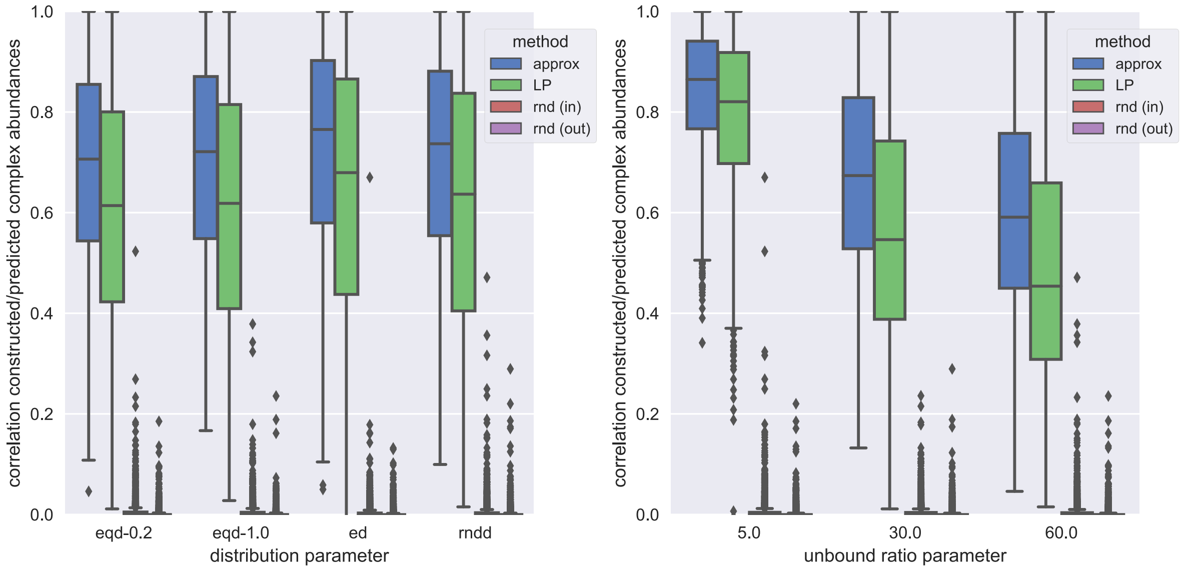


Figure 6.6: *Correlation of constructed complex abundances and predictions by different estimation methods depending on different modeling parameters. Results are shown in dependency of the distribution parameter (left) and the unbound ratio parameter (right).*

Differential transcription factor complexome of classical and non-classical monocytes

Finally, we applied CompleXChange to detect deregulated TF complexes (TFCs) between classical and non-classical monocytes whereby complexomes were predicted for each sample with PPIXpress and JDACO. This cellular transition was chosen because the expected differences should be comparably small and the number of samples in the dataset appeared sufficient. We used non-parametric testing, FDR 0.05 and default settings otherwise and enabled the option to assess if seed proteins (here: transcription factors) are enriched in up- or down-regulated complexes. CompleXChange reported 978 deregulated TFCs and 35 enriched TFs therein. Figure 6.7 shows a volcano plot of the complexes evaluated and the distributions of complexes involving the three most enriched TFs.

Comparison to differential expression results

Differentially expressed genes were determined using the quantified RNA-seq data (see Materials and Methods) and sleuth (v0.29.0) [83]. For this, transcript expression was summarized to the gene-level using matching Ensembl 90 data retrieved by biomaRt (v2.34.0) [446] and statistical significance was determined based on q-values below 0.05 in likelihood ratio- and Wald-tests. As result, 316 genes were found to be differentially expressed, 77 of those were TFs (47 upregulated, 30 downregulated). In the following, these genes are termed DE genes. We assume that the proteins encoded by them are deregulated as well.

We first studied to what extent DE genes overlapped with the 978 deregulated complexes. On average, about a third (37% ± 24%) of each reported deregulated

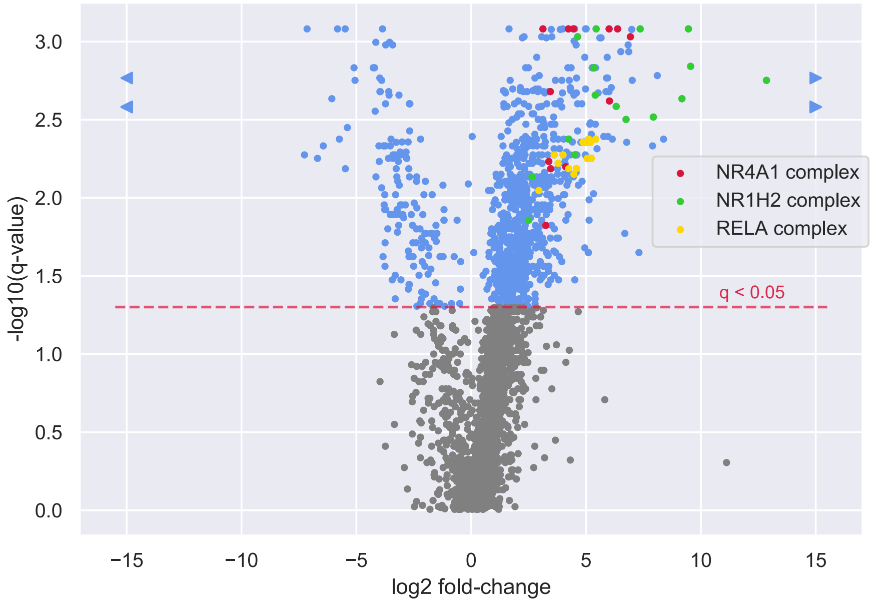


Figure 6.7: *Volcano plot of fold-changes in protein complex abundances. Significantly deregulated complexes between classical and non-classical monocytes are shown as blue points. Complexes below the significance threshold are colored grey. Additionally, complexes that contain one of the three most enriched TFs are shown in red (NR4A1), green (NR1H2) and yellow color (RELA), respectively. Fold-changes were computed as the ratios of mean abundances of respective complexes in the two groups. Complexes that exhibited border case fold-changes (zero mean abundance in one of the groups) were set to* ±15 *and the respective datapoints marked as triangles.*

complex consisted of proteins whose genes were deregulated between the two cell types. In 823 complexes (84.2% of all results) at least one protein-coding gene was deregulated and in 32 cases (3.3%) all were differentially expressed. These modes of action were also relevant in the 10 most deregulated complexes as can be seen in Figure 6.8. The significantly altered abundance of 155 complexes (15.8%) could not be inferred by differential expression analysis of protein-coding genes in isolation. Such events can be explained, on the one hand, by the effects of mutual dependence among the complexes since they share and compete for each protein product, and, on the other hand, by the dynamics of the neighborhood in the protein interactome which affect the cohesiveness measure in JDACO predictions (see [17]).

Next, we investigated the relationship between deregulated complexes and deregulated protein-coding genes in reverse direction. Of all 87, 945 complexes seen in any sample 54, 645 had at least one deregulated gene (62.1%). Among those complexes only 1.5% were detected as being deregulated. When complexes were filtered with CompleXChange to be present in at least 75% of either group, of the 2, 522 complexes, 1, 841 had a deregulated member (73.0%). Only 44.7% of those complexes were found to be deregulated by CompleXChange. We stored the 1, 841 complexes selected by DE for the analysis of information content in the next subsection.

At last, we compared the results in terms of TFs that were reported as deregulated. Whereas most of the TFs that were found to be enriched in deregulated complexes were also differentially expressed, 14 of the 35 (40%) enriched TFs were not significantly deregulated on the gene-level. Significance rankings between the two approaches cannot be compared, as can be seen in Table S2. Most noteworthy, NR4A1 is the TF with highest enrichment in CompleXChange, whereas in DE it is only the eighth TF when sorting by q-value and the 22nd TF when sorting by fold-change. Nr4a1 is the master regulator of non-classical monocytes in mice [447, 448]. In human, its ortholog NR4A1 is assumed to have the same regulatory function [449, 450].

Furthermore, DE TFs and TFs reported to be enriched in deregulated complexes were subjected to overrepresentation analyses using the web services GeneTrail2 [262] and PANTHER [261] against the background of all 601 TFs regarding five pathway annotation databases. Details concerning the analyses and the complete results are provided in Supplementary Section S2.3. Whereas the DE TFs showed only one rather unspecific enriched term in one database (PANTHER pathway "CCKR signaling map" 3.54-fold enriched, see Tab S3), the CompleXChange enriched TFs showed enrichment for all pathway annotation databases. The enriched annotations contained, for example, Toll(-like) receptor signaling (several terms across databases, see Tables S5, S7 and S8), TNF(-α) signaling (several terms across databases, see Tables S5 and S7), various specific interleukin signaling pathways (e.g. WikiPathways "IL-1 signaling pathway", 13.71-fold enriched, see Table S7) as well as more general terms such as "Inflammation mediated by chemokine and cytokine signaling pathway" (PANTHER pathway, 9.95-fold enriched, see Table S8). This matches the specialization that has been reported for these cell types [451–453].

Comparison to an alternative pipeline using CORUM and hu.MAP complexomes

For this comparison, we selected those protein complexes of CORUM and hu.MAP containing at least one TF instead of using predicted complexomes as the input. When we applied CompleXChange to the sample-specific subsets of the CORUM and hu.MAP complexomes using the same parameters as with the JDACO predictions, 77 CORUM complexes and 16 complexes of hu.MAP were identified as deregulated between classical and non-classical monocytes. The distribution of complex abundance changes appeared very one-sided in both datasets, see Figure 6.9. Due to the very small number of complexes assessed and due to the skewed distributions, the calculation of TFs enriched on the upper/lower end of the deregulation range is not really meaningful. This can be seen on the example of RREB1 that was found to be the only enriched TF for the hu.MAP data.

The overlap between results of CompleXChange analyses using the predicted JDACO complexes and complexomes of CORUM and hu.MAP was overall very small (see Table 6.3). Interestingly, the result derived from the predicted sample-specific complexomes was more similar to either result of the two complex databases than the overlap of the two databases (first three rows, Table 6.3). This also holds true when all TFCs are taken into account rather than only the

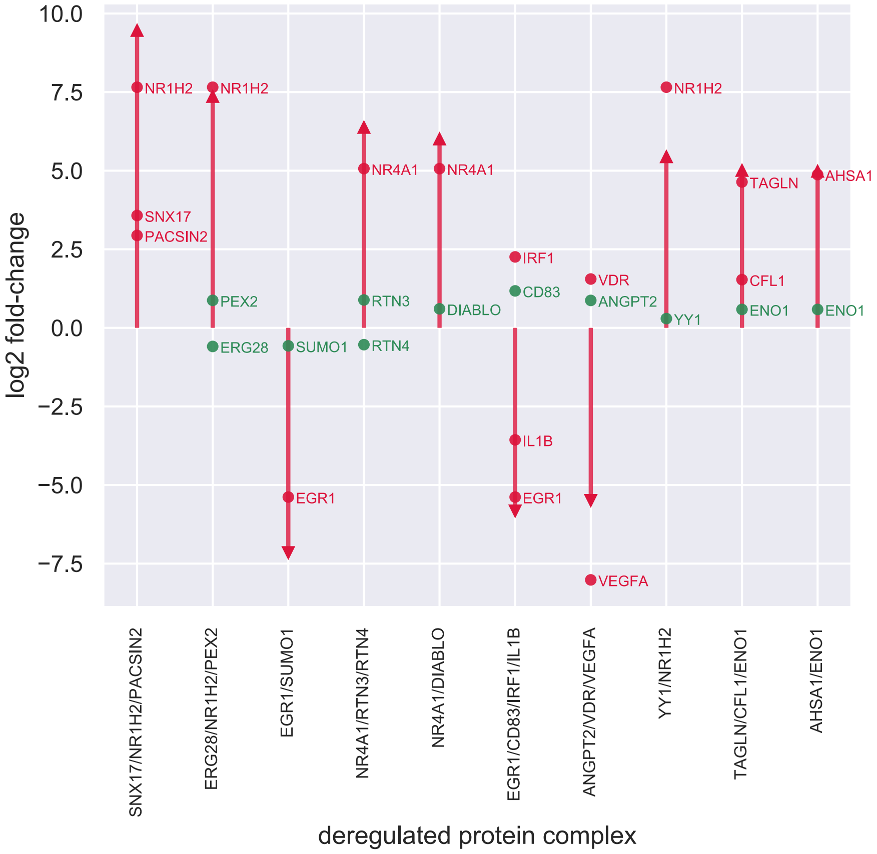


Figure 6.8: *Fold-changes of the top-10 deregulated transcription factor complexes and their members. The 10 most deregulated TFCs are shown in order of significance on the x-axis, the arrowtips on the y-axis depict their logarithmic fold-change. In addition, logarithmic fold-changes of their member proteins are overlayed on the respective columns whereby proteins coded by DE genes are colored red and those with non-DE genes associated are shown in green. Fold-changes were computed as the ratios of mean abundances of respective complexes and proteins in the two groups.*

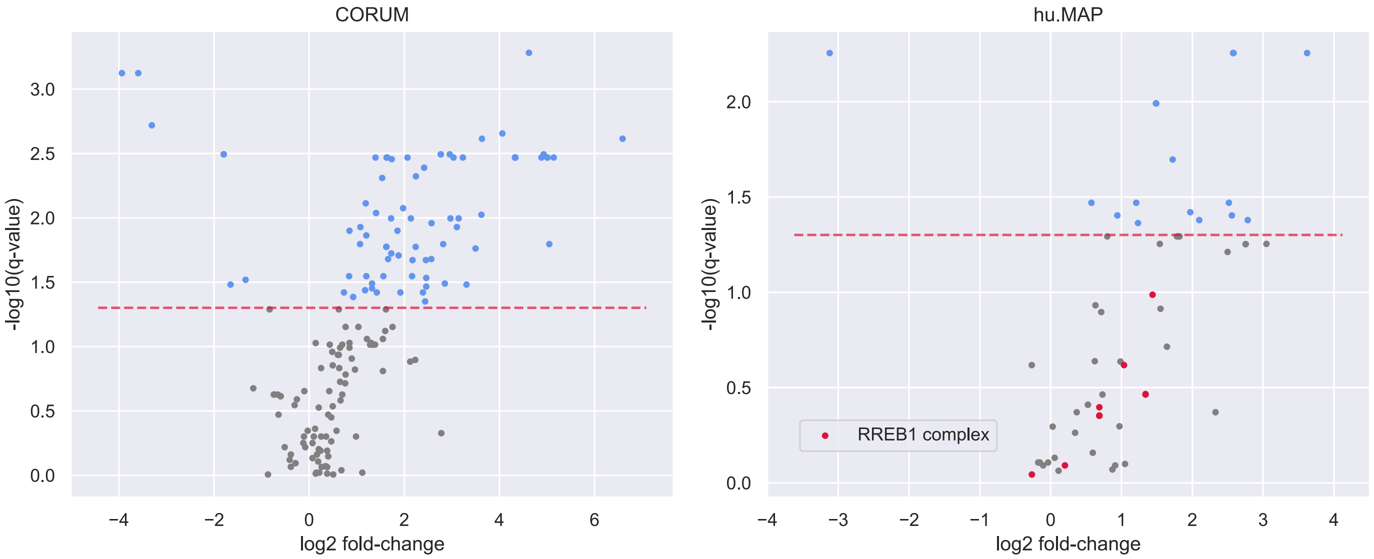


Figure 6.9: *Volcano plots of fold-changes in CORUM and hu.MAP protein complex abundances. The results for CORUM complexes are shown in the left plot and the results on the basis of hu.MAP complexes on the right. Significantly deregulated complexes between classical and non-classical monocytes are depicted as blue points and complexes below the significance threshold are colored grey. For the hu.MAP results, complexes that contain RREB1 are shown in red. Fold-changes were computed as the ratios of mean abundances of respective complexes in the two groups.*

ones deemed significant by CompleXChange (last row, Table 6.3). This is not very surprising since current experimentally-backed complexome libraries are still considered to be quite incomplete [188, 440]. Especially when taking into account the important interplay between complexes sharing proteins, input data of predicted complexomes seem more appropriate in this specific issue.

Abundances of reported deregulated complexes are meaningful descriptors of cell type

To assess the information content of deregulated complexes in an unbiased way, we tested their ability to act as descriptors in simple random forest models [454] that were trained to classify the monocyte data into classical and non-classical samples.

For each set of complexes (or TFs) tested we performed 100 iterations of stratified 10-fold cross-validation (CV) to account for randomness in dataset partitioning and tree building [454, 455]. The corresponding abundance values of complexes or TFs were used as the features in a random forest classifier with 32 trees (sufficient according to [429]). The number of features considered in each tree split was automatically set to the square root of the number of total input features according to the heuristic established by [428]. Other parameters were kept at the default setting as implemented in scikit-learn (v0.16) [426]. The performance for a set of complexes (or TFs) was then reported as the mean accuracy over all cross-validation iterations. We considered the following cases: (1) all complexes reported by CompleXChange applied to both predicted and reference complexomes, (2) two stricter sets for which we pruned the result

results sets compared	exact matches	average overlap	reasonable overlap [%]
CORUM / JDACO	0	0.24 ± 0.12	32.5
hu.MAP / JDACO	0	0.17 ± 0.16	18.8
hu.MAP / CORUM	1	0.08 ± 0.24	6.3
hu.MAP (all) / CORUM (all)	5	0.12 ± 0.19	11.9

Table 6.3: *Comparing deregulated complexes of JDACO, CORUM and hu.MAP. "hu.MAP (all)" and "CORUM (all)" depict the sets of all TFCs in the respective datasets whereas all other sets cover the reported deregulated complexes. Overlap between two protein complexes was quantified using the overlap score ω [194], "average overlap" between two result sets means the average of all best matches in terms of ω between the first (smaller) and second (larger) set of reported deregulated protein complexes. The percentage of complexes in the first (smaller) set with any reasonable match (ω > 0.25, as in [17, 179, 196]) in the second (larger) set is termed "reasonable overlap".*

for the predicted complexomes by demanding tighter q-values ($q < 0.01$ and $q < 0.001$), (3) permutation tests where we sampled random complex sets as well as for (4) DE complexes (complexes with at least one DE protein-coding gene associated, see previous subsection), and (5) DE/all TFs (using protein abundances). The results are summarized in Table 6.4.

