Vom Fachbereich Biologie der Universität Kaiserslautern

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften genehmigte Dissertation

Signature Topology: functional analysis of omics data

von

Dipl.-Biol. Nathan Mikhaylenko

Wissenschaftliche Aussprache: Kaiserslautern, 26.08.2022

Prüfungskommission:
1. Gutachter: Prof. Dr. Timo Mühlhaus
2. Gutachter: Prof. Dr. Stefan Kins
Vorsitzender: Prof. Dr. Thorsten Stoeck

Computational Systems Biology

Technical University of Kaiserslautern

Kaiserslautern

DE-386

Contents

Zusammenfassung **v**

Abstract **vii**

List of Tables **ix**

List of Figures **xi**

1 Introduction **1**

1.1 Measurement techniques and data generation 4

1.1.1 Measurement techniques for genome data 6

1.1.2 Measurement techniques for transcriptome data 7

1.1.3 Measurement techniques for proteome data 8

1.2 Patterns identification and dimensionality reduction for omics data sets 9

1.2.1 Clustering as a dimensionality reduction technique without prior knowledge 10

1.2.2 Enrichment analysis as a dimensionality reduction technique based on the prior knowledge 11

1.3 Incorporation of the prior knowledge for interpretation 12

1.3.1 MapMan as an ontology choice 16

1.4 Problem formulation 17

2 Method Description **19**

2.1 Tree graph as a layout for functional ontologies 20

2.2 Mapping data to ontology structure 20

2.3 Iterative implementation using dynamic programming algorithm 24

2.4 Optimization function 26

2.4.1 Experimental behavior relation expressed in distance measure 28

2.4.2 Functional relation expressed in complexity measure 28
2.5 Main Algorithm of Signature Topology approach 30
2.6 Optimal Configuration Problem 31
2.6.1 Preliminary grouping: Temporal Classification technique 33
2.6.2 Finding the optimal configuration: State Space Search Walk 34
2.6.3 Implementation of the SSSW 37
2.7 High-throughput data preprocessing 39
2.8 Code Source and Implementation 39

3 Materials **40**
3.1 Model organism - *Chlamydomonas reinhardtii* 40
3.2 Analysis of artificial datasets 40
3.2.1 Toy example data 41
3.2.2 Synthetic datasets with different noise levels 42
3.3 Analysis of real-world datasets 43
3.3.1 Proteome dataset 43
3.3.2 Transcriptome dataset 44

4 Results **45**
4.1 Toy example: Signature Topology approach illustration 45
4.2 Synthetic dataset: Noise robustness 48
4.3 Synthetic dataset: Minority revealing 50
4.4 Experiment dataset: Proteome 52
4.4.1 Bin 9: ST structure and grouping are concise 52
4.4.2 Bin 1: Calvin Cycle potential regulatory protein is revealed 56
4.4.3 Proteome overall performance: better functional purity and kinetic similarity for ST structure and resulted grouping 56
4.5 Experiment dataset: Transcriptome 59
4.5.1 Bin 9: ST grouping is concise 59
4.5.2 Bin 29: The ST grouping of the largest group could reveal single regulatory genes . . . 62

4.5.3 Transcriptome overall performance: better functional purity and kinetic similarity for ST structure and resulted grouping . . . 65
4.6 Inter-level Comparison . . . 66
4.6.1 Direct element-wise Pearson correlation . . . 67
4.6.2 Grouping similarity correlation . . . 69
4.7 Approach evaluation . . . 74
4.7.1 Robustness to data removal . . . 74
4.7.2 Balance between kinetic and functional information . . . 75
4.8 Biological application . . . 78
4.8.1 Regulators in Calvin Cycle . . . 78
4.8.2 Regulators in ribosomal proteins . . . 79
4.8.3 Signature Topology outcome - list of terminal groups . . . 80

5 Discussion 81
5.1 Data aggregation along the hierarchical structure . . . 84
5.2 Adjustable parameters for complexity measure . . . 84
5.3 Noise robustness . . . 85
5.4 Easy application to the real-world tasks . . . 86
5.5 Capability for inter system level comparison . . . 87
5.6 Conclusion . . . 88

Bibliography 91

A Supplementary Materials 102

Declaration 112

Acknowledgments 113

Curriculum vitae 114

Zusammenfassung

Eine der Hauptaufgaben der Molekularbiologie ist das Verständnis molekularbiologischer Prozesse. Dies bringt das Problem mit sich, Regulierungsnetzwerke zu kreieren und dazu wichtige Regulatoren zu finden. Damit ist es wichtig, eine solche Darstellung der Daten zu finden, die die unterschiedlichen Muster innerhalb der großen Gruppen aufzeigen kann. Auf der einen Seite gibt es zahlreiche experimentell ermittelte kinetische Informationen über die Veränderung der molekularen Präsenz im beobachteten System. Andererseits sind im Laufe der Jahre Beweise für die Beteiligung von Molekülen an verschiedenen biologischen Prozessen dokumentiert. Beide Informationsquellen haben ihre Nachteile: Experimentelle Daten spiegeln nur einen flüchtigen molekularen Zustand jedes einzelnen Organismus wider und sind daher oft variantenreich und verrauscht; Funktionelle Gruppen wurden als Verallgemeinerung bekannter Rollen von Molekülen in biologischen Prozessen bestimmt und können daher nicht vollständig und nur teilweise für bestimmte experimentelle Bedingungen und einzelne Organismen relevant sein. Unser Ziel ist es, einen Überblick über die experimentell beobachteten Moleküle zu erhalten und das Wissen aus beiden Quellen zu extrahieren, wobei Einschränkungen durch Rauschablenkung und Verallgemeinerung zu vermeiden. Die daraus resultierende optimale Darstellung der experimentellen Daten würde dann helfen, potenzielle Regulatoren zu lokalisieren.

Die vorgeschlagene Methode wird als Signature Topology (ST)-Ansatz bezeichnet, da sie die funktionale Topologie als Quelle des Vorwissens verwendet und eine spezifische Signatur für die gegebenen experimentellen Daten erstellt. Der ST-Ansatz basiert auf einem wissens- und datengesteuerten maschinellen Lernalgorithmus, der über einen dynamischen Programmieransatz implementiert wird. Der vorgeschlagene Ansatz basiert sowohl auf Vorwissen als auch auf dem Lernen aus den Daten und stellt eine Kombination aus überwachtem und unüberwachtem maschinellem Lernen dar. Die resultierende Netzwerkstruktur geht mit dem Datenüberfluss um und vermeidet eine zu detaillierte Beschreibung, die zu Fehlinterpretationen führen kann, und kann Elemente mit geringfügigen Verhaltensmustern herausgreifen.

Die Methode wird mit künstlichen Daten getestet und auf reale Massenspektrometrie-Proteomdaten und NGS-Transkriptomdaten von *Chlamydomonas reinhardtii* angewendet. Der vorgeschlagene Ansatz hilft bei der Identifizierung der potenziellen regulatorischen Gene, deren Rollen in der verwendeten funktionellen Ontologie nicht explizit vorgesehen sind. Darüber hinaus zeigt es eine erfolgreiche Reduzierung der Datenkomplexität unter Beibehaltung aller individuellen molekularen Informationen, die in der Literatur berichtet und in der funktionalen Ontologie gespeichert sind. Wenn der vorgeschlagene Ansatz verschiedene experimentelle Daten mit derselben Ontologie analysiert, sind die resultierenden Netzwerke einheitlich und können daher verglichen werden. Dies bietet die Möglichkeit, eine Vielzahl von experimentellen Bedingungen zu vergleichen, von verschiedenen Organismen bis hin zu verschiedenen Systemebenen.

Abstract

One of the main tasks of molecular biology is understanding the mechanisms of molecular biological processes. This brings the problem of creating regulatory networks and therefore finding key regulators. In order to do it, it is important to have such representation of the data that can reveal the distinct patterns within the big groups. On one side, there are numerous experimentally determined kinetic information about the alteration of molecular presence in the observed system. On the other side, there are documented throughout the years evidences of the involvement of molecules in different biological processes. Both sources of the information have their drawbacks: experimental data reflect only a fleeting molecular state of each individual organism and therefore are often high-variant and noisy; functional groups were determined as generalization of known roles of molecules in biological processes and therefore can be not complete and only partially relevant to certain experimental conditions and individual organisms. Our goal is to get the overview of the experimentally observed molecules and extract the knowledge from both sources, avoiding constrains of noise distractions and generalization bias. The resulted optimal representation of the experimental data then would help to pinpoint potential regulators.

The proposed method is called the Signature Topology (ST) approach, as it uses the functional topology as the prior knowledge source and creates a specific signature for the given experimental data. The ST approach is based on knowledge-and-data-driven machine learning algorithm, that is implemented via a dynamic programming approach. Based on both prior knowledge and learning from the data, the proposed approach represents a combination of supervised and unsupervised machine learning. The resulting network structure deals with data abundance and avoids an over-detailed description that may lead to misinterpretation and is able to pick out elements with minor behavior patterns.

The method is tested with artificial data and applied to real-world mass-spectrometry proteome data and

NGS-transcriptome data of *Chlamydomonas reinhardtii*. The proposed approach helps with identification of the potential regulatory genes, whose roles are not explicitly provided in the used functional ontology. Moreover, it shows a successful reduction in data complexity while preserving all individual molecular information reported in the literature and stored in the functional ontology. If the proposed approach analyzes different experimental data with the same ontology, the resulting networks are uniform and therefore can be compared. That gives an opportunity to compare between a great variety of experimental conditions, from different organisms to different system levels.

List of Tables

1.1 Omics techniques overview 5

3.1 Experimental data, assigned to fictional functional group bin 1 for toy example 41
3.2 Time points with temperature condition for proteome and transcriptome experiments 44

4.1 The whole Signature Topology algortihm in steps, applied to the toy example. 47
4.2 The pairwise Euclidean distance matrix for the elements in the toy example 48
4.3 *IC* values for any node in the tree with 5 elements in the root 48

A.1 DxI indices for proteome dataset 105
A.2 DxI indices for transcriptome dataset 106
A.3 Signature Topology Output list for proteome bin 9 110
A.4 Signature Topology Output list for transcriptome bin 9 111
A.5 Signature Topology Output list for transcriptome bin 29 111

List of Figures

1.1 Analysis concepts in systems biology 3
1.2 Example of a simple ontology 13
1.3 Types of ontology graphs 14
1.4 Schematic visualization of the Signature Topology approach 17

2.1 MapMan ontology structure of bin 16 21
2.2 Challenge of setting a threshold, illustrated on two hypothetical structures 22

2.3 Example of a tunnel in the original ontology structure, that is eliminated in a processed ontology structure . . . 24
2.4 Illustration of the DPA procedure . . . 25
2.5 Illustration of information content (IC) measure . . . 29
2.6 Block diagram for the Signature Topology algorithm . . . 31
2.7 Illustration of State Space Search Walk . . . 37

3.1 Protein kinetics of elements for toy example . . . 41
3.2 Synthetic data generation . . . 42
3.3 Heat acclimation stress experiment setup . . . 43

4.1 Tree structure changes during the implementation of the ST algorithm for the toy example . . . 46
4.2 Noise robustness trial . . . 49
4.3 Minority revealing trial setup . . . 50
4.4 Minority revealing trial - Number of type B errors . . . 51
4.5 Proteomics for bin 9: Time series of proteins . . . 53
4.6 Proteomics for bin 9: MapMan ontology and Signature Topology tree . . . 53
4.7 Proteomics for bin 9: DxI index . . . 54
4.8 Proteomics for bin 1: DxI index . . . 57
4.9 Proteomics of subbin "1.3" . . . 57
4.10 Proteomics overall performance for DxI index of structure complexity . . . 58
4.11 Proteomics overall performance for DxI index of grouping complexity . . . 59
4.12 Transcriptomics for bin 9: MapMan ontology and ST tree . . . 60
4.13 Transcriptomics for bin 9: Time series of proteins . . . 61
4.14 Transcriptomics for bin 9: DxI index . . . 62
4.15 Transcriptomics for bin 29: the original MapMan ontology and the Signature Topology tree . . . 63
4.16 Transcriptomics for bin 29: DxI index . . . 64
4.17 Transcriptomics for subbin "29.2.1.1.1.1": Ribosomal Proteins of chloroplast, subunit 30S . . . 65
4.18 Transcriptomics overall performance for DxI index of structure complexity . . . 66
4.19 Transcriptomics overall performance for DxI index of grouping complexity . . . 67
4.20 Pearson correlation coefficient (absolute value) between protein-transcript pairs . . . 68

4.21 Grouping similarity correlation, comparing proteome and transcriptome within 12 functional groups (MapMan bins) separately, calculated for original data and bootstrapped proteome data (shuffled within each bin) 71
4.22 Grouping similarity correlation, comparing proteome and transcriptome within 12 functional groups (MapMan bins) separately, calculated for original data and bootstrapped proteome data (shuffled within the whole proteome) 72
4.23 Robustness of the ST structure against data removal 75
4.24 Impact of the randomization of different input data on the output structure 77
4.25 Heat acclimation response in *Chlamydomonas reinhardtii* 78
4.26 Comparison of ST structure for the subbin "29.2.1.1.1.1" between proteome and transcriptome datasets 79

A.1 Standard deviation of the real-world data 102
A.2 Noise robustness trial, additional HC structure 103
A.3 Minority revealing trial, additional HC structure 104
A.4 Pearson correlation between protein-transcript pairs 107
A.5 Robustness of the ST structure against data removal: depth-dependent 108
A.6 Robustness of the ST structure against data removal: size-dependent 109

1. Introduction

Comprehensive understanding of any organism as a biological system requires interpretation of molecular complexity and its kinetic variations at multiple levels such as genome, epigenome, transcriptome, proteome, metabolome, and other system levels. With the development of high throughput techniques, it became possible to measure different biological entities simultaneously to gain a large scale view on these system levels. The availability of a constantly increasing number of data sets generated by high throughout techniques, named omics data, has revolutionized the field of medicine and biology with high demand for bioinformatics and computational tools (I. Subramanian et al. 2020).

These advances in biology to study multiple molecules simultaneously lead to a stronger focus on their interactions, and the molecular dynamic of the whole system itself. The approach to consider all elements of the system simultaneously can be described in terms of the interdisciplinary fundamental field such as “systems theory”. According to this paradigm, a system was initially described as “a set of related components that work together in a particular environment to perform required functions to achieve the systems’ perspective” (Bertalanffy 1945). “Systems Biology” is a combination of methods, ideas, and tools that evolved from this theory (Klipp et al. 2016). It is an interdisciplinary field that studies any biological system with a focus on a quantitative understanding of the organizational structure, dynamics, robustness, and functionality. Complex biological functions are considered as the hierarchical composition of interacting molecular compounds with individual properties (Klipp et al. 2016).

Any biological process, either caused by external influence as a response to environmental change or caused by internal agents as cell growth, is controlled by dynamic interactions between specific molecular entities such as genes, proteins, or metabolites. To study a biological system from this perspective, it is necessary to obtain a dataset of experimental measures of interacting entities on all these levels. The more data are available, the more complete information can be extracted and hence the more precise knowledge is concluded from the experiment. However, such an approach leads to the following challenges in Systems Biology (Kitano 2002; Nicholson and

Lindon 2008; Knepper 2012; Shahzad and Loor 2012; Wu et al. 2014; Sauer, Heinemann, and Zamboni 2007): (i) system-wide component identification and quantification at the observed system level; (ii) identification of interactions between observed molecular entities; (iii) quantitative influence of the structure, type, and quantity of component interactions on the biological process which means on a technical level the need for integration of heterogeneous data.

To face these challenges, systems biology requires a specific interdisciplinary toolset, that would include analytical, experimental, and computational methods. Analytical methods aim to identify and quantify observed biomolecules system-wide. They are summarized under the term „quantitative omics"-technologies. Their implementation addresses the first and the second challenges of identification and quantification of the individual components and their interaction. Among these technologies are genomics and transcriptomics based on microarrays/NGS (next-generation sequencing), and proteomics and metabolomics based on mass-spectrometry. These techniques were developed and have been constantly improving for more accurate profiling of gene/protein expression and metabolite concentrations, as well as the determination of protein modifications and physical interactions between proteins (Knepper 2012; Gomez-Cabrero et al. 2014; Pinu et al. 2019).

The third challenge regarding the determination of the varying influence of specific system levels or extraction methods refers to the development of various computational approaches to obtain new knowledge from multilevel experimental datasets. Statistical and machine learning methods are applied to propose a model abstraction, that can be used to explain observed phenomena and gain new information about the underlying principles and mechanisms of the system (Wu et al. 2014). These models can be further converted into a model that can be simulated in silico to provide predictions for more fine-tuned experiments (Shahzad and Loor 2012; Wu et al. 2014; Pinu et al. 2019).

Another approach to meet the challenges is searching through the numerous study reports, so called meta-analysis and data mining. For example, the elements with the identified similar behavior should be checked on sharing similar biologically relevant properties. To make it possible, such biological information about elements of the biological systems needs to be stored in a formalized way, building so-called ontologies. The task of creating the format readable and manipulable for machines and humans and assimilating the huge amount of differently structured data is often referred to as "big data challenge". Created ontologies are used as a source of prior knowledge to facilitate statistical and machine learning methods discussed further in the current work, such es enrichment analyses (Kitano 2002; Shahzad and Loor 2012; Gomez-Cabrero et al. 2014; Pinu et al. 2019).

The choice of computational analysis is determined by the experimental approach, which can take two

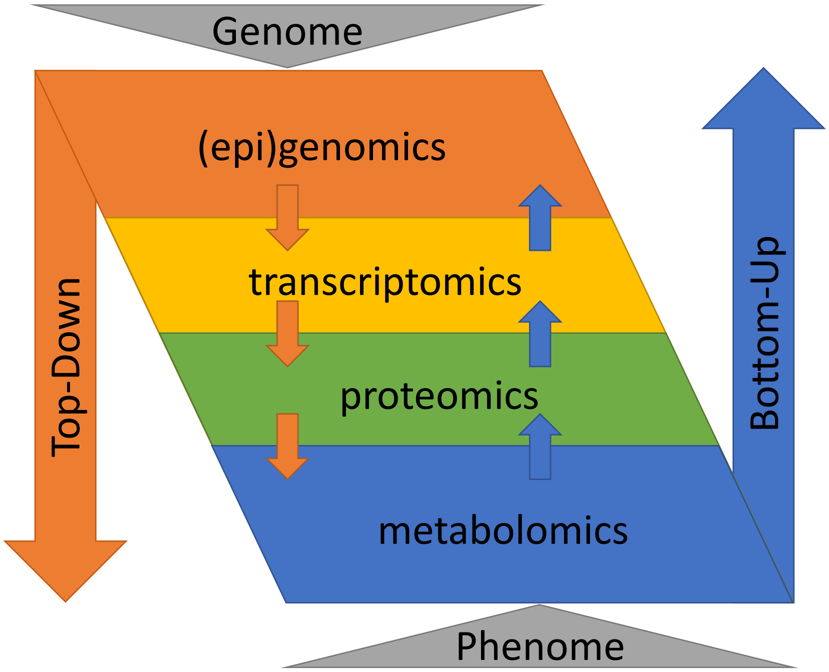


Figure 1.1: Analysis concepts in systems biology from (Pinu et al. 2019). Both approaches, Top-down and Bottom-up, aim to learn information out of the available data, but take different directions of interpretation of the data. Top-down approach considers genomic information (experimental omics data) as a starting point in the analysis and a source of a hypothesis. An alternative is bottom-up approach, which uses phenotype as an entry point and metabolomics as a source of a hypothesis. Metabolites represent the endpoint of gene-environment interactions, hence, representation of phenotypic differences.

major directions in Systems Biology: top-down or bottom-up (Figure 1.1). Both approaches use mathematic modeling and formulations, the difference is in their entry point for understanding biological systems. Bottom-up approaches start with the mathematical formulation of a biological process and the following simulation is based on quantitative experimental data. Top-down approaches, in contrast, start from experimental (high-throughput) omics data, from where knowledge is extracted and underlying mechanisms are proposed (Wu et al. 2014; Pinu et al. 2019).

Classical top-down system biology approaches, which are widely used in any molecular biology field, aim to apprehend the data of one omics level and extract some information out of it, that can help to understand the underlying processes during the experiment in the studied biological system. The approach can be in general represented as a pipeline of the following steps:

1. **experimental data generation**
2. **patterns identification and feature reduction**
3. **incorporation of the prior knowledge for interpretation**

1.1 Measurement techniques and data generation

The first relevant usage of the suffix -ome was around one hundred years ago for the word “genome”, in an article by German botanist Hans Winkler in 1920. In his examination of the relationship between parthenogenesis, polyploidy, and a number of chromosomes, Winkler proposed the term ‘genome’ to indicate “the haploid number of chromosomes that represent the material basis of the systematic unit in association with the respective protoplasm” (Winkler 1920; Noguera-Solano, Ruiz-Gutierrez, and Rodriguez-Caso 2013).

The term underwent a slow expansion of the meaning to include the complete set of genes of a cell as well as the structure information of their position in the chromosome or outside of the nuclei. With the beginning of the Human Genome Project and the blooming of gene sequencing techniques, it gained another important interpretation as a vast storehouse of information on the chemical and structural properties of the molecular building blocks of the cell (Noguera-Solano, Ruiz-Gutierrez, and Rodriguez-Caso 2013).

Curiously, one of the first public events focused on the genome was held in Kaiserslautern, in October 1978. It was the Symposium on the genome and chromatin with focus on their organization, evolution, and function. As follows from the name, discussed issues were plant genetics, chromatin, genomes, and chromosomes. Announced objectives were “(1) Orientation about current trends and results in our understanding of the organisation, evolution, and function of the plant genome at the level of DNA (gene), the level of chromatin, and the level of the karyotype, and (2) Presentation of hypotheses and models which may be stimulating for further research” (Nagl, Hemleben, and Ehrendorfer 1979). This meeting was especially significant for introducing the new field of molecular biology to give a new life to previous research on plants, where the concept of the genome emerged.

At the same time, development in molecular biology has required the emergence of new scientific terminology, related to the concept of the genome. ‘Proteome’, for instance, was proposed at a symposium in 1994 to describe the complete set of proteins that were expressed, and modified thereafter, by the complete genome throughout the lifetime of a cell (Wilkins 2009; Wasinger et al. 1995). But more often in practical usage, this term is used in a more specific sense for the group of proteins expressed by a cell at any particular given time. Following the analog, the ‘transcriptome’ refers to the set of all RNA molecules, transcripted either at a particular time point or throughout the lifetime of the organism.

Similarly, other terms appeared for reflecting information from other system levels of molecular components, such as metabolome, lipidome, up to a higher epimolecular level of phenome (a collection of phenotypical traits of the organism). Nowadays, the term “Omics” is referred to as a suffix signifying the measurement of the entire complement of a given level of biological molecules and information (Schneider and Orchard 2011).

Omics data technologies appeared with the first mentions of the omics terms. First, as a technique to collect measures from a specific set of genome or proteome, and later becoming more autonomous and expanding the coverage to thousands of molecules measured at once with increasing precision. Western blot in a sense can be also called an omics technique (Gasperskaja and Kučinskas 2017). Though, usually, it is more high-throughput methods that are referred to as omics-technologies. The most well-known and commonly-used ones are reviewed in Schneider and Orchard 2011, and the next subsections will give an overview of sequencing techniques for 3 different molecular components: genes, transcripts, and proteins, which are summarized in Table 1.1.

Table 1.1: Omics techniques overview

Technique	Advantage	Disadvantage	Reference
Genome			
Sanger	Robust and precise	Time- and resources-consuming	(Sanger and Coulson 1975; Kircher and Kelso 2010)
NGS	Requires low sample amount, fast for big number of samples	Expensive setup, complicated data analysis	(Jay Shendure et al. 2005)
TGS	Uninterrupted by cycles sequencing	Expensive setup, complicated data analysis	(Erwin L. van Dijk et al. 2018)
Transcriptome			
Sage	Direct and quantitative method, no prior knowledge, requires low sample amount. Simple data analysis	Low-throughput	(Velculescu et al. 1995; Hu and Polyak 2006)
cDNA microarray	High-throughput and quantitative method	Requires a reference genome, complicated data analysis	(Malone and Oliver 2011; Lockhart et al. 1996)
RNA-seq	Direct, quantitative and high-throughput	Expensive, difficulty in identification gene isoforms	(Marioni et al. 2008)
Proteome			
Mass-Spectrometry	High-throughput, reliable	Not all proteins are identifiable, complex analysis	(Yates, Ruse, and Nakorchevsky 2009)
Edman degradation	Precise and reliable	Slow, requires not-modified N-terminal amino acids	(Edman and Begg 1967)
Predicting from DNA/RNA	Expand the number of identified proteins	Expensive setup and complex analysis	(Sheynkman et al. 2016)

1.1.1 Measurement techniques for genome data

The identification of the DNA sequence became the main challenge for several decades after discovering the DNA molecule, its structure, and its role in heredity. The technique, that allowed to decipher the human genome in the first global project (Human Genome Project) was Sanger sequencing, that now considered the golden standard for genome sequencing, because of its reliability, reproducibility, and independence of the prior knowledge (Sanger and Coulson 1975; Kircher and Kelso 2010). But the method is relatively slow because it works with a single DNA molecule and is not easily automatized. As an improvement was developed a new approach called Next-Generation-Sequencing (NGS).

These two methods are based on the polymerase chain reaction. It consists of several cycles of sequential DNA replication. During each of them, DNA polymerase catalyzes the complementary incorporation of fluorescently-labeled deoxyribonucleoside 5'-triphosphates (dNTPs) into the DNA strain. The color of the labeled DNA fragment is recorded by a detector, thus determining the type and position of a nucleotide in the sequence.

The main difference between the Sanger technology and the NGS is that the latter is not limited to a single DNA fragment but analyzes millions of fragments in massively parallel sequencing technology (J. Shendure and Ji 2008). Both sequencing methods are widely used nowadays, so the Sanger method is more suitable for small-scale projects due to its accuracy and reliability, whereas the NGS is reserved for large-scale projects, where Sanger would be expensive, but what more important for modern research, too time-consuming. NGS is the common name for a bunch of sequencing methods, that differ in specifics of identifying individual nucleotides (Karl V Voelkerding, Shale A Dames, and Jacob D Durtschi 2009). For the last years, the most widely used platforms have been Applied Biosystems SOLiD (Jay Shendure et al. 2005), Ion Torrent: Proton / PGM sequencing (Rothberg et al. 2011), and Illumina Genome Analyzer (Bentley et al. 2008).

There are in development several different sequencing approaches that belong to the third generation of sequencing (TGS) paradigm, which is distinguished from NGS by focusing on uninterrupted sequencing of a single DNA or RNA molecule (not an ensemble, not cycle-based). This makes them more useful for some studies such as de novo assembly, improved genome annotations, and epigenome characterization (X. Wang et al. 2019). One example of TGS methods is Single Molecule Real-Time (SMRT) sequencing created by Pacific Biosciences (Ardui et al. 2018). It uses nanoscale optical waveguide technology to directly observe a single DNA polymerase molecule synthesizing a DNA strand. While it is in principle similar to the SNG Illumina technique as sequencing by synthesis, it does not depend on the "scan and wash" cycles and can sequence much longer reads. Also, it does not have the amplification bias associated with polymerase chain reaction (PCR) step.

The sequenced genome can be compared to the reference sequence stored in a library or to a germline cell of the same organism. It allows detecting mutations as large chromosomal aberrations as well as small variants (SV and SNV), which can be useful for studying phenotypically different organisms, in cases with chronic disease and cancers. For studying the reaction of genetically same organisms to changes in the environment, as in the current project a heat-shock acclimation experiment, the next level of functional genome analysis is more interesting – how the same set of genes are used by the organism adjusting to the changing conditions. At the next level, there can be considered RNA sequence analysis – transcriptome.

1.1.2 Measurement techniques for transcriptome data

The transcriptome is the set of all messenger RNA (mRNA) molecules, so-called "transcripts", produced in one cell or in a population of cells in one given time interval. Practically it means the amount and collection of transcripts, detected at the time point of the measurements. Substantial information about functional transcriptomics can be obtained through the analysis of the messenger RNA (mRNA) or complementary DNA (cDNA), which is copied from the mRNA by reverse transcription PCR. Another reason for researchers to choose testing the mRNA or cDNA rather than DNA is the fact, that RNA analysis may be more useful for a gene if it has many small exons, also such analysis can reveal abnormal splicing.

The first attempts to sequence the whole transcriptome of the cell were done with serial analysis of gene expression (SAGE) (Velculescu et al. 1995). The SAGE approach uses sequencing (initially Sanger sequencing) of long concatemers of small tags (initially ~10 bp) that uniquely identify different mRNAs. The method allows a direct transcript quantification and discovery of new genes. Over the years, variations of SAGE have been developed to identify mRNA more accurately by increasing tag length up to 26 (SuperSAGE by (Matsumura et al. 2005)). Another variation of SAGE is massively parallel signature sequencing (MPSS by (Brenner et al. 2000)).

Later, the widest implementation got another approach: DNA microarrays (Lockhart et al. 1996). DNA microarrays (or DNA chips) are based on the fluorescent measuring of the hybridization between the labeled target cDNA strands from a sample and the fixed on the chips probes. Because of their high throughput and lower cost, microarrays were widely used throughout the 2000s and still are important techniques for many studies. However, unlike SAGE, they require a reference genome for probing sets, that is a big limitation that has to be overcome by other techniques.

High throughput sequencing, NGS, beginning in the early 2000s, has sought to address the limitations inherent to previous approaches. Given the similarity in DNA and RNA molecule properties, NGS techniques could be applied to transcriptome sequencing, and the set of various NGS-based approaches were called RNA-Seq analysis (Marioni et al. 2008). More specifically, RNA-Seq is not limited by prior known gene sequences and supports both the discovery and quantification of transcripts using a single high-throughput sequencing assay.

After the RNA is synthesized, other regulatory mechanisms are involved to translate the RNA codons to a chain of amino acids in proteins. The difference between transcriptome and proteome levels can point to the translational regulatory system as a key player for the acclimation process.

1.1.3 Measurement techniques for proteome data

Proteomics is the large-scale study of proteins, particularly their expression patterns, structures, and functions. Essential information for this study is an amino acid chain sequence, that defines the protein properties.

Whereas the classical western blot techniques could give information about the absence and relative amount of presented proteins in the studied system, it always required a reference and could not give information about complete protein collection in the system, as well as the exact amino acid sequence of them.

Proteomics sequencing techniques are numerous nowadays, but the gold standard remains mass-spectrometry (MS). Similar to classical gel-based protein identification techniques, that allow separating the proteins based on the size and charge of the molecules, mass-spectrometry relies on the difference in the ratio between molecular masses of the peptides and its charge (*m/z* ratio), that is detected by high-sensitive hardware setup and following deciphering of the amino acid sequence by a software. The identification of parent proteins from derived peptides now relies almost entirely on the software of search engines, which can perform *in silico* digesting of protein sequence to generate peptides. Their molecular mass is then matched to the mass of the experimentally derived protein fragments.

Another less widely used, but preciser technique is based on the Edman degradation reaction (Edman and Begg 1967). This technique allows to sequence a protein by cleaving and identification of one amino acid at a time, operating on the longer peptide chains (30-50 amino acids) that makes a process of protein identification easier.

In the near future, the MS technique, that has its drawbacks as low sensitivity and complex protein identification analysis (without direct reading of a complete protein sequence), will likely be augmented by simultaneous genomics or transcriptomics profiling by matching specific protein and nucleic acid signatures (W. Timp and G. Timp 2020).

Other techniques will soon appear, that are now in the process of solving technical issues to achieve high-throughput performance, like fluorescent fingerprinting methods (Swaminathan et al. 2018) and nanopore selective filtering (Yusko et al. 2017), that will make proteomics data accumulation even faster than now, and bioinformatics tools should be ready to meet the increased demand for processing this amount of information.

1.2 Patterns identification and dimensionality reduction for omics data sets

Every omics dataset has a common challenge for bioinformatic analysis: high dimensionality and relatively high noise level (Bersanelli et al. 2016). Considering each change in abundance of a biological molecule individually can give very detailed information, but it can be difficult to convert this information into knowledge of the underlying biological processes. Also, such information can be unreliable because of noise or individual variability. In contrast, considering the whole functional group, for example, all molecules involved in Photosynthesis as an information carrier, can lead to high bias and loss of valuable knowledge of different patterns within the big group or individual key player. To make the data better interpretable, it is necessary to find the right balance between these two extrema, but how to determine the equilibrium point? It is a question for dimensionality reduction techniques to answer.

There are two main classical ways for dimensionality reduction: clustering without any prior knowledge, and grouping based on the pre-existed knowledge. For both approaches there are available numerous techniques, but the most known for molecular biology are hierarchical clustering and enrichment analysis, respectively.

1.2.1 Clustering as a dimensionality reduction technique without prior knowledge

Clustering is a method of grouping a set of variables without prior information about the group's properties and no mapping between any samples to a specific group, based only on correlation between the measured samples. On the other hand, one can you classification technique, where we know a set of groups and we know either rules for grouping or there is a training set of samples with known group assignment.

The most widely applied classic method is hierarchical clustering. That is why it will be used as a reference to compare with the proposed method. Hierarchical clustering initially considers each sample in a dataset as a separate cluster and then iteratively repeats the following two steps until all clusters are merged together: (i) identification of the two clusters with the smallest distance between them followed by (ii) merging of these two clusters into a joint cluster. These iterative merging steps allow to identify hierarchical relationships between the clusters that are stored in a tree-structure referred to as dendrogram (Zepeda-Mendoza and Resendis-Antonio 2013). This way, there is a list of clustering sets with number of clusters from one to n, where n is a number of elements in the dataset. This brings us to the question which clustering set (with how many clusters) to consider.

If there is no hint on the group number, then it is advised to use one of multiple elbow criteria to choose the number of groups for the data clustering. We use in the current work the elbow criterion based on the Sum of Squared Errors (SSE) (Nainggolan et al. 2019). The main idea of elbow criteria method is to check the quality of clustering for each number of clusters and plot the reflecting it quality index against the cluster number. The quality index is calculated in such a way, that with increased cluster number it goes either monotonically up or down (depending on the exact index) and at some point the curve is bent. This bending point - the "elbow" - determines the number of clusters. In this work we use the following elbow criterion based on the SSE:

$$SSE = \frac{\sum_{j=1}^{k} \frac{\sum_{i=1}^{n} dist(i,centroid(j))^2}{n}}{k}, \tag{1.1}$$

where k is a number of clusters, n is a number of elements in a cluster, $centroid$ of a cluster j is a matrix of all variables of all elements in the cluster, and a distance $dist$ between element i in the cluster j and its centroid is calculated as an average of Euclidean distances between a variable vector of an element i and each row in the centroid matrix as was suggested in Fesehaye et al. 2017 for grouping clustering.

The clustering technique outputs a set of groups of molecules, those groups later can be analyzed based on the reported functionality of the majority of the molecules in a group. The task can be achieved from the different perspective, if we consider the known properties of the molecules as prior knowledge and group the molecules accordingly.

1.2.2 Enrichment analysis as a dimensionality reduction technique based on the prior knowledge

Enrichment analysis is a technique to make the data more interpretable combined with previously available knowledge, so that later it is possible to apply it to statistical analysis. Pathway enrichment analysis is a crucial step in a pipeline for interpreting high-throughput data that incorporates current knowledge of genes and biological processes. A standard application determines statistical enrichment of molecular pathways, biological processes and other functional annotations for groups of experimentally identified genes. In contrast to them there are also numerous approaches for interpreting single gene lists.

The most well known enrichment analysis method is Gene Set Enrichment Analysis (A. Subramanian et al. 2005) - GSEA. The main idea is to consider difference in gene concentration between the control and the experimental conditions and to find out, which pathways are over-represented or under-represented in the subset of these different genes. First, the list of all differentially expressed genes is ordered from the most up-regulated (the difference is in favour of the experimental condition) or from the most down-regulated (control samples have higher concentration of these genes, compared to the experimental conditions). Then calculate the enrichment score for a specific gene set (a subset of the whole gene list, whose genes are associated with a certain molecular pathway or biological process) and by statistical testing determine, if this gene set is significantly enriched in the highly up-regulated or under-regulated part of the ordered list.

This way, the GSEA algorithm can detect up-regulated and down-regulated pathways in gene expression datasets. Some approaches allow analysis of multiple input gene lists however these primarily rely on visualization rather than data integration to evaluate the contribution of distinct gene lists towards each detected pathway (Kaimal et al. 2010; Reimand et al. 2007). A complete list of currently available methods for pathway analysis includes more than 30 methods and can be found in Alhamdoosh et al. 2016.

This way the dimensionality of the data is reduced by using the pre-defined groups of molecules, based on the available information about signaling pathways. But the proper incorporation requires specific storage of such functional information that can be achieved by different ways.

1.3 Incorporation of the prior knowledge for interpretation

Scientific knowledge discovery is driven forward by "standing on the shoulders of giants", meaning using the understanding gained by scientists from the past. This is especially true for Systems Biology and data generated by high throughput methods. Without a general understanding of a molecule's function or its role in the organism gained in precious previous mostly classical experiments, omics data are impossible to interpret. However, due to the size and comprehensiveness of the omics datasets this knowledge needs to be computationally accessible and readable.

Ontologies are used as a method to describe specific knowledge in a certain formalized way, following strict rules to allow easy access to the stored knowledge and to mediate further analysis. Originally, an ontology appeared as a term in philosophy, as a question about existence and being, and later was applied to more practical issues, as according to the philosopher Barry Smith, an ontology is “any theory or system that aims to describe, standardize or provide rigorous definitions for terminologies used in a domain” (Smith 2003). Nowadays, it is transferred to information science with a focus on the practical usage of an ontology as a system to represent properties of subjects and relations between subjects in an easily accessible way (J. B. L. Bard and Rhee 2004).

Common parts of an ontology are instances, classes, relations, and attributes (Gruber 1993; Uschold and Tate 1998; Guzzi 2019; Hoehndorf, Schofield, and Gkoutos 2015); examples of them are shown in Figure 1.2.

- Instance is a subject, a basic term of an ontology, its knowledge unit.
- Class is another term of an ontology that can group either instances or other classes or a mix of them.
- Relation describes an interaction between ontology's terms, either instances or classes. The most common relation is belonging of one term as a subset to another term, like an instance “circular DNA” belongs to class “DNA” and class “DNA” belongs to class “Molecules”. In principle, a relation can describe any possible interaction, for example, X catalyzes Y or receptor A activates pathway B.
- Attributes are properties of the classes, based on them the classes can be formed.

In a nutshell, an ontology helps to understand which properties are common for which subjects, what are relations between subjects of interest, and allows to integrate new information into the existing knowledge field.

Figure 1.2 shows a simple example of an extended ontology with two types of relations, with visualized common knowledge about 4 deoxyribonucleotides, their classification is based on molecular structure and ability to form hydrogen bonds. Arrows are relation, rectangles are instances, ovals are classes with specific properties. One-side directed black arrows represent a hierarchical relation IS_A: nucleotides A and G are purines and they belong to the class "Purine-based"; nucleotides T and C are pyrimidines and they belong to the class "pyrimidine-based"; and both Purines and Pyrimidines are Deoxyribonucleotides. The hierarchical nature of the relation allows us to state, based on the structure, that all nucleotides A, G, T, and C are Deoxyribonucleotides. Two-side directed (also called undirected) red arrows represent a non-hierarchical relation INTERACT: A and T interact with each other by forming a hydrogen bond.

The further constant development of the ontologies allowed to incorporate much more complex relations between terms. It made possible to achieve the most challenging goal: represent any information stored and expressed in natural language in ontology terms (van Dam et al. 2019). It became a separate research field in the information sciences, aiming to combine data mining and automatic information storage on the World Wide Web via Semantic Web vision (Eiter et al. op. 2006).