The complexes reported by CompleXChange when applied to the predicted complexomes showed monotonically increasing mean accuracy and decreasing variance with increasing stringency and thus decreasing set size (from 978 to 31, compare "sign. dereg. complexes" entries in Table 6.4). Whereas the significantly deregulated complexes with highest stringency gave the best overall accuracy of all feature sets tested, including DE TFs, most non-random feature sets basically showed similar performance within their standard deviations. The significantly deregulated complexes reported on the basis of the fixed protein complex datasets gave the lowest classification performance of non-randomized descriptors (compare non-randomized entries in Table 6.4). This strengthens the assumption that predicted complexomes are favorable currently.

As a baseline comparison to the CompleXChange results for the predicted complexomes of varying stringency, we evaluated how likely it is to get a similar performance by chance. For this, we drew 10,000 random complex sets of equal size from either all 87,945 predicted complexes seen in any sample or the filtered set of 2,522 complexes. In all tested scenarios, the accuracy of the corresponding CompleXChange result set was very unlikely to be achieved or exceeded by chance (all $p \leqslant 0.0003$, see Table S9 for statistics, Table 6.4 for averages and Figure 6.10 for observed distributions). The hu.MAP-derived deregulated complexes, on the other hand, were often even less predictive than random complexes on average which again encourages to employ complex prediction in this workflow (compare "sign. dereg. hu.MAP complexes" and all "random complexes" entries in Table 6.4).

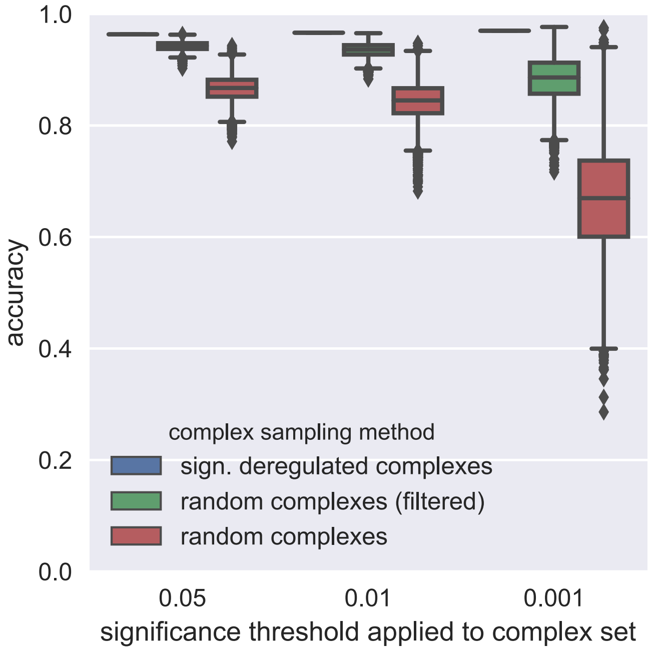


Figure 6.10: *Comparison of CompleXChange results of varying stringency with randomly selected deregulated complexes of equivalent size. Comparison of CompleXChange results of varying stringency with randomly selected deregulated complexes of equivalent size in terms of information content.*

feature set	set size	CV accuracy [%]
sign. dereg. complexes ($q < 0.05$)	978	96.4 ± 1.5
random complexes	978	86.6 ± 2.2
random complexes (filtered)	978	94.2 ± 0.7
sign. dereg. complexes ($q < 0.01$)	429	96.6 ± 1.1
random complexes	429	84.3 ± 3.4
random complexes (filtered)	429	93.5 ± 1.2
sign. dereg. complexes ($q < 0.001$)	31	97.0 ± 0.4
random complexes	31	66.8 ± 9.9
random complexes (filtered)	31	88.4 ± 3.9
sign. dereg. CORUM complexes	77	93.6 ± 2.6
sign. dereg. hu.MAP complexes	16	85.5 ± 3.9
DE complexes	1841	96.3 ± 1.5
DE TFs	77	96.2 ± 1.4
all TFs	601	94.9 ± 2.0

Table 6.4: *Cross-validation (CV) accuracies of feature sets examined. For the randomized complexes CV accuracy is reported as the mean across all permutations and its standard deviation. For all other evaluated sets CV accuracy depicts the mean and variance for the* 100 *iterations of CV.*

6.5 CONCLUSION

The increasing wealth of transcriptomic data and recently introduced computational tools enable to infer protein interactomes and complexomes in specific samples. With CompleXChange this information can be exploited to conduct differential analyses of the dynamic protein complexome in a quantitative manner. We showed for simulated data with known ground truth that its inferred complex abundances were in better agreement with the artificial reference and made more sense biologically than the runtime-intense mathematical optimization with linear programming. When tested in a realistic scenario, CompleXChange featured a performance regarding robustness and limited amounts of samples that is well-suitable for practical applications. Moreover, reported complexes held significant information content on cellular identity and partially orthogonal information to gene- and protein-centric analyses, which are not covering the physical interplay found in a cell. Hence, analysis of differential complexomes should become even more valuable in the future.

6.6 ADDENDUM

6.6.1 *Retrospective*

Most of my method development efforts, and CompleXChange in particular, suffered from one major obstacle in their respective evaluation phases: how to verify something for which, during the time of the project, was no (or even in the foreseeable future will not be) experimental data available for reasonable verification? Then one needs to fall back to what can be concisely termed ″biological sanity checks″[2] and has to provide sufficient evidence that the output of the approach is sound given the established knowledge on the matter at hand. In such a case it helps to work on a well-understood issue and, at best, independent and orthogonal data should be available that can be integrated into an assessment scheme that supports the biological meaningfulness of the results. I will exemplify that matter in the context of CompleXChange.

The GEUVADIS dataset was a very specific and good choice for the assessment of the differential analysis regarding the method's robustness against false-positive hits. Because the results are unlikely to change considerably depending on that, the exact choice of the dataset(s) should not matter for all other tests of the approach as well as the comparisons to the LP-based alternative method. But retrospectively, I often thought about reconsidering my selection of the study data would I redo this project. The dataset on classical and non-classical monocytes in human used here seemed very appealing for several reasons at first glance. First, it had the right size, which means there were enough samples to potentially achieve a reasonable statistical power without making the evaluation computationally tedious. And second, it concerned a very confined cellular transition, namely switching between two subtypes of the same cell type. However, the detailed knowledge on the inner workings of this

[2] I first heard the exact term "biological sanity check" in a talk by Jan Baumbach and acquired it gratefully.

transition was surprisingly sparse and there was no other data available for the subtypes that could have enabled to formulate testable hypotheses to strengthen the biological results derived by CompleXChange. A suitable alternative study dataset would have been the huge collection of ENCODE data that was used in the study that I will briefly address in Section 6.6.3. Using ENCODE's data often has the tremendous advantage of having plenty of additional genomic data readily available. In the context of CompleXChange, one could have used data on histone modifications of the samples to check if specific histone marks in potential binding regions of deregulated TF complexes which recruited corresponding histone modifying proteins to the region changed between the comparison groups, for example.

Ultimately, we decided to prioritize on the method assessment rather than the study. Integrating more data would have bloated the manuscript of the project and would have led to a loss of focus compared to such a clearer "method paper".

6.6.2 *Outlook*

As already stated in Section 6.3.1, CompleXChange does not consider exact stoichiometries when inferring complex abundances. It can clearly be argued that this could be considered a major flaw of the approach. In our area of interest, for example, homo-dimerization is described for many TF families [214, 456]. Although such homo-multimeric protein complexes are quite common, they are currently not marked out as such within the tool and accounting for these non-equal stoichiometries would certainly introduce shifts in the numerical results.

Unfortunately, and similar to the situation for binding affinities between individual proteins, there is only very limited data available on stoichiometries that can be worked with on a system-wide scale. Most established complexome database efforts lack this information completely, only the curated EMBL-EBI Complex Portal [457] has at least some complexes annotated with exact compositions.

Furthermore, while protein complex prediction methods that work on protein interaction networks are very helpful to fill the gaps in the incomplete knowledge that we have on protein complexomes [188, 440], they are by way of construction currently not able to detect any homo-multimerization or even exact stoichiometries. Homo-dimerization would be represented by the self-interaction of a protein in the protein interaction network. The notion of a self-interaction is not usable by the majority of computational approaches that aim at predicting protein complexes, though. The reason for this is that they are based on identification of dense modules in interactome networks and have no usage for this information. Thus, they are not able to uncover complexes in which multiple copies of a protein exist.

Still, assuming the necessary input data should become available in the future, either from curated knowledgebases or sophisticated predictions, the model and algorithms in CompleXChange could be extended to also cover stoichiometries in complexes.

6.6.3 *The thing that should not be: "Finding regulatory protein complexes defining pluripotency"*

An overarching research goal during my dissertation was always the question if one could reveal those TF complexes that are responsible, or at least important, for the transition and/or maintenance of the pluripotent state in cells. Pluripotency is the unique cellular capability to be able to differentiate into all three germ layers. Realizing and mastering how and why reprogramming into such a stem-cell-like state can be done, optimally in a safe and efficient way, is a key for understanding processes governing development and disorders, and even more so, may revolutionize regenerative medicine. At the very least, the research concerning these topics will be simplified when new insights lead to better approaches and improved protocols [211, 458]. While I hope to have worked out a valid methodology to obtain good guesses for candidates in that regard by conducting a differential complexome analysis between groups of samples that are pluripotent and samples that are not, my endeavor towards answering the issue did not reach a satisfying conclusion during my time.

A qualitative approach on pluripotency complexes

Early in 2015, right after PPIXpress and JDACO had their first productive versions, I performed my first attempt on such a study. I will roughly sketch out the elementary data and steps here. Using the publicly available, already processed and quantified transcript expression data on 5 H1 human embryonic stem cell (hESCs) samples from ENCODE [28], data on 16 terminally differentiated tissue samples from the Illumina Human BodyMap 2.0 (NCBI GEO accession GSE30611) and the human protein interactome of PrePPI [177], I constructed sample-specific interaction networks with PPIXpress using an expression value cutoff of 1.0, e.g. only transcript with an expression value above the threshold were considered abundant. Then, JDACO was used to predict complexes of TFs for each of the discretized stem cell and tissue interactomes. The 405 TFs annotated by the HOCOMOCO (version 9) motif knowledgebase [459] were utilized for that. As in the main project described in this Chapter, we already computed sample-specific complexomes from transcript expression data and unspecific protein interactions using PPIXpress and JDACO way ahead of CompleXChange. The important distinction here is that a fixed non-zero threshold was used to tailor the networks. The resulting TF complexes therefore lack a quantitative estimate and are just considered to be absent or abundant in each sample. As a consequence, a differential analysis could only be conducted qualitatively, say, for example, by applying Fisher's exact test.

To receive very concise results that would be suitable for the presentation at a conference I went an even more restricted route: instead of analyzing the differential abundance of protein complexes between the two groups of samples, the notion of the analysis was relaxed to only consider the change in TF combinations found in complexes rather than their exact protein compositions. 364 of such TF combinations were predicted in all H1hESC samples and 9,021 TF combinations found across all tissues samples. If one subtracted this union of combinations reported in the terminally differentiated tissue samples from

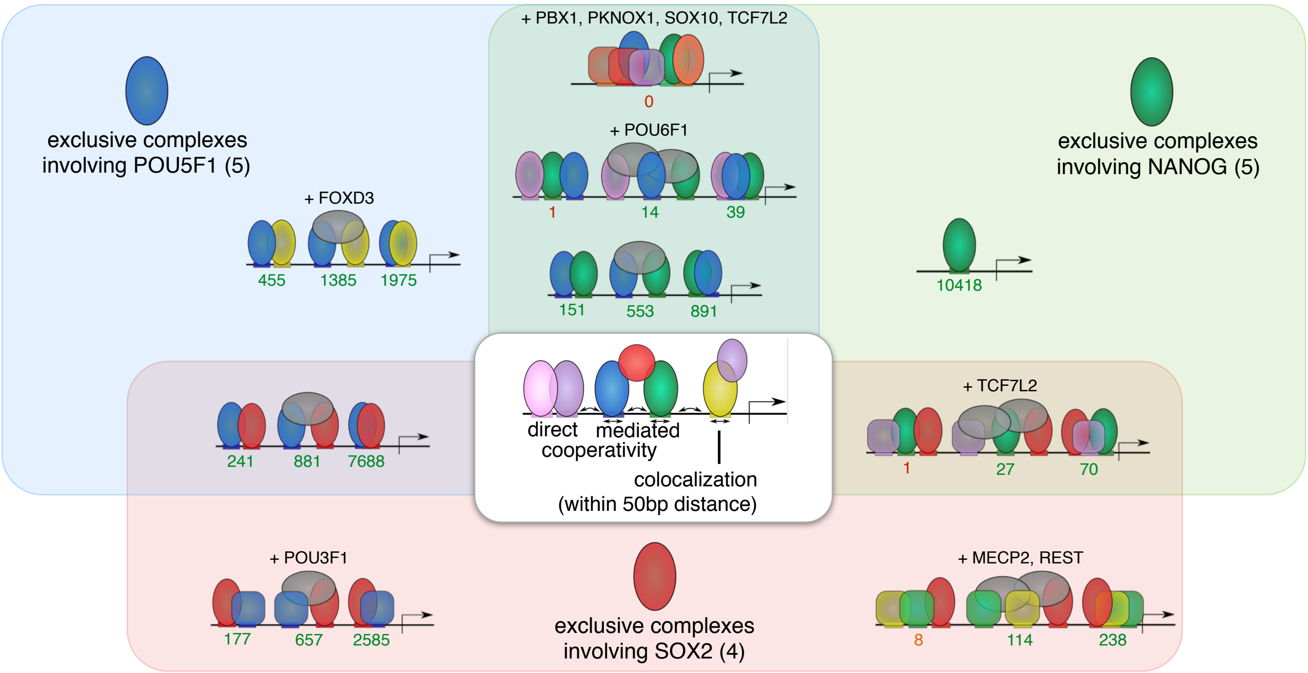

+ PBX1, PKNOX1, SOX10, TCF7L2
0
exclusive complexes involving POU5F1 (5)
+ POU6F1
1
14
39
exclusive complexes involving NANOG (5)
+ FOXD3
455
1385
1975
151
553
891
10418
direct
mediated
cooperativity
colocalization (within 50bp distance)
+ TCF7L2
241
881
7688
1
27
70
+ POU3F1
+ MECP2, REST
177
657
2585
exclusive complexes involving SOX2 (4)
8
114
238

Figure 6.11: *TF combinations involving classical master regulators of pluripotency. The blue area depicts all TF combinations in which POU5F1 (OCT4) was involved, the complexes with NANOG are shaded in green and the red background marks out all predicted TF complexes that had SOX2 as a member. Further TFs that were part of relevant combinations are noted textually. The numerical values below the schematic depictions of TFs binding to DNA mean the numbers of target genes of such TF combinations that possess binding sites for all the TFs obeying certain distance requirements. The exact number depends on the assumed type of cooperative interaction as outlined in the white box in the middle of the figure. Please refer to the main text for details. This graphic was originally used in several oral presentations on the topic that I gave over the years.*

the intersection of combinations that were seen in all stem cell samples, we were left with only 43 TF combinations that are solely found in all stem cells but never in any tissue sample.

Of these candidates, 9 contained one or several of the famous pluripotency factors OCT4 (POU5F1), NANOG or SOX2. Figure 6.11 shows that often two of the key factors occurred in the same complexes in our predictions, but never all three together. We also integrated data on the DNA sequences around promoters from the Eukaryotic Promoter Database [460], DNAse-seq data on H1 stem cells by ENCODE to select those promoters that are actually accessible in stem cells and computed binding sites using the respective HOCOMOCO motifs and the binding-site search tool FIMO [461]. TFs in complexes were then said to bind to a stretch of promoter DNA if they had alleged binding sites in direct adjacency along the sequence (pairwise distance within 0-10bp), if they could be cooperative using a mediated interaction (pairwise distance of sites within 10-50bp) or if they were colocalized (pairwise distance of sites within −50-50bp, the negative constraint allows for motif overlap). All but one of the 9 TF combinations that included a pluripotency factor indeed had target genes considering those very strict rulesets. We then used this knowledge on colo-

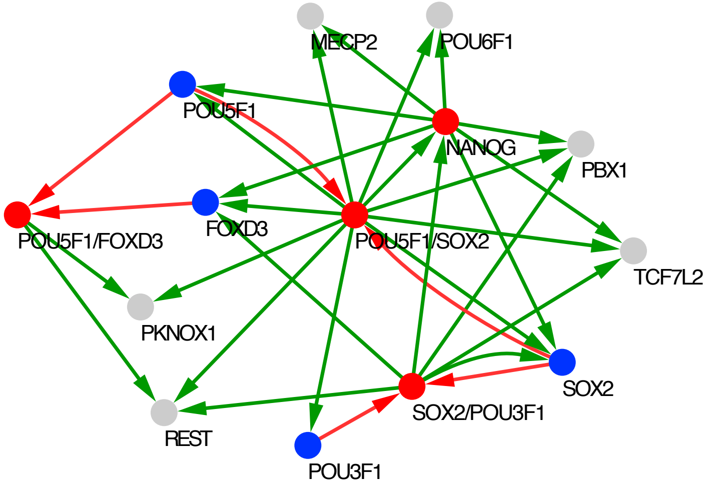


Figure 6.12: *Regulatory network among predicted TF combinations. To enable a very concise presentation, only the regulatory interactions between the 12 TFs that were part of the 9 TF combinations which were exclusively found in stem cells and involved a classical pluripotency factor are shown. Red nodes depict exclusive TF combinations (NANOG is a special case because it is the sole TF in at least one pluripotent exclusive complex), blue nodes are TF that are part of a complex in this network (and not exclusive combinations themselves) and grey nodes are TF that are not part of any complex in the network. Red arrows show a complex membership and green arrows denote a directed regulatory interaction. The network was visualized using Cytoscape [346].*

calization of interacting TFs to build a gene regulatory network among the 12 TFs found in this restricted set of TF combinations, see Figure 6.12. Complexes involving OCT4/SOX2 and NANOG showed an autoregulatory feed-forward loop in this network, for example. Given that both the OCT4/SOX2 complex and NANOG are needed for the activation of either OCT4 and SOX2, which is a reasonable assumption given that the three TFs are colocalizing significantly more often than by chance in such genomic regions of developmental relevance [210, 211], the dependency corresponds to a coherent type 1 feed-forward loop with AND-logic. Such a regulatory motif allows for a slow and thus robust activation of the loop but also a fast deactivation [462]. This core network of TFs and, more specifically in our case, TF complexes is thought to actively perpetuate this stable pluripotent cell state but, at the same time, also actively prevents a drift into alternative states which would lead to the inevitable expression of lineage specific factors [3].

While I still think the project was a valuable proof-of-concept application of the toolchain, there were flaws that I think rendered it unsuitable for a manuscript. First of all, I think the overall amount of samples in the data was too sparse and it is certainly arguable if the selected datasets were a good choice. While of high quality, the BodyMap data only considered terminally differentiated tissue samples, for example. In a differential analysis with such a very specific question a much broader ensemble of cellular samples from many different developmental stages would certainly suit the task better. When one was interested in a broad selection of transcript expression data of healthy human tissues and cell types at this time, the possibilities were fairly limited.

ENCODE only had released data of its first phase by then and the data of projects such as GTEx [283] (which anyhow only provides gene-quantified data and no raw data to the public), were not available yet. The ENCODE transcript expression data that was used was also not without its flaws. It originated from different labs (California Institute of Technology and Cold Spring Harbor Laboratory) that applied slightly different processing pipelines to the data. That this may affect the study became apparent when I clustered the complete set of ENCODE samples at that time in terms of their expression values. Notably, with the exception of the obviously very distinctive H1hESCs samples, all other samples clustered by lab and not by tissue. When the clustering was done on the basis of the discretized interactome, the problem was less pronounced but also not absent. This problem should be mostly curated by reprocessing of all data, something I did for the next iteration. Furthermore, I was always very critical about employing a discetized approach here, see also my respective discussion on that matter in Section 4.6.1.

A quantitative approach on pluripotency complexes

With CompleXChange ready to go, we eventually started a new iteration to tackle this question in 2017. Over time, the project was redefined and experienced significant changes until it has been finally concluded and abolished around summer 2019. The cornerstones of the new take on the issue should be employing a completely quantitative approach, to get rid of arbitrary but critical cutoffs, and the usage of way more data, that should be representative and processed identically this time. CompleXChange and its workflow satisfied the first requirement with ease. Luckily, the data question also became much simpler when ENCODE got way more data and a new portal to access its treasures [463]. The new interface allowed the programmatic access of all data and even incorporated additional new primary data sources, for example the data of the NIH Roadmap project [464].

I used the portal to filter and retrieve the raw read data for all paired-ended RNA-seq experiments that satisfied certain requirements. The data was supposed to be compliant to ENCODE's data quality standards, should be derived from complete cells (no fractions of cells) that had not been treated somehow, and their sequencing libraries should either had been created by a polyA-enriched (thus protein-coding enriched) or from a total RNA protocol. Suitable samples were then processed using kallisto and quantified to GENCODE protein-coding transcripts, just like in the CompleXChange project. After weeks of downloading and processing, a total of 442 transcript expression data samples were gathered and processed. Separating all samples tagged as induced pluripotent stem cells, H1 or H7 cell lines, the thus conceived dataset finally consisted of 11 pluripotent samples that were compared to 431 non-pluripotent samples of all developmental stages.

I then followed the same procedure that was described previously in Section 6.4.1 with slightly newer versions of PPIXpress and the necessary annotation data. Sample-specific interactomes were constructed on the basis of the human PrePPI data and PPIXpress whereby all transcripts with a non-zero expression value were taken into account and TF complexes were predicted with JDACO and the TF data by HOCOMOCO (version 11 this time) [220]. CompleXChange was then applied using its default options and with a new feature enabled that filters protein complexes in advance if they included proteins of the allosome. This obligatory filter is intended to correct for any sex-specific bias in the grouped data.

The pipeline predicted 3,023 TF complexes to be significantly deregulated (1,414 upregulated/1,609 downregulated in pluripotent cells) when the pluripotent samples and the broad set of non-pluripotent samples were compared using Wilcoxon rank-sum test. OCT4 and SOX2 were among the 5 highest enriched TFs in upregulated complexes whereas lineage-specific factors such as RUNX2, SMAD3 or NR2F2 were in the list of the 5 highest enriched TFs in complexes that were reported to be depleted in the pluripotent samples.

In earlier versions of this quantitative approach I also built gene regulatory networks of deregulated TF complexes by integrating the data on binding motifs, promoter sequences and chromatin accessibility as mentioned before. But in this setting those networks had, depending on the exact parameters used, in the range of half a million edges. Since such an order of magnitude in network size was hardly helpful, I pondered on how this information could be condensed in this context of core pluripotency regulators. A beneficial idea was that such a self-sufficient regulatory network should at the core consist of a strongly connected component, e.g. all pairs of nodes in the network can be reached by traversing directed edges. I could thus prune my complex-derived regulatory networks by selecting the largest strongly connected component. Algorithms to find strongly connected components in directed graphs, such as a classical depth-first algorithm by Robert Tarjan [465] that I implemented in my framework, only consider the topology of the network and have no notion of an edge or vertex type. In regulatory networks of TF complexes this is important, though, because a complex can certainly only exist if all its constituents, here the TFs, are still part of the network and can be removed completely if only one part is missing. Thus the condensation process iteratively selects the largest strongly connected component and removes nodes and edges of partially complete complexes until convergence is achieved. While this procedure indeed simplified the regulatory networks, they were still too large to nicely interpret and work with easily.

Anyhow, a main point of the study was to make use of the plethora of data generated by ENCODE. By integrating additional data sources, especially ChIP-seq data on histone modifications, we wanted to achieve the long-term goal of showing and validating regulatory effects in human that are exerted by predicted TF complexes. In the pluripotency case, H3K27ac signals seemed to be the most suitable candidate because they are the prime evidence of enhancer activity in general and in the developmental context alike [43, 210, 466]. The histone acetyltransferases EP300 and CREB-binding protein (CREBBP), both transcriptional coactivators that are recruited by TF complexes, are currently thought to be the main enzymes causing this specific histone acetylation at enhancers [467]. In the differential complexome results we indeed found 20 upregulated complexes including EP300 and 12 upregulated complexes that comprised CREBBP. These 32 complexes that included one of the cofactors were thus likely candidates that may be in charge of the activation and maintenance of the active state of pluripotency-specific enhancers. No deregulated TF complexes that included both histone acetyltransferases were reported by CompleXChange.

For the histone mark H3K27ac there was plenty of good data in ENCODE with 7 pluripotent vs 164 non-pluripotent samples (pluripotent being induced pluripotent cells and the cell lines H1, HUES6, HUES48, HUES64). Also, ENCODE had good ChIP-seq data on EP300 with a slightly smaller sample size of 2 pluripotent and 20 non-pluripotent samples. This dataset could be used as a more specific fallback to check for EP300 recruitment in particular. The data on CREBBP occurrences on the genome was not used due to its lower quality and inconsistency. We downloaded the samples uniformly processed in the popular Browser Extensible Data (BED) format, which is a convenient format to store scores or other values for defined genomic segments. Additionally we retrieved the enhancer region definitions of GeneHancer (version 4.4)[468]. We also retrieved the sequences of all the reference enhancer regions and computed binding sites for all TFs. These annotations served as a template to prune the genome-wide ChIP-seq data to a defined subset of known enhancers on which for every sample average scores were calculated when their peaks overlapped the reference regions. Since each sample then had the averaged scores for exactly the same genomic intervals, all samples could then be averaged within their group to finally obtain a mean score per defined enhancer for both pluripotent and non-pluripotent samples. These steps were conducted for both H3K27ac and EP300 ChIP-seq data individually.

For the evaluation, we first filtered for pluripotency enhancers by checking for which enhancers the average signal was zero in non-pluripotent samples and larger than zero in the pluripotent samples. In the H3K27ac data this was the case for 656 enhancer regions, and in the EP300 data 4391 enhancers were only bound in the pluripotent samples. Since these sets of enhancers are only active/bound in pluripotent samples and we determined which complexes were significantly upregulated there, we intended to assess if the upregulated TF complexes that included one of the histone modifiers of relevance were somehow outstanding regarding their average targeting of such regions or average ChIP-seq data scores in binding sites compared to downregulated or random TF complexes. Sadly that was only partially the case for the H3K27ac data and EP300/CREBBP-containing complexes that were upregulated in pluripotent samples and not at all the case for the EP300 data and the upregulated complexes that recruited the protein. Binding of a complex was again made dependent on the complete TF composition and only those binding events were considered in which all TFs of the complex had binding motifs in the enhancer sequence that in some way allowed for pairwise distances of −20-10bp between all TF pairs.

There are many reasons why these analyses could have failed. First of all, we determined TF binding from motifs rather than ChIP-seq data. With the ENCODE data this would have been possible for a small set of TFs for which we would then have experimentally verified binding events per sample rather than just context-free motif occurrences. This is important in the sense that we did not include additional data to aid the selection of relevant sites, e.g. on the openness and accessibility of the DNA (DNase-seq or ATAC-seq), which is generally helpful to judge the in-vivo activity of an event [217]. In the context of developmental processes and reprogramming this notion is blurred anyhow since most enhancer regions are generally accessible, especially when they are enriched in H3K27 acetylation [469], and special TFs that are called pioneer factors can even bind to regions of condensed chromatin [470, 471]. Furthermore, the assumptions that were tested were potentially too simplistic in terms of their restricted biological field of view. Only the very limited complexome of TF complexes that are upregulated in pluripotent cells was considered in the evaluation, although for each sample around $7,000$-$10,000$ TF complexes were predicted. We basically have no idea if other complexes may somehow prevent the recruitment of the tested complexes of interested to some of the the pluripotency enhancers we determined, e.g. by blocking binding regions, and thus would enable a finer grained view. More so, we did not take other classes of multiprotein complexes into account, for example, complexes of epigenetic reader and writer proteins [223] which may have a profound impact on posttranslational modifications of histone tails.

Nevertheless, the previous experiences thought us many things that can be built upon. Would I start over today with a third major iteration of finding key complexes in the regulation of pluripotency, I would certainly try to simplify the system being studied. To accomplish that, I would still rely on the vast resource of RNA-seq and accompanied data by ENCODE. But in contrast to my approaches before, I would ensure to turn every screw of the study design in a way that minimizes the search space of the problem in prior. A major tweak in that regard should be to only search for complexes of TFs for which we have ChIP-seq data on binding events in various cell types rather than to use the plethora of TFs for which we only have context-free probabilistic binding motif data. Moreover, the TF binding regions to be assessed at all could be even more refined by stringently employing differential analyses on the data. Such small adjustments to the overall pipeline should already tremendously improve the complexity of the result set and may thus be worth another try.

7 CONCLUSION AND OUTLOOK

Overall, the thesis follows a clear common thread and all intended methodical goals as outlined in Section 1.2 were ultimately met. Starting from the contextualization of protein interaction networks with PPIXpress, the differential analysis of such networks with PPICompare and the prediction of multiprotein complexes by JDACO, the theme of the thesis finally cumulates with the approach of CompleXChange, the differential complexome analysis which makes use of most tools previously developed in the context of this thesis.

7.1 CONCLUDING REMARKS TO ALL PROJECTS

The first method that we conceived in the course of this work was the tool PPIXpress [90] (see also Chapter 3) that allows to construct sample-specific protein-protein interaction networks (PPINs). In that function, it is the foundation for all my follow-up projects and the resulting differential interactome and differential complexome analysis pipelines.

PPINs are omnipresent in computational biology when it comes to the integration of network data. Still, it is often neglected that their composition of interactions is not a static entity but one that is, at the very least, dependent on the proteins that are abundant in the cellular state of interest. PPIXpress took this basic principle a step further and even considers the splicing state of the proteins to adjust individual edges of the network according to the composition of protein-coding transcripts that are expressed in the sample. This clear mapping of protein isoforms to viable protein interactions was achieved by dissecting all protein isoforms into the conserved protein domains that they include and by introducing data on the interactions between domain families. PPIXpress uses this domain-based interaction network of increased detail to relate protein interactions to domain interactions and then simply infers which protein interactions are actually supported in the sample from the domain-compositions of the most abundant isoforms of each protein.

We showed the benefit of the method by comparing personalized interaction networks of breast cancer patients from TCGA for which we constructed PPINs using both gene-based contextualization and our PPIXpress approach. Exploiting the transcript-level data enabled us to to detect a larger number of differential interactions between healthy and tumor tissues and among those, a significantly larger number of changes affected proteins that could be associated with cancerogenous processes. These positive effects and the overall difference between gene- and transcript-based results positively correlated with the amount of domain interaction data that was used during the construction. Furthermore, the results became clearly worse than even the gene-based adjustment when a random isoform was selected as the protein representative instead of the most abundant one, which further supported the model assumptions.

Eventually, when the first larger-scale data on isoform-sensitive interactomes [7] became available shortly after the publication of PPIXpress, the added value gained in the network contextualization by relating domain interactions to protein interactions could also be confirmed on the experimental dataset [207].

PPIXpress received quite a number of updates and some new features since its initial release. Besides necessary fixes regarding data that was retrieved, most new versions added support to new file formats or new data sources. But in addition to that, some new functionality was also added throughout its lifetime so far. Major examples are the option to reweight interactions instead of applying the discretized core approach that is described in the original manuscript or the integration of transcript biotype annotations to take further regulatory mechanisms into account that may affect the viability of the protein in the context. While they have not been pursued further yet, there are already elaborated ideas for the future of PPIXpress and its core principle of relating protein and domain interactions. PPIXpress itself could be expanded to also include data on short linear motifs and the interactions among them instead of only relying on data of Pfam domain families that is limited to highly conserved sequence segments. Besides the use-case of condition-specific PPINs, an algorithmic scheme that can associate modification of a particular sequence segment, or even a particular amino acid, with an influence on a defined interaction provides additional opportunities. One potential application that was already prototyped and tested by a student was the idea that such a mapping can also aid to infer if genome mutations affected protein interactions, for example.

Because we thought that the use case of identifying PPIN rewiring, as we showed for the case of breast cancer in the evaluation of PPIXpress, is likely of general interest, we heavily expanded the concept and designed PPICompare [92] (see also Chapter 4) as a dedicated tool for differential analysis of protein interactomes.

Since experimental characterization of proteomic rewiring events is currently not feasible on a larger scale, inferred protein interactomes with transcript-resolution likely present the best opportunity to somehow assess systematic changes in the interactome. Given two groups of PPIN samples, at best constructed with PPIXpress, the essential task of PPICompare is to identify all protein-protein interactions (PPIs) that are altered significantly often between the groups. The method does that by determining a general probability of rewiring across those samples which can then be used in a one-tailed binomial test to check if individual interactions are more often affected by rewiring events than expected by chance. Besides that, the tool monitors the transcriptomic cause that led to each individual change in connectivity and can report statistics on the matter. This relationship of causes and consequences, which links the change in expression data with the differential protein interactions, is then exploited to reveal a small set of alterations in the transcriptome that can explain all rewiring we see between groups and is the most likely of such sets given the data. PPICompare achieves that by relating rewiring events and transcriptomic reasons together in a bipartite graph and by then solving a weighted set-cover problem.

To test our new approach we selected the RNA-seq data on blood development produced by the BLUEPRINT project from which we constructed PPINs for 11 different cell types using PPIXpress. PPICompare was then applied to identify the changes made to the protein interactome during 10 developmental transitions in hematopoiesis. Using this dataset of developmental transitions, we not only evaluated what happened to the interactomes during such developmental leaps but, for the very first time, we could also systematically analyze the transcriptomic causes of each rewiring event and the biology behind that. A good example in that regard is the finding that interactions, which were exclusively deregulated by simultaneous up- or downregulation of both interacting partners, were more likely to participate in known protein complexes, had lower betweenness in the unpruned reference PPIN and were more likely to work on the same biological processes compared to interactions caused by regulatory modes in which only one protein was somehow deregulated or spliced. While it seems only natural that simultaneous deregulation of interaction partners is likely a means to control functional modules of proteins or stable multiprotein complexes, the study was able to deliver measurable facts that strongly indicated that. The contribution of alternative splicing (AS) only seemed to play a comparably minor role in our results in relation to other modes of regulation. Still, 871 rewiring events in all developmental transitions considered could only be fully explained by including AS and would have been missed by methods that only rely on gene expression. Furthermore, we found that the most important spliced proteins according to our optimization algorithm were strongly enriched in annotation terms that imply key roles of such isoform switches in the transcriptional regulation of cellular fate decisions.

In addition to the usage in fundamental research, a conceivable application of the tool could also have direct medical relevance. Although PPIs were often considered to be hardly feasible drug targets, the first PPI inhibitor medication Venetoclax was recently approved by the U.S. Food and Drug Administration [472] and others are currently facing clinical trials [403, 404]. We think that PPICompare's differential analysis on the network level could indeed be of great interest in this context because it allows drug researchers to vastly condense the relevant search space of potential target interactions in a very simple and fast manner. Concerning the methodology of PPICompare, it can be argued that the discretization that is applied to the PPINs in this model is certainly a critical parameter of the process which, when chosen rashly, may heavily affect the usability of the results. As broadly discussed in Section 4.6.1, this was done for reasons of conceptual simplicity and ultimately enabled the very clear and unambiguous relationship between interactomes and transcriptomes in the model, which could also be considered a strength of PPICompare. Because it works well for its envisaged tasks, there are no plans to alter the core methodology in that regard or to add major new features to the software.