Based on the task, ontologies can be classified into two big categories: (i) domain ontologies, whose aim is to describe the local field of knowledge, so-called domain, and (ii) foundational or top-level ontologies, also known as standard upper ontologies (SUO) (formalized by the IEEE P1600.1 standard (Guzzi 2019)), whose task is to describe a connection between ontologies of a lower level, domain ontologies, to form a high-level structure as an overview of available broad knowledge. The ontologies of the latter type are important for interdisciplinary analysis.

When an ontology of the domain type regards the biological, biomolecular, or biomedical domain, it is called

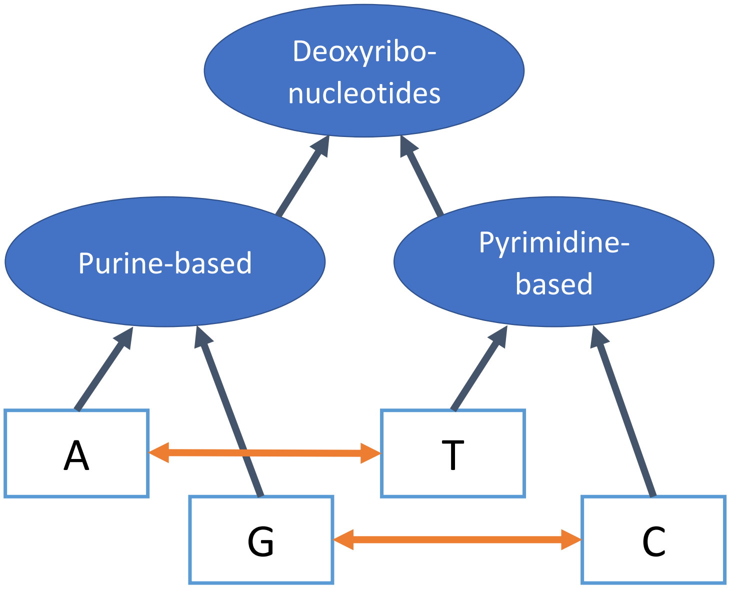


Figure 1.2: Example of a simple ontology. Boxes, circles and arrows represent instances, classes and relations, respectively. See the description in text.

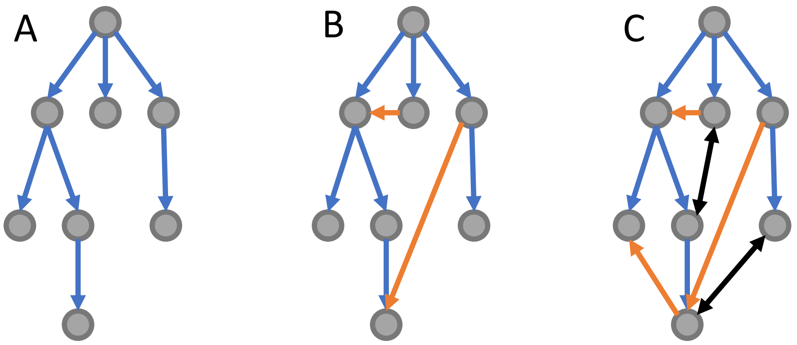


Figure 1.3: Types of ontology graphs. A. Tree. B. Directed Acyclic Graph (DAG). C. General Graph

a bio-ontology. The purpose of a bio-ontology is to characterize entities, that are subject of researches in the life sciences, which are of biological/biomedical significance; thus, a bio-ontology is concerned with the principled definition of biological, biomolecular, or biomedical classes, and the relations among them (Masseroli 2019).

Though ontologies are indeed an abstract concept, however, it is usually possible to illustrate them as graphs of different complexity, where their vertices (also called nodes) and edges (lines connecting the nodes) represent the terms and their relations of the ontology (J. Bard 2003), respectively, as it was visualized in an example on Figure 1.2. Depending on the rules, describing the relations between ontology terms, ontologies can be represented as different types of graph. A strictly hierarchical structure (tree) has one type of directed connection forming a parent-child pair. In this case, each term can have only one parent (see Figure 1.3, A). This type of graph is used further in the algorithm and described in the Methods section. Other types of ontologies can have a different type of connections when one term can have more than 1 parent, but still, all connections are directed. This type of graph is called Directed Acyclic Graph (DAG), Gene Ontology is an example of such (see Figure 1.3, B). In more complex cases, there can be a rule of interaction, where both elements are equal interacting players. This rule would be described as undirected connection, without the parent-child link, then such ontologies can be represented as general graphs (see Figure 1.3, C).

The main functions of ontologies are the following:

- Handling complex areas of knowledge – concise and precise language is needed to formalize knowledge, representing different levels of complexity.
- Interoperability – information stored in one ontology should be easily extracted and translated to enable comparison or be used in other ways with information from another ontology.
- Exploring large datasets – knowledge stored in the ontologies should help to make sense out of new data, especially high-throughput multi-dimensional datasets.

- Mapping knowledge domain – ontologies should be incorporated into a higher-level system, that would include all available knowledge about certain domains or fields of science. The system should give an easy interpretation for any new data, despite the possible complex nature of the explored biological system.

As complex systems, ontologies should be created and curated as a public project and be regularly updated to reflect the complete knowledge and state of the art of the field to be of use. Though they cannot completely avoid the subjectivism (Weiss 2020), it is important to make their terms be accepted and accommodate the needs of the majority of the scientific community.

To ensure that the ontologies satisfy given requirements the Open Biological and Biomedical Ontologies (OBO) Foundry was created (Smith et al. 2007). The mission of the OBO Foundry is to establish rules and provide a platform to develop a family of interoperable ontologies which would be both logically constructed, according to the informatic theory, and scientifically accurate to incorporate the biologically precise representation of the reality. With the aid of the OBO Foundry, a new paradigm for biomedical ontology development was created, as well as a set of ontologies for individual domains of investigation that became the gold standard (Masseroli 2019).

Far from a complete list of the well-known and wide-used OBO are the Gene Ontology (GO) (Ashburner et al. 2000), Cell Ontology (Meehan et al. 2011), Foundational Model of Anatomy (Rosse and Mejino 2008), Sequence Ontology (Cunningham et al. 2015), Protein Ontology (Natale et al. 2017), Relation Ontology (Guardia, Vêncio, and Farias 2012), Ontology for Biomedical Investigations (Bandrowski et al. 2016). The complete list can be found on the OBO Foundry website: `http://www.obofoundry.org/`.

The main goal of creating ontologies is easy access to the stored knowledge and easy implementation of the knowledge for new data analysis. As an example, consider one of the most common usages of a functional ontology: Gene Set Enrichment Analysis (GSEA). It is a simple, but a ubiquitous technique for highlighting biological processes that are crucial for understanding a mechanism of the studied system or process. It is achieved by gene category over-representation analysis, when genes are grouped into categories, based on some common biological property. Then the groups with high representation amongst the differentially expressed genes are highlighted for further detailed research (Young et al. 2010). The idea behind was to avoid the bias of over- or under-representation of single genes or transcripts, that appeared as an artifact of the measurement technique (Oshlack and Wakefield 2009).

1.3.1 MapMan as an ontology choice

MapMan ontology's annotations were initially manually curated from a set of various ontologies and libraries such as Gene Ontology Consortium (Ashburner et al. 2000), Kyoto Encyclopedia of Genes and Genomes (KEGG) database (Kanehisa and Goto 2000), The Arabidopsis Information Resource (TAIR) database (Huala et al. 2001) and many more others. At first, a model organism *Chlamydomonas reinhardtii* was chosen as a focus of this ontology, then later it was adjusted to a broader range of plant organisms as the MapMan system is mostly used in plant genomic researches.

The MapMan ontology system consists of 34 separate hierarchical trees, each of which represents a big functional group or so-called bin. Each term in the ontology corresponds to the biological process function with a certain level of specificity, which is reflected in the term's position on the tree. In contrast to KEGG Brite, the depth of the tree is not fixed and depends on the available information of detailed specifics of the given functional group. This way the least studied functional group tree (bin 3, metal handling) has only 1 level with 3 subbins in it, whereas one of the very well-studied functional group Protein is represented by a tree with 7 depth levels.

Connecting MapMan ontology terms to the terms of the golden standard Gene Ontology (GO) reveals the role of MapMan in integrating two independent GO ontologies: GO-BP (Biological Processes) corresponds to the higher, more general MapMan terms, when GO-MF (Molecular Functions) terms correspond to the lower, more specific MapMan terms (Klie and Nikoloski 2012). It means that, though they both have similar structural capabilities, MapMan has better coverage of the specificity of the terms.

MapMan ontology is further extended with more detailed information about pathways and processes, and with the last update in 2019 (Schwacke et al. 2019), nowadays, the ontology provides free access to functional information of a great range of organisms and system levels.

MapMan ontology was chosen for this work as one well described and broadly used in plant genetics ontology, with detailed diving into the specific role of biomolecules on different levels, such as genome, transcriptome, and proteome, that can provide further an opportunity to compare between different system levels. Moreover, MapMan was already used as a functional annotation source for previous studies in our group (Schroda, Hemme, and Mühlhaus 2015; Hemme et al. 2014), so it would be suitable for comparison between these and other studies in plant biology.

1.4 Problem formulation

The described classical top-down approach implemented via hierarchical clustering or enrichment analysis with its undoubtful benefits of universal and straightforward manner has a certain disadvantage, namely, it is static in terms of a combination of the experimental data with prior knowledge, stored in biomolecular functional ontologies. It means, that while the experimental data were obtained under different conditions, they will be considered with the same prior knowledge schemes. That would lead to a bias in analyzing the current experimental data. For example, the prior knowledge, based on the functionality of molecules in healthy conditions, can be misleading for the interpretation of pathological data. Ideally, the prior knowledge should be able to adapt to the exposed experimental data.

The proposed new approach is a combination of these two sides of the classical approach: clustering and pathway analysis, that are done not separately and independently, but iteratively on each level of the hierarchically organized topology of functional knowledge (see Figure 1.4). **The Signature Topology (ST) approach aims to extract the footprint signature out of the current experimental data and represent it as an optimal network (topology) according to the prior knowledge stored in the tree-organized functional ontology.** This way the clustering is assured to be dependent on the functional pathway information, and pathway set

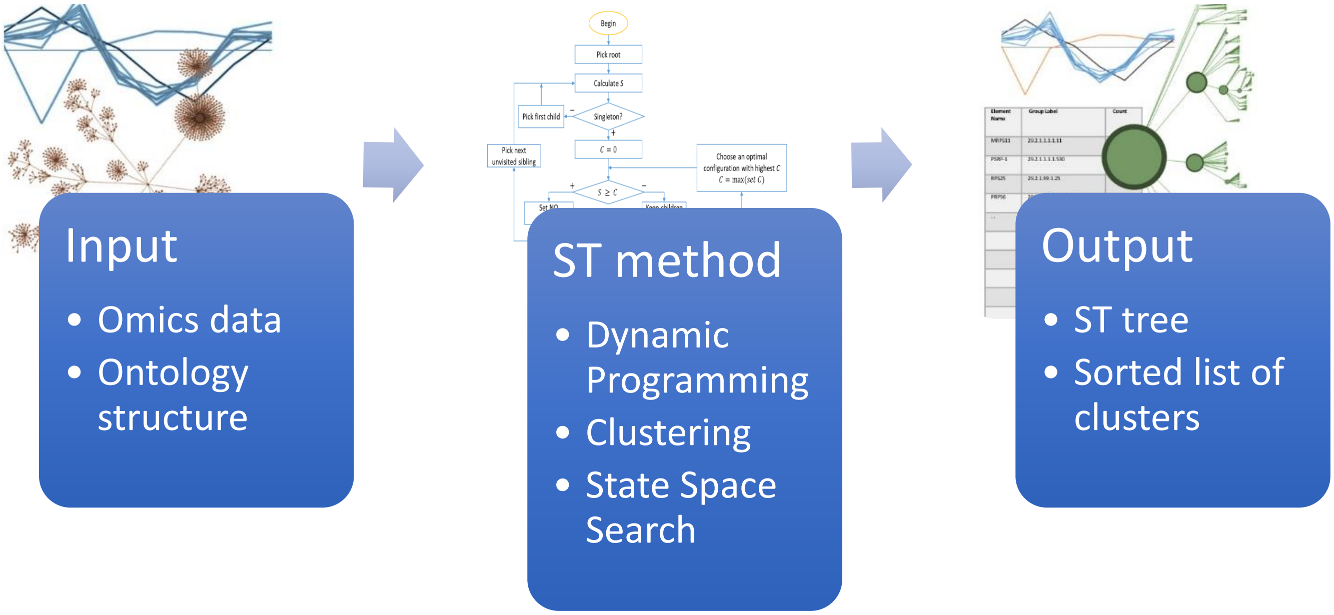


Figure 1.4: Schematic visualization of the Signature Topology approach.

grouping be immediately reflected in the clustering, hence, the static nature of the classical approach is avoided.

The implementation of the ST method is based on the dynamic programming algorithm; prior functional information is taken from the MapMan ontology (Thimm et al. 2004); the optimal grouping for each function specificity ontology term was found with a state-space search walk algorithm with initial starting point obtained from the hierarchical clustering.

The ST approach was tested on both synthetic, and real-world data. The synthetic dataset showed the robustness of the approach to the noise so that the algorithm is still reliable for the high level of variability, which is a common attribute of the molecular biological data; and suitable to the outlier-detection. The real-world data, proteome and transcriptome of the model organism *Chlamydomonas reinhardtii* under heat acclimation conditions, showed an ability of the ST approach to reduce the data complexity and to point out the regulatory genes.

As the ST approach combines two classical methods, clustering and pathway analysis, that brings us to a newly developing branch of data science - a knowledge-and-data-driven machine learning. It is a combination of well-known and broad-used supervised and unsupervised machine learning, where classes are known but used for an assignment only to some extent, combined with some insights from data directly.

This way, the proposed Signature Topology approach, as a data-and-knowledge-driven learning method, allows direct incorporation of experimental data into ontology knowledge structure, adapting the latter to reflect the currently involving biological mechanisms. It applies clustering and enrichment analysis iteratively on different functional specificity levels, resulting in the optimal observing point, given the experiment's noise and variability level.

2. Method Description

The goal of my work was to develop an approach for the automatized interpretation of the data under the current state of information, i.e. to describe the bio-elements such as genes, proteins, transcripts etc. based on their measured expression in an experiment with a system point of view, so that elements can be grouped, but not so rigid as in the classic ontology enrichment approach. The approach allows to fuse existing knowledge with newly measured data. That way the proposed approach would be data-oriented, based on experimental kinetic, but also would incorporate functional knowledge, that is stored in ontologies. Signature Topology (ST) approach aims to extract the footprint signature out of the current experimental data and represent it as an optimal network (topology) according to the prior knowledge stored in the tree-organized functional ontology. As an output of the analysis, a list of functionally relevant groups of bio-entities is created.

For fulfilling the task, the proposed approach works with a hierarchical tree-structured ontology (e.g. MapMan and KEGG) and uses its functional groups and subgroups division for finding an optimal grouping of biological entities from time-course omics experiments balancing between their kinetic similarities and complexity of the structure. As an outcome for further analyses the proposed approach provides a Signature Topology tree structure and a list of all resulted groups. The groups can be sorted by the size, similarity (cluster density) or parameter, characterizing how different the elements of the group are from their functional parent group. It gives an easy access to the most deviant elements, as they are represented as groups, containing only one element (singletons) or for groups with multiple elements with the biggest step gain parameter. This way researchers can focus on the most promising bio-entities for putative key regulators.

2.1 Tree graph as a layout for functional ontologies

Functional ontologies store the knowledge in a form of a graph. The mathematical definition of a graph is an ordered pair

$$G = (V, E), \tag{2.1}$$

where V is a set of vertices or nodes, and E is a set of edges, each of them is a 2-element subset of V, that represents a connection between two nodes.

The chosen for the task functional ontology is constructed as a specific type of graphs, namely trees, with a strict hierarchical structure. It is described as a graph with following conditions:

- Edges are directed: each edge has a starting node and a terminal node.
- Each node, except of a root, has exactly one edge pointed to this node. A node where this edge starts is called a parent of the node. It means that each node can have only one parent.
- A root is a node without a parent, and has the highest rank in hierarchy (corresponds to depth level 0). A tree can have only one root.
- Each node can have multiple edges, that have this node as a starting point. Terminal nodes of these edges are called children of the starting node. Meaning, that each node can have any number of children (or none at all). Nodes without children are called leaves. A rank of hierarchy in a tree decreases in the direction of edges as the depth level increases, so a child of a root would have depth level 1 and so on.

A hierarchical tree-structured graph is illustrated in Figure 2.1. As can be seen, all nodes of a tree can be ordered into levels or depths, according to the (minimal) number of edges from them to the root.

2.2 Mapping data to ontology structure

Described above is an empty structure, where entities of an experimental omics dataset from any system level (transcriptome, genome, proteome etc) can be mapped to. Each node is considered as an assignment for a group of elements. The main rule of the assignment states, that elements, that assigned to a child of a node, are also automatically assigned to the node itself and, therefore, to the parent of the node. It is possible, that an entity is assigned to a parent of a node but not to the node itself and accordingly not to the child of the current node. Each element has a property that defines its assignment to the specific node. So, if proteins are considered as entities, such property can be molecular or cellular functions, that previous studies have associated with the protein, and

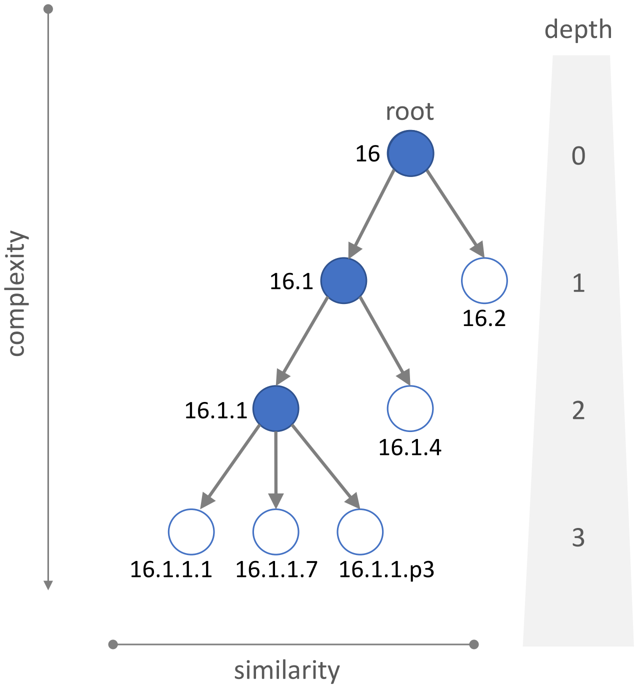
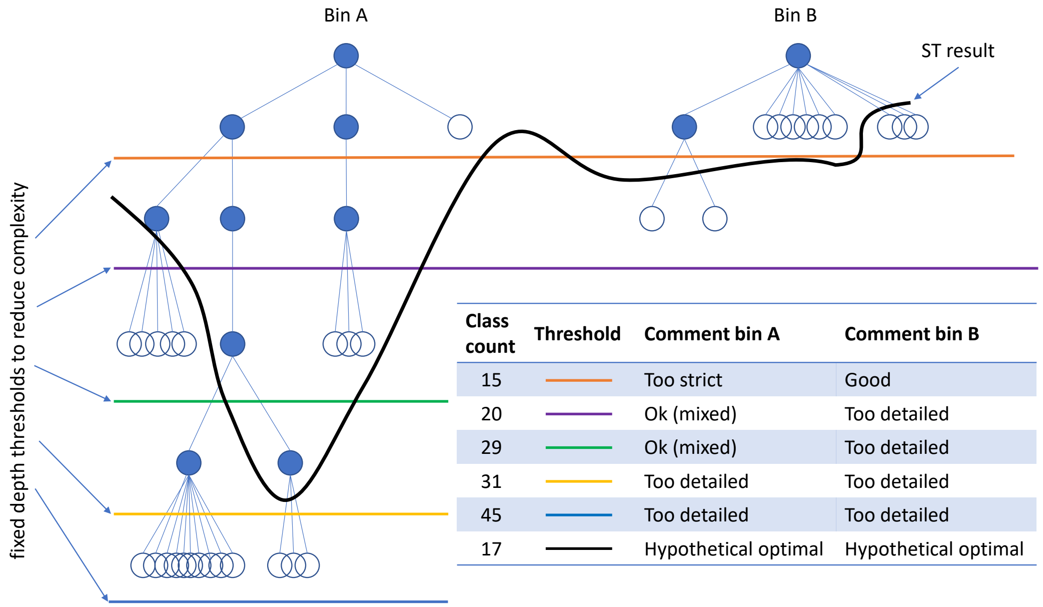


Figure 2.1: MapMan ontology structure of bin 16 (functional group: Secondary metabolism) for the given experimental dataset. Blue circles - nodes with more than 1 element; white circles - leaves, nodes with only 1 element.

this knowledge is reflected accordingly in functional ontology. In the current application of Signature Topology approach, the chosen ontology was MapMan (Thimm et al. 2004), that is why for better explanation of the approach algorithm, the details of the MapMan ontology were given, but in principle any strictly hierarchical ontology is suitable as a source of functional information.

MapMan ontology consists of 34 main functional groups or bins, that were initially manually collected from a set of various ontologies and libraries as Gene Ontology Consortium (GOC), Kyoto Encyclopedia of Genes and Genomes (KEGG) database, The Arabidopsis Information Resource (TAIR) and more (see, for example, (Thimm et al. 2004)). Each bin further was provided with more detailed information about pathways and processes, and was last updated in 2019 (Schwacke et al. 2019), and now is available for a great range of organisms and system levels. Each bin contains sub-classifications, as a tree with nodes, annotated as subbins from MapMan database. In MapMan ontology, a level in a tree, hence a level of specificity, is shown as a length of a sub-bin label. This way, a sub-bin with a label "16" means a root of a functional group "Secondary metabolism" with level 0, see in the Figure 2.1. All proteins, that are known as associated with secondary metabolism functions, are assigned to the root. On the next level 1 the main bin is divided into two subbins, namely "16.1" (label for the more specific function "secondary metabolism.isoprenoids") and "16.2" (label for the more specific function "secondary metabolism.phenylpropanoids"). All proteins, that were assigned to the root are assigned to only one subbin on this level (or none, which means, there is no information about more specific functions, that this protein provides). The corresponding label-function mapping can be downloaded from the MapMan webpage `https://mapman.gabipd.org/mapmanstore`.

As was explained in the introduction, the problem of interpretation of a hierarchically structured database is to determine a level of specificity. The least informative configuration would be where all elements of a bin are grouped together, this corresponds to the root of the tree. The most specific information would be if a tree is considered where each element is ultimately assigned to its own group, a singleton - a leaf with only one element inside. The details, available in ontologies differ between functional groups: if the pathway is of high interest and well-studied, the ontology tends to reach farther depths, because there is known more about the biological process, described in the bin. Without prior knowledge it is impossible to know at which level on the tree the analysis should focus to gain the most insight from the dataset, because this point lies between bias and noise. Sometimes it can be determined by the task of analysis, where, for instance, some specific functional groups are of interest, in this case some certain level of specificity is obvious. But such a situation is rare and in most cases a scientist has to struggle to artificially set the level or the size of leaves as a terminal condition, which is not optimal. So, an additional task of our approach is to set the complexity of the tree automatically, exclusively data-based. An illustration of this challenge is the Figure 2.2, where different thresholds for two hypothetical



Class count	Threshold	Comment bin A	Comment bin B
15	🟧	Too strict	Good
20	🟪	Ok (mixed)	Too detailed
29	🟩	Ok (mixed)	Too detailed
31	🟨	Too detailed	Too detailed
45	🟦	Too detailed	Too detailed
17	⬛	Hypothetical optimal	Hypothetical optimal

Figure 2.2: Challenge of setting a threshold, illustrated on two hypothetical structures. Blue circles - nodes with more than 1 element; white circles - leaves, nodes with only 1 element.

MapMan bins are shown, and corresponding resulting structures are either too general or too detailed and only more flexible and complex line is capable to capture the optimal functional scale of both datasets (black threshold line, corresponding to the ST resulting structure). The table on the figure gives a short description to the resulting structures.

The task to optimize the prior knowledge, encoded in the chosen functional ontology, requires first an adjustment of the ontology structure for further algorithm implementation. The MapMan ontology is filled with extracted experimental data, using prior mapping from dataset elements' (such as proteins, genes etc) connection to their functionality, and the resulting tree structure is processed with 2 following rules:

1. A MapMan tree has a functional specificity range from the most general point (root) to the most specific (singleton leaves). But the latter part of the structure is not always explicitly given, when the most specific functional annotated subbins still have several elements inside. In this case, the expansion of the tree is required: for each such element in a non-singleton terminal subbin, an additional child to the last annotated subbin is created. This child node is a singleton with a label "p" plus index of the element in an experimental database. See for example in Figure 2.1, the leaf "16.1.1.p3", where the element with index 3 has a MapMan annotation "16.1.1", but as the node 16.1.1 is not a singleton, an additional node is created, as a child to the subbin with the label "16.1.1.p3" to uniquely position the element 3 in the tree. This approach gives us an opportunity to separate the element p3 from its closest functional neighbors, saving the information about its potentially different role in biological processes.
2. Depending on the identified experimental items and their assignment to the MapMan ontology, there can be child-parent node pairs with the same sets of items, so-called tunnels (see Figure 2.3). Because these terms may obscure information about the actual functional hierarchy (by imposing artificial layers in the structure without bringing additional information), each mutually redundant parent-child pair is replaced with a single term saving both subbin functional labels but removing an excess level ("16.1-1" instead of the pair "16.1" - "16.1.1" as shown on the Figure 2.3.). The similar procedure of layer removal was applied e.g. by Dutkowski et al. 2013.

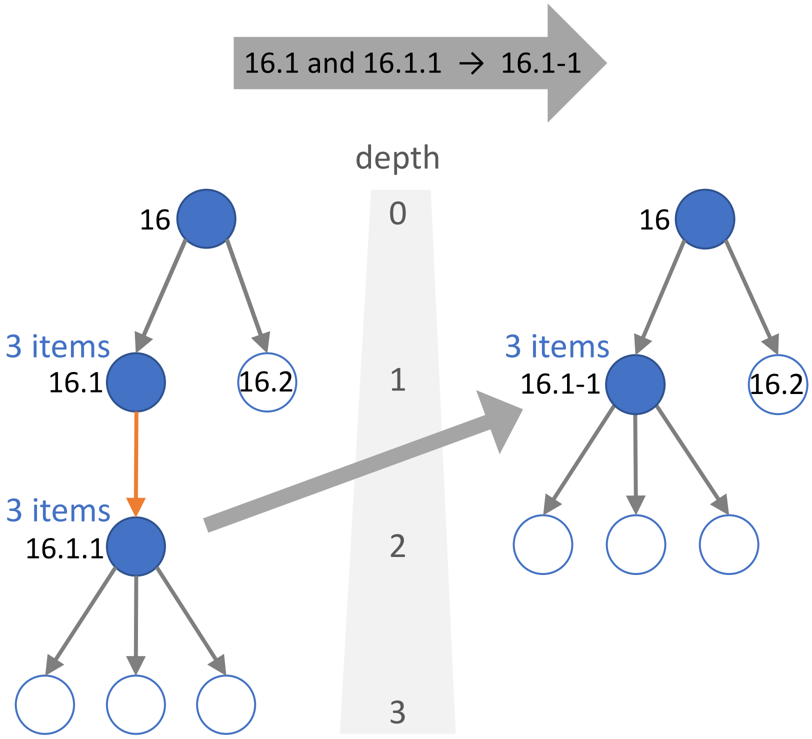


Figure 2.3: Example of a tunnel in the original ontology structure, that is eliminated in a processed ontology structure.

2.3 Iterative implementation using dynamic programming algorithm

With the task of the algorithm to find an optimal viewing point on the experimental data, the algorithm must go through the whole functional tree, through the range of varying functional specificity, to find the best annotation and corresponding grouping of the bio-entities, for example, proteins. And during the walk through the tree, the algorithm should always compare the information gain on each level against the similarity of the elements in the corresponding grouping set. Dynamic programming algorithm (DPA) is appropriate for such complex task as it can be implemented via iteratively solved simple identical sub-tasks.

DPA has two fundamental requirements (Sniedovich op. 2011): recursive substructure and overlapping sub-tasks. The tree structure provided by functional ontology intrinsically fulfills the first one; it has a suitable substructure. That means, that each node of a tree can be considered as a root of the sub-tree, independent of other branches of the tree. And the same algorithm, that is applied to the whole tree can be iteratively applied to each sub-tree.

The main idea of DPA is to break the whole task into sub-tasks, that can be performed at each substructure independently and the result can be used as an input to the next subtask calculation. If the given problem can be divided into such sub-tasks, the second requirement, namely a possibility to divide the whole task into a sequence of overlapping sub-problems, is fulfilled.

Walking through the whole structure starts from a terminal point (a leaf of the tree in the current case) and continues backward to an initial point (the root), carrying information about previous operations in an updating variable V from formula (2.2) (see Figure 2.4). DPA is often used to find the most optimal (shortest, cheapest etc.) path from an initial point to a terminal one.

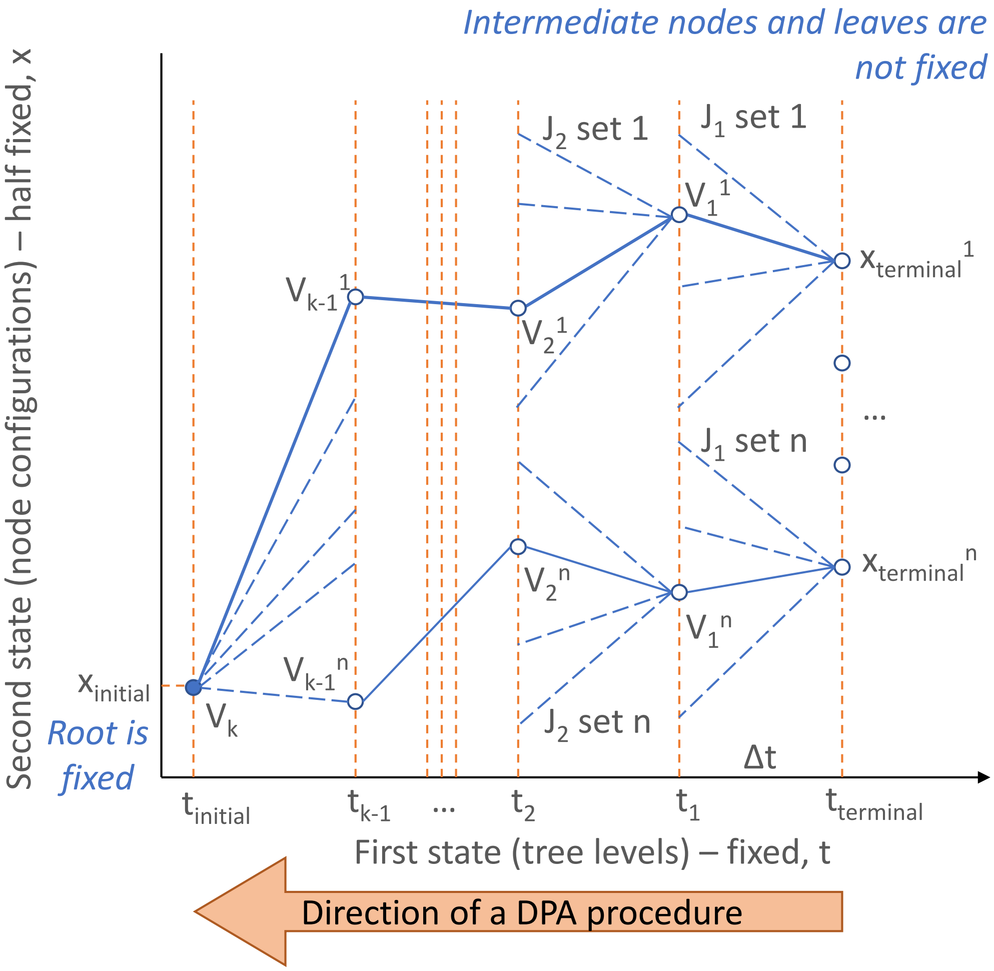


Figure 2.4: Illustration of the DPA procedure. The first state is fixed. In the classical implementation it is a temporal state, e.g. time steps. In the current case it is tree levels, and Δt is a child-parent link. The second state is half-fixed, meaning, it is known, where the trajectory starts (initial state is fixed), but it is not clear, where the optimal trajectory ends. In the current case, the second state is an actual set of nodes in their optimal configurations, that will be describing the grouping of the experimental data. In this case the fixed initial state is a root, and all other not fixed states - the desired structure in the rest of the tree. Set 1 (Set n) means a set of all possible steps that lead to state 1 (state n).

The main formula for DPA is a recursive value function V:

$$\begin{aligned} V_i &= optimal(J_i), \\ J_i &= RC_i + V_{i-1}, \end{aligned} \tag{2.2}$$

where J_i is a cost function at i step, RC_i is a running cost and V_{i-1} is a terminal cost, defined as a value function at a previous step. A running cost parameter depends on the current step. A terminal cost depends on the information prior to the current step, meaning the whole trajectory that was walked through before. The operator *optimal* can be *max* or *min*, depending on the optimization task.

To calculate the cost function V_k the whole trajectory is broken into k steps and starting with each possible terminal point goes backwards to all possible next points. At the first (terminal) step 1 we can calculate J_1, knowing the RC function and setting $V_0 = 0$. Comparing a set of all possible J_1 for each terminal point, we choose the biggest (or smallest) and save it as a $V_1 = optimal(set\ J_1)$. Then on the next step 2 this value will be used as a terminal cost and so on. Thus, we get an algorithm, where we focus on the calculations only for the current step i with sub-problem function, carrying information about previous steps in value function parameter V_{i-1} from previous steps. The procedure is illustrated on Figure 2.4. A detailed description of DPA can be found in Bertsekas 2012.

2.4 Optimization function

The proposed approach is DPA-based and adjusted to two additional features, due to the conditions of the current task:

- apply the back-stepping for a tree, starting with leaves, and save information about not only one trajectory but the whole branch of them. The similar idea is applied in Fibonacci spiral calculation. So, for one node V_{i-1} is defined as a sum of J_{i-1}, obtained from children of the node.
- support changing the tree structure by deciding between keeping the information about the node's children (save non-zero V_{i-1} as described above) or discarding it (set V_{i-1} to zero), thus declaring the current node as a terminal point (leaf of the tree). It corresponds to functional groups, that behave the same and do not give valuable information from the experiment. In that case you would group the bins together and discard the complexity information you obtained up to this step (more specific functional groups).

These conditions are summarized in

$$\begin{aligned} RC_i &= \begin{cases} S & \text{if } S > C \\ 0 & \text{if } S \leq C, \end{cases} \\ V_{i-1} &= \begin{cases} 0 & \text{if } S > C \text{ or } i = 1 \\ C & \text{if } S \leq C, \end{cases} \end{aligned} \tag{2.3}$$

where S, and C are step gain and configuration gain respectively, both are defined below.

Step gain S is the main parameter of a node, that is aimed to express a correlation between purity of a group (similarity of the elements in the group) and complexity of a structure:

$$S = \sum_{i=1}^{n} (d_{i,parent} - d_{i,current}) \cdot IC_{current}, \tag{2.4}$$

where d is a chosen distance measure between elements' kinetic (e.g. gene expression level changing over time) within one node (parent or current), and IC is a chosen measure of complexity, based on information content (IC) measurement, and n is a number of items in the current node.

Configuration gain C is defined as a cumulative function to remember the structure of the whole sub-tree branch:

$$C = \sum_{j=1}^{k} (V_j), \tag{2.5}$$

where k is a number of direct children of the current node, and V_j is a value function for the previous step, when child j was the current node.

With a change of a configuration of children (basically a partition of elements in the node) of the node, its step gain S stays the same, whereas its configuration gain C changes. That means we can rewrite the formulas (2.2) and (2.3) as following:

$$V_i = \max(S_i, C_i^1, \dots, C_i^m), \tag{2.6}$$

where m is the number of possible configurations, and operator *optimal* in our optimization task is *max*.

2.4.1 Experimental behavior relation expressed in distance measure

To estimate the relation between explored elements based on their experimental behavior, a distance measure between kinetic vectors should be specified. In the current project, the chosen distance measure d for a node is:

$$d_{i,node} = \max_{j}^{n}(D(i,j)), \tag{2.7}$$

where i is the current element in the node, n is the number of elements in the node and $D(i,j)$ is a Euclidean distance between vectors in multidimensional space, representing kinetic behavior (changing of protein/gene/etc abundance over time) of the elements i and j.

As the approach aims to pick out groups with different patterns among whose could be outliers, it was important to not mitigate the difference between the elements in the group, that is why the maximum of pairwise distances in the group is chosen as a distance measure.

2.4.2 Functional relation expressed in complexity measure

To express the complexity measure, first let us shortly dive into a separate information theory field - a theory of information content, that was taken as a base for the complexity measure. In a nutshell, information theory aims to quantify how much information is contained in a message. More generally, this can be used to quantify the information in an event and the corresponding variable, called entropy, is calculated using probability (Shannon 1948).

The complexity measure IC can be defined as a variation of information entropy (Shannon entropy), when we calculate the information contained in one group separated from the rest of elements:

$$IC(n) = -\left(\frac{n}{n_r} \cdot \log_2\left(\frac{n}{n_r}\right) + \frac{n_r - n}{n_r} \cdot \log_2\left(\frac{n_r - n}{n_r}\right)\right), \tag{2.8}$$

where n is a number of elements in the current node and n_r is a number of elements in the current functional group (number of elements in the root). The shape of the corresponding IC equation is illustrated on the Figure 2.5.A.

This way, we have the most information gain if we exclude the group with half the size of the initial group (size ratio $n/n_r = 0.5$), and least information gain if we exclude a singleton or all but one element in a group. But if we look at the optimization function from formula 2.4, there is another parameter to balance - a dissimilarity

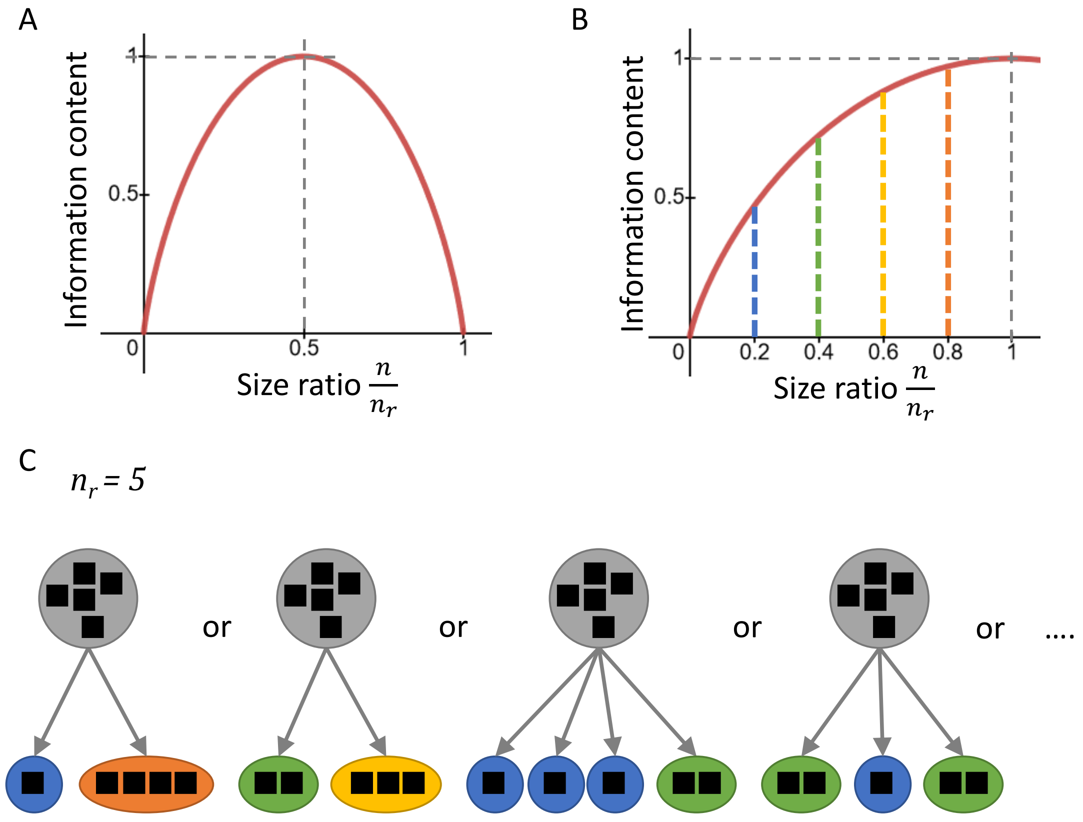


Figure 2.5: Illustration of information content (IC) measure. **A**. Information content (IC) from formula (2.8) depending on the ratio n/n_r where n_r is a fixed size of the functional group (number of elements in the root) and n is a number of elements in a current subgroup (a node in the tree). **B**. Adjusted information content (IC) measure used for balanced step gain calculation from formula (2.9). **C.** Illustration of the calculation of IC for a tree with five elements in the root. Each node in a tree has IC based on the node's size and independent on the number and size of the neighboring nodes. IC value for each size of the node corresponds to its color on the plot in sub-figure B and can be found in Table 4.3.

gain - which has the biggest value when we exclude a singleton and has the least value when we exclude all but one element. This way, to be able to balance between these two parameters, it was decided to use only half of the information content shape (monotonical increase), and we modified the plot accordingly:

$$IC(n) = -(\frac{n}{2 \cdot n_r} \cdot \log_2(\frac{n}{2 \cdot n_r}) + \frac{2 \cdot n_r - n}{2 \cdot n_r} \cdot \log_2(\frac{2 \cdot n_r - n}{2 \cdot n_r})), \tag{2.9}$$

The modified IC plot can be seen on the Figure 2.5.B, this way, IC counteracts the dissimilarity parameter, as it reaches the maximum when we consider the group with the biggest size ratio. In terms of functional group structure it means, that IC measure serves as a penalty to prevent unnecessary detailing of the structure, such as splitting the big group into a bunch of singletons. An example of calculating the IC measure for nodes of a tree with five elements in the root is shown in Figure 2.5.C.

2.5 Main Algorithm of Signature Topology approach

The Signature Topology algorithm aims to obtain the best ontology structure by changing the expanded original MapMan tree, filled with experimental data, so that the data elements will have the optimal annotation and grouping simultaneously. It is done by maximizing the gain by either preserving the current configuration with the cumulative parameter C or setting the current node as a new endpoint with parameter S, discarding the information about its children. To implement the idea, the algorithm includes the depth-first tree search and recursive DPA calculations.

The depth-first search of the tree is the spine of the ST approach: the algorithm walks through the whole tree from a root through the first branch directly to a leaf, and then by back-stepping DPA is applied until the whole tree is visited and processed.

Remarkable, that the approach does not just walk through the template tree structure, it also changes it according to the algorithm (see step 4 in the Algorithm 1 and formula (2.3)). So, DPA uses variable endpoints but always goes back to the root.

Algorithm 1 Signature Topology

```
1: Pick a root
2: For the current node calculate step gain with formula (2.4). Note, that for the root $S = 0$.
3: Check, if the current node is a singleton. If not, pick its first child and go to step 2. If yes, set configuration step $C = 0$, and go further.
4: Compare $S$ and $C$ of the current node and if $S \geq C$ then set a new endpoint, setting current node as a leaf (simplify the tree structure, if it was not a leaf in the original tree). If $C > S$ then keep the children configuration.
5: Check, if the current node has unvisited siblings. If yes, pick next unvisited sibling and go to step 2. If not, go further.
6: Check, if the current node is the root. If yes, END of the algorithm. If not, step back in the tree structure and pick the parent.
7: Generate configurations set out of the children of the current node. Depending on the approach, the size of the set can vary, but the calculation of the configuration step $C$ for each configuration is the same, see formula (2.5). Note, that if a group in the configuration is a child itself, then $V$ is chosen as maximum of its step gain $S$ and configuration step $C$, and depending on it, it may have or have not children, see formula (2.6). If a group in the configuration is a merging of children, then it may not have children, e.i. $C = 0$ and its $V$ is calculated as a sum of step gains of merged children (see formula (2.10)).
8: Choose an optimal configuration out of the set, meaning the one with the highest configuration step $C$. Go to step 4.
```

The proposed algorithm allows to walk through the whole tree and by saving information from previous steps it prevents recalculating gains even during finding the optimal configuration and changing the tree structure. The algorithm in a block diagram view is represented in Figure 2.6.

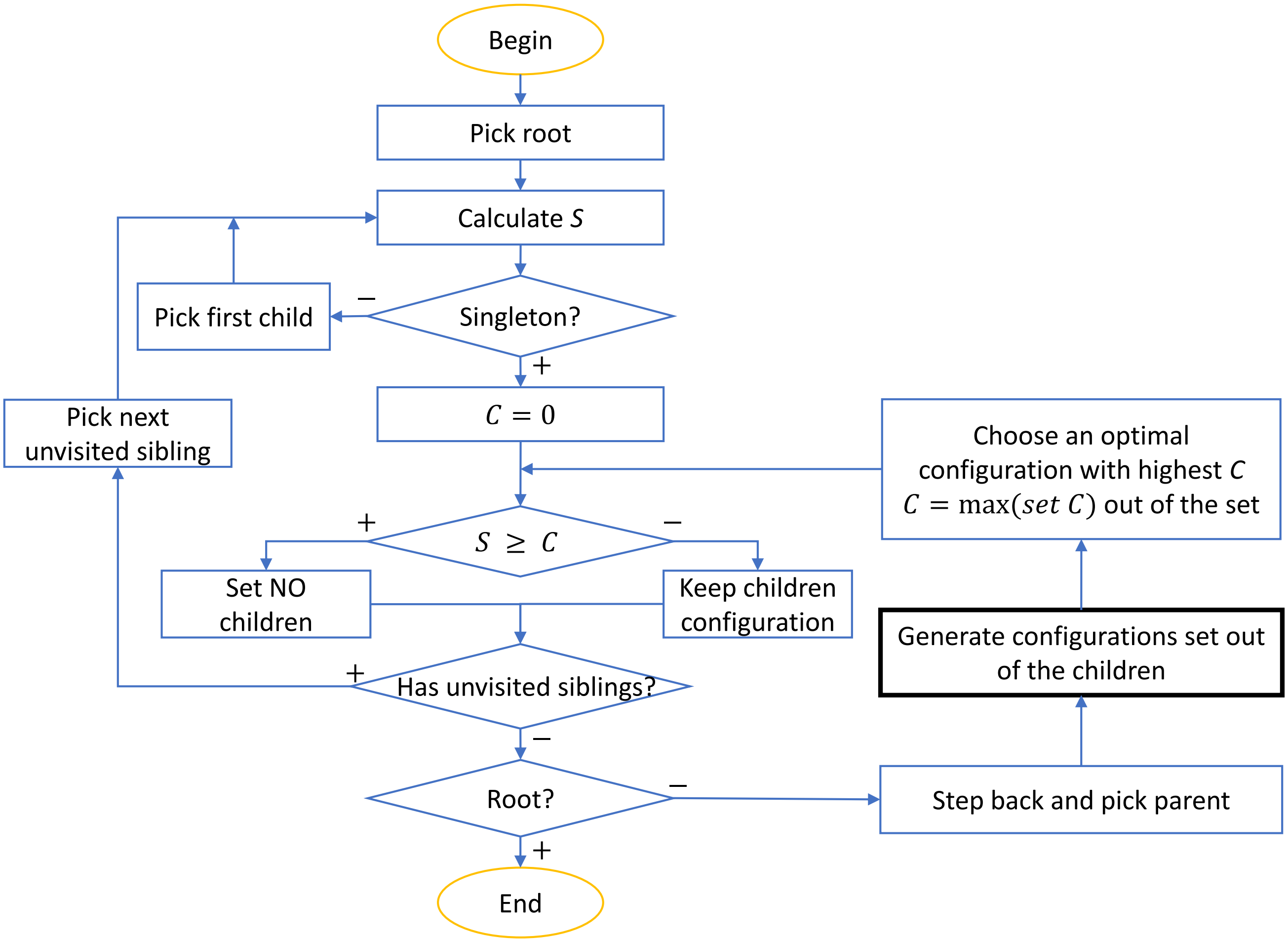

Begin
Pick root
Calculate S
Singleton?
−
Pick first child
+
$C = 0$
Pick next unvisited sibling
Choose an optimal configuration with highest C $C = \max(set\ C)$ out of the set
+
$S \geq C$
−
Set NO children
Keep children configuration
Generate configurations set out of the children
+
Has unvisited siblings?
−
Root?
−
Step back and pick parent
+
End

Figure 2.6: Block diagram for the Signature Topology algorithm. The step in the black border box may vary, as explained in the subsection Optimal Configuration Problem.

The most crucial and difficult part of the ST algorithm is finding the optimal configuration for the children of each non-terminal node out of the set of possible configurations (steps 7-8 in Algorithm 1, black box in the ST diagram in Figure 2.6). For the biological task it means finding which MapMan subbins can be merged or split to assure the best grouping of elements from the experimental dataset. It is also the most important step, because this step gives the opportunity to the ST approach to adjust the functional ontology to the experimental findings.

2.6 Optimal Configuration Problem

In the previous section we established the global algorithm to find the best structure balancing the experimental kinetics and functional annotation. In order to do it, for each node in a structure, an optimal configuration of children of the node should be determined.

The task of finding the optimal configuration is based on the same formulas for maximization of the configuration gain for each non-terminal node. But before collecting configuration sets, there should be established rules to build allowed configurations. In order to preserve the maximum of relevant prior knowledge, stored in the ontology structure, the following constrains for building the optimal configuration are required:

1. The smallest units for manipulation are the following: original children of the current node in the preprocessed MapMan structure (functional subbins) and extended children for individual elements (as the subbin "16.1.1.p3" in Figure 2.1).
2. Therefore, the most split configuration is the preprocessed original MapMan ontology structure.
3. According to the first postulate, only the solid units, and not their separate elements, can be merged to create new configurations.
4. The last of all possible configurations is the merging of all units together, so-called tunnel structure, when the node has only one child, that has the same size as the current node. According to rules of preprocessing original ontology structure (Section 2.1) it is the forbidden structure, so this configuration will not be considered.

According to these constrains, for example, a node with two original children (MapMan functional subbins or extended singletons one level deeper than the current node) has only one possible configuration - these two children themselves. A node with three original children would have four possible configurations - these three children separately as one configuration and three configurations where one child is separate and two others are merged. The idea behind building following sets of configurations is based on partitions of integers, an approach from field of number theory (Andrews 1998).

Due to the changed configuration of children groups, we have to define rules how to calculate configuration gain C for configurations with merged original children (see e.g. Figure 4.1, Final structure). This way, to calculate C with formula (2.5) and avoid unnecessary recalculations, we can state that because of the third constraint, the once merged group cannot have children and its group gain V will be equal its step gain S, that in its turn equals sum of the step gains of the merged elements:

$$V_{merged\ group} = \sum_{i=1}^{m}(S_i), \tag{2.10}$$

where m is a number of original children groups (units), that were merged to create a new group in a partition set.

It is important to find the optimal configuration in terms both complexity (information content) and kinetics similarity, that is expressed in formula (2.4). The classical clustering approaches are not applicable here, because the first component, complexity, is not considered within their optimal criteria based on dissimilarity.

The optimal configuration is defined by maximization of C that can be described as a combinatoric optimization problem. That in turn can be solved via explicit enumeration. Which means to compare C for all possible partitions of a set of units in the current node. In this case a black border step in the block-diagram in Figure 2.6 would be “Generate all possible partitions of a set of children elements”. The problem is that the number of all partitions, Bell number (Andrews 1998), is an exponentially generated function, that will eventually lead to combinatorics explosion problem.

As the amount of MapMan subbins on each level is fairly limited (average - 6, maximal - 76), the main problem occurs when big number of unannotated singletons (created on the preprocessing step, described in Section 2.1) is involved. It corresponds to the parts of functional ontologies, where many bio-molecules are identified with the same the most specific functional subgroup, and no further information is available to distinguish the roles of the molecules. To simplify the identification of the optimal configuration problem, a pre-clustering technique is used, that will create virtual units, that will be handled as units in finding the optimal configuration instead of singletons, but after the optimal configuration is found, the virtual groups will be substituted back with the original units. For such preliminary grouping any clustering technique can be applied, but the chosen one is Temporal Classification technique, developed by our colleagues (Leifeld et al. 2019) and available in FSharp.Stats library (version 0.3.0-beta).

2.6.1 Preliminary grouping: Temporal Classification technique

Preliminary grouping of bio-entities, that are assigned in the ontology structure to non-annotated singletons can help to reduce the number of configurations to check (to avoid the combinatoric outburst) without losing any functional information, because these bio-entities are siblings on the terminal level of specificity in the functional annotation. The only property that distinguishes them is experimental behavior (kinetics), and that is why using appropriate preliminary grouping will help to reduce the complexity without loss of functional knowledge.

Classification was used instead of clustering for preventing of combinatoric outburst and avoiding the necessary step to choose the best number of groups. For class notification there were chosen parameters of the kinetic vector, so-called signal (Lancaster and Sălkauskas 1990), meaning that the classification technique is shape-based. These parameters as combination of signal features like turning points and number of turning points were chosen as biologically relevant features.

The Temporal Classification approach incorporates the typical kinetic behavior of biological systems, especially tailored for short time series (that is often the case in biological studies because of cost reasons), and can work on a minimum of 4 time points. In the analyzed in the project real-world experiments there are 6 and 8 time points, that make the Temporal Classification approach being suitable for the given datasets. But it is worth knowing, that this step, preliminary grouping, can be implemented by any grouping approach, and be adapted by the researcher's needs.

The main idea is using constrained smoothing splines for curve fitting with assumptions of limited amount (4 in this case) of turning points (extrema), that has two focuses: minimizing flexibility energy by minimizing second derivative (fits to biological behavior) and constrained to possess a low amount of turning points. In other words, it is an enforced realization of extrema (Meyer 2012; Wood 1994; Lancaster and Sălkauskas 1990).

Model selection is realized on two levels. The best fit with robust generalized cross validation (rGCV) was used for low-level model selection (Lukas, Hoog, and Anderssen 2016). The best among the selected on the low-level models are found for each extremum possibility (how many extrema the model has, from zero to 4) by minimizing corrected Akaike information criterion (cAIC).

When the model for a given kinetic line is selected it is followed by isolation of extrema (spline function is known, so a 'simple' derivative resolves extrema positions) and rounding of extrema positions to discrete signal point.

This way a class in this classification approach is a position of maximum and minimum extrema for item's kinetic on the experiment timeline. The more detailed explanation of the Temporal Classification approach can be found in (Leifeld et al. 2019) or at the FSharp.Stats library (version 0.3.0-beta) introduction page.

2.6.2 Finding the optimal configuration: State Space Search Walk

Still, for a big experiment dataset with many time points, the number of preliminary groups is big enough to cause the combinatoric outburst by checking each possible configuration. One way to avoid it is to implement an algorithm that would allow us to find the optimal configuration by walking from a configuration with lower C to a configuration with higher C, ultimately reaching the optimal configuration. This algorithm can be implemented by the State Space Search Walk (SSSW) approach, and this way we minimize a number of checked configurations.

State space search (SSS) is a process used in the field of computer science, including artificial intelligence, in which successive configurations or states of a system are considered, with the intention of finding a goal state with a desired property (Poole and Mackworth 2017). In our case the goal state is a configuration with the maximum configuration gain C.

To implement the approach to the problem of finding the optimal configuration, we represent the whole set of possible configurations as a state space – a set of all possible states. In SSS approach the state space is a graph with nodes as states and edges as actions that allow to transform a starting node state into a target node state. In the current case scenario, we consider the action as a change in configuration, whereas one element is moved from its current group to another group. This representation allows to state, that all configurations are reachable from any chosen initial state.

As the property of the goal state is known: the optimal configuration should have the biggest gain, a heuristic approach for SSS problem can be applied. Because of the complex nature of the gain calculations, implementation of more efficient heuristic approaches such as A* search is not possible, that is why the chosen approach is the adjusting of a simpler greedy best-first search.

As known disadvantage of the greedy algorithm is sticking in local minima (Poole and Mackworth 2017), as a way to overcome this drawback, the implemented SSS Walk is a best-first search with expanded number of paths to check from the only best one to several (parameter A_{best}) and allowed situations when the gain in the next state is lower to some extent than in the current state (parameter δ_{sink}). To save time and memory there was also introduced a parameter N_{steps} to reduce the path length, as the goal state property is not recognized immediately by encountering the state but only comparing the states to each other, and by testing the system the optimal configurations are reached fast enough to be able to stop the search after performing walk of certain length.

Each configuration of a children partition is represented as a state and encoded as a symmetric adjacency matrix mA:

$$mA = \begin{bmatrix} v_{11} & v_{12} & \cdots \\ \vdots & \ddots & \\ v_{N1} & & v_{NN} \end{bmatrix}, \tag{2.11}$$

where cell values v_{ij} equal one when elements i and j belong to the same group and zero if they are in different groups. For example, an initial singleton configuration for children partition of the node "1.2" from the Figure 4.1 (Steps 1-3) will be encoded as a state with the following adjacency matrix:

$$mA_{1.2(singletons)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.12}$$

Here we can see, that only diagonal elements equal ones, that means that each element is in the same group only with itself, meaning all groups are singletons. Similarly, a final configuration for a children partition of the node "1.2" from the Figure 4.1 (Final structure) would be encoded as a state with the following adjacency matrix:

$$mA_{1.2(final)} = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix} \tag{2.13}$$

Here the first (p1) and the fourth (p4) elements are in the same group, that is why $v_{11} = v_{14} = v_{41} = v_{44} = 1$. Elements p2 and p3 are in singletons and consequently have zeroes everywhere outside the diagonal.

An action to transform one state into another is moving one element from its group to another group, and therefore an overview of all possible actions from one state can be shown as a transition matrix mT, whose cells are pairs (configuration gain, state matrix) as parameters of the new state. The Figure 2.7 shows an illustration for a children set containing 3 units. State matrix mA is a 3x3 adjacency matrix for 3 units (their values are color-coded for the groups they are pointing to: blue for a group of 2 elements and orange for a singleton). Transition matrix mT is shown including only gains as values for simplicity of illustration. The transition matrix is not necessary square and contains a number of rows equal to a number of units in the children set (in this case, 3) and a number of columns equal to a number of groups in the current state (in this case, 2) plus one. Each cell (i, j) in the transition matrix corresponds to a new state, that can be reached from the current state by moving element i to a group j. The last column represents an action of moving the element i outside of any existing group, i.e. excluding it to the singleton (in the Figure 2.7, black dotted box over matrix mT). Accordingly, the new configurations are depicted with corresponding state matrices. Note, that sometimes new states can repeat each other: states with matrices mA_{02} and mA_{12} were reached by pulling either the first or the second element

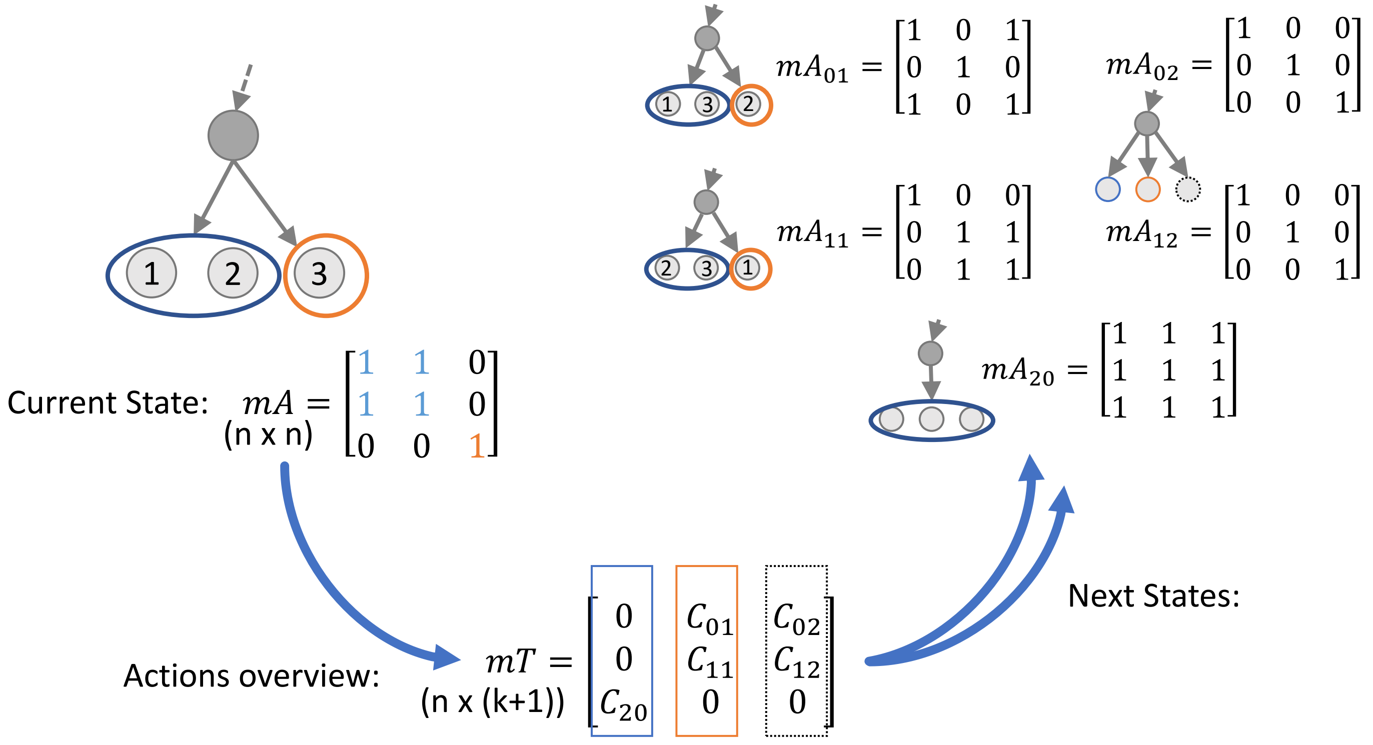

1
2
3
Current State: $mA = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
(n x n)
1 3 2
$mA_{01} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$
$mA_{02} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
2 3 1
$mA_{11} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 1 \end{bmatrix}$
$mA_{12} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
$mA_{20} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$
Next States:
Actions overview:
$mT = \begin{bmatrix} 0 & C_{01} & C_{02} \\ 0 & C_{11} & C_{12} \\ C_{20} & 0 & 0 \end{bmatrix}$
(n x (k+1))

Figure 2.7: Illustration of State Space Search Walk. One State Space Search step of a children partition for 3 elements. Explanation see in text.

outside of their group, and they encode the same configuration of singletons; it means that their configuration gains C_{02} and C_{12} are also equal.

For finding the best action in the matrix, the Priority Queue structure was used, implemented with the binary heap approach (the implementation is thoroughly explained, for example, in (Sedgewick and Wayne 2012)).

2.6.3 Implementation of the SSSW

The described above SSSW method is a good tool to speed up the calculations and avoid checking low-chance states. To apply the tool to our system, first, we need to understand the input, the goal and the output of the method implementation.

To understand what kind of input we should feed to the method, we must keep in mind, that defined in Section 2.6 requirements should be followed to preserve the functional information within each configuration. It means, that the goal of finding the optimal state is practically finding a balance between reflecting the most detailed functional specificity and grouping based on the similarity in experimental behavior for each node in the ontology tree.

Following these requirements, the method understands an element not as a single bio-molecule but as a group of them, determined by MapMan ontology on the corresponding tree level or preliminary defined virtual cluster (discussed in the previous section, we use Temporal Classification as a preliminary clustering technique, but it can be any other technique). To distinguish these undivided groups from general terms "group" and "grouping" within the configuration, let us call them units. It means, that state defined by matrix mA in formula 2.11, describes the grouping relations of these units, and the units can be moved from one group to another to form new state as defined in transition matrix mT from Figure 2.7. Similarly, the elements 1, 2, and 3 in Figure 2.7 are actually units (can contain many molecules each, but these molecules are not manipulable individually).

As a first step to launch the SSSW, a starting position - an initial state - should be chosen. In the current implementation, to avoid being stuck in the local minimum, several SSS Walks are launched independently, each initial configuration is the result of hierarchical clustering with given different number of clusters. To apply the hierarchical clustering to the units, a special distance measure between groups is required, for the current implementation pairwise correlation metric from Fesehaye et al. 2017 was used. Then each SSSW ends with the found optimal state, namely a specific configuration of the node. All potential optimal configurations are than compared based on their configuration step and the one with the biggest gain is chosen as an ultimate optimal configuration, that is then used in the main ST algorithm 1 as the optimal configuration with configuration step C. This way, the calculations for the black border box in diagram in Figure 2.6 are completed for the current node.

To avoid repetition in the walk and therefore loops, all encountered states are saved in the collection M and did not require recalculations each time they are encountered.

The application of the State Space Search Walk is summarized in Algorithm 2:

Algorithm 2 State Space Search Walk

```
1: Set terminal conditions: $\delta_{sink}, A_{best}, N_{steps}$ as a ratio of allowed gain sinking, a number of best actions to check, and a maximal length of the walk, respectively.
2: Apply to undivided solid units the Hierarchical clustering technique for each $k$ between 2 and $n-1$, where $n$ is number of units.
3: Use the first clustering configuration as an initial state, calculate the configuration gain $C$ for the state with formula (2.5), create an adjacency matrix for the state as in (2.11). Save the pair $(C, mA)$ in the collection $M$.
4: Create the transition matrix with values as pairs $(C, mA)$ of the resulting states.
5: Iteratively and recursively repeating steps 3-4 step from the current state to the best next state, according to the terminal conditions: check $A_{best}$ best actions, no longer than $N_{steps}$ steps in a path, not allowing repetitions in visited states (use collection $M$ to compare) and not sinking in gain $C$ lower than $\delta_{sink}$ allows.
6: Save the best state (with the biggest $C$) from all resulted walk paths (their number is $A_{best}^{N_{steps}}$).
7: Repeat steps 3-6 for each clustering configuration from the step 2.
8: Choose the best state (with the biggest $C$) over all states saved in step 6 as the optimal configuration.
```

By application of SSSW the path to the state with maximum configuration gain C is determined, which shortens the configuration search and finally leads to the optimal configuration without the necessity to test all possible configurations, which otherwise could easily lead to a combinatoric explosion. The SSSW method is applied on each node with unclear children configuration (more than two units in the node), thereby identifying the overall optimal structure step by step.

2.7 High-throughput data preprocessing

In this work the following data are analyzed: mass-spectrometry (MS) data from proteome measurements, transcriptome measurements and synthetic data with simulated different noise levels. Each dataset is a set of vectors, representing time-course changes in expression level of biological elements (genes, transcripts, etc) throughout the experiment. All kinetics data are processed with mean centering transformation (Z-score transformation). It gives a better representation of trends in behavior and diminishes the influence of absolute values.

According to researchers' needs, data can be additionally weighted to emphasize certain time points. Although, it is important to keep in mind, that it can obstruct the comparing of ST results between different experimental set-ups.

2.8 Code Source and Implementation

The code for the proposed approach is written in F# and implemented in Microsoft Visual Studio Community 2017. The main core of the code, implementing the Signature Topology algorithm, as well as used in this work analyses and plot functions, are available as Supplementary Source Files (.fs format).

All presented in the current work results and corresponding plots can be repeated by reader by running the Supplementary Script Files (.fsx format).

The code is open-sourced and can be found publicly available as a module 'ST' in BioFSharp library on GitHub page of the computational bioinformatics core unit CSBiology: https://github.com/CSBiology/BioFSharp/.

3. Materials

3.1 Model organism - *Chlamydomonas reinhardtii*

The chosen model organism green alga *Chlamydomonas reinhardtii* is a unicellular, soil-dwelling, $10\mu m$ sized organism with low biological complexity. That makes it a convenient model for studying stress responses on different system levels as it simplifies analysis (Harris, Stern, and Witman 2009; Schroda, Hemme, and Mühlhaus 2015). *Chlamydomonas* performs photosynthesis with a single chloroplast containing a photosynthetic apparatus that is similar to that of higher plants (Schroda, Hemme, and Mühlhaus 2015). The mitochondrial, nuclear, and chloroplast genome is already sequenced and its biomolecules have been studied extensively so that functional annotations are available (Merchant et al. 2007). Furthermore, it has a very short generation time of 5-8 hours and can be grown depending on the conditions as photo-, mixo- or heterotrophic (Harris, Stern, and Witman 2009). Following advantages over higher plants that make *Chlamydomonas* an excellent model system are (i) environmental changes can be rapidly and homogeneously applied to all cells in liquid culture; (ii) all cells in a culture are of the same type and different cell cycles have no influence as they can be averaged out by growing cells as asynchronous cultures; (iii) maintenance of cells under controlled conditions is possible; and (iv) gene families have generally fewer members than in other organisms (Hemme et al. 2014; Schroda, Hemme, and Mühlhaus 2015). For example, 18 land plant genes encoding heat shock factor genes compare with two in *Chlamydomonas* (Schulz-Raffelt, Lodha, and Schroda 2007) and the set of molecular chaperones in *Chlamydomonas* is only half of the set in land plants (Schroda 2004).

3.2 Analysis of artificial datasets

There were used two different approaches to generate artificial datasets created for each specific task.

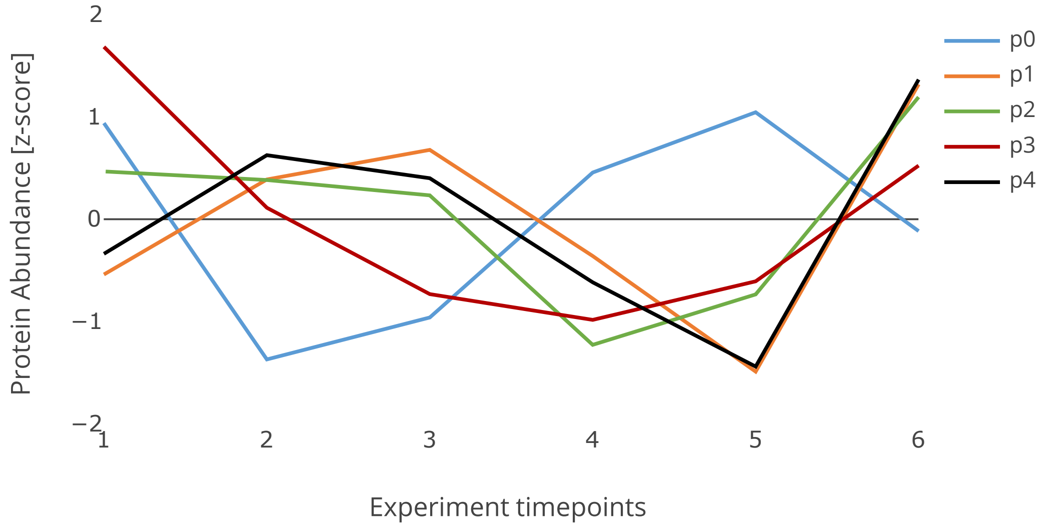


Figure 3.1: Protein kinetics of elements for toy example. Description of elements see in Table 3.1.

3.2.1 Toy example data

An illustration of the approach is done on the fictional functional group, represented as MapMan bin 1. As kinetic data, deliberately chosen measurements from *Chlamydomonas reinhardtii* proteome heat acclimation experiment were used; there are 5 elements, p0-p4, each one is a vector of 6 time-points, assigned to subbins of this fictional functional group. The five proteins are described in the Table 3.1: ID, that can be used for naming additional singleton subbin (used in subbin label "1.2.p4", see Figure 4.1, steps 1-3), protein identificator, MapMan functional annotation, preliminary grouping class. The proteins' experimental kinetic is shown in Figure 3.1. Note, that proteins p2 and p3 (green and red lines) belong to the same preliminary group 4, as they share the same shape in terms of extrema position of the curves, namely having minimum at timepoint 4. In contrast to them, elements p1 (orange line) and p4 (black line), despite being very similar, belong to different groups, because they have maximum at different timepoints (3 and 2 respectively). The original (fictional) ontology structure corresponding to the column "MapMan functional annotation" is shown in Figure 4.1, steps 1-3.

Table 3.1: Experimental data, assigned to fictional functional group bin 1 for toy example

ID	Protein identificator	MapMan subbin	Preliminary group id
p0	Cre08.g372950	1.1	1
p1	Cre12.g509650	1.2.1	2
p2	Cre01.g050950	1.2.2	4
p3	Cre07.g356350	1.2.3	4
p4	Cre03.g207800	1.2	3

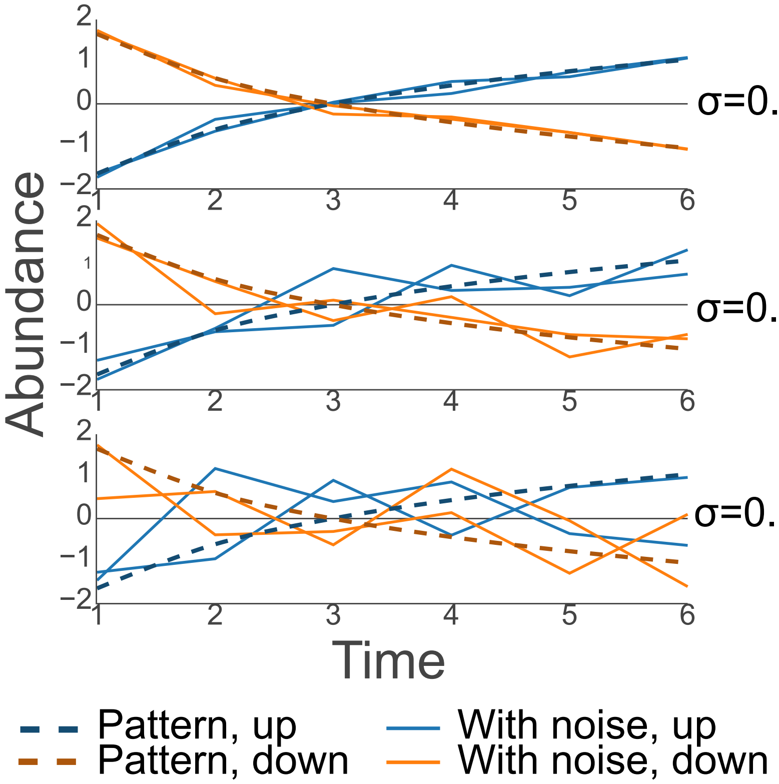


Figure 3.2: Synthetic data generation. Dashed lines are patterns without noise (σ = 0). Solid lines are patterns with added different noise levels, expressed in standard deviation σ.

3.2.2 Synthetic datasets with different noise levels

To evaluate the initial robustness and sensitivity of the proposed approach there were used synthetic data. Since the proposed algorithm is based on shape-based clustering, synthetic data were generated as a noise deviation from certain shape patterns. Two synthetic datasets were used to check two properties of the proposed algorithm: noise robustness and ability to reveal minority patterns (sensitivity). The logarithm kinetic $y = log(x^2)$ centered around mean was chosen to represent patterns for either up- or down-regulations sampled at six timepoints (see Figure 3.2, dashed lines). Noise was introduced as a random value of normal distribution around 0 with varying standard deviation σ added to the synthetic value at each time point (see Figure 3.2, solid lines).

Due to illustrative purposes of the task, small datasets were considered, so each synthetic dataset contains 10 elements. The small size allows to avoid using preliminary grouping.

Based on the specific task (noise robustness or minority revealing), the generated kinetic elements were mapped to specific ontology structures, that are explained in details in the corresponding Results sections.

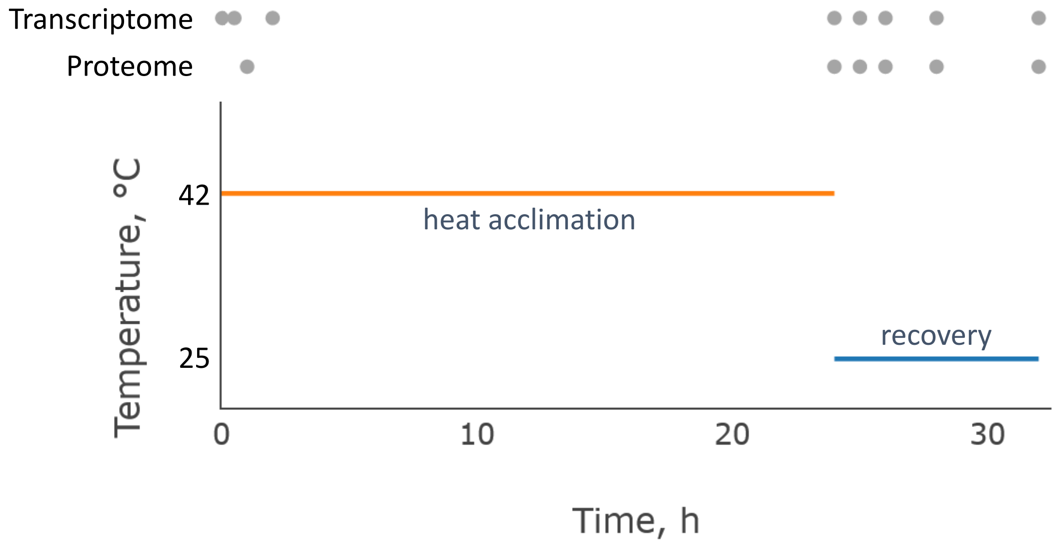


Figure 3.3: Heat acclimation stress experiment setup. The temperature change over time is shown, as well as time points when the proteome and transcriptome were sampled and measured. For detailed time-points description see Table 3.2.

3.3 Analysis of real-world datasets

For the illustration of the ST algorithm implementation on a real-world example, the following datasets were used: proteomics Mass-Spectrometry data and microarray transcriptome data from *Chlamydomonas reinhardtii* on heat acclimation stress experiment. The *Chlamydomonas reinhardtii* strain CF185 (CF185 cwd, mt+, arg7-, complemented with plasmid-derived ARG7 wild type gene) was used in all experiments.

MapMan ontology (Thimm et al. 2004) trees are used as the source of hierarchically structured functional information for both proteins and transcripts. The used mapping file is MapMan version for *Chlamydomonas reinhardtii* for Phytozome v9.0 identification system, is freely available and can be downloaded directly from the webpage: `https://mapman.gabipd.org/mapmanstore` (last called in March 2021).

Liquid algae cultures were exposed to the abiotic heat stress conditions (HS) as an abrupt temperature change from 25°C to 42°C for 24 hours, followed by a recovery phase at 25°C for another 8 hours, the experimental setup is illustrated on the Figure 3.3 and Table 3.2, specifying the sampling time-points for transcriptome and proteome measurements.

3.3.1 Proteome dataset

Experimental data were kindly provided by the Biotech laboratory of TU Kaiserslautern (Hemme et al. 2014). The proteome dataset was obtained by shotgun proteomics approach based on full 15N metabolic labeling.

Table 3.2: Time points with temperature condition for proteome and transcriptome experiments. Time counts after beginning of the experiment. HS=Heat shock.