The concept of the relevance of transcription factor (TF) complexes in transcriptional regulation and their prediction using the domain-aware cohesiveness optimization algorithm DACO [17, 18] are the foundation from which the central theme and ultimately all projects of direct relevance to my main research emerged. When we were able to construct PPINs that were tailored towards a specific sample, a new range of opportunities unfolded because these networks would also make it possible to search for the sample-specific complexes therein. While the DACO algorithm was up to the task in terms of the quality of the inferred complexes and already had some algorithmic tricks in charge, the runtime demands of the prototype implementation in Python may in fact impede its large-scale application in practice. We thus conceived the new Java implementation JDACO (see also Chapter 5) that is much faster than its predecessor, was adjusted for convenient usage with input data by PPIXpress and consequently was shed of its former data retrieval functionalities.

Besides switching to a faster programming language, the key consideration to gain performance was to increase the efficiency of parallel operations. For optimal utilization, the portion of non-parallel tasks was reduced heavily by redistributing the responsibilities of all individual components of the algorithm from a rather centralized organization to one in which each worker thread is empowered to also handle its share of data management overhead. Also, adjusted data structures were used on the interfaces between independent threads to diminish potential bottlenecks by minimizing the expected waiting times due to concurrent access of shared data. This internal redesign and its fresh Java implementation led to substantial savings in the compute time for practically relevant problem sizes.

Due to this effort, the prediction of complexes for even thousands of samples became feasible in appropriate time. We could thus regard the subproblem of being able to identify sample-specific complexomes as solved for our purposes. Still, we never considered protein complexes prediction as an endpoint in itself, but only as an important ingredient to be used in further data integration efforts or downstream analyses, like in the last approach shown in the thesis.

Finally, with the abilities to infer sample-specific interactomes and sample-specific protein complexes therein, the long-term objective of being able to determine differential protein complexomes could be concluded with CompleXChange [87] (see also Chapter 6).

My early approaches to that question handled the issue in a discretized way. By predicting protein interaction networks using a threshold on transcripts that was larger than zero, sample-specific complexes therein could be acquired by predicting the complexomes found in the contextualized interactomes. In this workflow, each complex could be either abundant or not abundant in the sample. When this discrete information is derived for groups of samples, Fisher's exact test could be utilized to allow a statistical assessment of differentially abundant protein complexes, for example. Depending on the availability of data, even more stringent and simpler solutions may be appropriate as we explained in Section 6.6.3. Still, it would be more elegant to have a quantitative approach that does not utilize such hard cutoffs. Although a large selection of quantitative data on gene and transcript expression or, increasingly more common, protein abundances are available, there are no quantitative measurements of protein complex abundances available yet. CompleXChange solves this problem by inferring abundance values from protein abundances (or rather gene expression measurements that we can associate with a protein) and the complexes that are present in a sample. The respective optimization algorithm works by iteratively distributing protein shares between complexes until convergence is attained. Thereby, two axiomatic constraints are satisfied, namely that the overall amount of each protein is fixed and that the abundance value assigned to a protein complex is determined by its least abundant member protein. With quantified complexes at hand, a numerical assessment of statistically relevant alterations is then straightforward.

Since it was the basis of the tool, we at first ensured that the iterative abundance estimation algorithm lived up to the task. For simulated data, our novel approach was shown to be much faster and to report better complex abundance approximations than the only comparable method based on linear programming. More so, when CompleXChange was applied to real biological data in the case of TF complexes in human monocyte subtypes and lymphoblastoids, we could demonstrate that the differential analysis methodology is rather robust concerning the detection of false-positive deregulation events. A comparison to differential gene expression results substantiated our basic premise that methods that somehow incorporate the notion of the physical interplay between proteins could potentially reveal information that cannot be gained by analyses that examine genes or proteins in isolation. We thus think analyzing differential complexomes, even if the latter are only inferred, has the capability to add insight to many transcriptomics studies even though the same experimental data constitutes the primary input.

The methodical foundation of CompleXChange is built on strong assumptions. Neither binding affinities between interacting proteins nor the exact stoichiometries of complex members are considered by the current implementation although they would certainly have a considerable impact on the outcomes of the abundance approximation step. However, if that information became widely available, it would be possible to adapt the algorithm correspondingly. As discussed in Section 6.6.2, it is just not very likely that the amount of relevant data will increase considerably in the foreseeable future.

7.2 OUTLOOK

Besides some enhancements to the tools and methods that were proposed previously or elaborated in depth in the corresponding Addenda, the huge untapped potential of the toolset presented in this thesis is hidden in the vastness of sequencing data that is already available. This is especially the case when one has concise scientific targets in mind for which candidate protein complexes or candidate protein interactions could now be detected easily by ready to use differential analysis pipelines. Only considering the broad area of gene regulation, protein complexes that are of combinatorial nature are found in and may be differentially abundant in distinct cellular conditions in the context of splicing, chromatin modifications or RNA-binding proteins.

One particular problem that has always been on our minds and did not reach a satisfying conclusion was to basically use the knowledge on differential complexomes to carve out the essence of pluripotency in terms of TF complexes. I briefly outlined two very different approaches on that manner in Section 6.6.3.

Adding to the point of data availability and the biological context of developmental processes, all methods that were presented in this thesis are able to utilize and, depending on the research question at hand, could benefit considerably from the usage of single-cell data. Although we never tested this assumption, there should generally be a certain improvement in the resolution of the data in practice because even in the most highly purified cell populations a high degree of heterogeneity will be present. When classical bulk measurements of samples are made, crucial details of individual cells are often blurred in the mixture and may go unnoticed because they were averaged out. This effect is especially pronounced in primed cells found in early developmental stages and certainly already affected our study on hematopoiesis in Chapter 4 [65, 355]. The availability of more data on individual cells may thus allow the revelation of otherwise concealed alterations and, what has not been mentioned yet, hugely increase the sample sizes of typical studies (with tradeoffs in variance and accuracy of each sample). With the progress that is currently made, being able to quantify even multiple different measurements in parallel for the very same cell, say expression data together with several histone mark annotations, may soon be a less extraordinary undertaking [473, 474].

Unfortunately, I did not even scratch on the surface of what I think could be achieved by the concept of gene regulatory networks of TF complexes as depicted in Figure 6.12. An essential motivation for the overall theme of my thesis was the assumption that knowledge on TF complexes would entail information on the logical wiring between TFs in the regulatory network and would aid to apprehend the regulatory mechanism of the complex. The functionality for many of these ideas is already part of the Java framework that I designed and implemented during my time as a doctoral student.

Given that all DNA-binding factors are relevant for the binding event and that all their binding sites must obey certain distance constraints to spatially allow for a stable interaction with the DNA, the composition of TFs in a complex candidate specifies very clearly where the candidate is able to bind. More so, because all member proteins of the complex are needed for such a specific regulatory targeting, the principle implements an AND-logic for gene regulatory circuits. Because we can predict TF complexes and have quite a diverse assortment of data on binding events and binding motifs, such rules can be derived automatically using the software we introduced and may certainly benefit the often handcrafted models used in Boolean network studies or stochastic simulations of regulatory networks. Due to the introduction of distinct node-types and the logical dependencies encoded, the topologies of regulatory networks of TF complexes are likely differing from established gene regulatory networks in the notion of TFs and their targets. It would be interesting to investigate the network motifs that are commonly found in the complexome-based networks, for example.

Furthermore, another benefit of TF complexes is that they can tell us about the potential recruitment of cofactors to defined genomic regions. With our current expertise on functional annotation of genes and proteins, as by the data of the Gene Ontology for example, assigning clear regulatory effects to the components of a complex became a rather simple task. How this information on the individual complex members, which could be very detailed functional depictions or just coarse-grained annotations, should be integrated to best categorize a TF complex is a topic that certainly needs further research.

In summary, there is already a plethora of uncharted territory and potential applications left for exploration in terms of differential interactomes, complexomes, and the software developed during my dissertation projects in general. Moreover, when the advent of single-cell techniques will soon be followed by a broad availability of respective datasets in the near future, it will be possible to eliminate a huge confounding factor in such analyses.

BIBLIOGRAPHY

[1] E. Bianconi et al. "An estimation of the number of cells in the human body." In: *Ann. Hum. Biol.* 40.6 (2013), pp. 463–471.

[2] B. Alberts. *Molecular Biology of the Cell*. 5th edition. Garland Science, 2008. ISBN: 9780815341062.

[3] E. H. Davidson. *The regulatory genome: gene regulatory networks in development and evolution*. Elsevier, 2010.

[4] M. Levine and R. Tjian. "Transcription regulation and animal diversity." In: *Nature* 424.6945 (2003), pp. 147–151.

[5] B. J. Blencowe. "Alternative splicing: new insights from global analyses." In: *Cell* 126.1 (2006), pp. 37–47.

[6] N. L. Barbosa-Morais et al. "The evolutionary landscape of alternative splicing in vertebrate species." In: *Science* 338.6114 (2012), pp. 1587–1593.

[7] X. Yang et al. "Widespread Expansion of Protein Interaction Capabilities by Alternative Splicing." In: *Cell* 164.4 (2016), pp. 805–817.

[8] K. M. Lelli, M. Slattery, and R. S. Mann. "Disentangling the many layers of eukaryotic transcriptional regulation." In: *Annu. Rev. Genet.* 46 (2012), pp. 43–68.

[9] R. Stadhouders, G. J. Filion, and T. Graf. "Transcription factors and 3D genome conformation in cell-fate decisions." In: *Nature* 569.7756 (May 2019), pp. 345–354.

[10] A. J. Bannister and T. Kouzarides. "Regulation of chromatin by histone modifications." In: *Cell Res.* 21.3 (2011), pp. 381–395.

[11] T. B. Miranda and P. A. Jones. "DNA methylation: the nuts and bolts of repression." In: *J. Cell. Physiol.* 213.2 (2007), pp. 384–390.

[12] B. Alberts. "The cell as a collection of protein machines: preparing the next generation of molecular biologists." In: *Cell* 92.3 (1998), pp. 291–294.

[13] F. Spitz and E. E. Furlong. "Transcription factors: from enhancer binding to developmental control." In: *Nat. Rev. Genet.* 13.9 (2012), pp. 613–626.

[14] G. A. Wray et al. "The evolution of transcriptional regulation in eukaryotes." In: *Mol. Biol. Evol.* 20.9 (2003), pp. 1377–1419.

[15] J. Zhao et al. "The network organization of cancer-associated protein complexes in human tissues." In: *Sci Rep* 3 (2013), p. 1583.

[16] D. S. Latchman. *Eukaryotic transcription factors*. 5th edition. Academic press, 2008.

[17] T. Will and V. Helms. "Identifying transcription factor complexes and their roles." In: *Bioinformatics* 30.17 (2014), pp. i415–421.

[18] T. Will. *Predicting Transcription Factor Complexes: A Novel Approach to Data Integration in Systems Biology*. Springer, 2014.

[19] H. M. Berman et al. "The Protein Data Bank." In: *Nucleic Acids Res.* 28.1 (2000), pp. 235–242.

[20] J. Kallen et al. "Structural States of Hdm2 and HdmX: X-ray Elucidation of Adaptations and Binding Interactions for Different Chemical Compound Classes." In: *ChemMedChem* 14.14 (2019), pp. 1305–1314.

[21] A. S. Rose et al. "NGL viewer: web-based molecular graphics for large complexes." In: *Bioinformatics* 34.21 (Nov. 2018), pp. 3755–3758.

[22] R. Akbani et al. "Genomic Classification of Cutaneous Melanoma." In: *Cell* 161.7 (2015), pp. 1681–1696.

[23] P. A. Ascierto et al. "The role of BRAF V600 mutation in melanoma." In: *J Transl Med* 10 (2012), p. 85.

[24] H. Davies, G. R. Bignell, C. Cox, et al. "Mutations of the BRAF gene in human cancer." In: *Nature* 417.6892 (2002), pp. 949–954.

[25] S. Aibar et al. "SCENIC: single-cell regulatory network inference and clustering." In: *Nat. Methods* 14.11 (2017), pp. 1083–1086.

[26] E. Shoshan et al. "NFAT1 Directly Regulates IL8 and MMP3 to Promote Melanoma Tumor Growth and Metastasis." In: *Cancer Res.* 76.11 (June 2016), pp. 3145–3155.

[27] J. Lonsdale et al. "The Genotype-Tissue Expression (GTEx) project." In: *Nat. Genet.* 45.6 (2013), pp. 580–585.

[28] ENCODE Project Consortium. "An integrated encyclopedia of DNA elements in the human genome." In: *Nature* 489.7414 (2012), pp. 57–74.

[29] J. D. Watson and F. H. Crick. "Molecular structure of nucleic acids; a structure for deoxyribose nucleic acid." In: *Nature* 171.4356 (1953), pp. 737–738.

[30] K. Luger, M. L. Dechassa, and D. J. Tremethick. "New insights into nucleosome and chromatin structure: an ordered state or a disordered affair?" In: *Nat. Rev. Mol. Cell Biol.* 13.7 (2012), pp. 436–447.

[31] International Human Genome Sequencing Consortium. "Finishing the euchromatic sequence of the human genome." In: *Nature* 431.7011 (2004), pp. 931–945.

[32] J. C. Venter et al. "The sequence of the human genome." In: *Science* 291.5507 (2001), pp. 1304–1351.

[33] D. M. Church et al. "Modernizing reference genome assemblies." In: *PLoS Biol.* 9.7 (2011), e1001091.

[34] S. Ballouz, A. Dobin, and J. A. Gillis. "Is it time to change the reference genome?" In: *Genome Biol.* 20.1 (Aug. 2019), p. 159.

[35] F. H. Crick. "On protein synthesis." In: *Symp. Soc. Exp. Biol.* 12 (1958), pp. 138–163.

[36] C. Willyard. "New human gene tally reignites debate." In: *Nature* 558.7710 (June 2018), pp. 354–355.

[37] A. Morillon and D. Gautheret. “Bridging the gap between reference and real transcriptomes.” In: *Genome Biol.* 20.1 (June 2019), p. 112.

[38] P. P. Amaral et al. “The eukaryotic genome as an RNA machine.” In: *Science* 319.5871 (2008), pp. 1787–1789.

[39] L. Ma, V. B. Bajic, and Z. Zhang. “On the classification of long non-coding RNAs.” In: *RNA Biol* 10.6 (2013), pp. 925–933.

[40] V. Haberle and A. Stark. “Eukaryotic core promoters and the functional basis of transcription initiation.” In: *Nat. Rev. Mol. Cell Biol.* 19.10 (Oct. 2018), pp. 621–637.

[41] G. A. Maston, S. K. Evans, and M. R. Green. “Transcriptional regulatory elements in the human genome.” In: *Annu Rev Genomics Hum Genet* 7 (2006), pp. 29–59.

[42] K. S. Manning and T. A. Cooper. “The roles of RNA processing in translating genotype to phenotype.” In: *Nat. Rev. Mol. Cell Biol.* 18.2 (Feb. 2017), pp. 102–114.

[43] J. Ernst et al. “Mapping and analysis of chromatin state dynamics in nine human cell types.” In: *Nature* 473.7345 (2011), pp. 43–49.

[44] Q. Pan et al. “Deep surveying of alternative splicing complexity in the human transcriptome by high-throughput sequencing.” In: *Nat. Genet.* 40.12 (2008), pp. 1413–1415.

[45] M. Buljan et al. “Tissue-specific splicing of disordered segments that embed binding motifs rewires protein interaction networks.” In: *Mol. Cell* 46.6 (2012), pp. 871–883.

[46] J. D. Ellis et al. “Tissue-specific alternative splicing remodels protein-protein interaction networks.” In: *Mol. Cell* 46.6 (2012), pp. 884–892.

[47] A. M. Miederer et al. “A STIM2 splice variant negatively regulates store-operated calcium entry.” In: *Nat Commun* 6 (2015), p. 6899.

[48] F. E. Baralle and J. Giudice. “Alternative splicing as a regulator of development and tissue identity.” In: *Nat. Rev. Mol. Cell Biol.* 18.7 (July 2017), pp. 437–451.

[49] X. Wang and J. Dai. “Concise review: isoforms of OCT4 contribute to the confusing diversity in stem cell biology.” In: *Stem Cells* 28.5 (2010), pp. 885–893.

[50] Y. Xu et al. “Alternative splicing links histone modifications to stem cell fate decision.” In: *Genome Biol.* 19.1 (Sept. 2018), p. 133.

[51] R. Corominas et al. “Protein interaction network of alternatively spliced isoforms from brain links genetic risk factors for autism.” In: *Nat Commun* 5 (2014), p. 3650.

[52] C. J. David and J. L. Manley. “Alternative pre-mRNA splicing regulation in cancer: pathways and programs unhinged.” In: *Genes Dev.* 24.21 (2010), pp. 2343–2364.

[53] J. P. Venables et al. “Cancer-associated regulation of alternative splicing.” In: *Nat. Struct. Mol. Biol.* 16.6 (2009), pp. 670–676.

[54] M. Danan-Gotthold et al. "Identification of recurrent regulated alternative splicing events across human solid tumors." In: *Nucleic Acids Res.* (2015).

[55] J. C. Alwine, D. J. Kemp, and G. R. Stark. "Method for detection of specific RNAs in agarose gels by transfer to diazobenzyloxymethyl-paper and hybridization with DNA probes." In: *Proc. Natl. Acad. Sci. U.S.A.* 74.12 (1977), pp. 5350–5354.

[56] M. Becker-Andre and K. Hahlbrock. "Absolute mRNA quantification using the polymerase chain reaction (PCR). A novel approach by a PCR aided transcript titration assay (PATTY)." In: *Nucleic Acids Res.* 17.22 (1989), pp. 9437–9446.

[57] V. E. Velculescu, L. Zhang, et al. "Serial analysis of gene expression." In: *Science* 270.5235 (1995), pp. 484–487.