Time, h	Proteome sampled	Transcriptome sampled	Description
0	No	Yes	HS start, set 42 °C
0.5	No	Yes	HS continued
1	Yes	No	HS continued
2	No	Yes	HS continued
24	Yes	Yes	HS end, set 25 °C
25	Yes	Yes	Recovery
26	Yes	Yes	Recovery
28	Yes	Yes	Recovery
32	Yes	Yes	Recovery

Proteins were identified and quantified with the IOMIQS framework (Integration of mass spectrometry identification and quantification software) (Mühlhaus et al. 2011). All samples were normalized method-specific to minimize technical influences. After applying differentially expressed proteins analysis, 688 identified proteins were determined as significantly different after being log2 transformed and tested with a one-way ANOVA over all time points using a significance threshold (p-value ≤ 0.05) after correction for multiple hypothesis testing according to (Benjamini and Hochberg 1995). In the following preprocessing step kinetic values for each protein were transformed with Z-score (around-mean centering) transformation (see the data distribution in the Supplementary Figure A.1.A).

In this experiment, mass-spectrometry proteome measurements were collected at 6 time points in the course of the acclimation experiment. Experimental setup is summarized on Figure 3.3 and in Table 3.2.

3.3.2 Transcriptome dataset

Experimental data were also kindly provided by the Biotech laboratory but not published yet. After application, samples were taken at different time points in three biological replicates respectively. All samples were hybridized to single-channel DNA microarrays and the measured fluorescence intensities generated gene expression profiles. In this experiment setup, transcriptome measurements were collected at 8 time points in course of the acclimation experiment. Experimental setup is summarized on Figure 3.3 and in Table 3.2. Overall, there were identified 5580 transcripts, determined as significantly changing within the time course using one-way ANOVA with an adjusted p-value threshold of 0.05 after correction for multiple testing (Benjamini and Hochberg 1995) (see the data distribution in the Supplementary Figure A.1.B).

4. Results

The ST approach was tested on different synthetic datasets to check the approach limitations regarding noise stability and outlier sensitivity, as well as on the real-world datasets. The real-world implementation included analysis of proteomics and transcriptomics of the model organism *Chlamydomonas reinhardtii*. Following inter-omics comparison allowed to estimate correlation between Signature Topology structures between the proteome and transcriptome datasets and compare it to the standard gene-wise Pearson correlation. In the end, the ST approach was validated with robustness analysis, and the balance between the influences of experimental information and prior knowledge was estimated.

4.1 Toy example: Signature Topology approach illustration

An illustration of the approach is done on the fictional functional group, represented as MapMan bin 1. As kinetic data we used proteome heat shock acclimation measurements from *Chlamydomonas reinhardtii* experiment; there are 5 elements, p0-p4, assigned to subbins of this fictional functional group. The five proteins are described in the Table 3.1: the index, that can be used for naming additional singleton subbin (used in subbin label $\underline{1.2.p4}$), protein identificator, MapMan functional annotation, preliminary grouping class. The proteins' experimental kinetic is shown in Figure 3.1 and in Figure 4.1, left panel. Note, that proteins p2 and p3 (green and red) belong to the same preliminary group 4, as they share the same shape in terms of extrema position of the curves. The original (fictional) ontology structure is shown in the Figure 4.1, left panel.

In the Table 4.1, the main ST algorithm is thoroughly followed for the toy dataset analysis. For each step in the applied algorithm, the table shows: the current node (its subbin label, temporal preliminary grouping marked with a letter T); indices of proteins in the current node; value for step gain (S) and configuration gain (C) of the current node; action that was done in the step; and transition to the next step. All changes in the ontology structure are illustrated in the Figure 4.1. Preliminary grouping is done according to the Temporal Classification

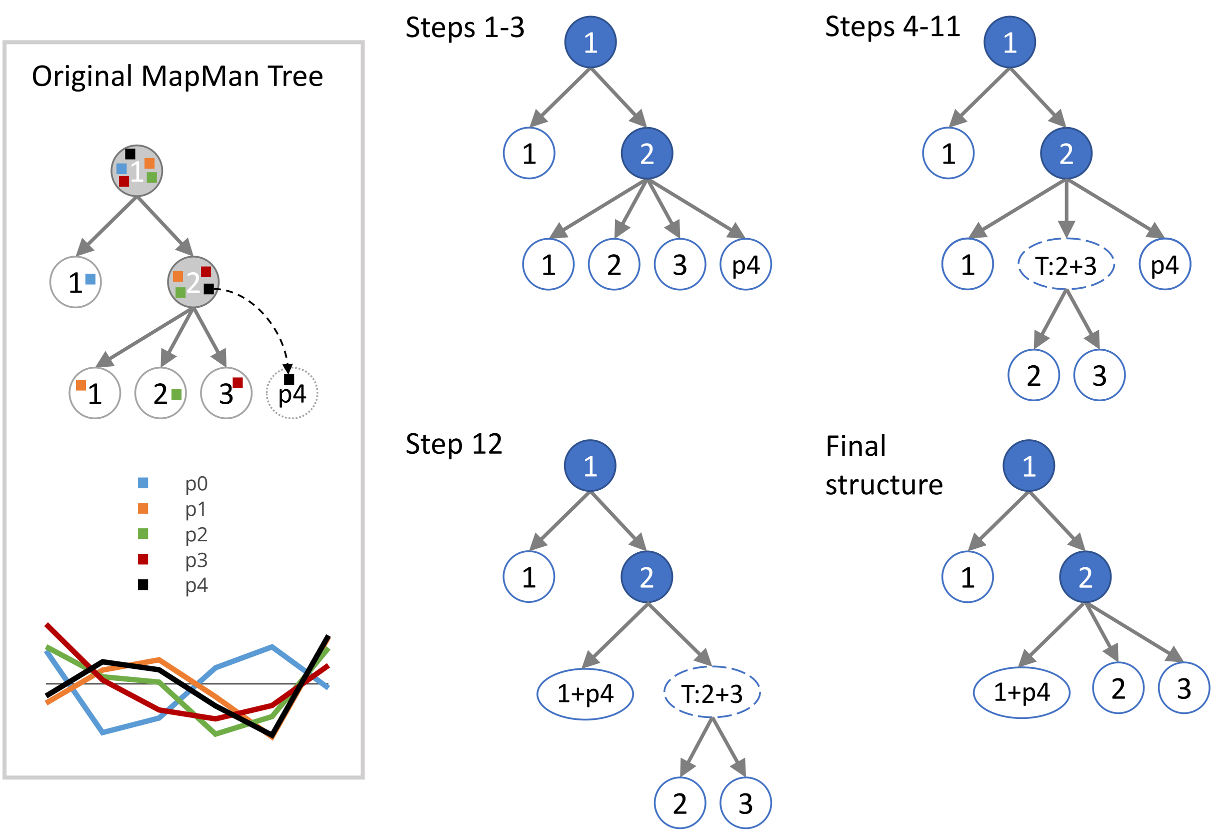


Figure 4.1: Tree structure changes during the implementation of the ST algorithm for the toy example. For detailed description of the toy dataset see Figure 3.1 and Table 3.1. For step-wise implementation of the ST algorithm see the Table 4.1. **Original MapMan Tree**: filled with elements from the toy dataset original MapMan ontology tree. Dashed line shows the preprocessing modification of pushing the element p4 without unique MapMan bin label to an artificial singleton node on the more specific level. Below is a scheme of kinetic behavior of the elements as a short reminder of their experimental similarity. **Steps 1-3**: an initial functional ontology structure with explicitly exposed hidden element p4 as a singleton leaf 1.2.p4. **Steps 4-11**: a structure after preliminary grouping of the children of the subbin 1.2 with the added virtual pre-grouped node 1.2.T:2+3. **Step 12**: a structure with the optimal children configuration of the subbin 1.2. **Final structure**: a structure with the optimal children configuration after removing the virtual node.

algorithm in Section 2.6.1.

Let us consider some steps and standard procedures more closely. So, in the step 1, it is stated that the optimal configuration for 2 children is clear, because an integer partition of two would be either (1),(1) or (2). The latter would give a tunnel, a forbidden structure, that is why the only possible, hence, optimal configuration would be to consider 2 children (nodes 1.1 and 1.2) as they are. Further, for example, to calculate step gain S of the node 1.1 at the step 2 we need to find the biggest distance between element p0 and all other elements in the root, according to formula (2.7):

$$d_{parent} = d_{(p0,parent)} = \max\big(D(p0,p1); D(p0,p2); D(p0,p3); D(p0,p4)\big), \tag{4.1}$$

Table 4.1: The whole Signature Topology algortihm in steps, applied to the toy example.

Step	Node	Proteins	S	C	Node state and action description	Transition
1	1	0,1,2,3,4	0 (root)	Unknown	Has only 2 children, optimal configuration is clear	Go to first child
2	1.1	0	1.94	0 (leaf)	S and C are calculated, end of branch	Go to its sibling
3	1.2	1,2,3,4	4.44	Unknown	Has 4 children, optimal configuration is unclear, there are singletons among children groups	Use preliminary grouping for singletons
4	1.2	1,2,3,4	4.44	Unknown	Preliminary groups are: (p1),(p2,p3),(p4). To apply SSSW we need to know S and C for each group	Go to first child in preliminary grouping
5	1.2.1	1	1.39	0 (leaf)	S and C are calculated, end of branch	Go to sibling
6	1.2.T:2+3	2,3	0.87	Unknown	Has only 2 children, optimal configuration is clear	Go to first child
7	1.2.T:2+3.p2	2	0.83	0 (leaf)	S and C are calculated, end of branch	Go to sibling
8	1.2.T:2+3.p3	3	0.83	0 (leaf)	S and C are calculated, end of branch	Back-step
9	1.2.T:2+3	2,3	0.87	1.66	$C > S$, keep the children	Go to sibling
10	1.2.p4	4	1.26	0 (leaf)	S and C are calculated, end of branch	Back-step
11	1.2	1,2,3,4	4.44	Unknown	All precalculations for finding optimal configuration for units (p1),(p2,p3),(p4) are ready	Apply SSSW
12	1.2	1,2,3,4	4.44	5.03	Optimal configuration: (p1,p4),(p2,p3). $C > S$, keep the children	Remove virtual pre-grouping layer
13	1.2	1,2,3,4	4.44	5.03	Final optimal configuration: (p1,p4),(p2),(p3).	Back-step
14	1	0,1,2,3,4	0 (root)	6.97	The whole tree is visited	End of ST algorithm

Note, that $d_{current} = 0$, because the current node is a singleton. For illustrative reasons, to be able to follow the calculations, the pairwise distance matrix between the elements is shown in Table 4.2 and IC values for any possible size of the node are shown in Table 4.3. This way,

$$S_{1.1} = d_{(p0,parent)} \cdot IC(\frac{1}{5}) = max(4.14, 3.48, 2.83, 4.13) \cdot 0.469 = 4.14 \cdot 0.469 = 1.94.$$

All calculations of step gains S were done analogously.

An example of the configuration step C calculations can be seen in step 9: the temporal node 1.2.T:2+3 has 2 children (see in Figure 4.1, Steps 4-11), and configuration step C according to formulas (2.5) and (2.6) equals the

Table 4.2: The pairwise Euclidean distance matrix for the elements in the toy example. Elements description can be found in Table 3.1 and in Figure 3.1.

		Element j				
		$p0$	$p1$	$p2$	$p3$	$p4$
Element i	$p0$	0	4.14	3.72	2.83	4.13
	$p1$	4.14	0	1.71	2.97	0.49
	$p2$	3.72	1.71	0	1.77	1.37
	$p3$	2.83	2.97	1.77	0	2.68
	$p4$	4.13	0.49	1.37	2.68	0

Table 4.3: IC values for any node in the tree with 5 elements in the root. $IC(i,j)$ means IC for a node with size i in a tree with j elements in the root. Note, $IC(j,j)$ represents IC for a forbidden tunnel structure, hence is not used.

$IC(\frac{1}{5})$	$IC(\frac{2}{5})$	$IC(\frac{3}{5})$	$IC(\frac{4}{5})$	$IC(\frac{5}{5})$ (not used)
0.469	0.722	0.881	0.971	1

sum of the biggest gains of its children, in this case step gains, as their configuration gains are zeros. This way:

$$C_{1.2.T:2+3} = S_{1.2.T:2+3.p2} + S_{1.2.T:2+3.p3} = 1.53 + 2.14 = 3.67.$$

This example covers most of possible actions, that the approach can perform during the construction of a ST tree. The only uncovered case is, when step gain S of the node is bigger than its biggest configuration gain C of the optimal configuration. Imagine, at the step 12, $S_{1.2} > C_{1.2}$. Then, according to the 2.3, the gain of splitting the node 1.2 would not be enough to overcome the punishment due to increasing complexity. That means that the tree structure should not keep the children of the node 1.2. Hence, the original ontology structure should be simplified by removing children of the node and the whole branch with them. In that case the resulted ST structure would contain only 3 nodes: the root 1 and its direct children 1.1 and 1.2.

4.2 Synthetic dataset: Noise robustness

The dataset for this trial was generated as described in Materials chapter: 5 elements with up-regulated pattern and 5 elements with down-regulated pattern. There were tested 16 datasets with increasing level of noise $\sigma = \{0.0, 0.1, 0.2 \ldots 1.5\}$. All elements were assigned only to the root bin, so the method had no prior functional information about the elements. The optimal configuration would be the root with 2 children, each with 5 corresponding elements inside. Because of the increasing level of noise, the method can decide to pick out some

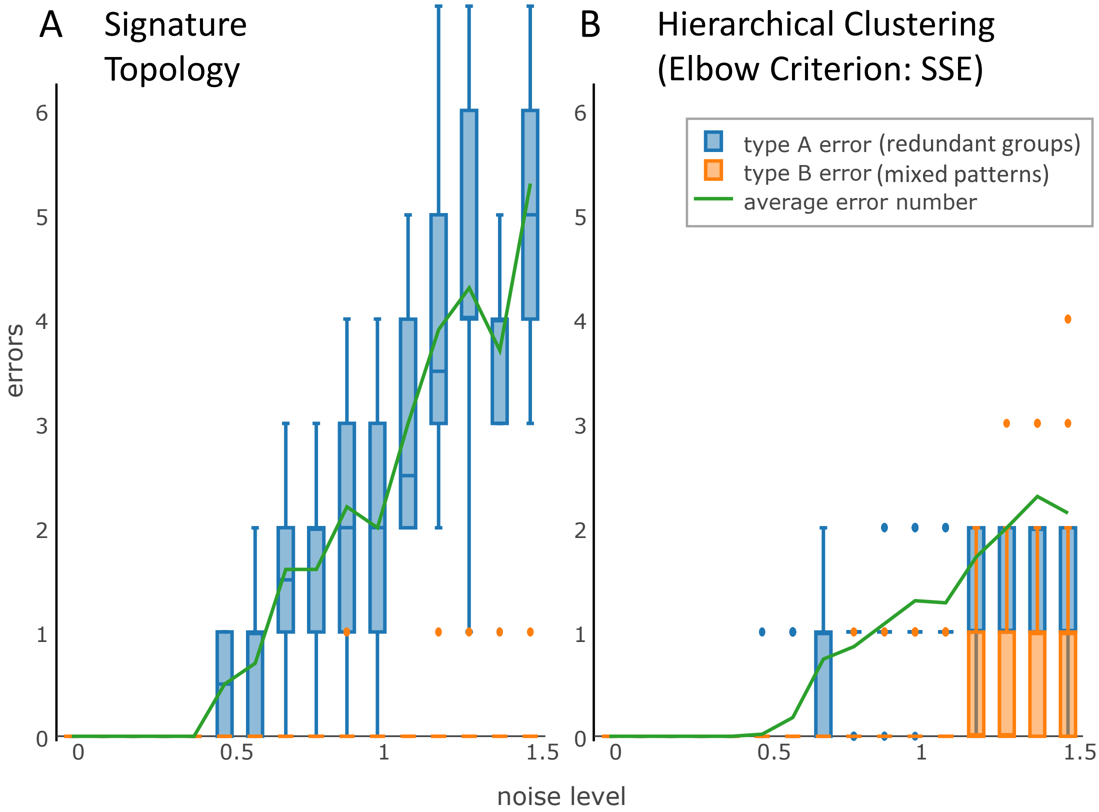


Figure 4.2: Noise robustness trial. Number of errors in constructing ST tree compared to optimal structure with increasing noise level σ. Error A (blue) means number of excess groups exclusive of two correct groups. Error B (orange) means number of elements of incorrect pattern within mixed groups, if such groups were created. Also an average of both error types (type A + type B) for each noise level is shown (green line). For each noise level number of elements $n = 10$. The number of errors were summed over 10 repetitions of the trial. **A.** Trial results for Signature Topology structures. **B.** Trial results for Hierarhical Clustering structures with number of clusters, chosen by elbow criterion based on sum of squared errors (SSE).

individual elements in their own groups. It can be considered as an error of type A (redundant groups). Also, at some noise level, elements with different patterns cannot be discerned anymore and then they can be assigned to the same group in ST. It can be considered as an error of type B (mixed patterns).

The results of this setup trial can be seen in the Figure 4.2, A. Up to noise level $\sigma = 0.5$ no errors appeared. From $\sigma = 0.6$ to $\sigma = 1.2$ an increased count of type A errors took place, whereas type B errors appeared only after high level of noise $\sigma = 0.9$.

The same dataset was clustered with the standard Hierarchical Clustering approach. Similar to the distance measure for the ST method (see Section 2.4.1), the complete linkage criterion was used. The optimal number of clusters is chosen with the standard elbow criterion based on SSE (Sum of Squared errors) index. In Figure 4.2, B, the results are presented. The number of redundant groups (error type A) is lower than for the ST approach, but errors type B (mixed patterns) occur at lower noise level and reach higher numbers. As the type B errors are responsible to the loss of information, vital to connection between kinetic patterns and functional role of the molecules in the organism, these errors are considered more crucial and, thus the ST approach is more noise robust then Hierarchical Clustering in terms of preserving the connection between experimental behavior and functional annotation.

4.3 Synthetic dataset: Minority revealing

To check the Signature Topology approach for the desired property of revealing the minor patterns, a dummy ontology structure was created. All elements of the dataset were assigned to one functional group without more specific information about their roles in the biological process. It corresponds to the tree structure with only one node, a root, with all elements inside. For simulating an existence of minority pattern, in the dataset some percentage of items is substituted by elements with opposite pattern, than the majority. With varying noise level σ, it is checked whether the proposed algorithm can reveal the injected elements within the original structure and expose them in newly created clusters on the level 1 in tree structure (see Figure 4.3).

The ratio of injected minor elements varied from 0.1, that corresponds to 9 elements with major pattern and 1 element with minor pattern, to 0.5, where up-regulated and down-regulated patterns are balanced and have 5 elements each. As was mentioned, error type B, number of elements with wrongly assigned patterns, is more crucial error for minority revealing trial, that is why this error will be in focus. The described trial was repeated 10 times for each approach: Signature Topology and Hierarchical Clustering with optimal cluster number determined by elbow criterion with SSE (sum of squared errors). The total number of type B errors for

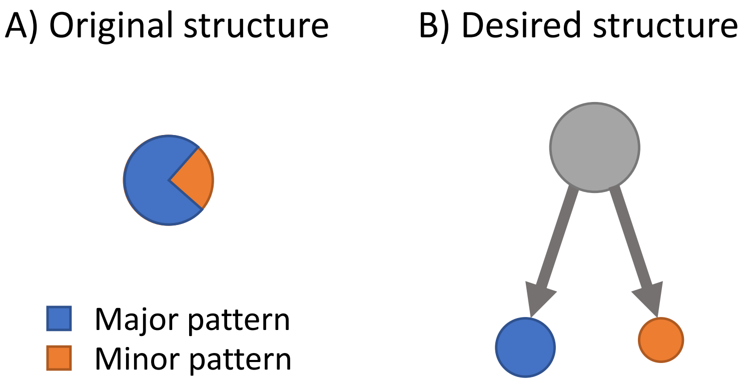


Figure 4.3: Minority revealing trial setup. **A.** Original structure with hidden minor pattern among the elements with the major pattern. Blue and orange colors represent the presence of elements with different kinetic patterns. **B.** Desired structure with two groups separating elements of minor and major patterns.

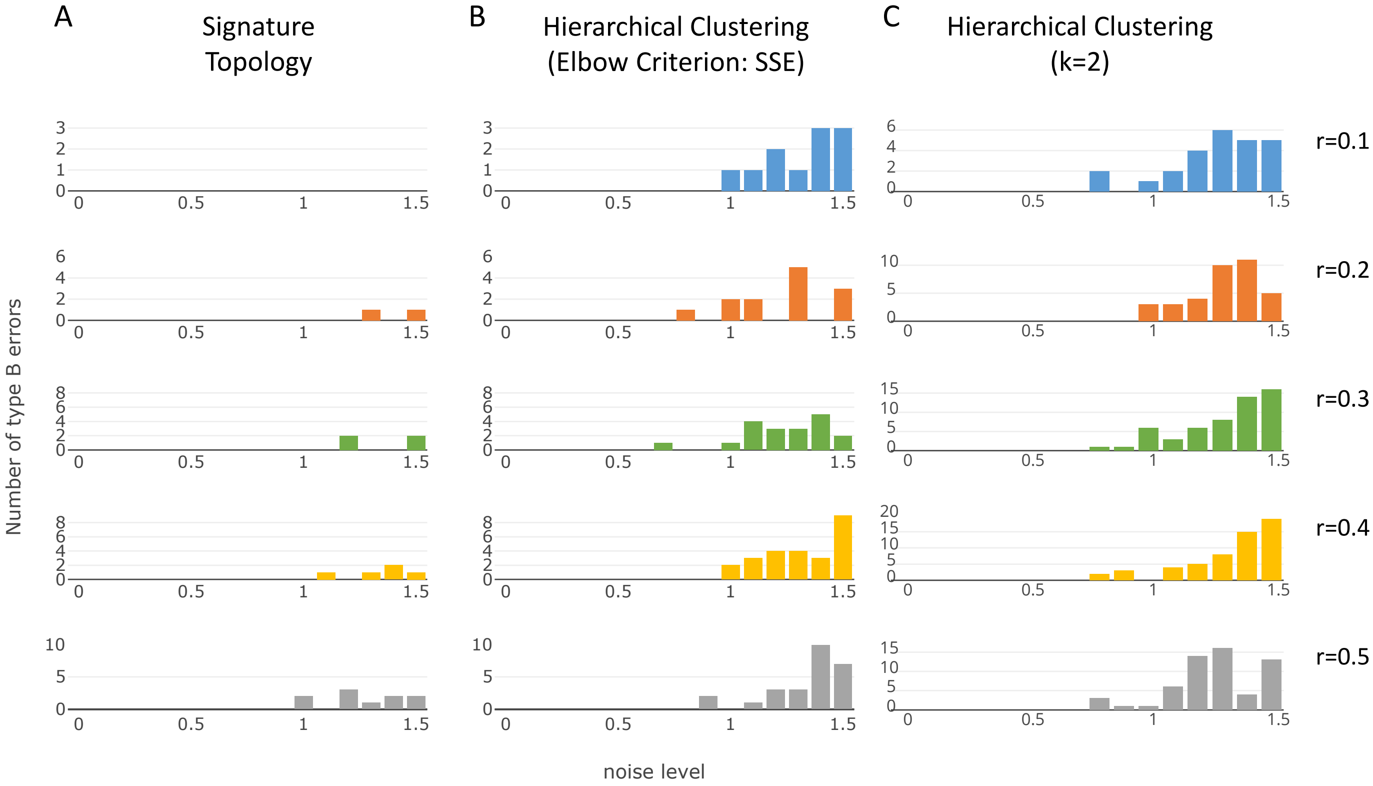


Figure 4.4: Minority revealing trial - Number of type B errors, considering 5 different ratios of minor pattern, comparing obtained with three different approaches structures to the optimal structure with increasing noise level σ. Type B error means a number of elements of minority pattern in mixed groups, if such groups were created. Noise level varied as $\sigma = \{0.0, 0.1, 0.2 \ldots 1.5\}$, minority ratio varied as $r = \{0.1, 0.2, 0.3, 0.4, 0.5\}$. For each parameter combinations, number of elements in the dataset is $n = 10$. The number of errors were summed over 10 repetitions of the trial. **A.** Signature Topology. **B.** Hierarchical Clustering with the number of clusters determined with Elbow criterion by SSE. **C.** Hierarchical Clustering with the given number of clusters $k = 2$.

each setup is shown in Figure 4.4.

Figure 4.4 shows that for all approaches, that despite varying minority level, up to noise level 0.5, all minority elements were revealed without mixing errors. The ST approach could avoid mixing errors for all minority ratios up to noise level $\sigma = 1$, especially good for minority ratio $r = 0.1$. The Hierarchical Clustering approach (with the same parameters as in the previous setup: complete linkage criterion, SSE index) showed poorer results. Interestingly, if to fix the number of clusters to the optimal $k = 2$, it did not improve the outcome (Figure 4.4, C), moreover, the number of errors doubled.

This trial showed that the proposed approach is suitable for revealing minority groups at least if they are 10% of the whole group and shows better results than the classical approach.

4.4 Experiment dataset: Proteome

To analyze the relation between the proteins, the ST approach was applied to elements from the proteome dataset from each functional group (MapMan bin) separately. First, we looked at the resulted structure for individual bins 9 and 1 and then made an overview for all bins in the whole proteome dataset. Bin 9 is convenient for the representative sake, as it has quite complex functional ontology structure (5 depth levels) yet not so many elements (21) so that the benefits of the ST structure can be easily explained. Bin 1 is larger (5 depth levels, 63 elements), so the comparison of the original and the ST structure is more complicated, however the corresponding functional group (Photosynthesis) is well studied and therefore offers possibilities to look at certain elements in more details that we will do in this and following chapters.

4.4.1 Bin 9: ST structure and grouping are concise

As a first illustration of the implementation of the Signature Topology approach, a part of proteomic dataset was considered, with proteins that were identified as belonging to a functional group "mitochondrial electron transport / ATP synthesis" (MapMan bin 9). The kinetic time-series of the identified proteins from bin 9 can be seen on Figure 4.5. The proteins kinetic is split into 3 plots for the sake of convenience to distinguish patterns. The original MapMan ontology, corresponding to the identified proteins, and the resulting ST tree can be seen on Figure 4.6. Color scheme on both figures is consistent. Notice, that there are no elements from subbins 9.3, 9.4 or 9.8, as no proteins from these functional subgroups were measured during the experiment.

The original MapMan ontology has 5 levels, with a very different distribution of kinetic patterns over the nodes on different levels. The obtained tree structure, after applying the ST approach, has merged elements, that could be combined without splitting functional groups, and newly divided groups, that are necessary to show the difference in kinetic even for elements from the most specific functional subbins. As an example of the former, is a subbin 9.5|6, that is a merge of functional subbins 9.5 and 9.6, marked orange in the middle section of Figure 4.6. In Figure 4.5, middle section, one can notice the similarity in kinetic patterns between orange elements. An example of the latter is shown in the upper section in Figure 4.6, it is a subbin 9.9, that was split in 2 additional groups: singleton 9.9.p2 and merged group 9.9.mix|p14|p16|p6|p15, whose different kinetic patterns can be seen in Figure 4.5. But even if the ST structure is justified for the observer, it is difficult to compare the original MapMan and the resulted ST structure, simultaneously keeping in mind the kinetic behavior. Therefore an objective and automated quality measure is required to compare the tree structures resulting from the ST

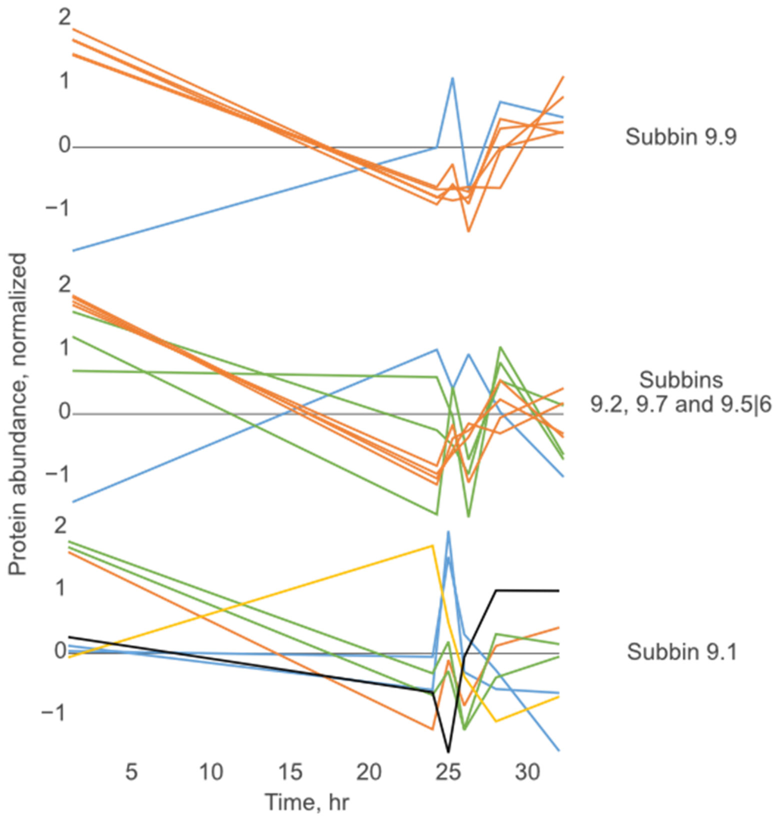


Figure 4.5: Proteomics for bin 9: Time series of proteins. Color code is consistent to Figure 4.6.

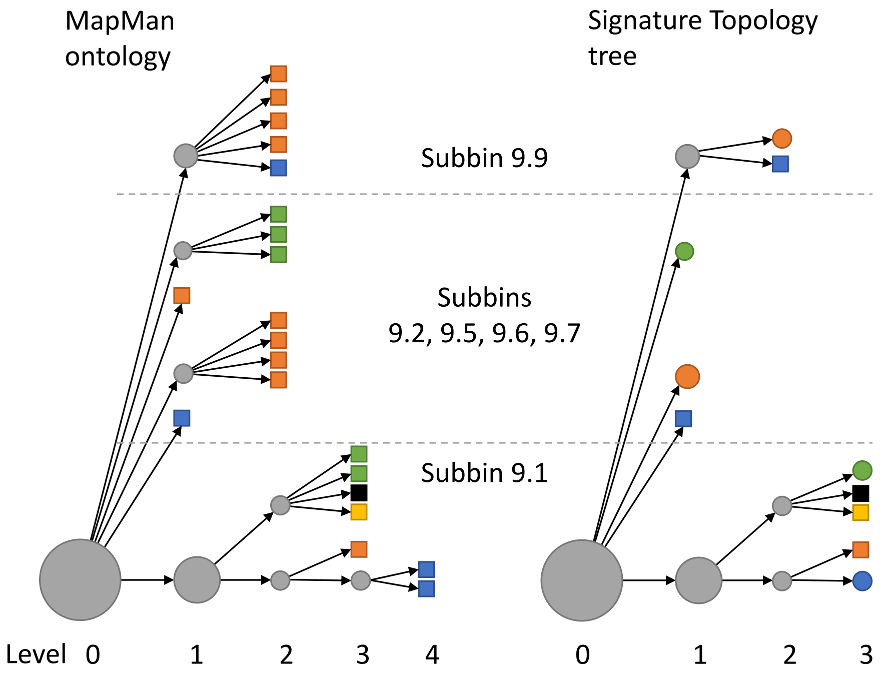


Figure 4.6: Proteomics for bin 9: MapMan ontology and Signature Topology tree. For convenience of coloring the structure is split to 3 groups. Color code is consistent to Figure 4.5. Circles are nodes with multiple elements inside, squares are singletons.

approach and the original tree.

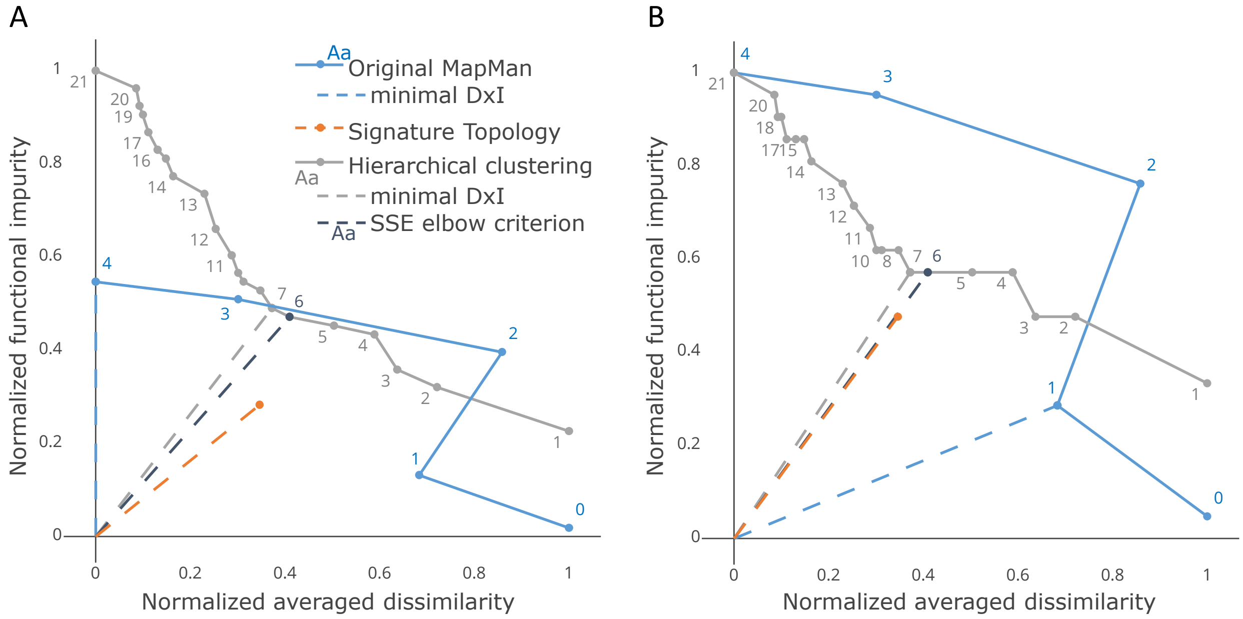


Figure 4.7: Proteomics for bin 9: DxI index for MapMan trees cut at different levels (blue line, label 0 means only root, label 1 means the root and its direct children etc), ST tree (orange point), and HC clusterings extended into functional trees (gray line with dark gray point for HC structure with optimal number of clusters according to elbow criterion with SSE). Dotted lines represent the minimal DxI indices of the corresponding structures: the shorter the line, the smaller the DxI index, meaning the structure is better balanced and has both functional purity and kinetic similarity in groups. **A.** Impurity is assessed as structure complexity, i.e. number of all functional subgroups in the structure, i.e. nodes in a tree. **B.** Impurity is assessed more practically meaningful, as grouping complexity, i.e. number of terminal functional subgroups in the structure, i.e. leaves in a tree.

As there are 2 main parameters to consider, purity of the grouping structure (defined by functional ontology) and similarity in groups (defined by kinetic patterns), it is not easy to observe a structure considering both parameters. For this reason, a special DxI index (Dissimilarity vs. Impurity) was introduced as a graphic representation of both parameters. These two parameters are formulated as normalized **average dissimilarity (D)** within a node for all leaves (or clusters for the compared method of Hierarchical Clustering) in a structure as a kinetic parameter on X axis; and as a normalized **number of functional subgroups as an impurity parameter on Y axis (I)**. Then the distance of the resulted point (dissimilarity, impurity) to the point of origin on the plane is a value of the given DxI index for an observed tree structure:

$$\begin{aligned} DxI &= \sqrt{(D/D_{max})^2 + (I/I_{max})^2} \\ D &= \frac{\sum d_{leaf}}{n_{leaves}} \\ I &= n_{functional\ subgroups} \end{aligned} \tag{4.2}$$

where d_{leaf} is a maximal pairwise distance within a leaf, and the normalization is done by dividing each component D and I over their maximum value from all structures for the current functional group.

There are two different ways to estimate the functional heterogeneity of the resulted structure of biological molecules (the I component in DxI index). One way is to consider the whole structure complexity - all nodes in the resulted tree, see the corresponding DxI plot in Figure 4.7, A. Another way is to consider only grouping complexity. That would include only functional subbins, that will be used for further analysis, meaning those, that mark the terminal nodes of the structure: count only leaves of a tree, see the corresponding DxI plot in Figure 4.7, B.

For the original MapMan Ontology (MM) structure, the DxI index is calculated for each depth level, considering terminal nodes as if the tree is truncated at this level. For Signature Topology structure we get exactly one point, as the structure is one-way defined here. And for comparison, a dendrogram with hierarchical clustering (HC) was built, and corresponding DxI indices were calculated for structures, obtained by cutting the dendrogram at different levels (considering different number of clusters k) and extending the grouping into tree with corresponding functional hierarchy.

First, look at the DxI index considering the structure complexity (Figure 4.7, A). Here we see, that MapMan structures have low impurity, but high dissimilarity, that results in the best $DxI_{MM4} = 0.55$. Hierarchical Clustering has a minimal DxI index at the structure with number of clusters $k = 7$ equals 0.62. Interestingly, the optimal prediction by using the elbow criterion (based on squared sum of errors (SSE)) corresponds to $k = 6$ and has similar DxI value. The ST structure has the smallest $DxI_{ST} = 0.45$, meaning it has the most concise structure with maximum functional purity and kinetic similarity.

Second, look at the DxI index considering the grouping complexity (Figure 4.7, B). Here we see, that by definition, the impurity values are the same for the deepest MapMan structure and the most detailed HC structure, because it is limited by number of elements in the structure. This way, the most balanced MapMan structure is on the level 1 with $DxI_{MM1} = 0.74$. The best HC structure is the same with number of clusters $k = 7$ and

$DxI_{HC7} = 0.68$. The ST structure has still the best $DxI_{ST} = 0.59$, meaning it has the most concise grouping with maximum functional purity and kinetic similarity in terms of this DxI index, too.

As the goal of a grouping approach is to preserve the functional relations (minimize the impurity) and include the current experiment kinetic relations (minimize the dissimilarity), the ST structure gives the best result.

4.4.2 Bin 1: Calvin Cycle potential regulatory protein is revealed

As the next step, proteins from functional group Photosynthesis (MapMan bin 1) were analyzed and though there is no gain in structure complexity for the ST structure over MapMan structure (see Figure 4.8, A: $DxI_{MM4} = 0.36$, $DxI_{ST} = 0.45$), it is still better than the best ST structure in both terms DxI index or SSE ($DxI_{HC13} = 0.54$ and $DxI_{HC6} = 0.57$)). For grouping complexity, however, the ST structure is still the best combination of functional purity and kinetic similarity (see Figure 4.8, B: $DxI_{ST} = 0.60$, $DxI_{MM2} = 0.74$, and $DxI_{HC11} = 0.76$). Moreover, for some specific branches of the tree, the advantage, that the ST structure has over MapMan tree, is more prominent.

One of these examples is a Calvin Cycle functional subgroup. A subtree starting with the corresponding node with functional subgroup Calvin Cycle (MapMan subbin 1.3) is considered more thoroughly. In the original MapMan structure (Figure 4.9, A) on the depth level 2 there are 9 groups, that have partly redundant information (several leaves with similar pattern represented by the same color) and partly impure (a leaf with black-orange coloring has proteins of 2 patterns inside). In the resulting ST structure, there are 5 groups on levels 2 and 3, that allow us to see pure and sufficient information about three patterns on level 2 and 2 patterns on level 3. Note, that the elements with similar patterns on the levels 2 and 3 (Figure 4.9, B, black and green, orange and yellow lines) could not be merged together, because that would lead to functional information getting lost.

4.4.3 Proteome overall performance: better functional purity and kinetic similarity for ST structure and resulted grouping

The ST approach was applied to the whole proteome dataset, separately for each functional group of the MapMan ontology. For each functional group (MapMan bin), the DxI plot was created and DxI indices for each structure were calculated and then plotted as histograms. Both approaches of Impurity estimation were considered: as structure complexity (Figure 4.10) and as grouping complexity (Figure 4.11). For each functional group there were taken four DxI indices: unique DxI index of ST tree (orange in both figures), the minimal DxI index over

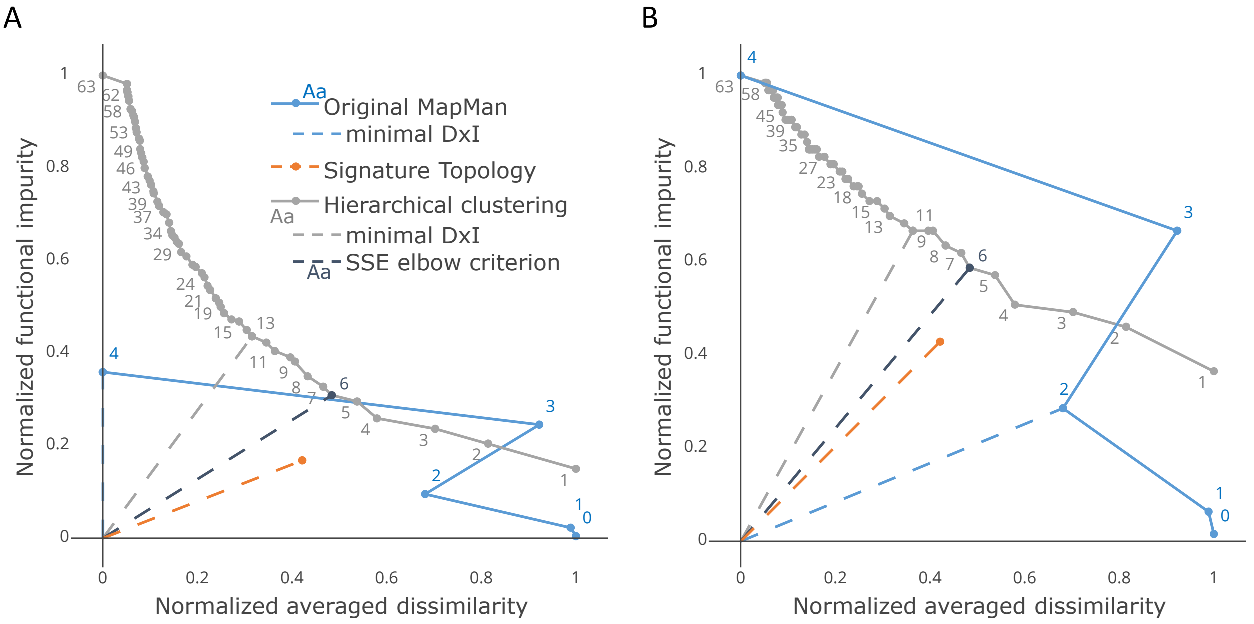


Figure 4.8: Proteomics for bin 1: DxI index for MapMan trees cut at different levels (blue line, label 0 means only root, label 1 means the root and its direct children etc), ST tree (orange point), and HC clusterings extended into functional trees (gray line with dark gray point for HC structure with optimal number of clusters according to elbow criterion with SSE). Dotted lines represent the minimal DxI indices of the corresponding structures: the shorter the line, the smaller the DxI index means the structure is better balanced and has both functional purity and kinetic similarity in groups. **A.** Impurity is assessed as structure complexity, i.e. number of all functional subgroups in the structure, i.e. nodes in a tree. **B.** Impurity is assessed more practically meaningful, as grouping complexity, i.e. number of terminal functional subgroups in the structure, i.e. leaves in a tree.

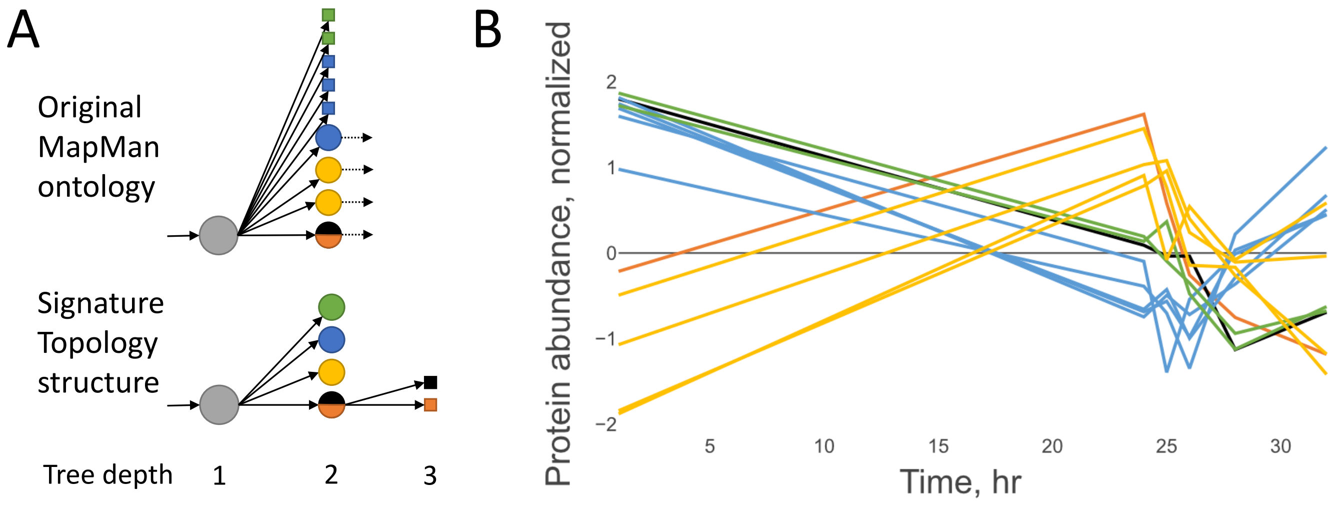


Figure 4.9: Proteomics of subbin 1.3. **A.** MapMan ontology and Signature Topology tree. The gray node is the subbin 1.3. Circles are nodes with multiple elements inside, squares are singletons. **B.** Time series of proteins from subbin 1.3. Coloring scheme is based on the ST leaves grouping.

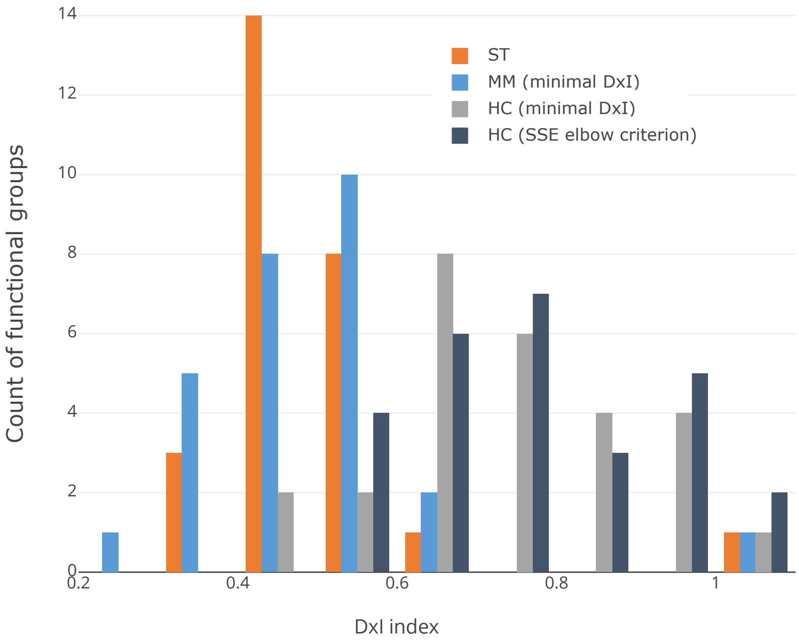


Figure 4.10: Proteomics overall performance for DxI index of structure complexity: For each functional group (MapMan bin) there were taken four DxI indices: of unique ST tree (orange, median 0.47), the minimal one of MapMan trees (blue, median 0.49), the minimal one of HC extended trees (light gray, median 0.72) and one of HC with optimal number of clusters according to SSE elbow criterion (dark gray, median 0.72). The impurity is assessed as number of all functional subgroups in the structure, i.e. nodes in a tree.

MapMan trees cut at different levels (blue in both figures), and two DxI indices from HC groupings extended into trees: one with minimal DxI index (light gray in both figures) and one with optimal number of clusters according to SSE elbow criterion (dark gray in both figures). From all 688 proteins, 550 were mapped to 27 MapMan functional groups, that were analyzed. Note, that to be analyzed, the group should have a size bigger than 2 proteins.

As observed the DxI distributions are not normal, non-parametrical Wilcoxon signed ranked test for paired sampling was used for statistical analysis. As can be seen from the DxI histograms, the ST structures have the smallest median DxI indices, compared to both original MapMan trees cut at any level, or classical HC grouping including functional information for both impurity estimations. For structure complexity of the impurity estimation (Figure 4.10), the difference between DxI indices for ST structure and both HC structures were found significant (p-values < 0.05). For grouping complexity of the impurity estimation (Figure 4.11), it was also the case for ST structure vs HC structures, as well as for ST structure vs MapMan structure.

All DxI indices and statistical test results are listed in Table A.1 in the Appendix.

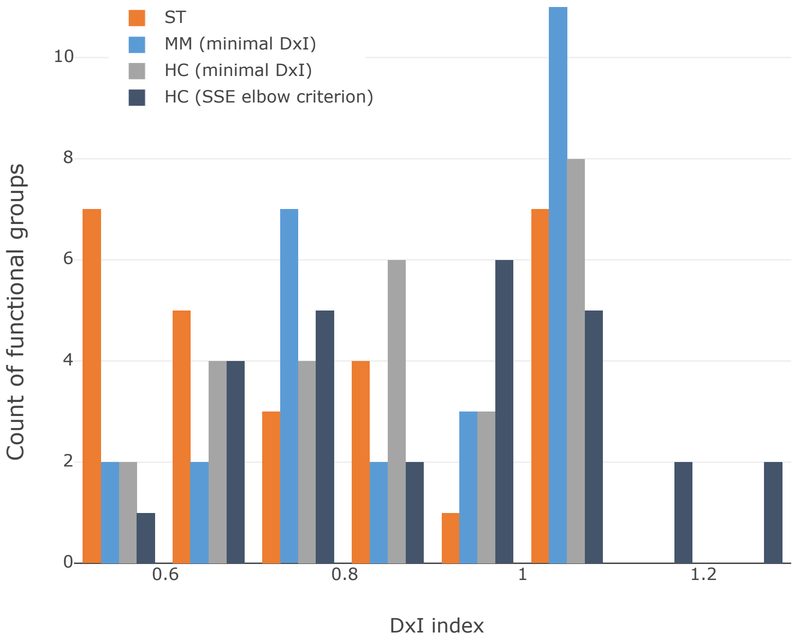


Figure 4.11: Proteomics overall performance for DxI index of grouping complexity: For each functional group (MapMan bin)there were taken four DxI indices: of unique ST tree (orange, median 0.75), the minimal one of MapMan trees (blue, median 0.93), the minimal one of HC extended trees (light gray, median 0.85) and one of HC with optimal number of clusters according to SSE elbow criterion (dark gray, median 0.91). Impurity is assessed more practically meaningful, as number of terminal functional subgroups in the structure, i.e. leaves in a tree.

4.5 Experiment dataset: Transcriptome

4.5.1 Bin 9: ST grouping is concise

As the first illustration of the implementation of the Signature Topology approach on a more complex and larger transcriptome dataset, we first considered genes that were identified as assigned to the same functional group "mitochondrial electron transport / ATP synthesis" (MapMan bin 9) as was illustrated in the first proteome example. The original MapMan ontology and the resulting ST tree can be seen on Figure 4.12. The kinetic time-series of the corresponding genes can be seen on Figure 4.13 split into 8 plots, A-H, for the sake of convenience to distinguish patterns. The color scheme on both figures is consistent.

On the first glance, it is clear, that the resulted ST structure is much more simple, than the original MapMan ontology. The MapMan tree has 5 levels with functional subbins (nodes) very different in size and kinetic similarity even on the same depth level. The ST structure has also different sizes of the terminal nodes, but the similarities of them are much higher (see Figure 4.13). But again, it is difficult to keep in mind both functional complexity and kinetic similarity simultaneously, just looking at the kinetic plots and ontology trees. That is why it is convenient to look at DxI indices.

The DxI indices are calculated the same way as described in the first example for proteome data by formula (4.2). Analogously, two DxI planes were created with two impurity estimation approaches (see Figure 4.14). The DxI indices for the original MapMan ontology were calculated for each tree, resulted by cutting the original ontology at different levels, then the minimal DxI index was shown on the plot. The Signature Topology approach gives the unambiguous structure, hence it is described by only one DxI index. As a comparison for the classical clustering approach, a dendrogram with Hierarchical Clustering (HC) was built. The DxI index was calculated for each k number of clusters by cutting the dendrogram at the corresponding level and extending the resulted grouping into a tree using the functional hierarchy from the MapMan ontology.

First, look at the DxI index considering the structure complexity (Figure 4.14, A). The ST structure with $DxI_{ST} = 0.48$ has not the best value in this case, giving way to the most detailed MapMan structure

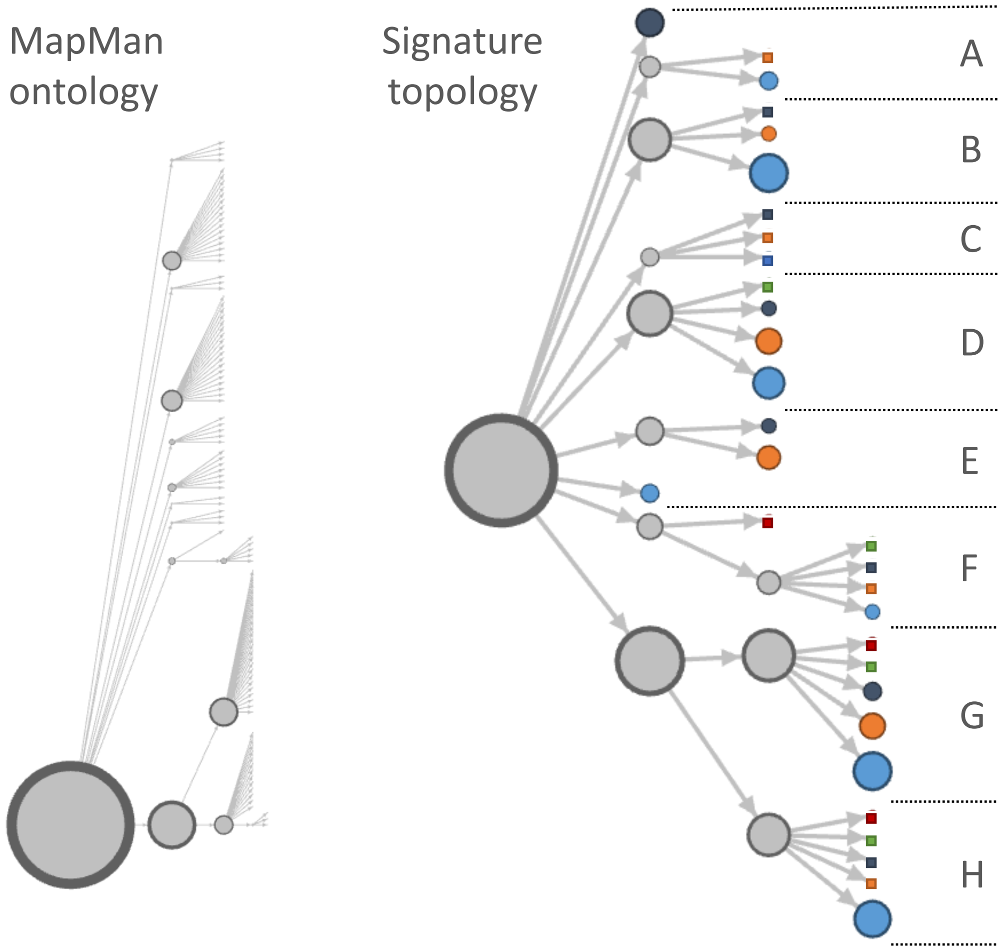


Figure 4.12: Transcriptomics for bin 9: MapMan ontology and ST tree. For convenience of coloring the structure is split into 8 groups, **A-H.** Circles are nodes with multiple elements inside, squares are singletons.

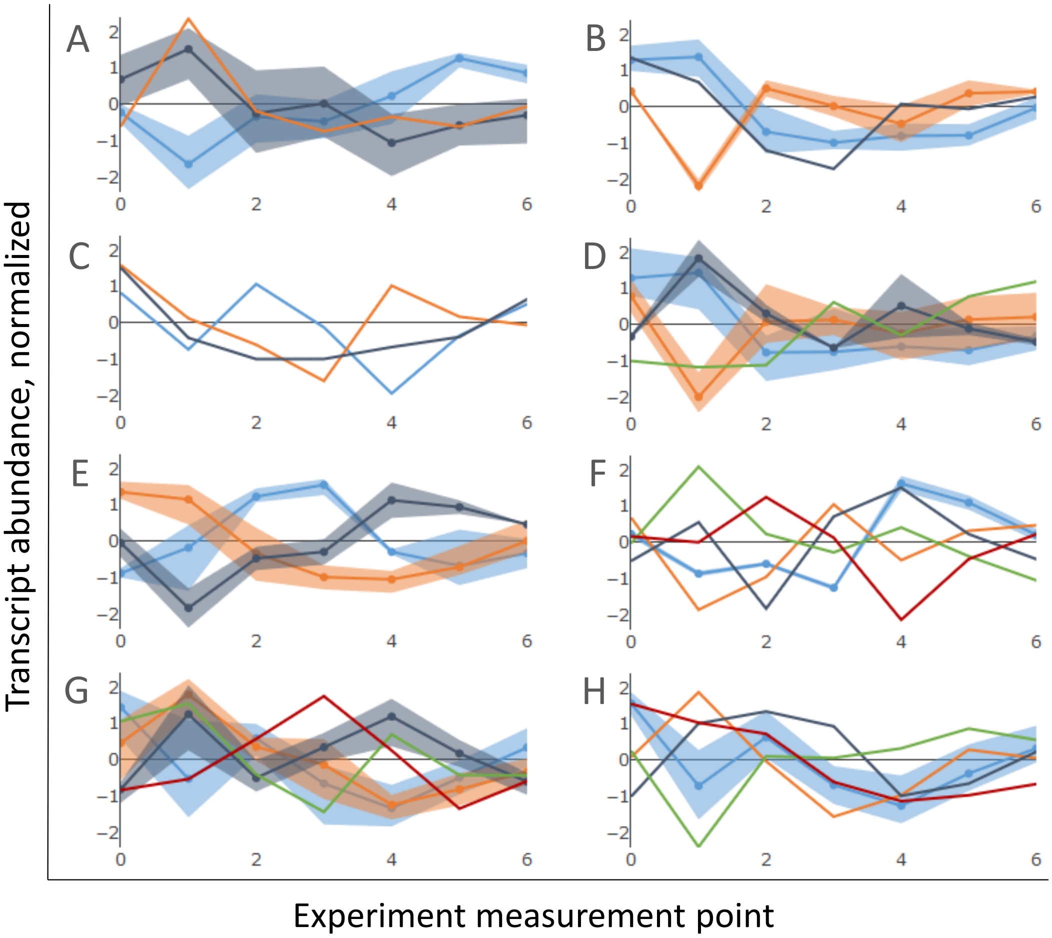


Figure 4.13: Transcriptomics for bin 9: Time series of proteins. Area shapes represent the maximal/minimal range of signal within a group. Color code is consistent to Figure 4.12.

($DxI_{MM4} = 0.46$) and being slightly better than any HC structures (minimal $DxI_{HC19} = 0.49$, SSE optimal $DxI_{HC17} = 0.51$). When looking at the DxI plane, it is obvious, that the MapMan structure won by neglecting the functional grouping (maximum impurity, meaning considering each element separately without grouping at all). This way, the ST grouping is more empirically meaningful for further analysis.

To prove it, look at the DxI index considering the grouping complexity (Figure 4.14, B). Here we see, that by definition, the impurity values are the same for the deepest MapMan structure and the most detailed HC structure, because it is limited by number of elements in the structure. This way, the most balanced MapMan structure is on the level 1 with $DxI_{MM1} = 0.83$. The best HC structure is the same with number of clusters $k = 19$ and $DxI_{HC19} = 0.58$. From this point of view, the ST structure has obviously the best parameter $DxI_{ST} = 0.54$, meaning it has the most concise grouping with maximum functional purity and kinetic similarity in terms of this DxI index.

As the goal of a grouping approach is to represent the experimental data while incorporating the known

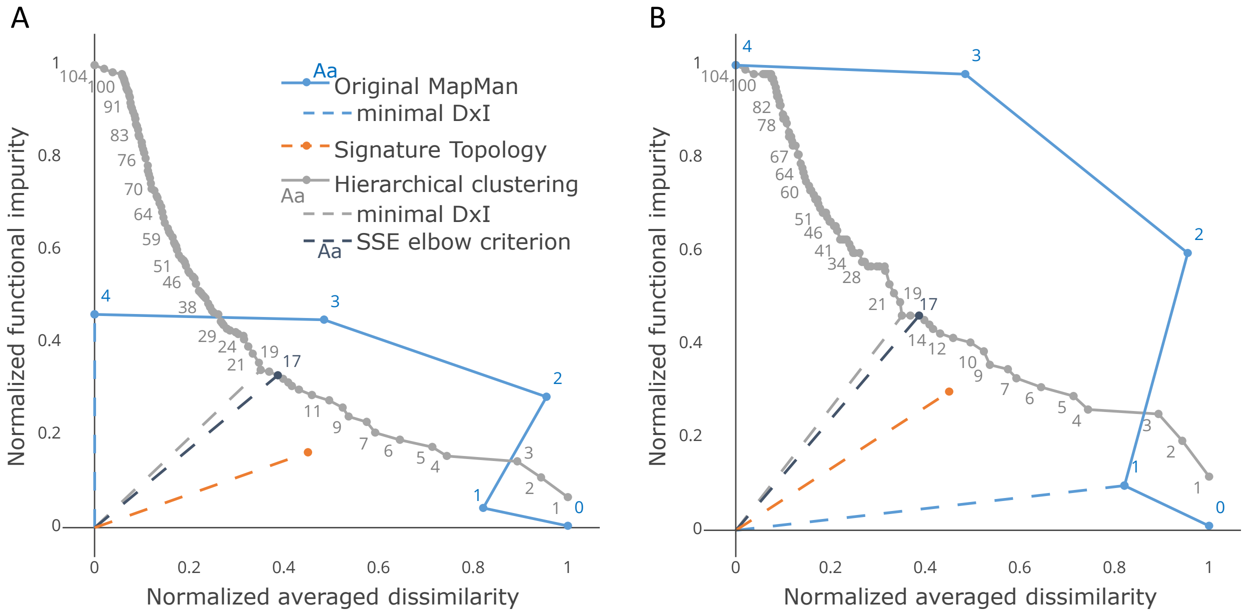


Figure 4.14: Transcriptomics for bin 9: DxI index for MapMan trees cut at different levels, ST tree, and HC clusterings extended into trees. Dotted lines represent the minimal DxI indices of the corresponding structures: the shorter the line, the smaller the DxI index means the structure is better balanced and has both functional purity and kinetic similarity in groups. **A.** Impurity is assessed as number of all functional subgroups in the structure, also nodes in a tree. **B.** Impurity is assessed more practically meaningful, as number of terminal functional subgroups in the structure, also leaves in a tree.

functional relations (minimize the impurity) and including the current experiment kinetic relations (minimize the dissimilarity), the ST structure gives the best result.

4.5.2 Bin 29: The ST grouping of the largest group could reveal single regulatory genes

The next step, the transcripts from the biggest functional group Ribosome Proteins (MapMan bin 29, 1418 elements) were analzyed and as can be seen from Figure 4.15 the ST structure is obviously conciser and simpler than MapMan ontology. Also in terms of grouping impurity and dissimilarity (see Figure 4.16, B), the DxI index for the ST structure ($DxI_{ST} = 0.58$) gives better results than the best MapMan structure ($DxI_{MM2} = 0.93$) or Hierarchical clusterings ($DxI_{HC91} = 0.67$). The DxI index for structure complexity (see Figure 4.16, A) does not favor the ST structure, because due to a high complexity and a large number of elements in the bin, the possible maximum of impurity (for HC structure with number of clusters $k = 1418$) makes the normalization insensitive to low impurity of the ST structure compared to MapMan and optimal HC structures. It makes an input from the similarity component weight more than from the purity component, that is why the ST structure looks non-balanced for these impurity assessment approach.

Analogously to the proteome example, let us look into details of the complex structure of the bin 29. There is a node with functional subgroup "Ribosomal proteins of chloroplasts, subunit 30S" (MapMan subbin 29.2.1.1.1.1), that was considered more thoroughly. In the original structure all 10 genes of the node are grouped either together in this node at the level 5 of the MapMan tree or each gene is separated in its own singleton one step deeper at the level 6 (Figure 4.17, A).

However, if take a look at the kinetic behavior of the genes (see Figure 4.17, B), one can see 3 different patterns of behavior, with black and blue colored patterns being similar. Note, that the ST approach could recognize all 3 patterns, even with 2 of them being so close in behavior and that for black and orange patterns there are only 1 element for each pattern. As an output of the implemented ST algorithm, there is obtained

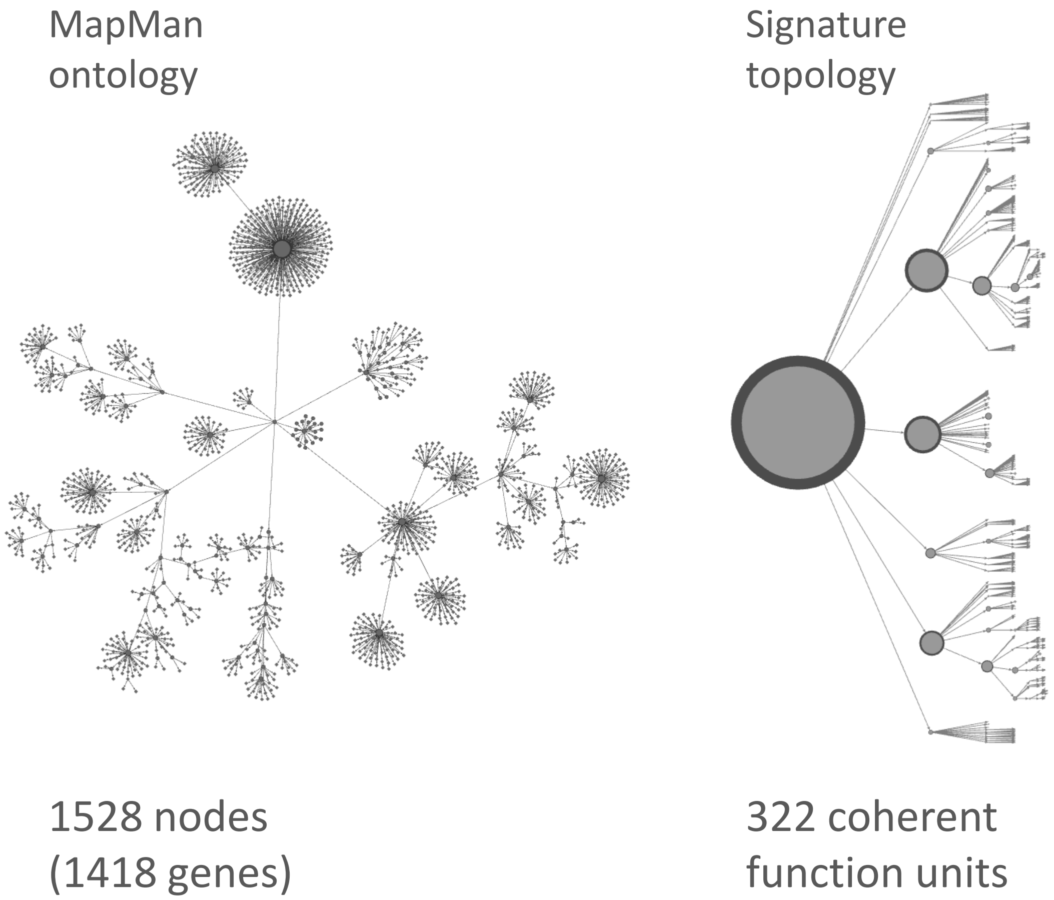


Figure 4.15: Transcriptomics for bin 29: the original MapMan ontology and the Signature Topology tree. The MapMan structure, despite being the tree as in previous examples, is shown as a general graph for the illustration reasons.

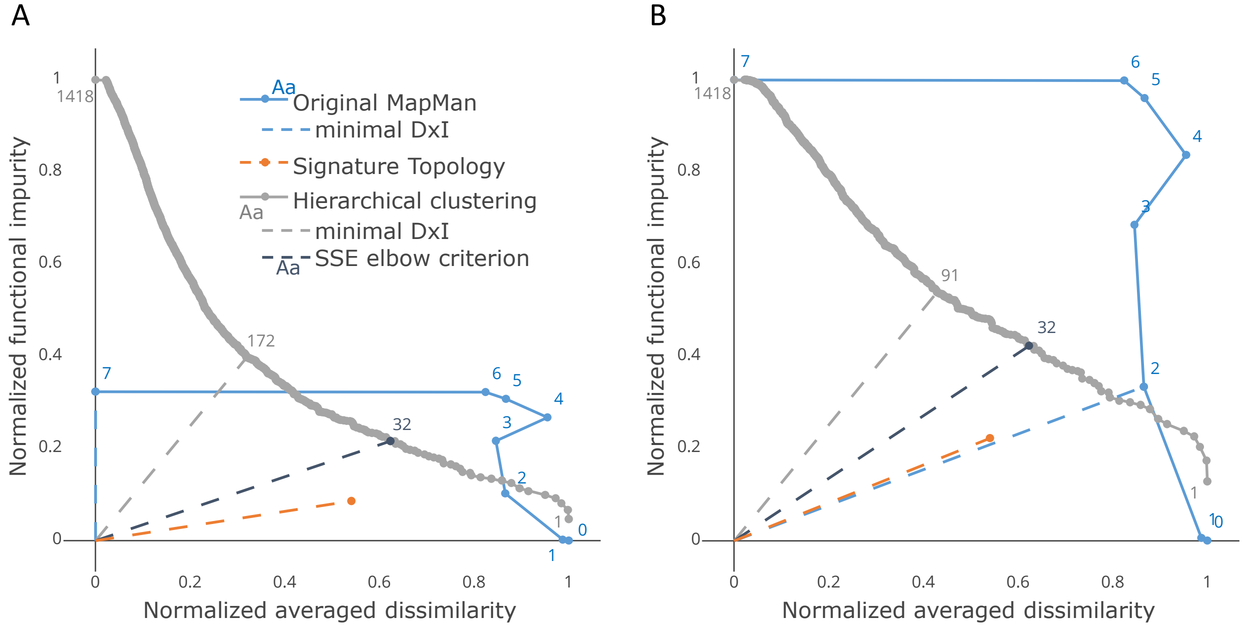


Figure 4.16: Transcriptomics for bin 29: DxI index for MapMan trees cut at different levels, ST tree, and HC clusterings extended into trees. Dotted lines represent the minimal DxI indices of the corresponding structures: the shorter the line, the smaller the DxI index means the structure is better balanced and has both functional purity and kinetic similarity in groups. A. Impurity is assessed as number of all functional subgroups in the structure, also nodes in a tree. B. Impurity is assessed more practically meaningful, as number of terminal functional subgroups in the structure, also leaves in a tree.

an ordered list of resulted terminal nodes of the ST tree, where genes MRPS11 (Cre06.g288400) and PSRP1 (Cre05.g237450), that were isolated as singletons from the majority pattern in the parent node, would be reported on top of the list with high priority (see Supplementary Table A.5), whereas the group with the other 8 elements will be in the middle of the list.

Look more closely at the output list (Supplementary Table A.5): it has 2 additional columns, that can be used for highlighting the relevant for the researchers properties of the groups: (i) cluster density that shows how similar elements are within the group, and (ii) step gain from the root calculated as defined in (2.4) that shows how different the elements in the group are from all elements in the MapMan Ontology functional bin. If the former is a more common parameter and it shows the densest groups, whose elements have the most similar kinetic behavior, this parameter is useless for singletons – it will be always equal to zero. The latter parameter is suitable for comparing singletons as well, and it shows the most deviant group, compared to the elements with similar functionality. This parameter can help to focus on finding the regulators, such as isolated genes in this example.

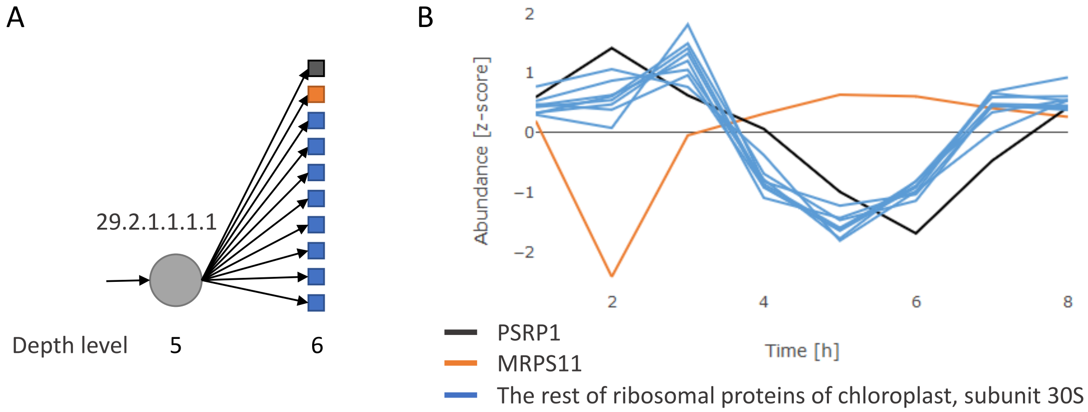


Figure 4.17: Transcriptomics for subbin 29.2.1.1.1.1: Ribosomal Proteins of chloroplast, subunit 30S. Elements with the same color were grouped together in the resulted ST structure. **A.** Original MapMan subtree. **B.** Kinetic experimental data of elements in the subbin.

4.5.3 Transcriptome overall performance: better functional purity and kinetic similarity for ST structure and resulted grouping

Analogously to the proteome example, the ST approach was implemented to the whole transcriptome dataset, separately for each functional group from MapMan ontology (MapMan bin). For each functional group (MapMan bin), the DxI plot was created and DxI indices for each structure were calculated and then plotted as histograms. Both approaches of Impurity estimation were considered: as structure complexity (Figure 4.18) and as grouping complexity (Figure 4.19). For each functional group there were taken four DxI indices: unique DxI index of ST tree (orange in both figures), the minimal DxI index over MapMan trees cut at different levels (blue in both figures), and two DxI indices from HC groupings extended into trees: one with minimal DxI index (light gray in both figures) and one with optimal number of clusters according to SSE elbow criterion (dark gray in both figures). From all 5580 transcripts, 5522 genes were mapped to 27 functional MapMan groups, that were analyzed. Note, that to be analyzed, the group should have size bigger than 2 genes. As observed samples do not represent normal distributions, non-parametrical Wilcoxon signed ranked test for paired sampling was used for statistical analysis.

As can be seen from the distribution of DxI indices, calculated with the structure complexity for impurity measure (Figure 4.18), the best MapMan structures have the smallest median DxI indices, compared to both resulted ST structure, or classical HC grouping including functional information. As explained before it is due to decreasing influence of the impurity parameter because of high impact of the HC structure in the normalization. But even for this calculations, the difference between DxI indices for ST structure and both HC structures were

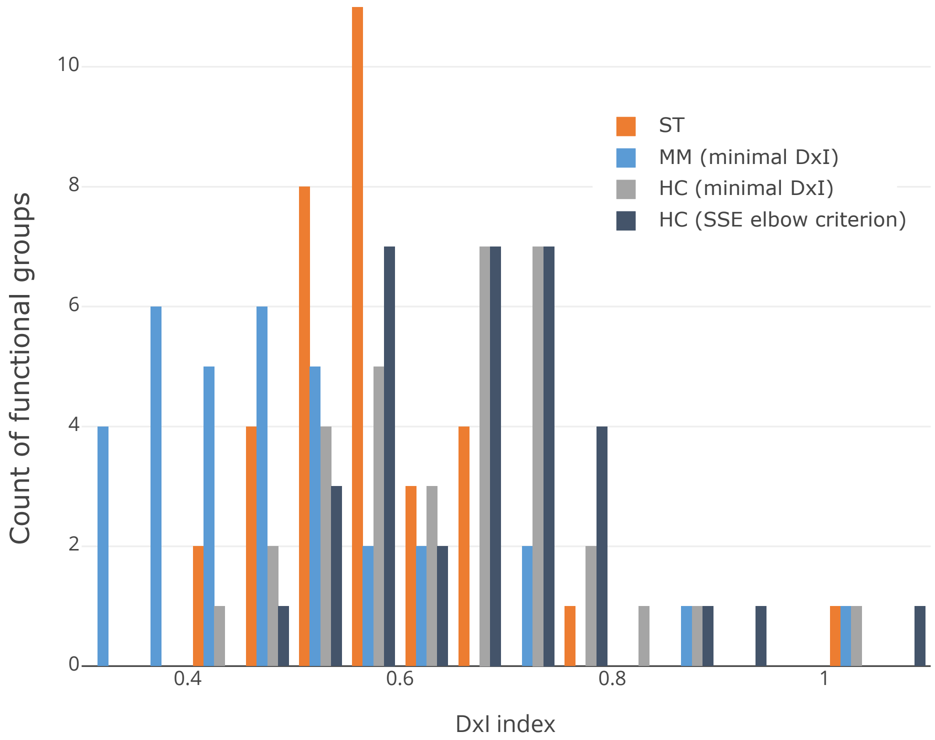


Figure 4.18: Transcriptomics overall performance for DxI index of structure complexity: For each functional group (MapMan bin) there were taken four DxI indices: of unique ST tree (orange, median 0.57), the minimal one of MapMan trees (blue, median 0.46), the minimal one of HC extended trees (light gray, median 0.66) and one of HC with optimal number of clusters according to SSE elbow criterion (dark gray, median 0.67). Impurity is assessed as number of all functional subgroups in the structure, i.e. nodes in a tree.

found significant (p-values < 0.05) showing that ST approach can create a structure being more functionally and kinetically pure.

For grouping complexity impurity estimation (Figure 4.11), it shows the ST structures have the lowest DxI index and the difference to other distributions being significant (p-value < 0.05), that proves, that the ST approach can provide grouping of elements, that balances the functional prior knowledge and incorporates the novel experimental data.

All DxI indices and statistical test results are listed in Table A.2 in the Appendix.