[58] M. Schena et al. "Quantitative monitoring of gene expression patterns with a complementary DNA microarray." In: *Science* 270.5235 (1995), pp. 467–470.

[59] F. Sanger, S. Nicklen, and A. R. Coulson. "DNA sequencing with chain-terminating inhibitors." In: *Proc. Natl. Acad. Sci. U.S.A.* 74.12 (1977), pp. 5463–5467.

[60] J. M. Heather and B. Chain. "The sequence of sequencers: The history of sequencing DNA." In: *Genomics* 107.1 (2016), pp. 1–8.

[61] O. Morozova and M. A. Marra. "Applications of next-generation sequencing technologies in functional genomics." In: *Genomics* 92.5 (2008), pp. 255–264.

[62] M. Margulies et al. "Genome sequencing in microfabricated high-density picolitre reactors." In: *Nature* 437.7057 (2005), pp. 376–380.

[63] J. Shendure et al. "Accurate multiplex polony sequencing of an evolved bacterial genome." In: *Science* 309.5741 (2005), pp. 1728–1732.

[64] Simon Bennett. "Solexa LTD." In: *Pharmacogenomics* 5.4 (2004), pp. 433–438.

[65] B. Hwang, J. H. Lee, and D. Bang. "Single-cell RNA sequencing technologies and bioinformatics pipelines." In: *Exp. Mol. Med.* 50.8 (Aug. 2018), p. 96.

[66] A. Mortazavi et al. "Mapping and quantifying mammalian transcriptomes by RNA-Seq." In: *Nat. Methods* 5.7 (2008), pp. 621–628.

[67] R. Lowe et al. "Transcriptomics technologies." In: *PLoS Comput. Biol.* 13.5 (May 2017), e1005457.

[68] A. Hatem et al. "Benchmarking short sequence mapping tools." In: *BMC Bioinformatics* 14 (2013), p. 184.

[69] D. R. Zerbino et al. "Ensembl 2018." In: *Nucleic Acids Res.* 46.D1 (Jan. 2018), pp. D754–D761.

[70] N. A. O'Leary et al. "Reference sequence (RefSeq) database at NCBI: current status, taxonomic expansion, and functional annotation." In: *Nucleic Acids Res.* 44.D1 (2016), pp. D733–745.

[71] J. Harrow et al. "GENCODE: the reference human genome annotation for The ENCODE Project." In: *Genome Res.* 22.9 (2012), pp. 1760–1774.

[72] P. Y. Wu, J. H. Phan, and M. D. Wang. "Assessing the impact of human genome annotation choice on RNA-seq expression estimates." In: *BMC Bioinformatics* 14 Suppl 11 (2013), S8.

[73] B. Li and C. N. Dewey. "RSEM: accurate transcript quantification from RNA-Seq data with or without a reference genome." In: *BMC Bioinformatics* 12 (2011), p. 323.

[74] N. L. Bray et al. "Near-optimal probabilistic RNA-seq quantification." In: *Nat. Biotechnol.* 34.5 (2016), pp. 525–527.

[75] R. Patro et al. "Salmon provides fast and bias-aware quantification of transcript expression." In: *Nat. Methods* 14.4 (2017), pp. 417–419.

[76] C. Trapnell et al. "Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation." In: *Nat. Biotechnol.* 28.5 (2010), pp. 511–515.

[77] B. Li et al. "RNA-Seq gene expression estimation with read mapping uncertainty." In: *Bioinformatics* 26.4 (2010), pp. 493–500.

[78] G. P. Wagner, K. Kin, and V. J. Lynch. "Measurement of mRNA abundance using RNA-seq data: RPKM measure is inconsistent among samples." In: *Theory Biosci.* 131.4 (2012), pp. 281–285.

[79] S. Anders and W. Huber. "Differential expression analysis for sequence count data." In: *Genome Biol.* 11.10 (2010), R106.

[80] F. Rapaport et al. "Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data." In: *Genome Biol.* 14.9 (2013), R95.

[81] I. Ullah et al. "Significance tests for analyzing gene expression data with small sample sizes." In: *Bioinformatics* 35.20 (2019), pp. 3996–4003.

[82] M. Crow et al. "Predictability of human differential gene expression." In: *Proc. Natl. Acad. Sci. U.S.A.* 116.13 (Mar. 2019), pp. 6491–6500.

[83] H. Pimentel et al. "Differential analysis of RNA-seq incorporating quantification uncertainty." In: *Nat. Methods* 14.7 (2017), pp. 687–690.

[84] T. Barrett et al. "NCBI GEO: archive for functional genomics data sets–update." In: *Nucleic Acids Res.* 41.Database issue (2013), pp. D991–995.

[85] Y. Kodama et al. "The Sequence Read Archive: explosive growth of sequencing data." In: *Nucleic Acids Res.* 40.Database issue (2012), pp. D54–56.

[86] N. Kolesnikov et al. "ArrayExpress update–simplifying data submissions." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D1113–1116.

[87] T. Will and V. Helms. "Differential analysis of protein complexes with CompleXChange." In: *BMC Bioinformatics* 20.1 (June 2019), p. 300.

[88] J. N. Weinstein et al. “The Cancer Genome Atlas Pan-Cancer analysis project.” In: *Nat. Genet.* 45.10 (2013), pp. 1113–1120.

[89] L. Chen et al. “Transcriptional diversity during lineage commitment of human blood progenitors.” In: *Science* 345.6204 (2014), p. 1251033.

[90] T. Will and V. Helms. “PPIXpress: construction of condition-specific protein interaction networks based on transcript expression.” In: *Bioinformatics* 32.4 (2016), pp. 571–578.

[91] X. Zhang, C. S. Gibhardt, T. Will, et al. “Redox signals at the ER-mitochondria interface control melanoma progression.” In: *EMBO J.* 38.15 (2019), e100871.

[92] T. Will and V. Helms. “Rewiring of the inferred protein interactome during blood development studied with the tool PPICompare.” In: *BMC Syst Biol* 11.1 (Apr. 2017), p. 44.

[93] M. Nazarieh et al. “TFmiR2: Constructing and analyzing disease-, tissue- and process-specific transcription factor and microRNA co-regulatory networks.” In: *Bioinformatics* (2019).

[94] Carl Ivar Branden and John Tooze. *Introduction to protein structure*. 2nd edition. Garland Science, 1999.

[95] A. K. Dunker et al. “Flexible nets. The roles of intrinsic disorder in protein interaction networks.” In: *FEBS J.* 272.20 (2005), pp. 5129–5148.

[96] V. N. Uversky. “Intrinsic Disorder, Protein-Protein Interactions, and Disease.” In: *Adv Protein Chem Struct Biol* 110 (2018), pp. 85–121.

[97] S. Prabakaran et al. “Post-translational modification: nature’s escape from genetic imprisonment and the basis for dynamic information encoding.” In: *Wiley Interdiscip Rev Syst Biol Med* 4.6 (2012), pp. 565–583.

[98] C. Chothia and J. Gough. “Genomic and structural aspects of protein evolution.” In: *Biochem. J.* 419.1 (2009), pp. 15–28.

[99] B. Smithers, M. Oates, and J. Gough. “’Why genes in pieces?’-revisited.” In: *Nucleic Acids Res.* 47.10 (June 2019), pp. 4970–4973.

[100] C. A. Ouzounis et al. “Classification schemes for protein structure and function.” In: *Nat. Rev. Genet.* 4.7 (2003), pp. 508–519.

[101] A. G. Murzin et al. “SCOP: a structural classification of proteins database for the investigation of sequences and structures.” In: *J. Mol. Biol.* 247.4 (1995), pp. 536–540.

[102] C. A. Orengo et al. “CATH - a hierarchic classification of protein domain structures.” In: *Structure* 5.8 (1997), pp. 1093–1108.

[103] E. L. Sonnhammer, S. R. Eddy, and R. Durbin. “Pfam: a comprehensive database of protein domain families based on seed alignments.” In: *Proteins* 28.3 (1997), pp. 405–420.

[104] G. Riddihough. “More meanders and sandwiches.” In: *Nat. Struct. Biol.* 1.11 (1994), pp. 755–757.

[105] S. Yellaboina et al. "DOMINE: a comprehensive collection of known and predicted domain-domain interactions." In: *Nucleic Acids Res.* 39.Database issue (2011), pp. D730–735.

[106] Y. Kim et al. "IDDI: integrated domain-domain interaction and protein interaction analysis system." In: *Proteome Sci* 10 Suppl 1 (2012), S9.

[107] R. Mosca et al. "3did: a catalog of domain-based interactions of known three-dimensional structure." In: *Nucleic Acids Res.* 42.Database issue (2014), pp. D374–379.

[108] R. D. Finn et al. "iPfam: a database of protein family and domain interactions found in the Protein Data Bank." In: *Nucleic Acids Res.* 42.Database issue (2014), pp. D364–373.

[109] UniProt Consortium. "UniProt: a worldwide hub of protein knowledge." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D506–D515.

[110] R. D. Finn et al. "The Pfam protein families database." In: *Nucleic Acids Res.* 38.Database issue (2010), pp. D211–222.

[111] J. Mistry et al. "The challenge of increasing Pfam coverage of the human proteome." In: *Database (Oxford)* 2013 (2013), bat023.

[112] S. El-Gebali et al. "The Pfam protein families database in 2019." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D427–D432.

[113] J. Cox and M. Mann. "Is proteomics the new genomics?" In: *Cell* 130.3 (2007), pp. 395–398.

[114] A. F. Altelaar, J. Munoz, and A. J. Heck. "Next-generation proteomics: towards an integrative view of proteome dynamics." In: *Nat. Rev. Genet.* 14.1 (2013), pp. 35–48.

[115] J. Renart, J. Reiser, and G. R. Stark. "Transfer of proteins from gels to diazobenzyloxymethyl-paper and detection with antisera: a method for studying antibody specificity and antigen structure." In: *Proc. Natl. Acad. Sci. U.S.A.* 76.7 (1979), pp. 3116–3120.

[116] W. N. Burnette. ""Western blotting": electrophoretic transfer of proteins from sodium dodecyl sulfate-polyacrylamide gels to unmodified nitrocellulose and radiographic detection with antibody and radioiodinated protein A." In: *Anal. Biochem.* 112.2 (1981), pp. 195–203.

[117] R. Aebersold and M. Mann. "Mass spectrometry-based proteomics." In: *Nature* 422.6928 (2003), pp. 198–207.

[118] O. T. Schubert et al. "Quantitative proteomics: challenges and opportunities in basic and applied research." In: *Nat Protoc* 12.7 (2017), pp. 1289–1294.

[119] S. E. Ong and M. Mann. "A practical recipe for stable isotope labeling by amino acids in cell culture (SILAC)." In: *Nat Protoc* 1.6 (2006), pp. 2650–2660.

[120] C. Vogel et al. "Sequence signatures and mRNA concentration can explain two-thirds of protein abundance variation in a human cell line." In: *Mol. Syst. Biol.* 6 (2010), p. 400.

[121] B. Schwanhäusser et al. "Global quantification of mammalian gene expression control." In: *Nature* 473.7347 (2011), pp. 337–342.

[122] M. S. Robles, J. Cox, and M. Mann. "In-vivo quantitative proteomics reveals a key contribution of post-transcriptional mechanisms to the circadian regulation of liver metabolism." In: *PLoS Genet.* 10.1 (2014), e1004047.

[123] M. Jovanovic et al. "Dynamic profiling of the protein life cycle in response to pathogens." In: *Science* 347.6226 (2015), p. 1259038.

[124] A. Ori et al. "Spatiotemporal variation of mammalian protein complex stoichiometries." In: *Genome Biol.* 17 (2016), p. 47.

[125] Z. Cheng et al. "Differential dynamics of the mammalian mRNA and protein expression response to misfolding stress." In: *Mol. Syst. Biol.* 12.1 (2016), p. 855.

[126] A. L. Bauernfeind and C. C. Babbitt. "The predictive nature of transcript expression levels on protein expression in adult human brain." In: *BMC Genomics* 18.1 (Apr. 2017), p. 322.

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

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

[129] M. Wilhelm et al. "Mass-spectrometry-based draft of the human proteome." In: *Nature* 509.7502 (2014), pp. 582–587.

[130] I. M. Nooren and J. M. Thornton. "Diversity of protein-protein interactions." In: *EMBO J.* 22.14 (2003), pp. 3486–3492.

[131] S. Jones and J. M. Thornton. "Principles of protein-protein interactions." In: *Proc. Natl. Acad. Sci. U.S.A.* 93.1 (1996), pp. 13–20.

[132] C. Yan et al. "Characterization of protein-protein interfaces." In: *Protein J.* 27.1 (2008), pp. 59–70.

[133] A. C. Gavin et al. "Functional organization of the yeast proteome by systematic analysis of protein complexes." In: *Nature* 415.6868 (2002), pp. 141–147.

[134] N. J. Krogan et al. "Global landscape of protein complexes in the yeast Saccharomyces cerevisiae." In: *Nature* 440.7084 (2006), pp. 637–643.

[135] U. Stelzl et al. "A human protein-protein interaction network: a resource for annotating the proteome." In: *Cell* 122.6 (2005), pp. 957–968.

[136] J. F. Rual et al. "Towards a proteome-scale map of the human protein-protein interaction network." In: *Nature* 437.7062 (2005), pp. 1173–1178.

[137] T. Rolland et al. "A proteome-scale map of the human interactome network." In: *Cell* 159.5 (2014), pp. 1212–1226.

[138] M. Y. Hein et al. "A human interactome in three quantitative dimensions organized by stoichiometries and abundances." In: *Cell* 163.3 (2015), pp. 712–723.

[139] G. T. Hart et al. "How complete are current yeast and human protein-interaction networks?" In: *Genome Biol.* 7.11 (2006), p. 120.

[140] M. P. Stumpf et al. "Estimating the size of the human interactome." In: *Proc. Natl. Acad. Sci. U.S.A.* 105.19 (2008), pp. 6959–6964.

[141] K. Luck et al. "Proteome-Scale Human Interactomics." In: *Trends Biochem. Sci.* 42.5 (May 2017), pp. 342–354.

[142] R. Oughtred et al. "The BioGRID interaction database: 2019 update." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D529–D541.

[143] R. Jansen, D. Greenbaum, and M. Gerstein. "Relating whole-genome expression data with protein-protein interactions." In: *Genome Res.* 12.1 (2002), pp. 37–46.

[144] J. D. Han et al. "Evidence for dynamically organized modularity in the yeast protein-protein interaction network." In: *Nature* 430.6995 (2004), pp. 88–93.

[145] P. M. Kim et al. "Relating three-dimensional structures to protein networks provides evolutionary insights." In: *Science* 314.5807 (2006), pp. 1938–1941.

[146] P. Aloy and R. B. Russell. "Structural systems biology: modelling protein interactions." In: *Nat. Rev. Mol. Cell Biol.* 7.3 (2006), pp. 188–197.

[147] I. Wohlgemuth et al. "Studying macromolecular complex stoichiometries by peptide-based mass spectrometry." In: *Proteomics* 15.5-6 (2015), pp. 862–879.

[148] A. Verger et al. "Twenty years of Mediator complex structural studies." In: *Biochem. Soc. Trans.* 47.1 (Feb. 2019), pp. 399–410.

[149] M. Dietzen et al. "Large oligomeric complex structures can be computationally assembled by efficiently combining docked interfaces." In: *Proteins* 83.10 (2015), pp. 1887–1899.

[150] J. Snider et al. "Fundamentals of protein interaction network mapping." In: *Mol. Syst. Biol.* 11.12 (2015), p. 848.

[151] S. Fields and O. Song. "A novel genetic system to detect protein-protein interactions." In: *Nature* 340.6230 (1989), pp. 245–246.

[152] M. Dreze et al. "High-quality binary interactome mapping." In: *Meth. Enzymol.* 470 (2010), pp. 281–315.

[153] N. Sahni et al. "Widespread macromolecular interaction perturbations in human genetic disorders." In: *Cell* 161.3 (2015), pp. 647–660.

[154] W. H. Dunham, M. Mullin, and A. C. Gingras. "Affinity-purification coupled to mass spectrometry: basic principles and strategies." In: *Proteomics* 12.10 (2012), pp. 1576–1590.

[155] T. Clancy and E. Hovig. "From proteomes to complexomes in the era of systems biology." In: *Proteomics* 14.1 (2014), pp. 24–41.

[156] B. Lehne and T. Schlitt. "Protein-protein interaction databases: keeping up with growing interactomes." In: *Hum. Genomics* 3.3 (2009), pp. 291–297.

[157] G. D. Bader and C. W. Hogue. “Analyzing yeast protein-protein interaction data obtained from different sources.” In: *Nat. Biotechnol.* 20.10 (2002), pp. 991–997.

[158] S. Orchard et al. “The MIntAct project–IntAct as a common curation platform for 11 molecular interaction databases.” In: *Nucleic Acids Res.* 42.Database issue (2014), pp. D358–363.

[159] L. Licata et al. “MINT, the molecular interaction database: 2012 update.” In: *Nucleic Acids Res.* 40.Database issue (2012), pp. D857–861.

[160] S. Razick, G. Magklaras, and I. M. Donaldson. “iRefIndex: a consolidated protein interaction database with provenance.” In: *BMC Bioinformatics* 9 (2008), p. 405.

[161] A. Calderone, L. Castagnoli, and G. Cesareni. “mentha: a resource for browsing integrated protein-interaction networks.” In: *Nat. Methods* 10.8 (2013), pp. 690–691.

[162] G. Alanis-Lobato, M. A. Andrade-Navarro, and M. H. Schaefer. “HIPPIE v2.0: enhancing meaningfulness and reliability of protein-protein interaction networks.” In: *Nucleic Acids Res.* 45.D1 (Jan. 2017), pp. D408–D414.

[163] J. I. Garzon et al. “A computational interactome and functional annotation for the human proteome.” In: *Elife* 5 (2016), e18715.

[164] D. Szklarczyk et al. “STRING v10: protein-protein interaction networks, integrated over the tree of life.” In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D447–452.