4.6 Inter-level Comparison

The integration of datasets from different system levels such as proteome and transcriptome can give information about differences and similarities of the biological processes on these levels, and the ST approach gives a suitable output as the resulting grouping sets, which can be compared between different experiments, system levels and

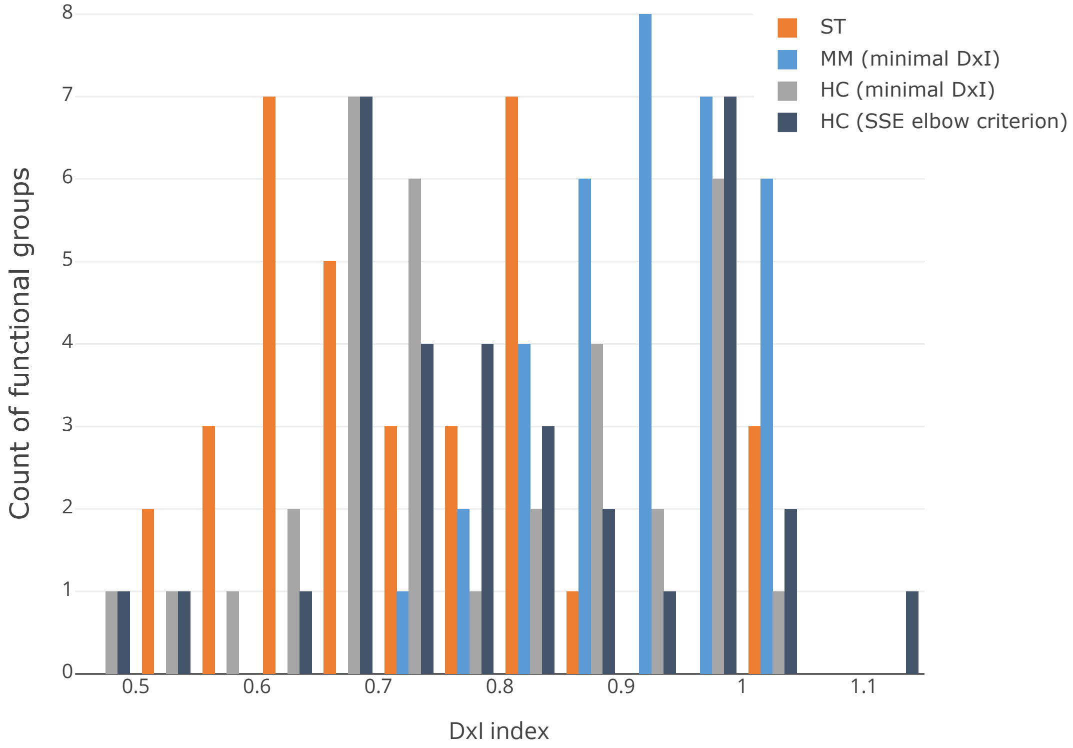


Figure 4.19: Transcriptomics overall performance for DxI index of grouping complexity: For each functional group (MapMan bin) there were taken four DxI indices: of unique ST tree (orange, median 0.71), the minimal one of MapMan trees (blue, median 0.92), the minimal one of HC extended trees (light gray, median 0.74) and one of HC with optimal number of clusters according to SSE elbow criterion (dark gray, median 0.78). Impurity is assessed more practically meaningful, as number of terminal functional subgroups in the structure, i.e. leaves in a tree.

organisms.

4.6.1 Direct element-wise Pearson correlation

An element-wise direct correlation of expression levels between proteome and transcriptome (Pearson correlation) is used to check the quality of ST grouping compared to the HC grouping (Figure 4.20). First, the common set of overlapping elements between proteome and transcriptome datasets should be determined, using the mapping information. With the used CRE identifiers for both datasets it is an easy task, as the protein and transcriptome labels are created identically. For differently identifying systems a mapping via e.g. sequence alignment is a possible solution.

As kinetic vectors for proteome and transcriptome experiments are different in length and exact time points, for gene-wise correlation there were chosen only 6 points from each kinetic vector. The first point is in the beginning of the experiment (1 hour and 0.5 hour for proteome and transcriptome respectively); the 5 others are the last 5 points from each vector, as they are coincided in both proteome and transcriptome experiment (see Table 3.2 and Figure 3.3).

The quality assessment is done as follows: (i) calculate the Pearson correlation coefficients for each protein-transcript pair from the common set (Figure 4.20, blue); (ii) calculate the Pearson correlation coefficients for only those protein-transcript pairs, that correspond to elements that were pair-wise grouped together in both transcriptome and proteome Signature Topology structures (Figure 4.20, orange) or hierarchical clustering (Figure 4.20, grey). This second part is a bit complicated, let us explain it in more details: Consider a functional bin, that has only 3 pairs in the common set: genes gA, gB and gC, and corresponding proteins pA, pB and pC. If in the proteome ST structure pA and pB are in the same leaf, and in the transcriptome ST structure gA and gB are in the same leaf (the label of the leaf is not necessary the same as for the proteome structure!), we include Pearson correlation coefficients for pairs gA-pA and gB-pB in the orange distribution. If gC and pC don't end up together with either gA and pA or gB and pB in the corresponding proteome and transcriptome ST structures,

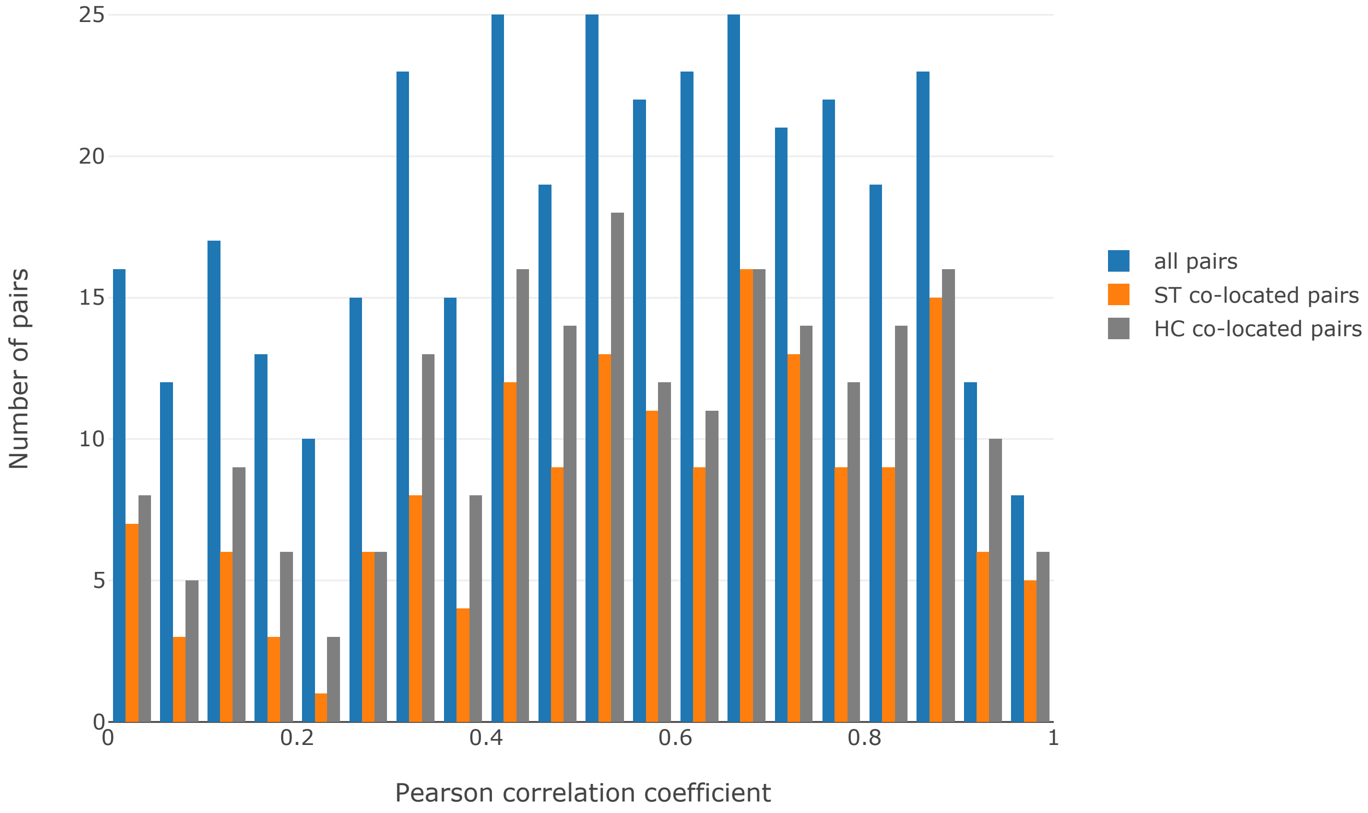


Figure 4.20: Pearson correlation coefficient (absolute value) between protein-transcript pairs, for the whole common set (blue), between elements that were grouped together in ST structure (orange) and in the optimal (Sum of Squared Error (SSE) elbow criterion) HC structure (gray). Distribution means are 0.52, 0.57, 0.55, and distribution medians are 0.52, 0.6, 0.56 for all pairs, pairs co-grouped in the ST structure, and pairs co-grouped in the HC structure, respectively.

we do not include gC-pC Pearson correlation coefficient in the orange distribution. This way we consider only gene-protein pairs, whose elements were found co-regulated by either ST or hierarchical clustering approach.

A distribution analysis shows, that the ST approach allows grouping elements, that are similar in expression, meaning potentially co-regulated, better, than the HC approach (T-test for comparing all pairs and ST co-located pairs gives p-value < 0.05, whereas comparing all pairs and HC co-located pairs gives bigger $p - value = 0.1$). Also, for the most of the bins the average correlation for the co-located pairs was stronger (larger in amplitude, independent from the sign) than in the whole common set (Supplementary Figure A.4). It means, that the ST approach allows better finding of the co-regulatory groups than just element-wise Pearson correlation analysis can suggest.

4.6.2 Grouping similarity correlation

A direct comparison of the ST structures is implemented with the most straight-forward algorithm that is based on Jaccard index and can show a correlation between two clustering sets, in terms of whether common elements of both datasets are grouped together in both clustering sets. The presence of not common elements is ignored. To calculate the grouping similarity correlation index, for each of the clustering structure the adjacency matrix is created, as described in State Space Search Walk for (2.11). Remember, the matrix consists of values $v_{i,j}$ such that $v = 1$ if elements i and j are in one group and $v_{i,j} = 0$ otherwise. Then, create an AND_matrix as a matrix which elements are ones, if in the cell position in both adjacency matrices are ones (logical AND) but leave the diagonal zero. Similarly, create an OR_matrix:

$$
\begin{aligned}
m_{AND} &= (mA_A \wedge mA_B) * \begin{bmatrix} 0 & 1 & \cdots & 1 & 1 \\ 1 & 0 & \cdots & 1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \cdots & 0 & 1 \\ 1 & 1 & \cdots & 1 & 0 \end{bmatrix}, \\
m_{OR} &= (mA_A \vee mA_B) * \begin{bmatrix} 0 & 1 & \cdots & 1 & 1 \\ 1 & 0 & \cdots & 1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \cdots & 0 & 1 \\ 1 & 1 & \cdots & 1 & 0 \end{bmatrix}.
\end{aligned}
\tag{4.3}
$$

Then the grouping similarity correlation index is calculated as a ratio of L1 norms of matrices AND_matrix

and *OR_matrix* (note, here L1 norm is a simple count of non-zero cells in a matrix):

$$\begin{aligned} G &= \frac{n_{AND}}{n_{OR}}, \\ n_{AND} &= ||m_{AND}||^1, \\ n_{OR} &= ||m_{OR}||^1 \end{aligned} \tag{4.4}$$

Additionally, to compare the grouping correlation to the value by chance, a bootstrapping analysis was performed, when the protein MapMan bin identificators were shuffled among the kinetic data, to destroy the kinetic-functional links in the original structure of one of the two compared datasets. The proteome dataset was chosen as the smaller one, to speed up the bootstrapping analysis.

Grouping correlation for the single bin 9

For illustrative reasons, the comparative analysis of the same functional group "mitochondrial electron transport / ATP synthesis" (MapMan bin 9), that was used as the first example for the proteome and transcriptome analyses, was considered first in more details. The proteome of bin 9 contains 21 proteins; the transcriptome of bin 9 contains 104 genes. The intersection between the datasets gives 21 protein-gene pairs in the common set.

Based on the inter-level grouping similarity correlation (4.4), the ST approach gives the correlation index 0.5. As described above, it is hypothesized, that the correlation between the proteome and transcriptome levels is biologically related, and the ST approach has an ability to show this correlation. To check the hypothesis, the grouping correlation index is compared for the same algorithm, applied to the randomized dataset. The randomization breaks the original protein-transcript pairs in the common set, creating fictional functional relations. As the Signature Topology approach considers the functional ontology layout, the resulting structure is also heavily affected by shuffling the functional labels.

Two randomized versions of the proteome dataset were used: shuffling among the kinetic dataset for the bin 9 only, that allows to preserve the group-specific majority pattern and variability; as well as shuffling within the whole proteome dataset, that allows to keep the variation/noise level the same for all functional groups, but destroys the functional relation between the elements inside the dataset and with the transcriptome elements simultaneously.

The average results of the bootstrapping analysis with 1000 repeats are the following: the grouping correlation index equals 0.25 for shuffling within a functional group (the forth box-plot in the Figure 4.21, A), and 0.17

for shuffling within the whole proteome dataset (the forth box-plot in the Figure 4.22, A). That shows that the observed correlation between the proteome and transcriptome Signature Topology structure is significantly different from the random.

For comparison, the same procedure was applied to the Hierarchical clustering approach (number of clusters

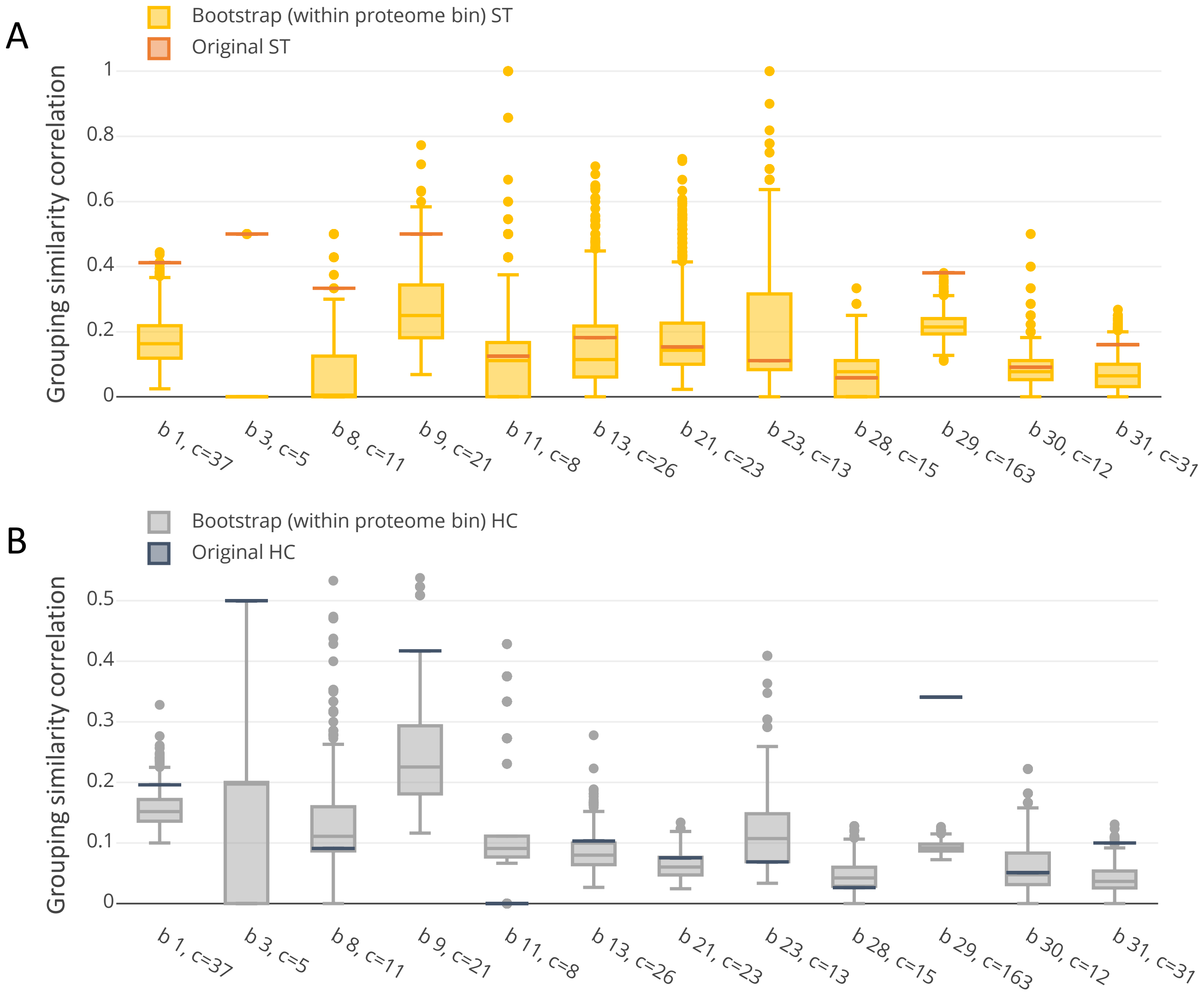


Figure 4.21: Grouping similarity correlation, comparing proteome and transcriptome within 12 functional groups (MapMan bins) separately, calculated for original data and bootstrapped proteome data (shuffled within each bin). Shuffling was repeated 1000 times for functional labels (MapMan tree label) within the proteome elements of the investigated bin. X-axis is labeled for functional bin number (b) and size of its common set (c). **A.** Comparison between Signature Topology structures, built for the original proteome and transcriptome datasets (orange lines) and built for the original transcriptome elements and the randomly shuffled proteome elements (yellow box-plots). **B.** Comparison between Hierarchical clustering, built for the original proteome and transcriptome datasets (dark-gray lines), and built for the original transcriptome elements and the randomly shuffled proteome elements (gray box-plots). Note, the Y-axis range is twice less than for the ST structures from the figure A.

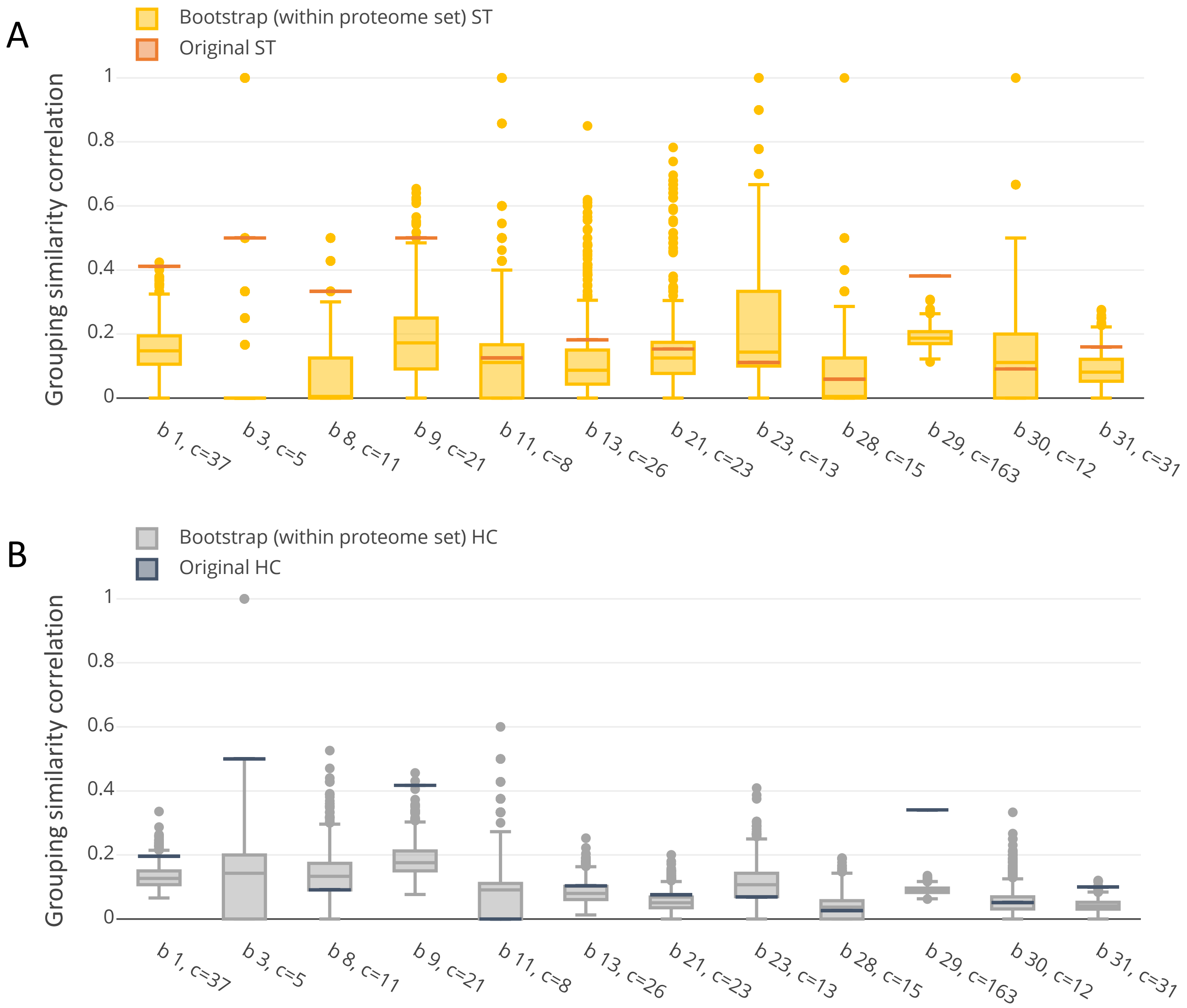


Figure 4.22: Grouping similarity correlation, comparing proteome and transcriptome within 12 functional groups (MapMan bins) separately, calculated for original data and bootstrapped proteome data (shuffled within the whole proteome). Shuffling was repeated 1000 times for functional labels (MapMan tree label) within the the whole proteome dataset. X-axis is labeled for functional bin number (b) and size of its common set (c). **A.** Comparison between Signature Topology structures, built for the original proteome and transcriptome datasets (orange lines) and built for the original transcriptome elements and the randomly shuffled proteome elements (yellow box-plots). **B.** Comparison between Hierarchical clustering, built for the original proteome and transcriptome datasets (dark-gray lines), and built for the original transcriptome elements and the randomly shuffled proteome elements (gray box-plots).

defined by elbow criterion based on Sum of Squared Errors (SSE)). Hierarchical clustering considers only kinetic information, so shuffling within the bin affects only protein-transcript relation in common set and not clustering structure itself for proteome elements, and it gives a baseline for random correlation for elements in the observed proteome and transcriptome bins. Shuffling within the whole proteome set completely randomizes the grouping correlation, showing the difference between the given proteome bin elements and the whole proteome dataset variability.

The average random grouping correlation for HC structure is 0.23 for shuffling within the bin 9 ((the forth box-plot in the Figure 4.21, B), and is 0.18 for shuffling within the whole proteome dataset (the forth box-plot in the Figure 4.22, B). Note, that the original HC structures give grouping correlation index 0.42, that is smaller, than ST index, but still higher, than both random values. Comparing these two approaches, we can see, that random ST structures have much bigger range of possible values than HC structures, but both original structures gave grouping correlation significantly bigger than average random.

Let us see if the same pattern is true for all other functional groups (MapMan bins).

Overall proteome-transcriptome correlation analysis

This grouping correlation calculation (4.4) was implemented for inter-level correlation evaluation for all functional groups (MapMan bins) between proteome and transcriptome datasets, where grouping correlation was non-zero. It means, that at least two gene-protein pairs should be in the same group both in the proteome and transcriptome ST structures as well as the HC structures (chosen with SSE elbow criterion). It gave 12 functional groups, see in Figures 4.21 and 4.22. On the x-axis of the plot, the label shows a bin number followed by a number of gene-protein pairs as a size of the common set. For the comparison to functionally and protein-transcript relation random data, the shuffling was done within the bin (Figure 4.21) and within the whole proteome dataset (Figure 4.22).

The shuffling within the given proteome bin ensures random connection between the kinetic and functional information, that affects the ST structure, but does not affect the HC structure. It also creates random mapping for common set of protein-transcript pairs. In half of the functional groups the original ST structures give the grouping correlation index significantly higher than the random distribution (bins 1, 3, 8, 9, 29, 31 in Figure 4.21, A). For HC approach it gives similar results except of the bin 8 (Figuree 4.21, B), where the original HC structures gave the grouping correlation index even lower, than the random median. It can be interpreted as the contribution of the functional information allowed better correlation between proteome and transcriptome, than considering only kinetic information.

The shuffling within the whole proteome dataset increases the variability level of the obtained random sets, and also affects the HC structures, as random elements from outside of the given functional group are contained in the random sets. For this shuffling method, 7 groups had significantly higher grouping correlation index for

ST structure than for random distribution (Figure 4.22, A). The results for HC approach did not differ from the shuffling within the current bin.

This is only one example of how the direct comparison of the Signature Topology structures could be done and it opens a broad field of interpretation.

4.7 Approach evaluation

Because of the complex data structure and functional knowledge involvement, the classical cluster validation techniques such as various internal clustering validation indices and external clustering validation (Charrad et al. 2014, Brock et al. (2008), Theodoridis and Koutroumbas (2008)) are not applicable in this case. Instead, the proposed approach was checked on (i) the method robustness by randomly removing fraction of elements out of the bin (up to 50%) and comparing the resulted grouping with the grouping of the remaining elements in the original clustering structure with Jaccard index, and (ii) the functionality-kinetic connection by comparing Hierarchical clustering, the ST approach, and classical enrichment based on MapMan ontology on simulated data, that were obtained by randomly assigning subbin labels to dataset kinetic elements.

4.7.1 Robustness to data removal

The first evaluation step is to check, how robust is the proposed method in terms of maintaining the connections between elements with a changing set of available elements. For the task, a simulated dataset was prepared by randomly removing a fraction of elements from the bin (up to 50%) and comparing resulted structure to the original structure. The comparison idea is the same, as was used for the inter-omics comparison in Chapter 4.6.2, with one structure is the original ST tree, and another structure is the ST tree for the reduced simulated dataset, and a common set is the latter, the reduced simulated dataset. The higher the grouping correlation is, the better the proposed approach can maintain the connection between analyzed elements under the changing circumstances, the more robust the ST method is.

The results for bins with a size bigger than 20 elements, both proteome and transcriptome datasets, show, that the clustering is robust for any removed fraction (up to 50%) for big groups (with bin size bigger than 200 elements inside), and less robust for smaller groups (Figure 4.23). But the robustness is also dependent on the MapMan layout structure – for some bins the average grouping correlation stayed above 90 for all tested removing fractions and repetitions. Interesting, that for the most proteome bins the robustness is higher than for most transcriptome bins, independent from their sizes. That can lead to speculations about the noise difference between these two omics data. Additional comparisons were made for bin structure parameters as tree depth (see Supplementary Figure A.5) and maximum number of MapMan children (see Supplementary Figure A.6), but no dependency was observed.

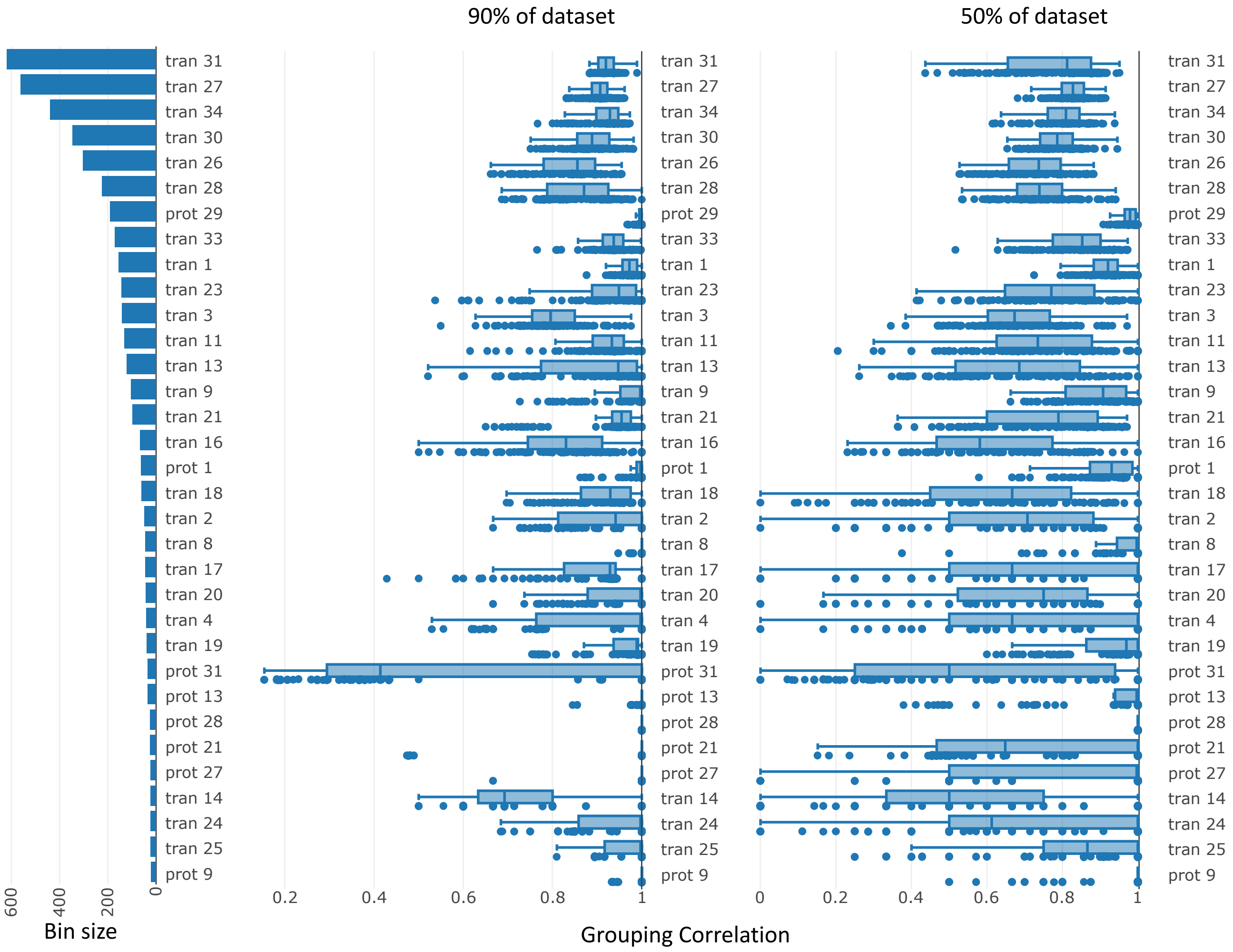


Figure 4.23: Robustness of the ST structure against data removal: The left plot shows the size of the corresponding bins from proteome (prot) and transcriptome (tran) datasets. The middle and the right plots show the group correlation between the original ST structure and ST structure of the randomly reduced dataset to 90% and 50% of the original size, correspondingly. 100 repetitions for each bin and each reducing fraction are presented.

4.7.2 Balance between kinetic and functional information

The second evaluation step is to check, how big is the influence on the structure caused by the disruption between the kinetic and functional information of the analyzed elements. It is done by comparing two structures:

the structure built on the original data (this will be called the original structure) against the structure built on randomized data. The randomized data were obtained with following randomization scheme: the kinetics of a protein is randomly assigned to an existing functional subbin. This way all slots in the MapMan structure are original, but proteins are assigned to the slots in a random order. This balanced shuffling scheme is performed for each functional bin first within the functional bin ("Shuffle bin" method in Figure 4.24) and then with the whole proteome dataset ("Shuffle all" method in Figure 4.24).

The analysis of the approach' ability to recover the original structure after the shuffling is done on four different grouping approaches: hierarchical clustering with optimal cluster number defined by SSE (HC_{SSE}), hierarchical clustering with optimal cluster number defined by DxI (HC_{DxI}), the Signature Topology approach (ST), and a grouping based on fixed MapMan ontology tree (MM). For the last case, up-regulated groups (defined by majority vote) on each depth level were compared between the original and randomized structures. As a comparison parameter, Jaccard Index (Levandowsky and Winter 1971) was used for evaluating the similarities of the original and randomized sets. Jaccard Index equal 1 means the compared groups have identical elements.

The comparison was performed on two alignment criteria:

1. Kinetic information: How similar are groups in terms of containing the elements with the same kinetic information (functional labels are ignored). This method is denoted as "Align kinetics" in Figure 4.24.
2. Functional information: How similar are groups in terms of containing the elements with the same subbin notations (kinetics of elements is ignored). This method is denoted as "Align functions" in Figure 4.24.

The described algorithm of the analysis was applied to the functional bin 9: considering two different shuffling techniques (within the bin 9 and among the whole proteome dataset) for each aligning criterion, 4 different testing methods were applied.

The analysis goal is to see how much influence the randomization of the elements will make for each grouping approach and aligning criterion. The bigger the difference between the original structure (based on the original data) and the grouping on the randomized dataset, the more information the original structure contained, that is why as a comparison parameter was chosen (1-Jaccard Index). If no difference between original and randomized structure were found, it would give 0, meaning the randomization did not affect the structure, that would mean that structure is not dependent on the information that was randomized. If the recovery of the original structure is not possible, it would give 1 as a maximum value, that would mean that the original structure is highly dependent on the randomized information.

The analysis showed that all three grouping methods (ST, MM, HC) were unable to recover kinetic infor-

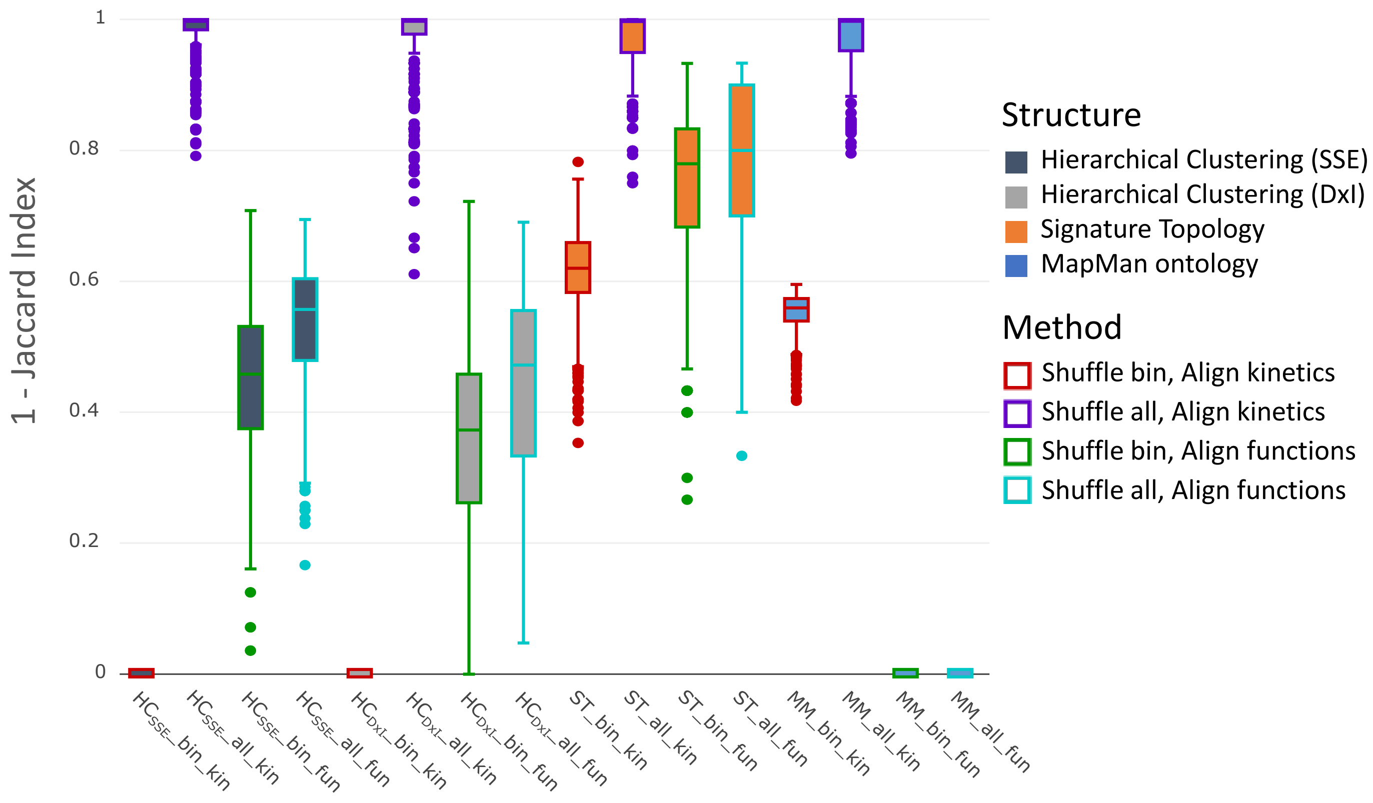


Figure 4.24: Impact of the randomization of different input data on the output structure. (1 - Jaccard index) between original and randomized data (bin 9) for four approaches of grouping: hierarchical clustering with optimum k by SSE (HC_{SSE}, dark-grey), hierarchical clustering with optimum k by DxI (HC_{DxI}, light-grey), the Signature Topology approach (ST, orange), and MapMan ontology (MM, blue) is calculated. Randomization was done either by shuffling given functional subbin labels within the bin elements ("shuffle bin" method) or within all proteome elements ("shuffle all" method). Jaccard index was calculated by aligning the kinetic elements or functional notations, $Jaccard\ index = 1$, means, that the original and shuffled groups have the same set of elements. $(1 - Jaccard\ index) = 1$, means, that the recovering of the information after shuffling was not possible (sets of elements are completely different). The procedure was repeated 1000 times and presented as a box plot for each case.

mation after shuffling within the whole proteome dataset (Figure 4.24 method "Shuffle all, Align kinetics", purple border: average values are close to ones for all structures $HC_{SSE}\_all\_kin$, $HC_{DxI}\_all\_kin$, ST_all_kin, and MM_all_kin). The alignment of kinetics with shuffling within the bin (method "Shuffle bin, Align kinetics", red border) gave a score of 0 for both Hierarchical clustering structures, because this clustering technique is deterministic and it is always the same structure, as functional information gives no impact on the clustering. For the same method, the difference between randomized and original structures is more for Signature Topology than for MapMan (median 0.62 against 0.55), that is an evidence, that the randomization brings more impact on the ST structure, meaning this structure holds more kinetic information. Comparing schemes, recovering functional information (green and cyan borders), we can see, that for MapMan structure the recovery was complete $(1 - Jaccard\ index = 0)$ because by definition the shuffling could not change the subbin labels within the MapMan tree. Comparing Hierarchical clusterings and Signature Topology, that were influenced by randomization of kinetic-functional connection, we can see, that again the ST structure got bigger impact than HC for both shuffling options (dissimilarity correlation 0.78 and 0.8 for ST against 0.45 and 0.55 for HC_{SSE}, and 0.37 and 0.47 for HC_{DxI}, for shuffling within the bin 9 and within the whole proteome dataset respectively), meaning the ST structure contains more functional information.

4.8 Biological application

The ST approach, applied to the real-world datasets sampled from *Chlamydomonas reinhardtii* during the heat acclimation experiment, could reveal regulatory genes.

4.8.1 Regulators in Calvin Cycle

If we look at the Calvin Cycle proteins (see Figure 4.9), we can notice, that the revealed yellow group has a minor pattern in contrast to the blue and green major pattern. If we would consider this group, Calvin Cycle proteins, as a solid unit and consider its behavior according to the major pattern, those yellow genes would be neglected and information about their experimental behavior would be lost.

The ST approach distinguished a minor behavior pattern of these proteins from the major pattern in

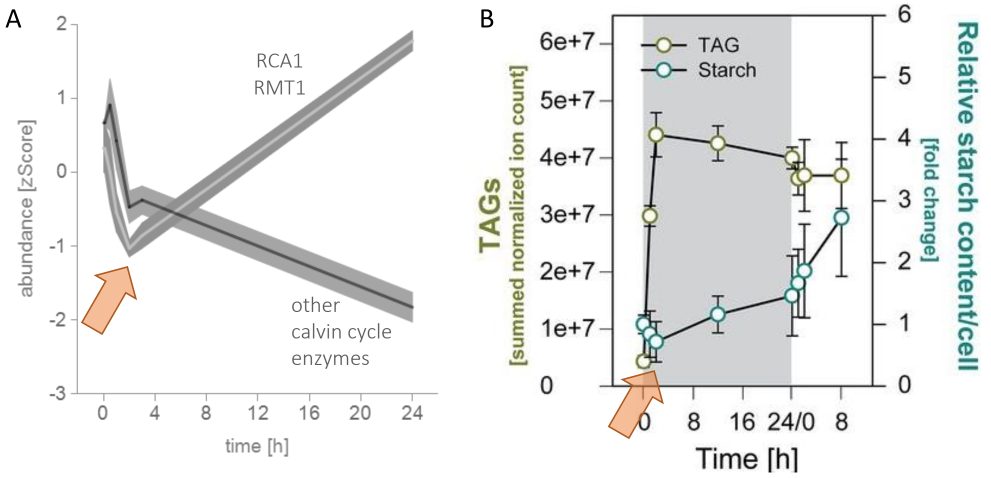


Figure 4.25: Heat acclimation response in *Chlamydomonas reinhardtii*. A. Microarray data for Calvin Cycle proteins from Hemme et al. 2014 and Mühlhaus et al. 2011. B. Cytological and Ultrastructural Data from (Hemme et al. 2014).TAG and starch content. The TAG content was determined from summed normalized ion counts of all TAG species measured by mass spectrometry. The relative starch content per cell was determined by colorimetry.

the functional group. Among them there are known regulators rubisco activase RCA1 and protein-arginine methyltransferase RMT1. If we compare kinetic of these two proteins with TAG and starch synthesis during heat acclimation (Figure 4.25, B, from (Hemme et al. 2014)), we may notice, that around the time point, when RCA1 and RMT1 became active (Figure 4.25, A), the synthesis activity from TAG to starch has been changed. It is a pure speculation, but it gives a hint to look closely on these molecules' role as switchers in a cell metabolism.

4.8.2 Regulators in ribosomal proteins

In the functional group Ribosome proteins (MapMan bin 29), ribosome-binding factor PSRP1 has functional annotation "protein.synthesis.ribosomal protein.prokaryotic.chloroplast.30S subunit.S30A" and belongs to the subbin 29.2.1.1.1.530. In both transcriptome and proteome data (Figure 4.26) the molecule behaves differently from other elements with the parent functional specificity 29.2.1.1.1, and in both ST structures this molecule is assigned as a singleton (Figure 4.26, A, C). Despite having different behavior in proteome and transcriptome experiment (Figure 4.26, B, D), in both datasets the ST approach managed to determine, that the deviation from the majority of the functionally similar elements is important. The known role of the RPCP1 as a gene regulator

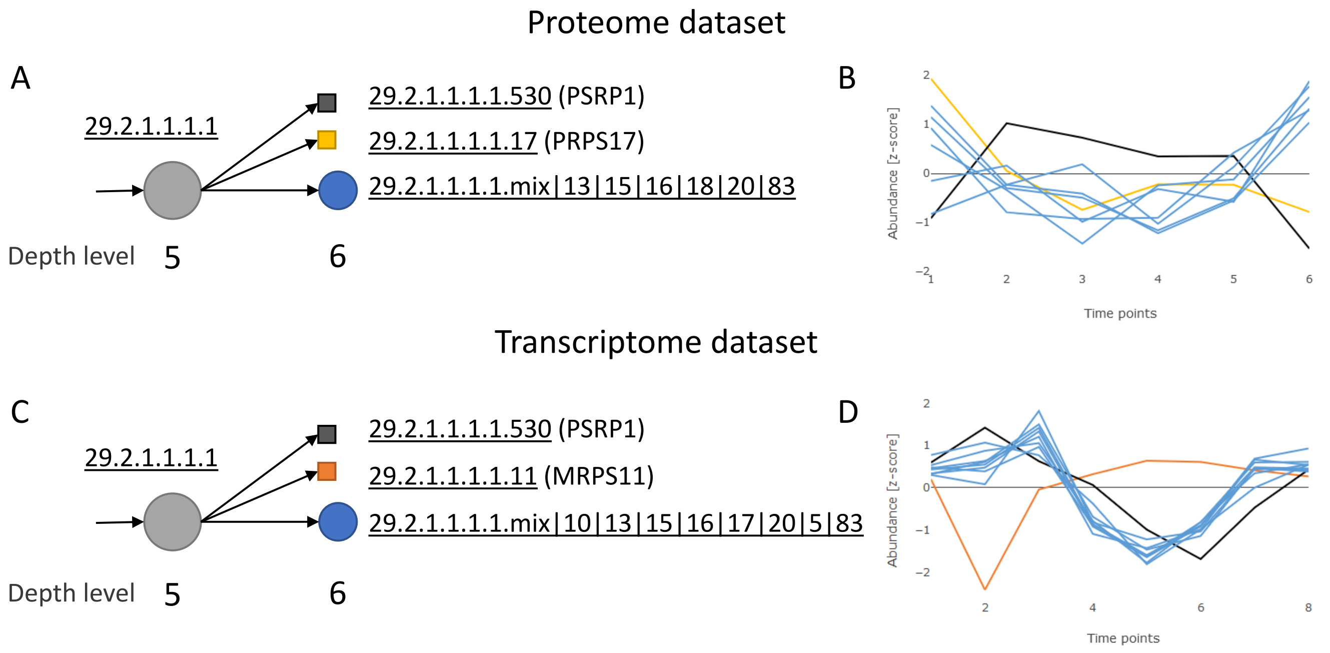


Figure 4.26: Comparison of ST structure for the subbin 29.2.1.1.1.1 between proteome and transcriptome datasets. Proteome dataset has 8 elements, 6 of them were clustered in a mixed group (blue), 2 proteins were assigned to singletons (PSRP1, black; PRPS17, yellow). Transcriptome dataset has 10 elements, 8 of them were clustered together (blue), 2 proteins were assigned to singletons (PSRP1, black; MRPS11, orange). A, The Signature Topology structure for proteome data. B, Kinetic behavior of the 8 proteome elements. C, The Signature Topology structure for transcriptome data. B, Kinetic behavior of the 10 transcriptome elements.

(Sharma et al. 2009) proves the point of the minor pattern being important to be distinguished from the majority. Another transcript from this group, that was picked out as a singleton is mitochondrial ribosomal protein S11 (MRPS11) that is a structural component (Ishiguchi et al. 2004). Another singleton in the proteome structure is PRPS17 (30S ribosomal protein S17, chloroplastic) that is not thoroughly studied yet and can be an interesting goal for further regulatory studies.

4.8.3 Signature Topology outcome - list of terminal groups

The ST approach outputs a list of terminal groups (leaves in the tree) for better overview and further analysis of the resulted structure. See example of the output list for proteome bin 9 in Table A.3 and for transcriptome bin 9 in Table A.4. Each table includes following parameters of the terminal groups:

- group label - modified MapMan subbin label,
- group size - number of elements in the group,
- step gain from root pro element - an average step gain by extracting the group from the root node
- group density - maximal distance for each element in the group, averaged on the group size
- elements - list of identificators of elements in the group.

A group size can be used to select singletons, a group density can show co-regulated groups of similar functions, and a step gain can highlight the most deviant children of the functional group. The last can help revealing the specific regulators, that is not co-regulated as the elements with similar functionality, and that is why often overlooked by classic approaches.

5. Discussion

The goal of creating the Signature Topology (ST) approach was to make the data more interpretable by incorporating the functional ontology information into the pattern identification process. This goal was achieved by implementing the iterative dynamic programming within the paradigm of data-and-knowledge-driven machine learning. The necessity for the new approach extracting information out of the omics data was motivated by the drawbacks of the existing approaches.

One of such well-known methods, described in the Introduction, gene set enrichment analysis - GSEA - was invented to overcome the limitations of the more straight-forward and simple single-gene analysis (A. Subramanian et al. 2005). Single gene analysis focused on the individual genes, that can be found on the top or bottom of the ranked list of differentially expressed genes, as the most deviant, hence the most crucial for the process in the experiment. In contrary to the single gene, GSEA operates with a gene set, defined as a group of genes, that share common biological function, chromosomal location, or regulation. GSEA can successfully overcome the limitations of the single-gene approach, resulting in following advantages: 1) results are more robust to the natural noise; 2) decrease of the complexity; 3) concentration on the pathway effect; 4) comparability between different studies due to universal gene sets databases, that are constantly expanding. Still, it lacks the main advantage of the single-gene analysis, namely it neglects the individuals with a diverse pattern, which distinguishes the key role players in the regulatory mechanisms. Also, it does not reflect the given study without bias to existed division of genes into the static gene sets. Also not clear specificity depth of the genes division can strongly affect results.

The new proposed approach aims to combine the advantages of both classical analyses. It may seem unfeasible because the GSEA and single-gene approaches are mutually exclusive, but it is possible as the algorithm can iteratively move from the more global functional annotations of the GSEA-like approaches to more detailed single-gene descriptions in one pipeline.

Highlighted pitfalls of the classical approach (clustering + GSEA), and how they were avoided by the newly proposed in the current work Signature Topology method are listed here:

1. Consecutive steps - first clustering and then the majority vote of GSEA - neglect outliers, that represent the minority of experimentally detected kinetic patterns, that lead to omitting some possible key players. Whereas the ST grouping allows singletons (a group, consisted of only one element) with specific and distinct patterns. It is important, because often the key enzymes do not follow the majority behavior and can be easily overseen otherwise.
2. Considering clusters obtained from only kinetic relations mixes available functional features and dims the possible gain of information that can be obtained from the experiment. Also, it is often the case, that the GSEA approach considers only large functional groups, and a huge layer of available detailed information is not considered at all. Whereas the ST approach secures the functional purity of the obtained groups on the optimal level of function's specificity.
3. The real dynamic experimental picture is not reflected in the analysis, so the captured picture is static. By applying the ST approach the prior knowledge is actualized by the analyzed experiment, so the analysis adjusts the ontology by pulling the suitable ontology level to show a fingerprint of the current experimental conditions.
4. Besides, the classical approach performs kind of double work - first cluster everything and then find a sense in the functional combination of elements within a cluster. Independence between these two steps makes it impossible to adjust parameters both of clustering and enrichment analysis to get the best explanation of the data. That drawback can be overcome by introducing an approach that would apply these two steps in a continuous process that was successfully implemented in the ST approach.

As a result, the ST approach gives an optimal overview of the explained omics data from the point of view of their involvement in the biological processes. The method's output is dual, including a ST tree and an ordered list of terminal nodes of the tree. The list shows functional pure gene groups from the most divert from their functional neighbors to the least. It can be used for choosing a potential key role player for further research. Also, the results of the ST approach can be compared between different omics levels (multi-omics analysis), experimental conditions, and organisms.

There are numerous ways to extend and improve the classical approach, such as clustering elements based on several experimental parameters or integrating multiple omics levels (Tini et al. 2019) in e.g. network-based genes clustering (Gladitz, Klink, and Seifert 2018; Yan et al. 2018; Haynes et al. 2013). Drawbacks of such approaches are data and level specificity, whereas the proposed Signature Topology approach is universal to any experiment setup and any omics-data.

One way of the improvements goes in the same direction as the ST approach - into the data-and knowledge-driven machine learning, aiming to integrate experimental data with the prior functional knowledge and represent the obtained information in an easily understandable way. In recent years, the implementation of this idea appeared in (Kramer et al. 2014; Dutkowski et al. 2013) and in (Farré et al. 2017). In these works, the focus was on finding the ideal overall ontology that would be unique and reflect the correlation between proteins for each and all situations. Multiple datasets are used so that each individual experiment would slightly influence the global final picture. The drawback of such approaches is a lack of insight into the current experiment and still the high inertness of the global ontology structure. It can definitely help to gather basic correlations between proteins but it is biased to the strongest exhibition of protein interactions. Also, it requires an external threshold to filter weaker protein correlations. In our turn, we concentrated on more instantaneous correlations and the threshold is defined automatically based on the intrinsic data variability itself.

Another similar idea appeared as ActivePathways method in Paczkowska et al. 2020. The method integrates multiple molecular datasets and pathway annotations. It detects significantly enriched pathways across multiple datasets, including those pathways that are not detected as significant in any individual dataset. It is suitable for coding and non-coding and uses hypergeometric tests for multiple enrichment analysis. But this method uses the given pathway annotations without rearrangement, reflecting each dataset or combination of them. Additionally, this approach is not suitable for comparing and integration of datasets that lay in wide size range (metabolites in tens, proteins in hundreds, genes in thousand of elements).

As the current work have shown, the ST approach implements the data-and knowledge-driven machine learning in a unique way with following properties:

1. Data aggregation along the hierarchical structure
2. Adjustable parameters for complexity measure
3. Robustness to high noise level
4. Easy application to the real tasks
5. Capability to wide range comparisons

In the following we consider each property in more details.

5.1 Data aggregation along the hierarchical structure

The ST approach is implemented within a paradigm of the data-and knowledge-driven machine learning and pulls the knowledge part from the hierarchical ontology and therefore both flexible (can be adjusted to the needs) and rigid, as the ontology defines the initial structure of the data. It can be considered as a drawback and the limitation of the approach, but as there is a wide spectrum of the ontologies, one can always find the one, suitable for the needs. This idea, using the knowledge stored in the ontologies, is also implemented in other approaches, for example in Elmarakeby et al. 2021 for defying specific architecture of Deep Neural Network based on the ontologies on different system levels.

The chosen MapMan ontology is one well described and broadly used, with detailed diving into the specific role of biomolecules on different levels, such as genome, transcriptome, and proteome, that can provide further an opportunity to compare between different system levels. Moreover, MapMan was already used as a functional annotation source for previous studies in our group (Schroda, Hemme, and Mühlhaus 2015; Hemme et al. 2014), so it would be suitable for comparison between these and other studies in plant biology.

Note, that independent of the ontology choice, the algorithm remains the same and is applicable to any tree-like hierarchical structure. This way, the ST approach can be used for cancer researches with the KEGG Brite tree as an ontology. It is also possible to incorporate DAG-like ontologies (e.g. Gene Ontology) as an outlined hierarchical structure, that would widen the application range of the ST approach.

Guiding the data aggregation along the hierarchical structure leads to an apparent disadvantage as the result would be biased by the hierarchical structure. Hierarchical structure encodes the prior knowledge and as the ST algorithm goes along its lines, it is heavily biased by the ontology. To analyze how exactly the knowledge incorporation affects the data structure, we compared the ST approach to the simple hierarchical clustering (no prior knowledge, no bias, flexible interpretation) and to the MapMan labels (maximum bias, no experiment interpretation). Our results (Figure 4.24) have shown, that the apparent drawback is actually a great compromise between two extrema.

5.2 Adjustable parameters for complexity measure

Many approaches include in their algorithm possibility to adjust some parameters to better fit the model to the data. In case of clustering methods, it is usually a number of clusters. Because of the unsupervised paradigm, it is unknown how many groups the data should be divided into, and choosing different number of clusters heavily affects the outcome. However, despite obvious influence on a result, a number of clusters little tells us about the data structure properties, namely why there are exactly this number of groups. In other words, this parameter lacks physical interpretability.

Contrary to the number of clusters, the ST approach gives an opportunity to adjust the gain formula by changing the Information Content (IC) part (here is calculated by formula (2.9)). As IC parameter corresponds to the complexity impact and therefore is responsible for a penalty for too high complexity, it is possible to use different penalty approaches and to modify the IC formula accordingly. This parameter represents a complexity of the data structure and therefore has better interpretability for the understanding of the data, compared to the voluntary chosen number of clusters or calculated components of the Principal Component Analysis.

The code also allows using weighted experiment data, that can be helpful when the data points have different importance for the studied phenomena. For example, there can be hints, that some specific time points are more meaningful than others in the time series data, then it can be wise to introduce a weighting vector as an additional parameter to the ST algorithm, whereas by default the weighting vector is ones.

However possible, adjusting these parameters brings the limitation to the comparison with different experiments and system levels so we advice to use the standard settings.

5.3 Noise robustness

Analyzing biological molecular data is challenging because of its high noise level. It is caused by various reason, such as technical imperfection of the measuring technology as well as individual variability of the living organisms. To deal with this challenge, the data interpretation techniques must be robust to the noise in order to catch the real pattern within the data, partially hidden by the noise.

According to works of Klebanov and Yakovlev 2007; Hong et al. 2013 and supported by observed experiment data, standard deviation in omics data due to biology data specificity is usually around 0.5, as well as variability of the used in this project proteome and transcriptome datasets (Figure A.1).

Defined here type B error (mixing patterns in one group) is considered more hazardous than the type A error (redundant groups) as the former will lead to the loss of the experimental information that can be crucial to mapping experimental pattern to the functional information. On the other hand, type A error will just increase the complexity without loss of the information.

The chosen setups for artificial data trial (see Figures 3.2 and 4.3) could distinguish between type A error (redundant groups) and type B error (mixed patterns), allowing to check, which method makes more benign errors (type A) or more crucial errors (type B). It was shown, that the ST approach was able to make the least number of type B errors, though allowing relative high amount of type A errors (see Figures 4.2 and 4.4). Interestingly, the hierarchical clustering approach could not achieve the low level of type B error even by applying the punishment for high complexity (choosing number of clusters by minimal DxI index) neither for the noise robustness setup (Figure A.2, C) nor for the minority revealing setup (Figure A.3, C).

Such results show that the proposed ST approach is more conservative in terms of more crucial type B error than the hierarchical clustering approach. Overall analysis of the synthetic datasets has proven that the proposed ST approach is appropriate to apply for real experimental data as it is robust to the noise level, characteristic for the omics data.

5.4 Easy application to the real-world tasks

Observing different experimental behavior patterns for the molecules from the same functional group, can give a hint to the different involvement of the molecules into the carrying of the function. This way, the specific role of the regulatory genes and their rarity, compared to, for example, structural molecules, often cause them to behave differently from the majority pattern of molecules in the same functional group. Finding such minor patterns at every function specificity level is a promising approach to highlight the potential candidates in regulatory genes.

As the ST approach allows detailed grouping based on the experimental behavior of individual molecules, it can reveal minor patterns in the dataset, as was demonstrated by both synthetic (Figure 4.4) and experiment data (Figures 4.9 and 4.26), it is a suitable method for pointing to the regulatory molecules in the data. In these real-world examples, proteome data analysis has shown, that the well known cell cycle regulator RCA1 (Dong et al. 1997) has indeed a minor pattern within the Calvin Cycle functional group and was successfully revealed and was separated from the majority pattern group. Both proteome and transcriptome data analyses of the ribosomal protein functional group have managed to separate the regulatory molecule PSRP1 (Sharma et al. 2009) from the majority of structural molecules in the group. It also pointed to other molecules with minor patterns that could be promising candidates for the future researches.

As highly relevant for different experiment data, for the unsupervised learning part (finding the optimal configuration for each functional ontology term) a shape-based clustering technique is used. Shape-based clustering is considered the most simple, yet quite reliable despite its simplicity (Aghabozorgi, Seyed Shirkhorshidi, and Ying Wah 2015). In this project, real time-series data were analyzed, but with a shape-based approach, the usage is not restricted by them and the same algorithm can work with, for example, experiment-series data. In contrast, an implementation of, e.g. a model-based approach, would require a necessary adjustment step, that could be difficult to automatize. This algorithm further can be used as a part of a feature-based approach if proceeded by a feature-selection method like Principal Component Analysis (PCA) (Meng et al. 2016).

5.5 Capability for inter system level comparison

Biological interpretation of changes in discrete omic-domains is challenging in the face of complex biochemical regulation such as organismal versus tissue versus cellular-level processes, epigenetics, and mRNA or protein post-translational modification (Wanichthanarak, Fahrmann, and Grapov 2015) because of incomplete knowledge of the underlying biological processes. That is why it is important to overcome such discreteness and to be able to compare and combine observed results between the system levels.

The connections between different system levels are complex and often have non-linear relationships, that are difficult to incorporate into the model and learn from the data. This way, the inter system level comparison is hampered by the different timings of the events on different levels. That is the reason why, for example, the direct co-clustering of proteome and transcriptome data usually does not provide the insight into the data relations ((Maier, Güell, and Serrano 2009; B. Wang et al. 2019)). The advantage of the proposed approach is, that the ST algorithm outputs the tree structures, that are time-invariant and can be directly compared.

The provided ST tree structures are easy to compare, because of the common 2 parameters, that are necessary for the comparison: the group labels bearing the functional specificity and the experimental elements, that have direct inter-experiment mapping due to the biological relevance (genes vs proteins vs metabolites between system levels or homologues mapping between organisms). That is why the most simple and direct approach was used, though the comparison techniques must not be limited to only this one.

This direct comparison of different omics levels is possible if we have knowledge about connection of elements via ontology terms. Here, as an example, we can see, that there are similarities and differences in the proteome and transcriptome ST structures (see Tables A.3 and A.4). Observed similarities (3 elements from proteome subbin 9.7 are also together in transcriptome subbin 9.7.mix|p68|p1|p86|p47|p19|p74|p93|p59|p95; 4 elements from proteome subbin 9.9.mix|p14|p16|p6|p15 are together in transcriptome subbin 9.9.mix|p46|p12|p33|

p40|p28|p73|p27|p78|p44|p6|p69|p97|p100) show that the transcripts and corresponding proteins are in direct translation mechanisms. Observed differences are the evidence, that the gene translation is more complex. Thus, for example, 2 elements from proteome subbin 9.1.1.5 and 1 element that was a singleton in the proteome structure ended up in one merged group in transcriptome structure. Whereas elements from the proteome subbin 9.mix|5|6 were separated into 2 transcriptome subbins 9.mix|4|6 and 9.5.mix|p56|p23|p71|p11|p14.

If the functional component would not allow direct alignment of the groups, the application of the described in Results section 4.6.2 grouping correlation would not be possible. Then one could try to use a direct element-wise correlation of kinetic vectors between mapped elements between datasets. Though direct correlation of expression levels between proteome and transcriptome is reportedly not strong (Maier, Güell, and Serrano 2009; B. Wang et al. 2019), it still can be used. Thus, Pearson correlation for protein-transcript pairs of elements that were grouped together in both analyzed datasets was in most cases stronger than for all pairs (see Figure 4.20 and Supplementary Figure A.4).

Grouping correlation is more reliable than element-wise Pearson correlation, because it will give a higher score for the case, when 2 transcripts and 2 proteins are grouped together, even when these 2 transcripts are up-regulated and 2 proteins are down-regulated, that would be considered as low element-wise correlation. The results of grouping correlation between system level comparing HC structures and ST structures (Figures 4.21 and 4.22) have shown, that including the functional information in the structure gives higher correlation between the proteome and transcriptome levels for more than half functional groups of molecules.

More complex approach to study inter system level dependencies between structures can be integrated with help of the broad experience of information theory regarding graphs, namely multi-layer networks. Suitable for the needs would be Multilayer Community Detection approaches (Didier, Valdeolivas, and Baudot 2018; Newman and Girvan 2004; Huang et al. 2020), which was able, for example, to detect protein complexes and protein function modules (Chen et al. 2018).

5.6 Conclusion

The proposed Signature Topology approach constructs signature footprints for any experiment omics data, structured along hierarchical functional ontology. The obtained ST structure and corresponding output list of terminal groups give a representation of new experiment data, incorporated with current knowledge about the biological functions of the molecules.

The demonstrated results of implementation of the Signature Topology approach to artificial and real-world experiment data have proved, that the proposed method has following features: (i) Robustness to noise, (ii) Sensitivity to minor patterns, (iii) Comparability between different experimental systems, (iv) Revealing the kinetic patterns with relevant, data-specific functional information.

The proposed approach allows to get an unbiased representation of the experimental data because it does not require an outside voluntary threshold and is bound to the current experimental conditions. It also allows direct comparison of different systems by comparing Signature Topology trees because of the same ontology terms of the trees. The demonstrated comparison is correlation-based, but more flexible and have functional knowledge incorporated in the comparison, that makes it more reliable than either pure pathway-based or correlation-based approaches.

Currently manual steps are necessary to perform a comparison. However, to automatize this step, multilayer graph analysis should be incorporated into the algorithm. Fortunately, the algorithm foundation and functional-oriented programming language F# allows this with ease.

The ST approach relies on the tree-like ontologies as the prerequisites. Although most of the present ontologies are DAG-structured, they can be straightforwardly reduced to tree. Overcoming nuisances of this transformation (creating double terms or reducing available information by omitting double terms) can expand the number of suitable for the ST approach ontologies.

The output list of terminal groups allows experimentalists direct interpretation of the results, as the naming of the revealed groups is based on the ontology terms. Aggregating existed functional groups is denoted by concatenation of the existing labels. In future it is possible to extend the understanding of revealed groups by incorporating the text mining tools to search through recent publications and finding more precise common functions of the newly aggregated groups.

A possibility for adjustment the model to the data via the complexity measure component of the gain function is a part of the more global task - hyper-parameter exploration for the IC landscape. The implemented in the current work formula for the IC shows good results, but with solved in future hyper-parameter adjustment task the model will automatically suggest the optimal parameter.

The proposed method is superior to the classic clustering approach in terms of revealing the minor patterns and is going together with similar approaches in the direction of finer adjustment the data-and-knowledge-driven machine learning approaches to analysis of complex omics data.

The Signature Topology approach is implemented as a part of an open-source FSharpBio library, detailed test examples are easy to follow, which makes the usage of the approach a simple task. The ST approach is actively developing and will have even more useful tools in future.

Bibliography

Aghabozorgi, Saeed, Ali Seyed Shirkhorshidi, and Teh Ying Wah (2015). “Time-series clustering – A decade review”. In: *Information Systems* 53, pp. 16–38. ISSN: 03064379. DOI: 10.1016/j.is.2015.04.007.

Alhamdoosh, Monther et al. (2016). “Combining multiple tools outperforms individual methods in gene set enrichment analyses”. In: *Bioinformatics* 33.3, pp. 414–424. ISSN: 1367-4803. DOI: 10.1093/bioinformatics/btw623.

Andrews, George E. (1998). *The theory of partitions*. 1st paperback ed. Cambridge mathematical library. Cambridge: Cambridge Univ. Press. ISBN: 052163766X.

Ardui, Simon et al. (2018). “Single molecule real-time (SMRT) sequencing comes of age: applications and utilities for medical diagnostics”. In: *Nucleic acids research* 46.5, pp. 2159–2168. DOI: 10.1093/nar/gky066.

Ashburner, M. et al. (2000). “Gene ontology: tool for the unification of biology. The Gene Ontology Consortium”. In: *Nature genetics* 25.1, pp. 25–29. ISSN: 1061-4036. DOI: 10.1038/75556.

Bandrowski, Anita et al. (2016). “The Ontology for Biomedical Investigations”. In: *PloS one* 11.4, e0154556. DOI: 10.1371/journal.pone.0154556.

Bard, Jonathan (2003). “Ontologies: Formalising biological knowledge for bioinformatics”. In: *BioEssays : news and reviews in molecular, cellular and developmental biology* 25.5, pp. 501–506. ISSN: 0265-9247. DOI: 10.1002/bies.10260.

Bard, Jonathan B. L. and Seung Y. Rhee (2004). “Ontologies in biology: design, applications and future challenges”. In: *Nature reviews. Genetics* 5.3, pp. 213–222. ISSN: 1471-0056. DOI: 10.1038/nrg1295.

Benjamini, Yoav and Yosef Hochberg (1995). “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, pp. 289–300. ISSN: 00359246. DOI: 10.1111/j.2517-6161.1995.tb02031.x.

Bentley, David R. et al. (2008). “Accurate Whole Human Genome Sequencing using Reversible Terminator Chemistry”. In: *Nature* 456.7218, pp. 53–59. ISSN: 0028-0836. DOI: `10.1038/nature07517`.

Bersanelli, Matteo et al. (2016). “Methods for the integration of multi-omics data: mathematical aspects”. In: *BMC bioinformatics* 17 Suppl 2, p. 15. DOI: `10.1186/s12859-015-0857-9`.

Bertalanffy, L. Von (1945). “Zu einer allgemeinen Sytemlehre”. In: 18.

Bertsekas, Dimitri P. (2012). *Approximate dynamic programming*. Fourth edition. Vol. / Dimitri P. Bertsekas ; Volume 2. Dynamic programming and optimal control. Belmont, Massachusetts: Athena Scientific. ISBN: 9781886529083.

Brenner, S. et al. (2000). “Gene expression analysis by massively parallel signature sequencing (MPSS) on microbead arrays”. In: *Nature biotechnology* 18.6, pp. 630–634. DOI: `10.1038/76469`.

Chen, Shi et al. (2018). “Global vs local modularity for network community detection”. In: *PloS one* 13.10, e0205284. DOI: `10.1371/journal.pone.0205284`.

Cunningham, Fiona et al. (2015). “Improving the Sequence Ontology terminology for genomic variant annotation”. In: *Journal of biomedical semantics* 6, p. 32. ISSN: 2041-1480. DOI: `10.1186/s13326-015-0030-4`.

Didier, Gilles, Alberto Valdeolivas, and Anaïs Baudot (2018). “Identifying communities from multiplex biological networks by randomized optimization of modularity”. In: *F1000Research* 7. DOI: `10.12688/f1000research.15486.2`.

Dong, X. et al. (1997). “Control of G1 in the developing Drosophila eye: rca1 regulates Cyclin A”. In: *Genes & development* 11.1, pp. 94–105. ISSN: 0890-9369. DOI: `10.1101/gad.11.1.94`.

Dutkowski, Janusz et al. (2013). “A gene ontology inferred from molecular networks”. In: *Nature biotechnology* 31.1, pp. 38–45. DOI: `10.1038/nbt.2463`.

Edman, P. and G. Begg (1967). “A Protein Sequenator”. In: *European Journal of Biochemistry* 1.1, pp. 80–91. ISSN: 0014-2956. DOI: `10.1111/j.1432-1033.1967.tb00047.x`.

Eiter, Thomas et al. (op. 2006). “Reasoning with Rules and Ontologies”. In: *Reasoning Web*. Ed. by David Hutchison et al. Vol. 4126. Lecture Notes in Computer Science. Berlin: Springer, pp. 93–127. ISBN: 978-3-540-38409-0. DOI: `10.1007/11837787{\textunderscore}4`.

Elmarakeby, Haitham A. et al. (2021). “Biologically informed deep neural network for prostate cancer discovery”. In: *Nature* 598.7880, pp. 348–352. ISSN: 1476-4687. DOI: `10.1038/s41586-021-03922-4`.

Erwin L. van Dijk et al. (2018). “The Third Revolution in Sequencing Technology”. In: *Trends in Genetics* 34.9, pp. 666–681. ISSN: 0168-9525. DOI: `10.1016/j.tig.2018.05.008`. URL: `https://www.cell.com/trends/genetics/pdf/S0168-9525(18)30096-9.pdf`.

Farré, Jean-Claude et al. (2017). “Active Interaction Mapping as a tool to elucidate hierarchical functions of biological processes”. In: *Autophagy* 13.7, pp. 1248–1249. DOI: `10.1080/15548627.2017.1313946`.

Fesehaye, Debessay et al. (2017). “Group Clustering Using Inter-Group Dissimilarities”. In: *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*. [S.l.]: IEEE, pp. 1011–1021. ISBN: 978-1-5386-1792-2. DOI: `10.1109/ICDCS.2017.299`.

Gasperskaja, Evelina and Vaidutis Kučinskas (2017). “The most common technologies and tools for functional genome analysis”. In: *Acta medica Lituanica* 24.1, pp. 1–11. ISSN: 1392-0138. DOI: `10.6001/actamedica.v24i1.3457`.

Gladitz, Josef, Barbara Klink, and Michael Seifert (2018). “Network-based analysis of oligodendrogliomas predicts novel cancer gene candidates within the region of the 1p/19q co-deletion”. In: *Acta neuropathologica communications* 6.1, p. 49. DOI: `10.1186/s40478-018-0544-y`.

Gomez-Cabrero, David et al. (2014). “Data integration in the era of omics: current and future challenges”. In: *BMC systems biology* 8 Suppl 2, p. I1. DOI: `10.1186/1752-0509-8-S2-I1`.

Gruber, Thomas R. (1993). “A translation approach to portable ontology specifications”. In: *Knowledge Acquisition* 5.2, pp. 199–220. ISSN: 10428143. DOI: `10.1006/knac.1993.1008`.

Guardia, Gabriela D. A., Ricardo Z. N. Vêncio, and Cléver R. G. de Farias (2012). “A UML profile for the OBO relation ontology”. In: *BMC genomics* 13 Suppl 5, S3. DOI: `10.1186/1471-2164-13-S5-S3`.

Guzzi, Pietro H. (2019). “Ontology in Bioinformatics”. In: *Encyclopedia of Bioinformatics and Computational Biology*. Ed. by Shoba Ranganathan, Kenta Nakai, and Christian Schönbach. Amsterdam: Elsevier, pp. 809–812. ISBN: 9780128114322. DOI: `10.1016/B978-0-12-809633-8.20490-1`.

Harris, Elizabeth H., David B. Stern, and George Witman (2009). *The Chlamydomonas sourcebook*. 2nd ed. Amsterdam and Boston: Academic Press. ISBN: 9780123708762.

Haynes, Brian C. et al. (2013). “Mapping functional transcription factor networks from gene expression data”. In: *Genome Research* 23.8, pp. 1319–1328. ISSN: 1088-9051. DOI: `10.1101/gr.150904.112`.

Hemme, Dorothea et al. (2014). “Systems-wide analysis of acclimation responses to long-term heat stress and recovery in the photosynthetic model organism Chlamydomonas reinhardtii”. In: *The Plant cell* 26.11, pp. 4270–4297. DOI: `10.1105/tpc.114.130997`.

Hoehndorf, Robert, Paul N. Schofield, and Georgios V. Gkoutos (2015). “The role of ontologies in biological and biomedical research: a functional perspective”. In: *Briefings in Bioinformatics* 16.6, pp. 1069–1080. ISSN: 1467-5463. DOI: `10.1093/bib/bbv011`.

Hong, Huixiao et al. (2013). “Estimating relative noise to signal in DNA microarray data”. In: *International journal of bioinformatics research and applications* 9.5, pp. 433–448. ISSN: 1744-5485. DOI: `10.1504/IJBRA.2013.056085`.

Hu, Min and Kornelia Polyak (2006). “Serial analysis of gene expression”. In: *Nature protocols* 1.4, pp. 1743–1760. DOI: `10.1038/nprot.2006.269`.

Huala, E. et al. (2001). “The Arabidopsis Information Resource (TAIR): A Comprehensive Database and Web-Based Information Retrieval, Analysis, and Visualization System for a Model Plant”. In: *Nucleic acids research* 29.1. ISSN: 0305-1048. DOI: `10.1093/nar/29.1.102`. URL: `https://pubmed.ncbi.nlm.nih.gov/11125061/`.

Huang, Xinyu et al. (2020). “A survey of community detection methods in multilayer networks”. In: *Data Mining and Knowledge Discovery*. ISSN: 1384-5810. DOI: `10.1007/s10618-020-00716-6`.

Ishiguchi, Hiroshi et al. (2004). “ZNF143 activates gene expression in response to DNA damage and binds to cisplatin-modified DNA”. In: *International journal of cancer* 111.6, pp. 900–909. ISSN: 0020-7136. DOI: `10.1002/ijc.20358`.

Kaimal, Vivek et al. (2010). “ToppCluster: a multiple gene list feature analyzer for comparative enrichment clustering and network-based dissection of biological systems”. In: *Nucleic acids research* 38.Web Server issue, W96–W102. DOI: `10.1093/nar/gkq418`.

Kanehisa, M. and S. Goto (2000). “KEGG: kyoto encyclopedia of genes and genomes”. In: *Nucleic acids research* 28.1, pp. 27–30. ISSN: 0305-1048. DOI: `10.1093/nar/28.1.27`.

Karl V Voelkerding, Shale A Dames, and Jacob D Durtschi (2009). “Next-Generation Sequencing: From Basic Research to Diagnostics”. In: *Clinical Chemistry* 55.4, pp. 641–658. ISSN: 1530-8561. DOI: `10.1373/clinchem.2008.112789`. URL: `https://www.researchgate.net/publication/24043867_Next-Generation_Sequencing_From_Basic_Research_to_Diagnostics`.

Kircher, Martin and Janet Kelso (2010). “High-throughput DNA sequencing–concepts and limitations”. In: *BioEssays : news and reviews in molecular, cellular and developmental biology* 32.6, pp. 524–536. ISSN: 0265-9247. DOI: `10.1002/bies.200900181`.

Kitano, Hiroaki (2002). “Computational systems biology”. In: *Nature* 420.6912, pp. 206–210. ISSN: 0028-0836. DOI: `10.1038/nature01254`.

Klebanov, Lev and Andrei Yakovlev (2007). “How high is the level of technical noise in microarray data?” In: *Biology direct* 2, p. 9. DOI: `10.1186/1745-6150-2-9`.

Klie, Sebastian and Zoran Nikoloski (2012). “The Choice between MapMan and Gene Ontology for Automated Gene Function Prediction in Plant Science”. In: *Frontiers in genetics* 3, p. 115. DOI: `10.3389/fgene.2012.00115`.

Klipp, E. et al. (2016). *Systems biology: A textbook / Edda Klipp, Wolfram Liebermeister, Christoph Wierling, Axel Kowald*. Second, completely revised and enlarged edition. Weinheim: Wiley-VCH. ISBN: 9783527675678.

Knepper, Mark A. (2012). “Systems biology in physiology: the vasopressin signaling network in kidney”. In: *American journal of physiology. Cell physiology* 303.11, pp. C1115–24. DOI: `10.1152/ajpcell.00270.2012`.

Kramer, Michael et al. (2014). “Inferring gene ontologies from pairwise similarity data”. In: *Bioinformatics* 30.12, pp. i34–42. ISSN: 1367-4803. DOI: `10.1093/bioinformatics/btu282`.

Lancaster, Peter and Kştutis Sălkauskas (1990). *Curve and Surface Fitting. An Introduction*. London: Academic Press. ISBN: 0124360602.

Leifeld, T. et al. (2019). “Curve form based quantization of short time series data”. In: *2019 18th European Control Conference (ECC)*, pp. 3710–3715. DOI: `10.23919/ECC.2019.8795870`.

Levandowsky, Michael and David Winter (1971). “Distance between Sets”. In: *Nature* 234.5323, pp. 34–35. ISSN: 1476-4687. DOI: `10.1038/234034a0`.

Lockhart, D. J. et al. (1996). “Expression monitoring by hybridization to high-density oligonucleotide arrays”. In: *Nature biotechnology* 14.13, pp. 1675–1680. DOI: `10.1038/nbt1296-1675`.

Lukas, Mark A., Frank R. de Hoog, and Robert S. Anderssen (2016). “Practical use of robust GCV and modified GCV for spline smoothing”. In: *Computational Statistics* 31.1, pp. 269–289. ISSN: 0943-4062. DOI: `10.1007/s00180-015-0577-7`.

Maier, T., M. Güell, and L. Serrano (2009). “Correlation of mRNA and protein in complex biological samples”. In: *FEBS letters* 583.24. ISSN: 1873-3468. DOI: `10.1016/j.febslet.2009.10.036`. URL: `https://pubmed.ncbi.nlm.nih.gov/19850042/`.

Malone, John H. and Brian Oliver (2011). “Microarrays, deep sequencing and the true measure of the transcriptome”. In: *BMC biology* 9, p. 34. DOI: `10.1186/1741-7007-9-34`.

Marioni, John C. et al. (2008). “RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays”. In: *Genome research* 18.9, pp. 1509–1517. ISSN: 1088-9051. DOI: 10.1101/gr.079558.108.

Masseroli, Marco (2019). “Biological and Medical Ontologies: Introduction”. In: *Encyclopedia of Bioinformatics and Computational Biology*. Ed. by Shoba Ranganathan, Kenta Nakai, and Christian Schönbach. Amsterdam: Elsevier, pp. 813–822. ISBN: 9780128114322. DOI: 10.1016/B978-0-12-809633-8.20395-6.

Matsumura, Hideo et al. (2005). “SuperSAGE”. In: *Cellular microbiology* 7.1, pp. 11–18. ISSN: 1462-5814. DOI: 10.1111/j.1462-5822.2004.00478.x.