[165] A. Bossi and B. Lehner. “Tissue specificity and the human protein interaction network.” In: *Mol. Syst. Biol.* 5 (2009), p. 260.

[166] T. J. Lopes et al. “Tissue-specific subnetworks and characteristics of publicly available human protein interaction databases.” In: *Bioinformatics* 27.17 (2011), pp. 2414–2421.

[167] A. Sinha and H. A. Nagarajaram. “Nodes occupying central positions in human tissue specific PPI networks are enriched with many splice variants.” In: *Proteomics* 14.20 (2014), pp. 2242–2248.

[168] R. Barshir et al. “Comparative analysis of human tissue interactomes reveals factors leading to tissue-specific manifestation of hereditary diseases.” In: *PLoS Comput. Biol.* 10.6 (2014), e1003632.

[169] O. Magger et al. “Enhancing the prioritization of disease-causing genes through tissue specific protein interaction networks.” In: *PLoS Comput. Biol.* 8.9 (2012), e1002690.

[170] C. von Mering et al. “Comparative assessment of large-scale data sets of protein-protein interactions.” In: *Nature* 417.6887 (2002), pp. 399–403.

[171] E. Sprinzak, S. Sattath, and H. Margalit. “How reliable are experimental protein-protein interaction data?” In: *J. Mol. Biol.* 327.5 (2003), pp. 919–923.

[172] K. Luck et al. "Proteome-Scale Human Interactomics." In: *Trends Biochem. Sci.* 42.5 (May 2017), pp. 342–354.

[173] T. C. Ings et al. "Ecological networks–beyond food webs." In: *J Anim Ecol* 78.1 (2009), pp. 253–269.

[174] P. L. Kastritis and A. M. Bonvin. "On the binding affinity of macromolecular interactions: daring to ask why proteins interact." In: *J R Soc Interface* 10.79 (2013), p. 20120835.

[175] L. V. Bozhilova et al. "Measuring rank robustness in scored protein interaction networks." In: *BMC Bioinformatics* 20.1 (2019), p. 446.

[176] Q. C. Zhang et al. "Structure-based prediction of protein-protein interactions on a genome-wide scale." In: *Nature* 490.7421 (2012), pp. 556–560.

[177] Q. C. Zhang et al. "PrePPI: a structure-informed database of protein-protein interactions." In: *Nucleic Acids Res.* 41.Database issue (2013), pp. D828–833.

[178] R. Jansen et al. "A Bayesian networks approach for predicting protein-protein interactions from genomic data." In: *Science* 302.5644 (2003), pp. 449–453.

[179] Y. Ozawa et al. "Protein complex prediction via verifying and reconstructing the topology of domain-domain interactions." In: *BMC Bioinformatics* 11 (2010), p. 350.

[180] W. Ma et al. "Protein complex prediction based on maximum matching with domain-domain interaction." In: *Biochim. Biophys. Acta* 1824.12 (2012), pp. 1418–1424.

[181] D. Emig et al. "AltAnalyze and DomainGraph: analyzing and visualizing exon expression data." In: *Nucleic Acids Res.* 38.Web Server issue (2010), W755–762.

[182] M. Deng et al. "Inferring domain-domain interactions from protein-protein interactions." In: *Genome Res.* 12.10 (2002), pp. 1540–1548.

[183] R. Riley et al. "Inferring protein domain interactions from databases of interacting proteins." In: *Genome Biol.* 6.10 (2005), R89.

[184] K. S. Guimaraes et al. "Predicting domain-domain interactions using a parsimony approach." In: *Genome Biol.* 7.11 (2006), R104.

[185] H. Lee et al. "An integrated approach to the prediction of domain-domain interactions." In: *BMC Bioinformatics* 7 (2006), p. 269.

[186] M. Giurgiu et al. "CORUM: the comprehensive resource of mammalian protein complexes-2019." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D559–D563.

[187] S. Pu et al. "Up-to-date catalogues of yeast protein complexes." In: *Nucleic Acids Res.* 37.3 (2009), pp. 825–831.

[188] K. Drew et al. "Integration of over 9,000 mass spectrometry experiments builds a global map of human protein complexes." In: *Mol. Syst. Biol.* 13.6 (June 2017), p. 932.

[189] S. Srihari et al. "Methods for protein complex prediction and their contributions towards understanding the organisation, function and dynamics of complexes." In: *FEBS Lett.* 589.19 Pt A (2015), pp. 2590–2602.

[190] J. Zahiri et al. "Protein complex prediction: A survey." In: *Genomics* (2019).

[191] V. Spirin and L. A. Mirny. "Protein complexes and functional modules in molecular networks." In: *Proc. Natl. Acad. Sci. U.S.A.* 100.21 (2003), pp. 12123–12128.

[192] B. Zhang et al. "From pull-down data to protein interaction networks and complexes with biological relevance." In: *Bioinformatics* 24.7 (2008), pp. 979–986.

[193] A. J. Enright, S. Van Dongen, and C. A. Ouzounis. "An efficient algorithm for large-scale detection of protein families." In: *Nucleic Acids Res.* 30.7 (2002), pp. 1575–1584.

[194] G. D. Bader and C. W. Hogue. "An automated method for finding molecular complexes in large protein interaction networks." In: *BMC Bioinformatics* 4 (2003), p. 2.

[195] D. J. Watts and S. H. Strogatz. "Collective dynamics of 'small-world' networks." In: *Nature* 393.6684 (1998), pp. 440–442.

[196] T. Nepusz, H. Yu, and A. Paccanaro. "Detecting overlapping protein complexes in protein-protein interaction networks." In: *Nat. Methods* 9.5 (2012), pp. 471–472.

[197] A. D. King, N. Przulj, and I. Jurisica. "Protein complex prediction via cost-based clustering." In: *Bioinformatics* 20.17 (2004), pp. 3013–3020.

[198] X. L. Li, C. S. Foo, and S. K. Ng. "Discovering protein complexes in dense reliable neighborhoods of protein interaction networks." In: *Comput Syst Bioinformatics Conf* 6 (2007), pp. 157–168.

[199] L. Liang et al. "Integrating data and knowledge to identify functional modules of genes: a multilayer approach." In: *BMC Bioinformatics* 20.1 (2019), p. 225.

[200] S. Tornow and H. W. Mewes. "Functional modules by relating protein interaction networks and gene expression." In: *Nucleic Acids Res.* 31.21 (2003), pp. 6283–6289.

[201] U. de Lichtenberg et al. "Dynamic complex formation during the yeast cell cycle." In: *Science* 307.5710 (2005), pp. 724–727.

[202] M. T. Dittrich et al. "Identifying functional modules in protein-protein interaction networks: an integrated exact approach." In: *Bioinformatics* 24.13 (2008), pp. i223–231.

[203] S. Keretsu and R. Sarmah. "Weighted edge based clustering to identify protein complexes in protein-protein interaction networks incorporating gene expression profile." In: *Comput Biol Chem* 65 (2016), pp. 69–79.

[204] S. H. Jung et al. "Protein complex prediction based on simultaneous protein interaction network." In: *Bioinformatics* 26.3 (2010), pp. 385–391.

[205] R. Mosca, A. Ceol, and P. Aloy. "Interactome3D: adding structural details to protein networks." In: *Nat. Methods* 10.1 (2013), pp. 47–53.

[206] S. Rizzetto et al. "Qualitative and Quantitative Protein Complex Prediction Through Proteome-Wide Simulations." In: *PLoS Comput. Biol.* 11.10 (2015), e1004424.

[207] M. A. Ghadie et al. "Domain-based prediction of the human isoform interactome provides insights into the functional impact of alternative splicing." In: *PLoS Comput. Biol.* 13.8 (2017), e1005717.

[208] S. Rizzetto et al. "Context-dependent prediction of protein complexes by SiComPre." In: *NPJ Syst Biol Appl* 4 (2018), p. 37.

[209] I. Simon et al. "Serial regulation of transcriptional regulators in the yeast cell cycle." In: *Cell* 106.6 (2001), pp. 697–708.

[210] J. Göke et al. "Combinatorial binding in human and mouse embryonic stem cells identifies conserved enhancers active in early embryonic development." In: *PLoS Comput. Biol.* 7.12 (2011), e1002304.

[211] K. Hochedlinger and K. Plath. "Epigenetic reprogramming and induced pluripotency." In: *Development* 136.4 (2009), pp. 509–523.

[212] N. K. Wilson et al. "Combinatorial transcriptional control in blood stem/progenitor cells: genome-wide analysis of ten major transcriptional regulators." In: *Cell Stem Cell* 7.4 (2010), pp. 532–544.

[213] A. Z. Ansari and A. K. Mapp. "Modular design of artificial transcription factors." In: *Curr Opin Chem Biol* 6.6 (2002), pp. 765–772.

[214] S. A. Lambert et al. "The Human Transcription Factors." In: *Cell* 172.4 (Feb. 2018), pp. 650–665.

[215] D. Talavera, M. Orozco, and X. de la Cruz. "Alternative splicing of transcription factors' genes: beyond the increase of proteome diversity." In: *Comp. Funct. Genomics* 2009 (2009), p. 905894.

[216] W. W. Wasserman and A. Sandelin. "Applied bioinformatics for the identification of regulatory elements." In: *Nat. Rev. Genet.* 5.4 (2004), pp. 276–287.

[217] F. Schmidt et al. "Combining transcription factor binding affinities with open-chromatin data for accurate gene expression prediction." In: *Nucleic Acids Res.* 45.1 (Jan. 2017), pp. 54–66.

[218] D. S. Johnson et al. "Genome-wide mapping of in vivo protein-DNA interactions." In: *Science* 316.5830 (2007), pp. 1497–1502.

[219] T. L. Bailey. "DREME: motif discovery in transcription factor ChIP-seq data." In: *Bioinformatics* 27.12 (2011), pp. 1653–1659.

[220] I. V. Kulakovskiy et al. "HOCOMOCO: towards a complete collection of transcription factor binding models for human and mouse via large-scale ChIP-Seq analysis." In: *Nucleic Acids Res.* 46.D1 (2018), pp. D252–D259.

[221] O. Fornes et al. "JASPAR 2020: update of the open-access database of transcription factor binding profiles." In: *Nucleic Acids Res.* (2019).

[222] E. Morgunova and J. Taipale. "Structural perspective of cooperative transcription factor binding." In: *Curr. Opin. Struct. Biol.* 47 (Dec. 2017), pp. 1–8.

[223] Y. A. Medvedeva et al. "EpiFactors: a comprehensive database of human epigenetic factors and complexes." In: *Database (Oxford)* 2015 (2015), bav067.

[224] Larry Wasserman. *All of statistics: a concise course in statistical inference*. Springer, 2004. ISBN: 0387402721.

[225] R. A. Fisher. *Statistical methods for research workers*. Genesis Publishing Pvt Ltd, 2006.

[226] J. Neyman and E. S. Pearson. "On the problem of the most efficient tests of statistical hypotheses." In: *Philosophical Transactions of the Royal Society of London* 231.694-706 (1933), pp. 289–337.

[227] R. L. Wasserstein et al. "The ASA's statement on p-values: context, process, and purpose." In: *The American Statistician* 70.2 (2016), pp. 129–133.

[228] J. H. McDonald. *Handbook of biological statistics*. Vol. 3. Sparky House Publishing, 2014.

[229] F. Wilcoxon. "Individual Comparisons by Ranking Methods." In: *Biometrics Bulletin* 1.6 (1945), pp. 80–83.

[230] H. B. Mann and D. R. Whitney. "On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other." In: *The Annals of Mathematical Statistics* 18.1 (1947), pp. 50–60.

[231] Student. "The Probable Error of a Mean." In: *Biometrika* 6.1 (1908), pp. 1–25.

[232] B. L. Welch. "The generalization of 'Student's' problem when several different population variances are involved." In: *Biometrika* 34 (1947), pp. 28–35. ISSN: 0006-3444.

[233] A. Subramanian et al. "Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles." In: *Proc. Natl. Acad. Sci. U.S.A.* 102.43 (2005), pp. 15545–15550.

[234] C Bonferroni. "Teoria statistica delle classi e calcolo delle probabilita." In: *Pubblicazioni del R Istituto Superiore di Scienze Economiche e Commericiali di Firenze* 8 (1936), pp. 3–62.

[235] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning: data mining, inference and prediction*. 2nd edition. Springer, 2009.

[236] Y. Benjamini and Y. Hochberg. "Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing." In: *Journal of the Royal Statistical Society. Series B (Methodological)* 57.1 (1995), pp. 289–300. ISSN: 00359246.

[237] W. M. Gelbart et al. "The FlyBase database of the Drosophila Genome Projects and community literature." In: *Nucleic Acids Res.* 27.1 (1999), pp. 85–88.

[238] J. A. Blake et al. "The Mouse Genome Database (MGD): expanding genetic and genomic resources for the laboratory mouse." In: *Nucleic Acids Res.* 28.1 (2000), pp. 108–111.

[239] C. A. Ball et al. "Integrating functional genomic information into the Saccharomyces genome database." In: *Nucleic Acids Res.* 28.1 (2000), pp. 77–80.

[240] M. Ashburner et al. "Gene ontology: tool for the unification of biology." In: *Nat. Genet.* 25.1 (2000), pp. 25–29.

[241] D. Binns et al. "QuickGO: a web-based tool for Gene Ontology searching." In: *Bioinformatics* 25.22 (2009), pp. 3045–3046.

[242] Gene Ontology Consortium. "The Gene Ontology Resource: 20 years and still GOing strong." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D330–D338.

[243] C. Pesquita. "Semantic similarity in the gene ontology." In: *The Gene Ontology Handbook*. Humana Press, New York, NY, 2017, pp. 161–173.

[244] F. Schmidt et al. "An ontology-based method for assessing batch effect adjustment approaches in heterogeneous datasets." In: *Bioinformatics* 34.17 (2018), pp. i908–i916.

[245] W. A. Haynes, A. Tomczak, and P. Khatri. "Gene annotation bias impedes biomedical research." In: *Sci Rep* 8.1 (Jan. 2018), p. 1362.

[246] P. Gaudet and C. Dessimoz. "Gene Ontology: Pitfalls, Biases, and Remedies." In: *Methods Mol. Biol.* 1446 (2017), pp. 189–205.

[247] D. Hanahan and R. A. Weinberg. "The hallmarks of cancer." In: *Cell* 100.1 (2000), pp. 57–70.

[248] D. Hanahan and R. A. Weinberg. "Hallmarks of cancer: the next generation." In: *Cell* 144.5 (2011), pp. 646–674.

[249] R. A. Weinberg. "Coming full circle-from endless complexity to simplicity and back again." In: *Cell* 157.1 (2014), pp. 267–271.

[250] Y. A. Fouad and C. Aanei. "Revisiting the hallmarks of cancer." In: *Am J Cancer Res* 7.5 (2017), pp. 1016–1036.

[251] A. Suzuki et al. "Aberrant transcriptional regulations in cancers: genome, transcriptome and epigenome analysis of lung adenocarcinoma cell lines." In: *Nucleic Acids Res.* 42.22 (2014), pp. 13557–13572.

[252] S. Carbon et al. "AmiGO: online access to ontology and annotation data." In: *Bioinformatics* 25.2 (2009), pp. 288–289.

[253] M. Kanehisa and S. Goto. "KEGG: kyoto encyclopedia of genes and genomes." In: *Nucleic Acids Res.* 28.1 (2000), pp. 27–30.

[254] H. Ogata et al. "KEGG: Kyoto Encyclopedia of Genes and Genomes." In: *Nucleic Acids Res.* 27.1 (1999), pp. 29–34.

[255] M. Kanehisa et al. "New approach for understanding genome variations in KEGG." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D590–D595.

[256] G. Joshi-Tope et al. "Reactome: a knowledgebase of biological pathways." In: *Nucleic Acids Res.* 33.Database issue (2005), pp. D428–432.

[257] A. Fabregat et al. "The Reactome Pathway Knowledgebase." In: *Nucleic Acids Res.* 46.D1 (2018), pp. D649–D655.

[258] d. a. W. Huang, B. T. Sherman, and R. A. Lempicki. "Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists." In: *Nucleic Acids Res.* 37.1 (2009), pp. 1–13.

[259] A. Marco-Ramell et al. "Evaluation and comparison of bioinformatic tools for the enrichment analysis of metabolomics data." In: *BMC Bioinformatics* 19.1 (Jan. 2018), p. 1.

[260] d. a. W. Huang et al. "Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources." In: *Nat Protoc* 4.1 (2009), pp. 44–57.

[261] H. Mi et al. "PANTHER version 11: expanded annotation data from Gene Ontology and Reactome pathways, and data analysis tool enhancements." In: *Nucleic Acids Res.* 45.D1 (Jan. 2017), pp. D183–D189.

[262] D. Stockel et al. "Multi-omics enrichment analysis using the GeneTrail2 web service." In: *Bioinformatics* 32.10 (May 2016), pp. 1502–1508.

[263] S. Draghici et al. "Global functional profiling of gene expression." In: *Genomics* 81.2 (2003), pp. 98–104.

[264] I. Rivals et al. "Enrichment or depletion of a GO category within a class of genes: which test?" In: *Bioinformatics* 23.4 (2007), pp. 401–407.

[265] B. Efron, R. Tibshirani, et al. "On testing the significance of sets of genes." In: *The annals of applied statistics* 1.1 (2007), pp. 107–129.

[266] J. Han et al. "ESEA: Discovering the Dysregulated Pathways based on Edge Set Enrichment Analysis." In: *Sci Rep* 5 (2015), p. 13044.

[267] R. Fielding et al. *Hypertext Transfer Protocol – HTTP/1.1*. RFC 2616. Internet Engineering Task Force, 1999.

[268] J. Postel and J. Reynolds. *File Transfer Protocol*. RFC 0959. Internet Engineering Task Force, 1985.

[269] A. P. Felt et al. "Measuring HTTPS Adoption on the Web." In: *26th USENIX Security Symposium 2017)*. USENIX Association. 2017, pp. 1323–1338.

[270] E. Schechter. 2018. URL: https://security.googleblog.com/2018/02/a-secure-web-is-here-to-stay.html.

[271] B. Braschi et al. "Genenames.org: the HGNC and VGNC resources in 2019." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D786–D792.

[272] International Organization for Standardization. *ISO/IEC 9075 Database languages - SQL*.

[273] E. F. Codd. "A Relational Model of Data for Large Shared Data Banks." In: *Commun. ACM* 13.6 (1970), pp. 377–387. ISSN: 0001-0782.

[274] M. Haeussler et al. "The UCSC Genome Browser database: 2019 update." In: *Nucleic Acids Res.* 47.D1 (2019), pp. D853–D858.

[275] M. Vidal et al. "Interactome networks and human disease." In: *Cell* 144.6 (2011), pp. 986–998.

[276] J. Harrow et al. "GENCODE: producing a reference annotation for ENCODE." In: *Genome Biol.* 7 Suppl 1 (2006), pp. 1–9.

[277] D. Talavera et al. "Alternative splicing and protein interaction data sets." In: *Nat. Biotechnol.* 31.4 (2013), pp. 292–293.

[278] J. M. Rodriguez et al. "APPRIS: annotation of principal and alternative splice isoforms." In: *Nucleic Acids Res.* 41.Database issue (2013), pp. D110–117.

[279] Q. Zhong et al. "Edgetic perturbation models of human inherited disorders." In: *Mol. Syst. Biol.* 5 (2009), p. 321.

[280] N. Sahni et al. "Edgotype: a fundamental link between genotype and phenotype." In: *Curr. Opin. Genet. Dev.* 23.6 (2013), pp. 649–657.

[281] M. Gonzalez-Porta et al. "Transcriptome analysis of human tissues and cell lines reveals one dominant transcript per gene." In: *Genome Biol.* 14.7 (2013), R70.

[282] I. Ezkurdia et al. "Most highly expressed protein-coding genes have a single dominant isoform." In: *J. Proteome Res.* 14.4 (2015), pp. 1880–1887.

[283] M. Mele et al. "The human transcriptome across tissues and individuals." In: *Science* 348.6235 (2015), pp. 660–665.

[284] D. Smedley et al. "The BioMart community portal: an innovative alternative to large, centralized data repositories." In: *Nucleic Acids Res.* (2015).

[285] A. Bateman, A., et al. "UniProt: a hub for protein information." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D204–212.

[286] F. Cunningham et al. "Ensembl 2015." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D662–669.

[287] R. D. Finn et al. "Pfam: the protein families database." In: *Nucleic Acids Res.* 42.Database issue (2014), pp. D222–230.

[288] P. Jones et al. "InterProScan 5: genome-scale protein function classification." In: *Bioinformatics* 30.9 (2014), pp. 1236–1240.

[289] A. Chatr-Aryamontri et al. "The BioGRID interaction database: 2015 update." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D470–478.

[290] K. A. Gray et al. "Genenames.org: the HGNC resources in 2015." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D1079–1085.

[291] D. C. Koboldt et al. "Comprehensive molecular portraits of human breast tumours." In: *Nature* 490.7418 (2012), pp. 61–70.

[292] D. Diez et al. "Systematic identification of transcriptional regulatory modules from protein-protein interaction networks." In: *Nucleic Acids Res.* 42.1 (2014), e6.

[293] M. Kanehisa et al. "Data, information, knowledge and principle: back to metabolism in KEGG." In: *Nucleic Acids Res.* 42.Database issue (2014), pp. 199–205.

[294] J. A. Blake et al. "Gene Ontology Consortium: going forward." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D1049–1056.

[295] A. Goncearenco et al. "Coverage of protein domain families with structural protein-protein interactions: current progress and future trends." In: *Prog. Biophys. Mol. Biol.* 116.2-3 (2014), pp. 187–193.

[296] D. Esch et al. "A unique Oct4 interface is crucial for reprogramming to pluripotency." In: *Nat. Cell Biol.* 15.3 (2013), pp. 295–301.

[297] I. Ezkurdia et al. "Comparative proteomics reveals a significant bias toward alternative protein isoforms with conserved structure and function." In: *Mol. Biol. Evol.* 29.9 (2012), pp. 2265–2283.

[298] S. Razick, G. Magklaras, and I. M. Donaldson. "iRefIndex: a consolidated protein interaction database with provenance." In: *BMC Bioinformatics* 9 (2008), p. 405.

[299] G. Wedler. *Lehrbuch der Physikalischen Chemie*. 5th. Wiley, 2004. ISBN: 9783527310661.

[300] N. E. Davey et al. "Attributes of short linear motifs." In: *Mol Biosyst* 8.1 (2012), pp. 268–281.

[301] K. Van Roey et al. "Short linear motifs: ubiquitous and functionally diverse protein interaction modules directing cell regulation." In: *Chem. Rev.* 114.13 (2014), pp. 6733–6778.

[302] F. Diella et al. "Understanding eukaryotic linear motifs and their role in cell signaling and regulation." In: *Front. Biosci.* 13 (2008), pp. 6580–6603.

[303] H. Dinkel et al. "ELM 2016–data update and new functionality of the eukaryotic linear motif resource." In: *Nucleic Acids Res.* 44.D1 (2016), pp. 294–300.

[304] D. Sarkar, T. Jana, and S. Saha. "LMPID: a manually curated database of linear motifs mediating protein-protein interactions." In: *Database (Oxford)* 2015 (2015).

[305] J. C. Obenauer, L. C. Cantley, and M. B. Yaffe. "Scansite 2.0: Proteome-wide prediction of cell signaling interactions using short sequence motifs." In: *Nucleic Acids Res.* 31.13 (2003), pp. 3635–3641.

[306] K. F. Lyon et al. "Minimotif Miner 4: a million peptide minimotifs and counting." In: *Nucleic Acids Res.* 46.D1 (2018), pp. D465–D470.

[307] T. S. Chen et al. "Predicting peptide-mediated interactions on a genome-wide scale." In: *PLoS Comput. Biol.* 11.5 (2015), e1004248.

[308] E. Petsalaki and R. B. Russell. "Peptide-mediated interactions in biological systems: new discoveries and applications." In: *Curr. Opin. Biotechnol.* 19.4 (2008), pp. 344–350.

[309] P. Kumar, S. Henikoff, and P. C. Ng. "Predicting the effects of coding non-synonymous variants on protein function using the SIFT algorithm." In: *Nat Protoc* 4.7 (2009), pp. 1073–1081.

[310] I. A. Adzhubei et al. "A method and server for predicting damaging missense mutations." In: *Nat. Methods* 7.4 (2010), pp. 248–249.

[311] R. Mosca et al. "dSysMap: exploring the edgetic role of disease mutations." In: *Nat. Methods* 12.3 (2015), pp. 167–168.

[312] A. Gress et al. "StructMAn: annotation of single-nucleotide polymorphisms in the structural context." In: *Nucleic Acids Res.* 44.W1 (2016), W463–468.

[313] S. Henikoff and J. G. Henikoff. "Amino acid substitution matrices from protein blocks." In: *Proc. Natl. Acad. Sci. U.S.A.* 89.22 (1992), pp. 10915–10919.

[314] H. E. Driver and A. L. Kroeber. "Quantitative Expression of Cultural Relationships." In: *University of California Publications in American Archaeology and Ethnology* Quantitative Expression of Cultural Relationships (1932), pp. 211–256.

[315] R. C. Tryon. *Cluster Analysis: Correlation Profile and Orthometric (factor) Analysis for the Isolation of Unities in Mind and Personality*. Edwards brother, Incorporated, lithoprinters and publishers, 1939.

[316] R. B. Cattell. "The description of personality: Basic traits resolved into clusters." In: *The journal of abnormal and social psychology* 38.4 (1943), p. 476.

[317] V. Estivill-Castro. "Why So Many Clustering Algorithms: A Position Paper." In: *SIGKDD Explor. Newsl.* 4.1 (2002), pp. 65–75.

[318] S. P. Lloyd. "Least squares quantization in pcm." In: *IEEE Transactions on Information Theory* 28 (1982), pp. 129–137.

[319] E. Forgy. "Cluster analysis of multivariate data: efficiency versus interpretability of classifications." In: *Biometrics* 21 (1965), pp. 768–780.

[320] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. 2nd edition. Wiley-Interscience, 2000. ISBN: 0471056693.

[321] M. Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." In: *Kdd*. Vol. 96. 34. 1996, pp. 226–231.

[322] A. Y. Ng, M. I. Jordan, and Y. Weiss. "On spectral clustering: Analysis and an algorithm." In: *Advances in neural information processing systems*. 2002, pp. 849–856.

[323] A. K. Jain, M. N. Murty, and P. J. Flynn. "Data Clustering: A Review." In: *ACM Comput. Surv.* 31.3 (1999), pp. 264–323. ISSN: 0360-0300.

[324] J. H. Ward Jr. "Hierarchical grouping to optimize an objective function." In: *Journal of the American statistical association* 58.301 (1963), pp. 236–244.

[325] P. H. A. Sneath, R. R. Sokal, et al. *Numerical taxonomy. The principles and practice of numerical classification*. W. H. Freeman, 1973.

[326] B. King. "Step-wise clustering procedures." In: *Journal of the American Statistical Association* 62.317 (1967), pp. 86–101.

[327] R. R. Sokal. "A statistical method for evaluating systematic relationship." In: *University of Kansas science bulletin* 28 (1958), pp. 1409–1438.

[328] T. H. Cormen et al. *Introduction to Algorithms*. 3rd. The MIT Press, 2009. ISBN: 9780262533058.

[329] R. M. Karp. "Reducibility among combinatorial problems." In: *Complexity of computer computations*. Springer, 1972, pp. 85–103.

[330] N. E. Young. "Greedy set-cover algorithms." In: *Encyclopedia of algorithms*. Springer, 2008, pp. 379–381.

[331] V. Chvatal. "A greedy heuristic for the set-covering problem." In: *Mathematics of Operations Research* 4.3 (1979), pp. 233–235.

[332] E. Yeger-Lotem and R. Sharan. "Human protein interaction networks across tissues and diseases." In: *Front Genet* 6 (2015), p. 257.

[333] A. Grossmann et al. "Phospho-tyrosine dependent protein-protein interaction network." In: *Mol. Syst. Biol.* 11.3 (2015), p. 794.

[334] J. Song, Z. Wang, and R. M. Ewing. "Integrated analysis of the Wnt responsive proteome in human cells reveals diverse and cell-type specific networks." In: *Mol Biosyst* 10.1 (2014), pp. 45–53.

[335] K. Lage et al. "A large-scale analysis of tissue-specific pathology and gene expression of human disease genes and complexes." In: *Proc. Natl. Acad. Sci. U.S.A.* 105.52 (2008), pp. 20870–20875.

[336] T. Ideker and N. J. Krogan. "Differential network biology." In: *Mol. Syst. Biol.* 8 (2012), p. 565.

[337] A. de la Fuente. "From 'differential expression' to 'differential networking' - identification of dysfunctional regulatory networks in diseases." In: *Trends Genet.* 26.7 (2010), pp. 326–333.

[338] J. Ji et al. "A powerful score-based statistical test for group difference in weighted biological networks." In: *BMC Bioinformatics* 17 (2016), p. 86.

[339] A. Reverter et al. "Simultaneous identification of differential gene expression and connectivity in inflammation, adipogenesis and cancer." In: *Bioinformatics* 22.19 (2006), pp. 2396–2404.

[340] R. Gill, S. Datta, and S. Datta. "A statistical framework for differential network analysis from microarray data." In: *BMC Bioinformatics* 11 (2010), p. 95.

[341] D. Ruan, A. Young, and G. Montana. "Differential analysis of biological networks." In: *BMC Bioinformatics* 16 (2015), p. 327.

[342] S. V. Landeghem et al. "Diffany: an ontology-driven framework to infer, visualise and analyse differential molecular networks." In: *BMC Bioinformatics* 17.1 (2016), p. 18.

[343] S. H. Orkin and L. I. Zon. "Hematopoiesis: an evolving paradigm for stem cell biology." In: *Cell* 132.4 (2008), pp. 631–644.

[344] J. H. Martens and H. G. Stunnenberg. "BLUEPRINT: mapping human blood cell epigenomes." In: *Haematologica* 98.10 (2013), pp. 1487–1489.

[345] *BLUEPRINT Epigenome Project 7th Data Release*. 2015. URL: {http://dx.doi.org/10.6019/blueprint_20150910}.

[346] P. Shannon et al. "Cytoscape: a software environment for integrated models of biomolecular interaction networks." In: *Genome Res.* 13.11 (2003), pp. 2498–2504.

[347] C. A. Gallo et al. "Discretization of gene expression data revised." In: *Brief. Bioinformatics* 17.5 (2016), pp. 758–770.

[348] M. Levandowsky and D. Winter. "Distance between sets." In: *Nature* 234.5323 (1971), pp. 34–35.

[349] D. Lara-Astiaso et al. "Chromatin state dynamics during blood formation." In: *Science* 345.6199 (2014), pp. 943–949.

[350] C. Bock et al. "DNA methylation dynamics during in vivo differentiation of blood and skin stem cells." In: *Mol. Cell* 47.4 (2012), pp. 633–647.

[351] N. Novershtern et al. "Densely interconnected transcriptional circuits control cell states in human hematopoiesis." In: *Cell* 144.2 (2011), pp. 296–309.

[352] S. Doulatov et al. "Hematopoiesis: a human perspective." In: *Cell Stem Cell* 10.2 (2012), pp. 120–136.

[353] R. Yamamoto et al. "Clonal analysis unveils self-renewing lineage-restricted progenitors generated directly from hematopoietic stem cells." In: *Cell* 154.5 (2013), pp. 1112–1126.

[354] L. Perie et al. "The Branching Point in Erythro-Myeloid Differentiation." In: *Cell* 163.7 (2015), pp. 1655–1662.

[355] R. A. Nimmo, G. E. May, and T. Enver. "Primed and ready: understanding lineage commitment through single cell analysis." In: *Trends Cell Biol.* 25.8 (2015), pp. 459–467.

[356] J. Adolfsson et al. "Identification of Flt3+ lympho-myeloid stem cells lacking erythro-megakaryocytic potential a revised road map for adult blood lineage commitment." In: *Cell* 121.2 (2005), pp. 295–306.

[357] F. Notta et al. "Distinct routes of lineage development reshape the human blood hierarchy across ontogeny." In: *Science* 351.6269 (2016), aab2116.

[358] A. Conesa et al. "A survey of best practices for RNA-seq data analysis." In: *Genome Biol.* 17.1 (2016), p. 13.

[359] B. Li et al. "RNA-Seq gene expression estimation with read mapping uncertainty." In: *Bioinformatics* 26.4 (2010), pp. 493–500.

[360] A. Yates et al. “Ensembl 2016.” In: *Nucleic Acids Res.* 44.D1 (2016), pp. D710–716.

[361] A. Ruepp et al. “CORUM: the comprehensive resource of mammalian protein complexes–2009.” In: *Nucleic Acids Res.* 38.Database issue (2010), pp. 497–501.

[362] G. Yu et al. “GOSemSim: an R package for measuring semantic similarity among GO terms and gene products.” In: *Bioinformatics* 26.7 (2010), pp. 976–978.

[363] M. Carlson. *org.Hs.eg.db: Genome wide annotation for Human*. R package version 3.2.3.

[364] A. A. Hagberg, D. A. Schult, and P. J. Swart. “Exploring network structure, dynamics, and function using NetworkX.” In: *Proceedings of the 7th Python in Science Conference (SciPy2008)*. Pasadena, CA USA, 2008, pp. 11–15.

[365] M. Kanehisa et al. “KEGG as a reference resource for gene and protein annotation.” In: *Nucleic Acids Res.* 44.D1 (2016), pp. D457–462.

[366] A. Fabregat et al. “The Reactome pathway Knowledgebase.” In: *Nucleic Acids Res.* 44.D1 (2016), pp. D481–487.

[367] P. Khatri and S. Draghici. “Ontological analysis of gene expression data: current tools, limitations, and open problems.” In: *Bioinformatics* 21.18 (2005), pp. 3587–3595.

[368] H. Han et al. “TRRUST: a reference database of human transcriptional regulatory interactions.” In: *Sci Rep* 5 (2015), p. 11432.

[369] C. Trapnell. “Defining cell types and states with single-cell genomics.” In: *Genome Res.* 25.10 (2015), pp. 1491–1498.

[370] M. Etzrodt, M. Endele, and T. Schroeder. “Quantitative single-cell approaches to stem cell research.” In: *Cell Stem Cell* 15.5 (2014), pp. 546–558.

[371] F. Paul et al. “Transcriptional Heterogeneity and Lineage Commitment in Myeloid Progenitors.” In: *Cell* 163.7 (2015), pp. 1663–1677.

[372] J. Zhu, H. Yamane, and W. E. Paul. “Differentiation of effector CD4 T cell populations (*).” In: *Annu. Rev. Immunol.* 28 (2010), pp. 445–489.

[373] T. Hong et al. “A simple theoretical framework for understanding heterogeneous differentiation of CD4+ T cells.” In: *BMC Syst Biol* 6 (2012), p. 66.

[374] S. Yona and S. Jung. “Monocytes: subsets, origins, fates and functions.” In: *Curr. Opin. Hematol.* 17.1 (2010), pp. 53–59.

[375] N. Cabezas-Wallscheid et al. “Identification of regulatory networks in HSCs and their immediate progeny via integrated proteome, transcriptome, and DNA methylome analysis.” In: *Cell Stem Cell* 15.4 (2014), pp. 507–522.

[376] J. Z. Ni et al. "Ultraconserved elements are associated with homeostatic control of splicing regulators by alternative splicing and nonsense-mediated decay." In: *Genes Dev.* 21.6 (2007), pp. 708–718.

[377] A. L. Saltzman et al. "Regulation of multiple core spliceosomal proteins by alternative splicing-coupled nonsense-mediated mRNA decay." In: *Mol. Cell. Biol.* 28.13 (2008), pp. 4320–4330.

[378] H. B. Fraser. "Modularity and evolutionary constraint on proteins." In: *Nat. Genet.* 37.4 (2005), pp. 351–352.

[379] X. Chang et al. "Dynamic modular architecture of protein-protein interaction networks beyond the dichotomy of 'date' and 'party' hubs." In: *Sci Rep* 3 (2013), p. 1691.

[380] T. Narayanan et al. "Modularity detection in protein-protein interaction networks." In: *BMC Res Notes* 4 (2011), p. 569.

[381] R. Dunn, F. Dudbridge, and C. M. Sanderson. "The use of edge-betweenness clustering to investigate biological function in protein interaction networks." In: *BMC Bioinformatics* 6 (2005), p. 39.

[382] M. Shi et al. "Cell cycle progression following naive T cell activation is independent of Jak3/common gamma-chain cytokine signals." In: *J. Immunol.* 183.7 (2009), pp. 4493–4501.

[383] K. Theilgaard-Monch et al. "The transcriptional program of terminal granulocytic differentiation." In: *Blood* 105.4 (2005), pp. 1785–1796.

[384] R. van Furth, J. A. Raeburn, and T. L. van Zwet. "Characteristics of human mononuclear phagocytes." In: *Blood* 54.2 (1979), pp. 485–500.

[385] G. Fossati et al. "The mitochondrial network of human neutrophils: role in chemotaxis, phagocytosis, respiratory burst activation, and commitment to apoptosis." In: *J. Immunol.* 170.4 (2003), pp. 1964–1972.

[386] P. A. Kramer et al. "A review of the mitochondrial and glycolytic metabolism in human platelets and leukocytes: implications for their use as bioenergetic biomarkers." In: *Redox Biol* 2 (2014), pp. 206–210.

[387] D. Graczyk, R. J. White, and K. M. Ryan. "Involvement of RNA Polymerase III in Immune Responses." In: *Mol. Cell. Biol.* 35.10 (2015), pp. 1848–1859.

[388] Y. H. Chiu, J. B. Macmillan, and Z. J. Chen. "RNA polymerase III detects cytosolic DNA and induces type I interferons through the RIG-I pathway." In: *Cell* 138.3 (2009), pp. 576–591.

[389] N. Tamassia et al. "IFN-Beta expression is directly activated in human neutrophils transfected with plasmid DNA and is further increased via TLR-4-mediated signaling." In: *J. Immunol.* 189.3 (2012), pp. 1500–1509.

[390] I. W. Taylor et al. "Dynamic modularity in protein interaction networks predicts breast cancer outcome." In: *Nat. Biotechnol.* 27.2 (2009), pp. 199–204.

[391] J. D. Pinon et al. "Bim and Bmf in tissue homeostasis and malignant disease." In: *Oncogene* 27 Suppl 1 (2008), pp. 41–52.

[392] G. A. Blobel. "CREB-binding protein and p300: molecular integrators of hematopoietic transcription." In: *Blood* 95.3 (2000), pp. 745–755.

[393] E. Shaulian and M. Karin. "AP-1 as a regulator of cell life and death." In: *Nat. Cell Biol.* 4.5 (2002), E131–136.

[394] L. Steinmuller et al. "Regulation and composition of activator protein 1 (AP-1) transcription factors controlling collagenase and c-Jun promoter activities." In: *Biochem. J.* 360.Pt 3 (2001), pp. 599–607.

[395] A. Wilson et al. "c-Myc controls the balance between hematopoietic stem cell self-renewal and differentiation." In: *Genes Dev.* 18.22 (2004), pp. 2747–2763.

[396] J. Skokowa et al. "LEF-1 is crucial for neutrophil granulocytopoiesis and its expression is severely reduced in congenital neutropenia." In: *Nat. Med.* 12.10 (2006), pp. 1191–1197.

[397] B. T. MacDonald, K. Tamai, and X. He. "Wnt/beta-catenin signaling: components, mechanisms, and diseases." In: *Dev. Cell* 17.1 (2009), pp. 9–26.

[398] G. Genovese et al. "The tumor suppressor HINT1 regulates MITF and beta-catenin transcriptional activity in melanoma cells." In: *Cell Cycle* 11.11 (2012), pp. 2206–2215.

[399] A. Bauer, O. Huber, and R. Kemler. "Pontin52, an interaction partner of beta-catenin, binds to the TATA box binding protein." In: *Proc. Natl. Acad. Sci. U.S.A.* 95.25 (1998), pp. 14787–14792.

[400] C. Soza-Ried et al. "Essential role of c-myb in definitive hematopoiesis is evolutionarily conserved." In: *Proc. Natl. Acad. Sci. U.S.A.* 107.40 (2010), pp. 17304–17308.

[401] C. Orelio and E. Dzierzak. "Bcl-2 expression and apoptosis in the regulation of hematopoietic stem cells." In: *Leuk. Lymphoma* 48.1 (2007), pp. 16–24.

[402] J. Koipally and K. Georgopoulos. "A molecular dissection of the repression circuitry of Ikaros." In: *J. Biol. Chem.* 277.31 (2002), pp. 27697–27705.

[403] D. E. Scott et al. "Small molecules, big targets: drug discovery faces the protein-protein interaction challenge." In: *Nat Rev Drug Discov* (2016).

[404] X. Ran and J. E. Gestwicki. "Inhibitors of protein-protein interactions (PPIs): an analysis of scaffold choices and buried surface area." In: *Curr Opin Chem Biol* 44 (June 2018), pp. 75–86.

[405] C. R. Williams et al. "Empirical assessment of analysis workflows for differential expression analysis of human samples using RNA-Seq." In: *BMC Bioinformatics* 18.1 (2017), p. 38.

[406] W. Hawkins and F. Ryder. "Shared and unique mechanisms of macrophage-like neutrophils in EAE." Université Laval, 2019.

[407] G. E. Moore. "Cramming more components onto integrated circuits." In: *Electronics* 38.8 (1965), pp. 114–117.

[408] D. Geer. “Chip makers turn to multicore processors.” In: *Computer* 38.5 (2005), pp. 11–13.

[409] M. M. Waldrop. “The chips are down for Moore’s law.” In: *Nature* 530.7589 (2016), pp. 144–147.

[410] T. Wouters. 2017. URL: https://wiki.python.org/moin/GlobalInterpreterLock.

[411] T. Lappalainen et al. “Transcriptome and genome sequencing uncovers functional variation in humans.” In: *Nature* 501.7468 (2013), pp. 506–511.

[412] D. T. Chang et al. “YPA: an integrated repository of promoter features in Saccharomyces cerevisiae.” In: *Nucleic Acids Res.* 39.Database issue (2011), pp. D647–652.

[413] S. Ashkiani et al. “GPU LSM: A dynamic dictionary data structure for the GPU.” In: *2018 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. IEEE. 2018, pp. 430–440.

[414] K. Kaczmarski and A. Wolant. “GPU R-Trie: Dictionary with ultra fast lookup.” In: *Concurrency and Computation: Practice and Experience* (2019), e5027.

[415] M. C. Wahl, C. L. Will, and R. Luhrmann. “The spliceosome: design principles of a dynamic RNP machine.” In: *Cell* 136.4 (2009), pp. 701–718.

[416] A. C. Paoletti et al. “Quantitative proteomic analysis of distinct mammalian Mediator complexes using normalized spectral abundance factors.” In: *Proc. Natl. Acad. Sci. U.S.A.* 103.50 (2006), pp. 18928–18933.

[417] T. F. Lou et al. “Integrated analysis of RNA-binding protein complexes using in vitro selection and high-throughput sequencing and sequence specificity landscapes (SEQRS).” In: *Methods* 118-119 (Apr. 2017), pp. 171–181.

[418] G. B. Dantzig. “Reminiscences about the origins of linear programming.” In: *Operations Research Letters* 1.2 (1982), pp. 43 –48. ISSN: 0167-6377.

[419] G. B. Dantzig. *Origins of the simplex method*. Tech. rep. STANFORD UNIV CA SYSTEMS OPTIMIZATION LAB, 1987.

[420] M. Berkelaar et al. “lpsolve: Open source (mixed-integer) linear programming system.” In: *Eindhoven U. of Technology* 63 (2004).

[421] F. A. Potra and S. J. Wright. “Interior-point methods.” In: *Journal of Computational and Applied Mathematics* 124.1-2 (2000), pp. 281–302.

[422] R. A. Fisher. “The use of multiple measurements in taxonomic problems.” In: *Annals of Eugenics* 7.2 (1936), pp. 179–188.

[423] M. Sokolova and G. Lapalme. “A systematic analysis of performance measures for classification tasks.” In: *Information Processing & Management* 45.4 (2009), pp. 427–437.

[424] T. K. Ho. “Random decision forests.” In: *Proceedings of 3rd international conference on document analysis and recognition*. Vol. 1. IEEE. 1995, pp. 278–282.

[425] L. Breiman. “Random forests.” In: *Machine learning* 45.1 (2001), pp. 5–32.

[426] F. Pedregosa et al. "Scikit-learn: Machine Learning in Python." In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[427] L. Breiman et al. "CART: Classification and Regression Trees." In: *Wadsworth, Belmont, CA* (1984).

[428] P. Geurts et al. "Extremely randomized trees." In: *Machine Learning* 63.1 (2006), pp. 3–42.

[429] T. M. Oshiro et al. "How Many Trees in a Random Forest?" In: *Proceedings of the 8th International Conference on Machine Learning and Data Mining in Pattern Recognition*. MLDM'12. Berlin, Germany: Springer-Verlag, 2012, pp. 154–168.

[430] A. C. Gavin et al. "Proteome survey reveals modularity of the yeast cell machinery." In: *Nature* 440.7084 (2006), pp. 631–636.

[431] A. C. Gingras et al. "Analysis of protein complexes using mass spectrometry." In: *Nat. Rev. Mol. Cell Biol.* 8.8 (2007), pp. 645–654.

[432] E. L. Rudashevskaya et al. "Global profiling of protein complexes: current approaches and their perspective in biomedical research." In: *Expert Rev Proteomics* 13 (10 2016), pp. 951–964.

[433] A. H. Smits et al. "Stoichiometry of chromatin-associated protein complexes revealed by label-free quantitative mass spectrometry-based proteomics." In: *Nucleic Acids Res.* 41.1 (2013), e28.

[434] R. van Nuland et al. "Quantitative dissection and stoichiometry determination of the human SET1/MLL histone methyltransferase complexes." In: *Mol. Cell. Biol.* 33.10 (2013), pp. 2067–2077.

[435] A. Celaj et al. "Quantitative analysis of protein interaction network dynamics in yeast." In: *Mol. Syst. Biol.* 13.7 (2017), p. 934.

[436] E. K. Papachristou et al. "A quantitative mass spectrometry-based approach to monitor the dynamics of endogenous chromatin-associated protein complexes." In: *Nat Commun* 9.1 (2018), p. 2311.

[437] S. Srihari et al. "Complex-based analysis of dysregulated cellular processes in cancer." In: *BMC Syst Biol* 8 Suppl 4 (2014), S1.

[438] B. E. Barker et al. "A robust and efficient method for estimating enzyme complex abundance and metabolic flux from expression data." In: *Comput Biol Chem* 59 Pt B (2015), pp. 98–112.

[439] S. H. Lee et al. "Global organization of protein complexome in the yeast Saccharomyces cerevisiae." In: *BMC Syst Biol* 5 (2011), p. 126.

[440] X. Li et al. "Computational approaches for detecting protein complexes from protein interaction networks: a survey." In: *BMC Genomics* 11 Suppl 1 (2010), S3.

[441] UniProt Consortium. "UniProt: the universal protein knowledgebase." In: *Nucleic Acids Res.* 45.D1 (2017), pp. D158–D169.

[442] I. V. Kulakovskiy et al. "HOCOMOCO: expansion and enhancement of the collection of transcription factor binding sites models." In: *Nucleic Acids Res.* 44.D1 (2016), pp. D116–125.

[443] D. Fletcher et al. "Modelling skewed data with many zeros: a simple approach combining ordinary and logistic regression." In: *Environmental and ecological statistics* 12.1 (2005), pp. 45–54.

[444] A. Gleiss et al. "Two-group comparisons of zero-inflated intensity values: the choice of test statistic matters." In: *Bioinformatics* 31.14 (2015), pp. 2310–2317.

[445] Z. H. Zhang et al. "A comparative study of techniques for differential expression analysis on RNA-Seq data." In: *PLoS ONE* 9.8 (2014), e103207.

[446] S. Durinck et al. "Mapping identifiers for the integration of genomic datasets with the R/Bioconductor package biomaRt." In: *Nat Protoc* 4.8 (2009), pp. 1184–1191.

[447] L. M. Carlin et al. "Nr4a1-dependent Ly6C(low) monocytes monitor endothelial cells and orchestrate their disposal." In: *Cell* 153.2 (2013), pp. 362–375.

[448] R. N. Hanna et al. "The transcription factor NR4A1 (Nur77) controls bone marrow differentiation and the survival of Ly6C- monocytes." In: *Nat. Immunol.* 12.8 (2011), pp. 778–785.

[449] J. Cros et al. "Human CD14dim monocytes patrol and sense nucleic acids and viruses via TLR7 and TLR8 receptors." In: *Immunity* 33.3 (2010), pp. 375–386.

[450] R. N. Hanna et al. "NR4A1 (Nur77) deletion polarizes macrophages toward an inflammatory phenotype and increases atherosclerosis." In: *Circ. Res.* 110.3 (2012), pp. 416–427.

[451] K. L. Wong et al. "Gene expression profiling reveals the defining features of the classical, intermediate, and nonclassical human monocyte subsets." In: *Blood* 118.5 (2011), pp. 16–31.

[452] K. L. Wong et al. "The three human monocyte subsets: implications for health and disease." In: *Immunol. Res.* 53.1-3 (2012), pp. 41–57.

[453] E. Idzkowska et al. "The Role of Different Monocyte Subsets in the Pathogenesis of Atherosclerosis and Acute Coronary Syndromes." In: *Scand. J. Immunol.* 82.3 (2015), pp. 163–173.

[454] L. Breiman. "Random Forests." In: *Machine Learning* 45.1 (2001), pp. 5–32.

[455] R. Kohavi. "A Study of Cross-validation and Bootstrap for Accuracy Estimation and Model Selection." In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2*. IJCAI'95. Montreal, Quebec, Canada: Morgan Kaufmann Publishers Inc., 1995, pp. 1137–1143. ISBN: 1-55860-363-8.

[456] K. A. Lee. "Dimeric transcription factor families: it takes two to tango but who decides on partners and the venue?" In: *J. Cell. Sci.* 103 (Pt 1) (1992), pp. 9–14.

[457] B. H. Meldal et al. "The complex portal–an encyclopaedia of macromolecular complexes." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D479–484.

[458] A. Yilmaz and N. Benvenisty. "Defining Human Pluripotency." In: *Cell Stem Cell* 25.1 (2019), pp. 9–22.

[459] I. V. Kulakovskiy et al. "HOCOMOCO: a comprehensive collection of human transcription factor binding sites models." In: *Nucleic Acids Res.* 41.Database issue (2013), pp. 195–202.

[460] R. Dreos et al. "The Eukaryotic Promoter Database: expansion of EPDnew and new promoter analysis tools." In: *Nucleic Acids Res.* 43.Database issue (2015), pp. D92–96.

[461] C. E. Grant, T. L. Bailey, and W. S. Noble. "FIMO: scanning for occurrences of a given motif." In: *Bioinformatics* 27.7 (2011), pp. 1017–1018.

[462] U. Alon. "Network motifs: theory and experimental approaches." In: *Nat. Rev. Genet.* 8.6 (2007), pp. 450–461.

[463] C. A. Davis et al. "The Encyclopedia of DNA elements (ENCODE): data portal update." In: *Nucleic Acids Res.* 46.D1 (Jan. 2018), pp. D794–D801.

[464] A. Kundaje et al. "Integrative analysis of 111 reference human epigenomes." In: *Nature* 518.7539 (2015), pp. 317–330.

[465] R. Tarjan. "Depth-First Search and Linear Graph Algorithms." In: *SIAM Journal on Computing* 1.2 (1972), pp. 146–160.

[466] M. P. Creyghton et al. "Histone H3K27ac separates active from poised enhancers and predicts developmental state." In: *Proc. Natl. Acad. Sci. U.S.A.* 107.50 (2010), pp. 21931–21936.

[467] R. Raisner et al. "Enhancer Activity Requires CBP/P300 Bromodomain-Dependent Histone H3K27 Acetylation." In: *Cell Rep* 24.7 (Aug. 2018), pp. 1722–1729.

[468] S. Fishilevich et al. "GeneHancer: genome-wide integration of enhancers and target genes in GeneCards." In: *Database (Oxford)* 2017 (Jan. 2017).

[469] S. Fu et al. "Differential analysis of chromatin accessibility and histone modifications for predicting mouse developmental enhancers." In: *Nucleic Acids Res.* 46.21 (Nov. 2018), pp. 11184–11201.

[470] A. L. Todeschini, A. Georges, and R. A. Veitia. "Transcription factors: specific DNA binding and specific gene regulation." In: *Trends Genet.* 30.6 (2014), pp. 211–219.

[471] A. Soufi et al. "Pioneer transcription factors target partial DNA motifs on nucleosomes to initiate reprogramming." In: *Cell* 161.3 (2015), pp. 555–568.

[472] B. L. Lampson and M. S. Davids. "The Development and Current Use of BCL-2 Inhibitors for the Treatment of Chronic Lymphocytic Leukemia." In: *Curr Hematol Malig Rep* 12.1 (Feb. 2017), pp. 11–19.

[473] I. C. Macaulay, C. P. Ponting, and T. Voet. "Single-Cell Multiomics: Multiple Measurements from Single Cells." In: *Trends Genet.* 33.2 (Feb. 2017), pp. 155–168.

[474] K. Rooijers et al. “Simultaneous quantification of protein-DNA contacts and transcriptomes in single cells.” In: *Nat. Biotechnol.* 37.7 (July 2019), pp. 766–772.