Meehan, Terrence F. et al. (2011). “Logical development of the cell ontology”. In: *BMC bioinformatics* 12, p. 6. DOI: 10.1186/1471-2105-12-6.

Meng, Chen et al. (2016). “Dimension reduction techniques for the integrative analysis of multi-omics data”. In: *Briefings in Bioinformatics* 17.4, pp. 628–641. ISSN: 1467-5463. DOI: 10.1093/bib/bbv108.

Merchant, Sabeeha S. et al. (2007). “The Chlamydomonas Genome Reveals the Evolution of Key Animal and Plant Functions”. In: *Science (New York, N.Y.)* 318.5848, pp. 245–250. ISSN: 0036-8075. DOI: 10.1126/science.1143609.

Meyer, Mary C. (2012). “Constrained penalized splines”. In: *Canadian Journal of Statistics* 40.1, pp. 190–206. ISSN: 03195724. DOI: 10.1002/cjs.10137.

Mühlhaus, Timo et al. (2011). “Quantitative shotgun proteomics using a uniform ^{15}N-labeled standard to monitor proteome dynamics in time course experiments reveals new insights into the heat stress response of Chlamydomonas reinhardtii”. In: *Molecular & cellular proteomics : MCP* 10.9, p. M110.004739. DOI: 10.1074/mcp.M110.004739.

Nagl, Walter, Vera Hemleben, and Friedrich Ehrendorfer (1979). *Genome and Chromatin: Organization, Evolution, Function: Symposium, Kaiserslautern, October 13-15, 1978*. Vol. 2. Plant Systematics and Evolution, Entwicklungsgeschichte und Systematik der Pflanzen, 0172-6668. Vienna: Springer Vienna. ISBN: 3709185580.

Nainggolan, Rena et al. (Nov. 2019). “Improved the Performance of the K-Means Cluster Using the Sum of Squared Error (SSE) optimized by using the Elbow Method”. In: *Journal of Physics: Conference Series* 1361.1, p. 012015. DOI: 10.1088/1742-6596/1361/1/012015. URL: https://doi.org/10.1088/1742-6596/1361/1/012015.

Natale, Darren A. et al. (2017). “Protein Ontology (PRO): enhancing and scaling up the representation of protein entities”. In: *Nucleic Acids Research* 45.D1, pp. D339–D346. ISSN: 0305-1048. DOI: `10.1093/nar/gkw1075`. URL: `https://academic.oup.com/nar/article/45/D1/D339/2605841?login=true`.

Newman, M. E. J. and M. Girvan (2004). “Finding and evaluating community structure in networks”. In: *Physical review. E, Statistical, nonlinear, and soft matter physics*, p. 026113. ISSN: 1539-3755. DOI: `10.1103/PhysRevE.69.026113`.

Nicholson, Jeremy K. and John C. Lindon (2008). “Systems biology: Metabonomics”. In: *Nature* 455.7216, pp. 1054–1056. ISSN: 0028-0836. DOI: `10.1038/4551054a`.

Noguera-Solano, Ricardo, Rosaura Ruiz-Gutierrez, and Juan Manuel Rodriguez-Caso (2013). “Genome: twisting stories with DNA”. In: *Endeavour* 37.4, pp. 213–219. DOI: `10.1016/j.endeavour.2013.05.003`.

Oshlack, Alicia and Matthew J. Wakefield (2009). “Transcript length bias in RNA-seq data confounds systems biology”. In: *Biology direct* 4, p. 14. DOI: `10.1186/1745-6150-4-14`.

Paczkowska, Marta et al. (2020). “Integrative pathway enrichment analysis of multivariate omics data”. In: *Nature communications*, p. 735. DOI: `10.1038/s41467-019-13983-9`.

Pinu, Farhana R. et al. (2019). “Systems Biology and Multi-Omics Integration: Viewpoints from the Metabolomics Research Community”. In: *Metabolites* 9.4. ISSN: 2218-1989. DOI: `10.3390/metabo9040076`.

Poole, David L. and Alan K. Mackworth (2017). *Artificial intelligence: Foundations of computational agents*. 2nd edition. Cambridge et al.: Cambridge University Press. ISBN: 9781107195394.

Reimand, Jüri et al. (2007). “g:Profiler—a web-based toolset for functional profiling of gene lists from large-scale experiments”. In: *Nucleic acids research* 35.Web Server issue, W193–200. DOI: `10.1093/nar/gkm226`.

Rosse, Cornelius and José L. V. Mejino (2008). “The Foundational Model of Anatomy Ontology”. In: *Anatomy ontologies for bioinformatics*. Ed. by Albert G. Burger, Duncan Davidson, and Richard Baldock. Vol. 6. Computational Biology. London: Springer, pp. 59–117. ISBN: 978-1-84628-884-5. DOI: `10.1007/978-1-84628-885-2{\textunderscore}4`.

Rothberg, Jonathan M. et al. (2011). “An integrated semiconductor device enabling non-optical genome sequencing”. In: *Nature* 475.7356, pp. 348–352. ISSN: 0028-0836. DOI: `10.1038/nature10242`.

Sanger, F. and A. R. Coulson (1975). “A rapid method for determining sequences in DNA by primed synthesis with DNA polymerase”. In: *Journal of Molecular Biology* 94.3, pp. 441–448. ISSN: 0022-2836. DOI: `10.1016/0022-2836(75)90213-2`. URL: `http://www.sciencedirect.com/science/article/pii/0022283675902132`.

Sauer, Uwe, Matthias Heinemann, and Nicola Zamboni (2007). “Genetics. Getting closer to the whole picture”. In: *Science (New York, N.Y.)* 316.5824, pp. 550–551. DOI: 10.1126/science.1142502.

Schneider, Maria V. and Sandra Orchard (2011). “Omics technologies, data and bioinformatics principles”. In: *Methods in molecular biology (Clifton, N.J.)* 719, pp. 3–30. DOI: 10.1007/978-1-61779-027-0{\textunderscore}1.

Schroda, Michael (2004). “The Chlamydomonas genome reveals its secrets: chaperone genes and the potential roles of their gene products in the chloroplast”. In: *Photosynthesis research* 82.3, pp. 221–240. DOI: 10.1007/s11120-004-2216-y.

Schroda, Michael, Dorothea Hemme, and Timo Mühlhaus (2015). “The Chlamydomonas heat stress response”. In: *The Plant journal : for cell and molecular biology* 82.3, pp. 466–480. DOI: 10.1111/tpj.12816.

Schulz-Raffelt, Miriam, Mukesh Lodha, and Michael Schroda (2007). “Heat shock factor 1 is a key regulator of the stress response in Chlamydomonas”. In: *The Plant journal : for cell and molecular biology* 52.2, pp. 286–295. DOI: 10.1111/j.1365-313X.2007.03228.x.

Schwacke, Rainer et al. (2019). “MapMan4: A Refined Protein Classification and Annotation Framework Applicable to Multi-Omics Data Analysis”. In: *Molecular plant* 12.6, pp. 879–892. DOI: 10.1016/j.molp.2019.01.003. URL: http://www.sciencedirect.com/science/article/pii/S1674205219300085.

Sedgewick, Robert and Kevin Wayne (2012). *Algorithms*. 4. ed., 3. printing. Upper Saddle River, NJ: Addison-Wesley. ISBN: 9780321573513.

Shahzad, Khuram and Juan J. Loor (2012). “Application of Top-Down and Bottom-up Systems Approaches in Ruminant Physiology and Metabolism”. In: *Current Genomics* 13.5, pp. 379–394. ISSN: 1389-2029. DOI: 10.2174/138920212801619269.

Shannon, C. E. (1948). “A Mathematical Theory of Communication”. In: *Bell System Technical Journal* 27.4, pp. 623–656. ISSN: 00058580. DOI: 10.1002/j.1538-7305.1948.tb00917.x.

Sharma, Manjuli R. et al. (2009). “PSRP1 Is Not a Ribosomal Protein, but a Ribosome-binding Factor That Is Recycled by the Ribosome-recycling Factor (RRF) and Elongation Factor G (EF-G)*”. In: *The Journal of Biological Chemistry* 285.6, pp. 4006–4014. ISSN: 0021-9258. DOI: 10.1074/jbc.M109.062299.

Shendure, J. and H. Ji (2008). “Next-generation DNA sequencing”. In: *Nature biotechnology* 26.10. DOI: 10.1038/nbt1486. URL: https://pubmed.ncbi.nlm.nih.gov/18846087/.

Shendure, Jay et al. (2005). “Accurate multiplex polony sequencing of an evolved bacterial genome”. In: *Science (New York, N.Y.)* 309.5741, pp. 1728–1732. DOI: 10.1126/science.1117389.

Sheynkman, Gloria M. et al. (2016). “Proteogenomics: Integrating Next-Generation Sequencing and Mass Spectrometry to Characterize Human Proteomic Variation”. In: *Annual review of analytical chemistry (Palo Alto, Calif.)* 9.1, pp. 521–545. ISSN: 1936-1327. DOI: `10.1146/annurev-anchem-071015-041722`.

Smith, Barry (2003). “Ontology. In Luciano Floridi (ed.), Blackwell Guide to the Philosophy of Computing and Information.” In: pp. 155–166.

Smith, Barry et al. (2007). “The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration”. In: *Nature biotechnology* 25.11, pp. 1251–1255. DOI: `10.1038/nbt1346`.

Sniedovich, Moshe (op. 2011). *Dynamic programming: Foundations and principles*. 2nd ed. Monographs and textbooks in pure and applied mathematics. Boca Raton: Chapman & Hall/CRC Press. ISBN: 978-0-8247-4099-3.

Subramanian, Aravind et al. (2005). “Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles”. In: *Proceedings of the National Academy of Sciences of the United States of America* 102.43, pp. 15545–15550. ISSN: 0027-8424. DOI: `10.1073/pnas.0506580102`.

Subramanian, Indhupriya et al. (2020). “Multi-omics Data Integration, Interpretation, and Its Application”. In: *Bioinformatics and biology insights* 14, p. 1177932219899051. ISSN: 1177-9322. DOI: `10.1177/1177932219899051`.

Swaminathan, Jagannath et al. (2018). “Highly parallel single-molecule identification of proteins in zeptomole-scale mixtures”. In: *Nature biotechnology*. DOI: `10.1038/nbt.4278`.

Thimm, Oliver et al. (2004). “mapman: a user-driven tool to display genomics data sets onto diagrams of metabolic pathways and other biological processes”. In: *The Plant Journal* 37.6, pp. 914–939. ISSN: 09607412. DOI: `10.1111/j.1365-313X.2004.02016.x`.

Timp, Winston and Gregory Timp (2020). “Beyond mass spectrometry, the next step in proteomics”. In: *Science advances* 6.2, eaax8978. DOI: `10.1126/sciadv.aax8978`.

Tini, Giulia et al. (2019). “Multi-omics integration-a comparison of unsupervised clustering methodologies”. In: *Briefings in Bioinformatics* 20.4, pp. 1269–1279. ISSN: 1467-5463. DOI: `10.1093/bib/bbx167`.

Uschold, Mike and Austin Tate (1998). “Putting ontologies to use”. In: *The Knowledge Engineering Review* 13.1, pp. 1–3. ISSN: 0269-8889. DOI: `10.1017/S0269888998001027`.

van Dam, Jesse C. J. et al. (2019). “The Empusa code generator and its application to GBOL, an extendable ontology for genome annotation”. In: *Scientific data* 6.1, p. 254. DOI: `10.1038/s41597-019-0263-7`.

Velculescu, V. E. et al. (1995). “Serial analysis of gene expression”. In: *Science (New York, N.Y.)* 270.5235, pp. 484–487. DOI: `10.1126/science.270.5235.484`.

Wang, Bo et al. (2019). “Reviving the Transcriptome Studies: An Insight Into the Emergence of Single-Molecule Transcriptome Sequencing”. In: *Frontiers in genetics* 10, p. 384. ISSN: 1664-8021. DOI: `10.3389/fgene.2019.00384`.

Wang, Xinlei et al. (2019). “Combined transcriptomics and proteomics forecast analysis for potential genes regulating the Columbian plumage color in chickens”. In: *PloS one* 14.11, e0210850. DOI: `10.1371/journal.pone.0210850`.

Wanichthanarak, Kwanjeera, Johannes F. Fahrmann, and Dmitry Grapov (2015). “Genomic, Proteomic, and Metabolomic Data Integration Strategies”. In: *Biomarker insights* 10.Suppl 4, pp. 1–6. ISSN: 1177-2719. DOI: `10.4137/BMI.S29511`.

Wasinger, V. C. et al. (1995). “Progress with gene-product mapping of the Mollicutes: Mycoplasma genitalium”. In: *Electrophoresis* 16.7, pp. 1090–1094. ISSN: 0173-0835. DOI: `10.1002/elps.11501601185`.

Weiss, Kenneth M. (2020). “The Four Horsemen of the ’Omicsalypse’: ontology, replicability, probability and epistemology”. In: *Human genetics* 139.1, pp. 115–120. DOI: `10.1007/s00439-019-02007-7`.

Wilkins, Marc (2009). “Proteomics data mining”. In: *Expert review of proteomics* 6.6, pp. 599–603. DOI: `10.1586/epr.09.81`.

Winkler, Hans (1920). *Verbreitung und Ursache der Parthenogenesis im Pflanzen und Tierreiche*. Jena: Fischer.

Wood, S. N. (1994). “Monotonic Smoothing Splines Fitted by Cross Validation”. In: *SIAM Journal on Scientific Computing* 15.5, pp. 1126–1133. ISSN: 1064-8275. DOI: `10.1137/0915069`.

Wu, Xindong et al. (2014). “Data mining with big data: IEEE Transactions on Knowledge and Data Engineering, 26(1), 97-107”. In: DOI: `10.1109/TKDE.2013.109`.

Yan, Jingwen et al. (2018). “Network approaches to systems biology analysis of complex disease: integrative methods for multi-omics data”. In: *Briefings in Bioinformatics* 19.6, pp. 1370–1381. ISSN: 1467-5463. DOI: `10.1093/bib/bbx066`.

Yates, John R., Cristian I. Ruse, and Aleksey Nakorchevsky (2009). “Proteomics by mass spectrometry: approaches, advances, and applications”. In: *Annual review of biomedical engineering* 11, pp. 49–79. DOI: `10.1146/annurev-bioeng-061008-124934`.

Young, Matthew D. et al. (2010). “Gene ontology analysis for RNA-seq: accounting for selection bias”. In: *Genome biology* 11.2, R14. DOI: `10.1186/gb-2010-11-2-r14`.

Yusko, Erik C. et al. (2017). “Real-time shape approximation and fingerprinting of single proteins using a nanopore”. In: *Nature nanotechnology* 12.4, pp. 360–367. DOI: `10.1038/nnano.2016.267`.

Zepeda-Mendoza, Marie Lisandra and Osbaldo Resendis-Antonio (2013). “Hierarchical Agglomerative Clustering”. In: *Encyclopedia of Systems Biology*. Ed. by Werner Dubitzky et al. New York, NY: Springer New York, pp. 886–887. ISBN: 978-1-4419-9863-7. DOI: `10.1007/978-1-4419-9863-7_1371`. URL: `https://doi.org/10.1007/978-1-4419-9863-7_1371`.

A. Supplementary Materials

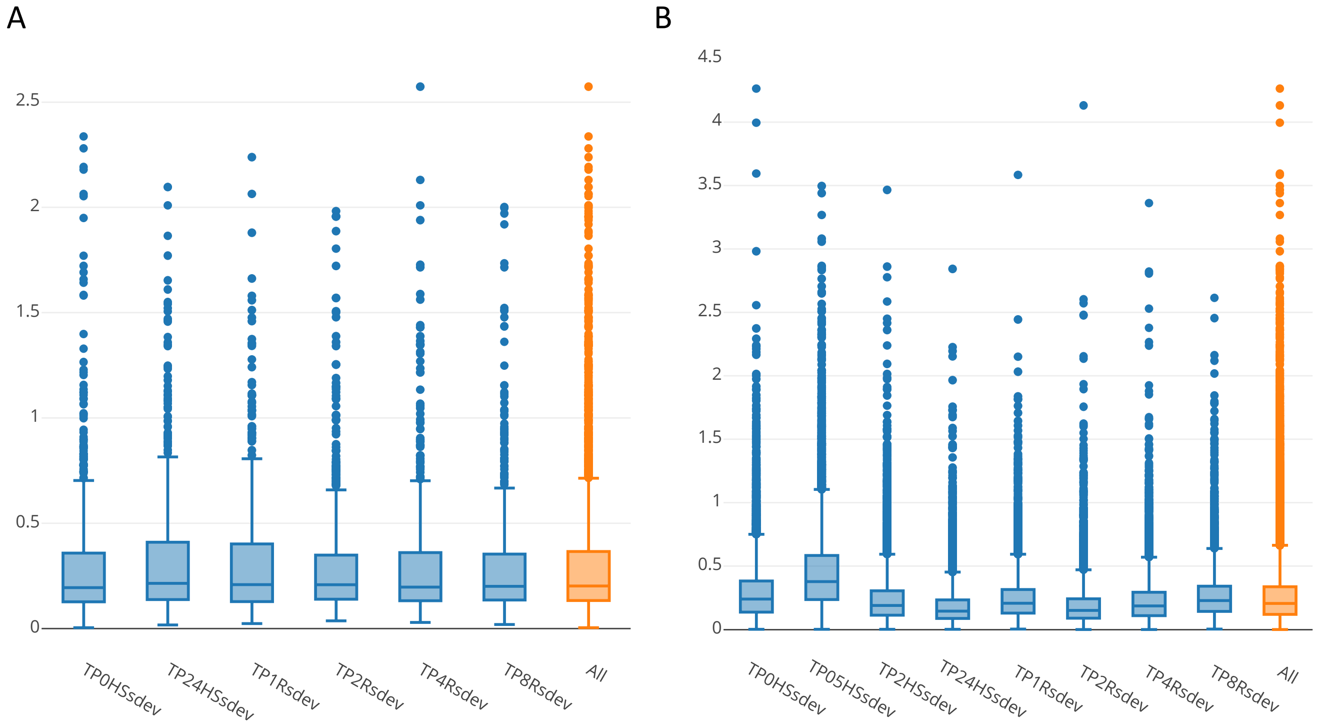


Figure A.1: Standard deviation of the real-world data. Label explanation: TP - time point, HS - heat shock, R - recovery. **A.** Standard deviation of the raw proteome data. **B.** Standard deviation of the raw transcriptome data.

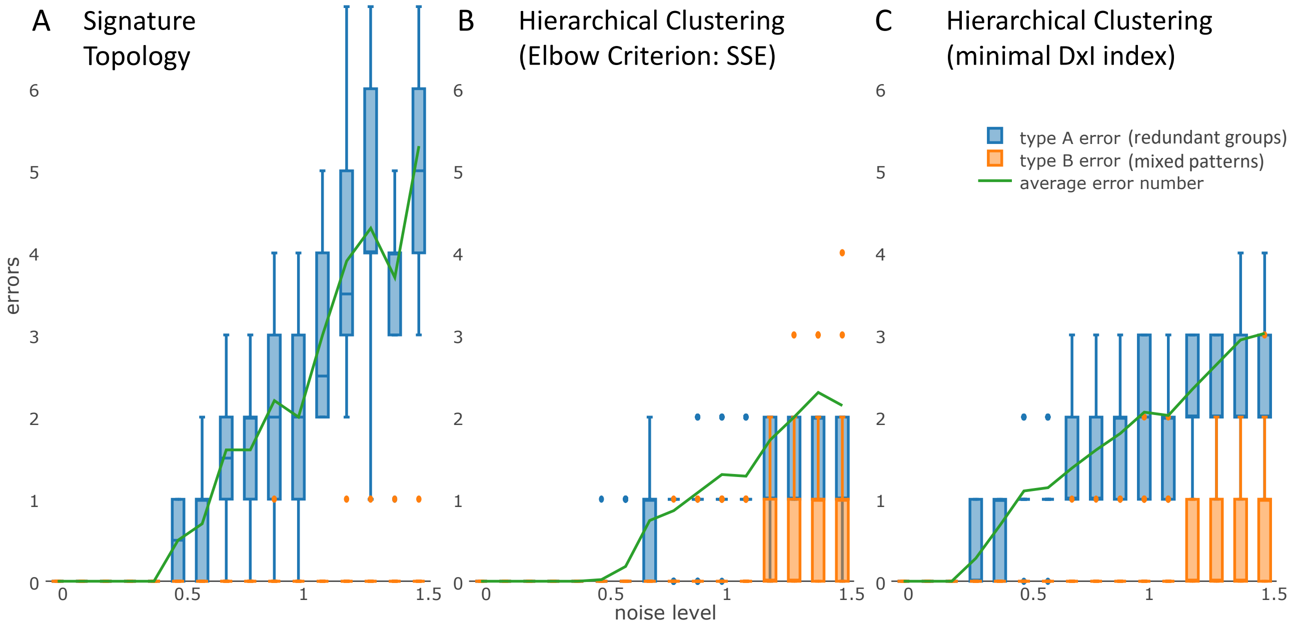


Figure A.2: Noise robustness trial. Number of errors in constructing ST tree compared to optimal structure with increasing noise level σ. Error A (blue) means number of excess groups exclusive of two correct groups. Error B (orange) means number of elements of minority (incorrect) pattern in mixed group, if such group was created. For each noise level number of elements $n = 10$. The number of errors were summed over 10 repetitions of the trial. **A.** Trial results for Signature Topology structures. **B.** Trial results for Hierarhical Clustering structures with number of clusters, chosen by elbow criterion based on sum of squared errors (SSE). **C.** Trial results for Hierarhical Clustering structures with number of clusters, chosen by minimum DxI index.

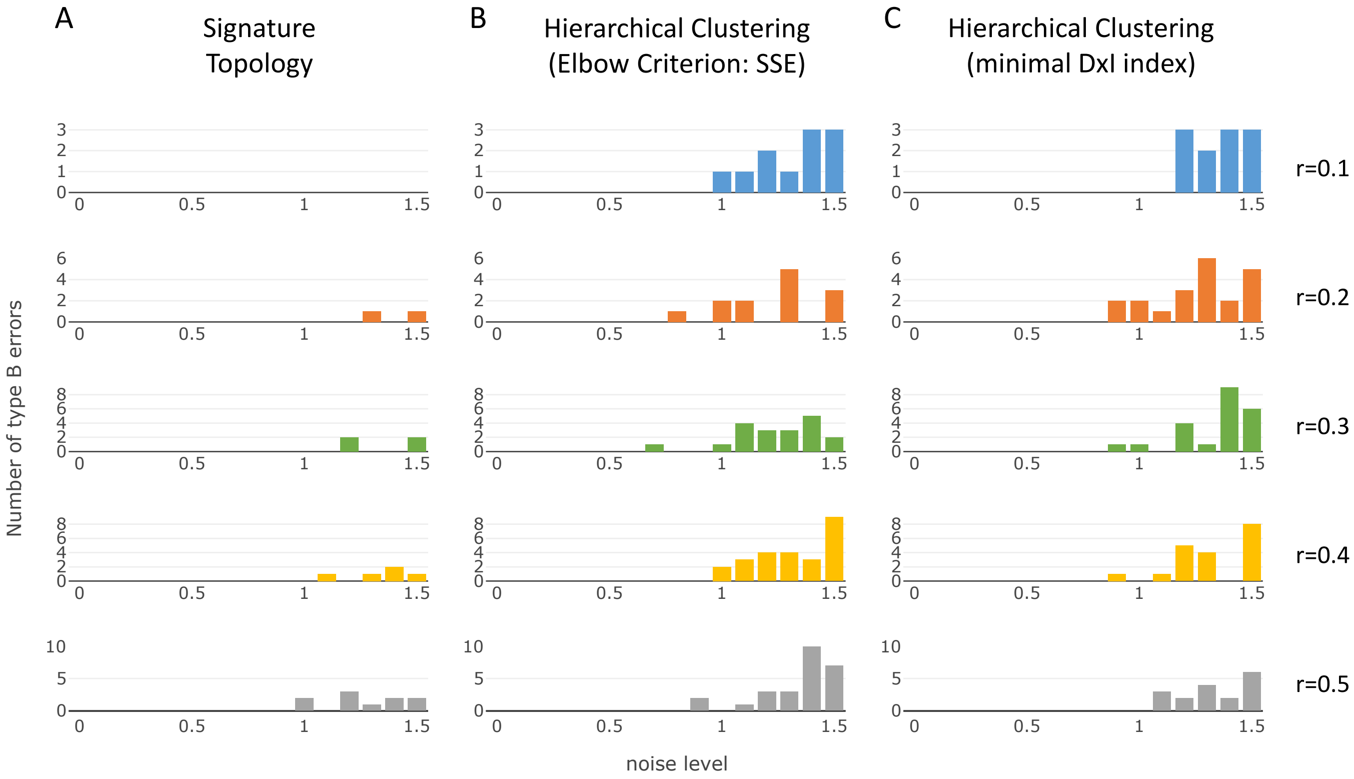


Figure A.3: Minority revealing trial - Number of type B errors, considering 5 different ratios of minor pattern, comparing obtained with three different approaches structures to the optimal structure with increasing noise level σ. Type B error means a number of elements of minority pattern in mixed group, if such group was created. For each noise level and minority ratio $n = 10$. Noise level varied as $\sigma = \{0.0, 0.1, 0.2 \ldots 1.5\}$, minority ratio varied as $r = \{0.1, 0.2, 0.3, 0.4, 0.5\}$. For each parameter combinations, number of elements in the dataset is $n = 10$. The number of errors were summed over 10 repetitions of the trial **A**. Signature Topology. **B**. Hierarchical Clustering with the number of clusters determined with Elbow criterion by SSE. **C**. Hierarchical Clustering with the number of clusters with minimal DxI index (see formula (4.2)).

Table A.1: DxI indices for proteome dataset. DxI index was calculated for each functional group (MapMan bin) separately for following structures: MapMan trees, cut at different levels, Signature Topology structure, Hierarchical clustering (HC) extended into functional tree. For structure complexity, difference in paired samples was found significant between ST tree and minimal HC tree (p-value=1.3e-06), and between ST tree and HC SSE optimal tree (p-value=2.8e-07) according to Wilcoxon signed rank test. For grouping complexity, difference in paired samples was found significant between ST tree and MapMan (p-value=0.001) as well as between ST tree and both HC trees (p-values equal 0.03 and 0.001 for minimal HC and SSE HC, respectively) according to Wilcoxon signed rank test.

	DxI (structure complexity)				DxI (grouping complexity)			
Functional group (bin)	minimal MapMan	Signature Topology	minimal HC	HC (SSE)	minimal MapMan	Signature Topology	minimal HC	HC (SSE)
1	0.454	0.359	0.538	0.574	0.601	0.739	0.759	0.761
2	0.442	0.429	0.905	0.985	0.851	0.791	1.000	1.209
3	0.453	0.545	0.650	0.650	0.658	1.000	0.613	0.613
4	0.317	0.375	0.775	0.792	0.631	1.000	1.000	1.112
5	0.437	0.543	0.664	0.664	0.548	0.780	0.548	0.548
6	0.556	0.556	0.970	0.970	1.000	1.000	1.000	1.155
7	0.538	0.538	1.000	1.079	1.000	0.955	1.000	1.203
8	0.408	0.489	0.723	0.723	0.597	0.549	0.930	0.938
9	0.447	0.547	0.616	0.625	0.589	0.741	0.682	0.703
11	0.357	0.395	0.618	0.618	0.569	0.716	0.741	0.766
12	0.475	0.500	0.715	0.715	0.850	1.000	0.850	0.912
13	0.454	0.335	0.683	0.683	0.646	0.797	0.947	0.968
16	0.400	0.400	0.849	0.904	1.000	0.934	0.855	0.983
17	0.380	0.333	0.960	0.960	0.725	1.000	1.000	1.040
19	0.615	0.615	0.851	0.851	1.000	1.000	1.000	1.064
20	0.400	0.400	0.994	0.994	1.000	1.000	0.789	0.789
21	0.596	0.560	0.581	0.581	0.632	0.594	0.645	0.645
23	0.489	0.420	0.786	0.807	0.919	0.833	1.000	1.078
25	1.000	1.000	0.828	0.837	1.000	1.000	0.768	0.768
26	0.519	0.600	0.783	0.790	0.749	1.000	1.000	1.086
27	0.496	0.493	0.661	0.705	0.787	0.928	0.889	0.911
28	0.443	0.441	0.471	0.567	0.555	0.691	0.553	0.606
29	0.504	0.282	0.476	0.585	0.538	0.689	0.842	0.892
30	0.473	0.563	0.611	0.649	0.535	0.795	0.666	0.697
31	0.574	0.426	0.697	0.716	0.805	0.820	0.835	0.888
33	0.545	0.545	0.861	1.018	1.000	1.000	0.835	0.995
34	0.569	0.583	0.762	0.782	0.862	1.000	0.948	1.024
median	0.473	0.493	0.723	0.723	0.749	0.928	0.850	0.912

Table A.2: DxI indices for transcriptome dataset. DxI index was calculated for each functional group (MapMan bin) separately for following structures: MapMan trees, cut at different levels, Signature Topology structure, Hierarchical clustering (HC) extended into functional tree. For structure complexity, difference in paired samples was found significant between ST tree and MapMan (p-value 0.0002) and minimal HC tree (p-value=0.002), and between ST tree and HC SSE optimal tree (p-value=2.9e-05) according to Wilcoxon signed rank test. For grouping complexity, difference in paired samples was found significant between ST tree and MapMan (p-value=3.4e-06) as well as between ST tree and both HC trees (p-values equal 0.01 and 0.0007 for minimal HC and SSE HC, respectively) according to Wilcoxon signed rank test.

	DxI (structure complexity)				DxI (grouping complexity)			
Functional group (bin)	minimal MapMan	Signature Topology	minimal HC	HC (SSE)	minimal MapMan	Signature Topology	minimal HC	HC (SSE)
1	0.414	0.340	0.494	0.505	0.518	0.831	0.636	0.658
2	0.457	0.366	0.662	0.669	0.801	0.851	0.834	0.841
3	0.615	0.502	0.573	0.575	0.681	0.965	0.687	0.687
4	0.529	0.455	0.692	0.700	0.824	0.927	0.988	1.032
5	0.589	0.706	0.786	0.786	0.697	1.000	0.679	0.748
6	0.590	0.643	0.840	0.855	0.635	1.000	0.879	0.977
7	0.671	0.492	0.742	0.743	1.032	0.895	0.969	0.990
8	0.423	0.436	0.665	0.665	0.694	0.735	0.882	0.882
9	0.479	0.461	0.489	0.508	0.540	0.827	0.580	0.602
10	0.555	0.424	0.738	0.738	0.729	0.968	0.808	0.808
11	0.640	0.392	0.705	0.705	0.850	0.860	0.945	0.948
12	0.489	0.489	0.883	0.904	1.000	0.913	1.000	1.145
13	0.530	0.308	0.678	0.695	0.771	0.869	0.979	0.997
14	0.582	0.556	0.648	0.655	0.814	1.000	0.689	0.693
15	0.566	0.563	0.656	0.666	0.720	1.000	0.656	0.685
16	0.535	0.378	0.702	0.711	0.773	0.814	0.959	0.983
17	0.534	0.388	0.760	0.775	0.839	0.830	0.956	0.966
18	0.679	0.546	0.723	0.753	0.879	0.949	0.870	0.884
19	0.579	0.629	0.680	0.712	0.647	0.793	0.908	0.963
20	0.516	0.424	0.603	0.626	0.803	0.978	0.708	0.722
21	0.569	0.468	0.617	0.618	0.657	0.908	0.717	0.774
22	0.496	0.440	0.746	0.778	0.840	0.891	0.889	1.028
23	0.596	0.374	0.585	0.589	0.649	0.799	0.725	0.727
24	0.757	0.867	0.651	0.677	0.780	1.000	0.633	0.651
25	0.651	0.711	0.718	0.742	0.734	0.993	0.747	0.843
26	0.569	0.524	0.590	0.594	0.623	0.935	0.772	0.780
27	0.532	0.397	0.517	0.587	0.584	0.922	0.742	0.754
28	0.596	0.447	0.501	0.514	0.632	0.995	0.546	0.546
29	0.547	0.323	0.510	0.660	0.585	0.928	0.689	0.753
30	0.565	0.466	0.567	0.570	0.618	0.978	0.682	0.682
31	0.603	0.335	0.519	0.587	0.622	0.879	0.669	0.682
32	1.000	1.000	1.000	1.051	1.000	1.000	0.993	0.993
33	0.652	0.507	0.438	0.463	0.665	0.917	0.451	0.473
34	0.548	0.514	0.567	0.590	0.591	0.959	0.735	0.744
median	0.567	0.464	0.659	0.667	0.708	0.925	0.745	0.777

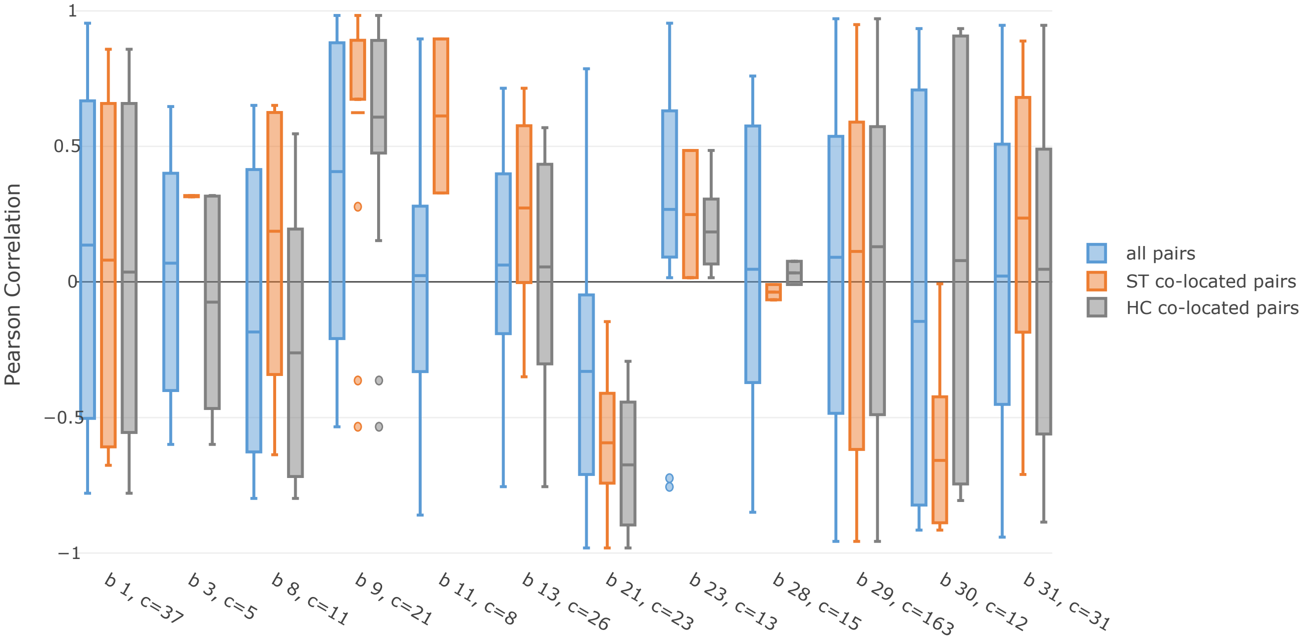


Figure A.4: Pearson correlation between protein-transcript pairs, for the whole common set (blue) and between elements that were grouped together in Signature Topology structure (orange) or in clusters after applying hierarchical clustering (gray). X-axis is labeled for functional bin number (b) and size of its common set (c).

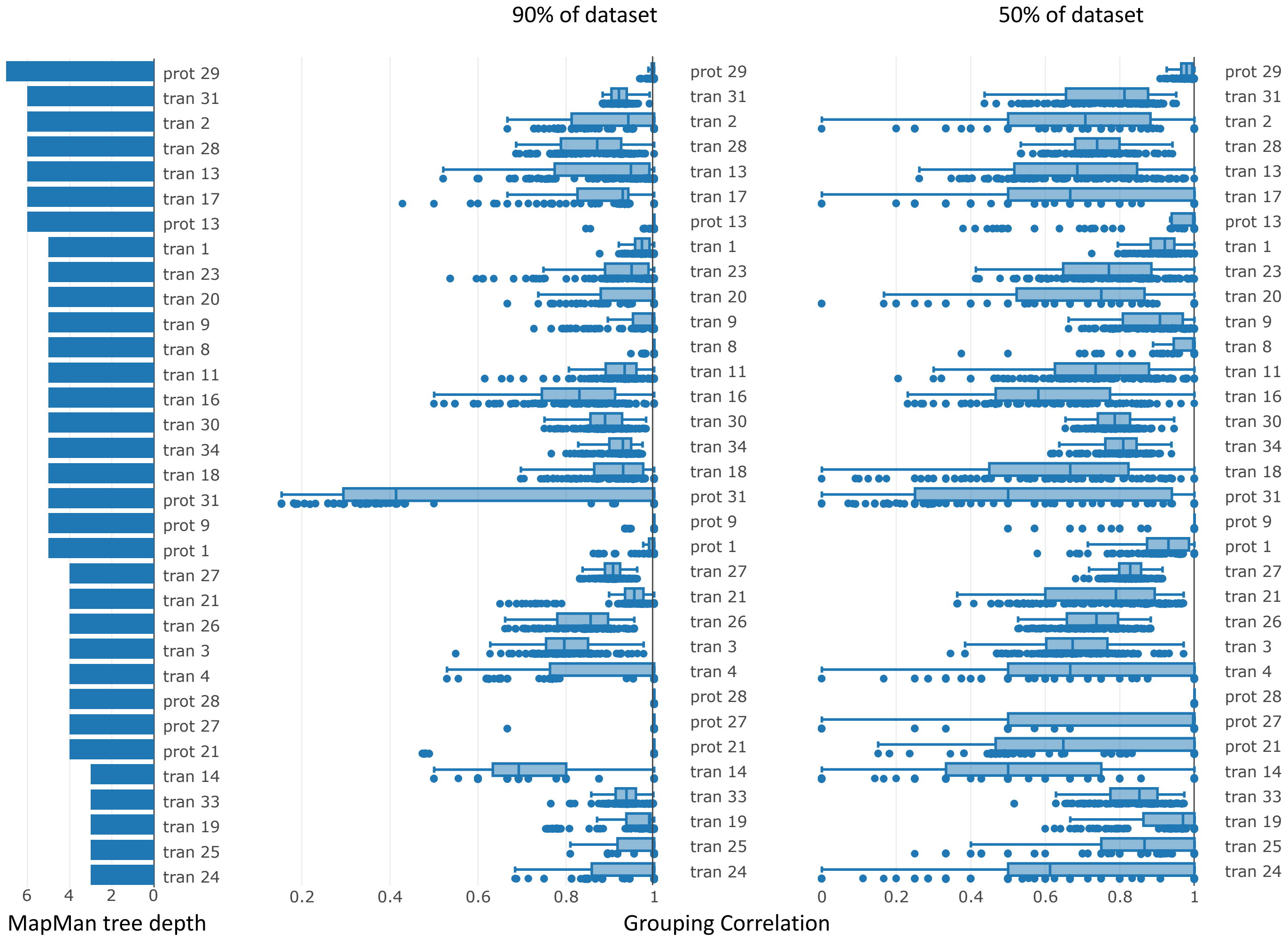


Figure A.5: Robustness of the ST structure against data removal: The left plot shows the depth of the corresponding bin tree from proteome (prot) and transcriptome (tran) datasets. The middle and the right plots show the group correlation between the original ST structure and ST structure of the randomly reduced dataset to 90% and 50% of the original size, correspondingly. 100 repetitions for each bin and each reducing fraction are presented.

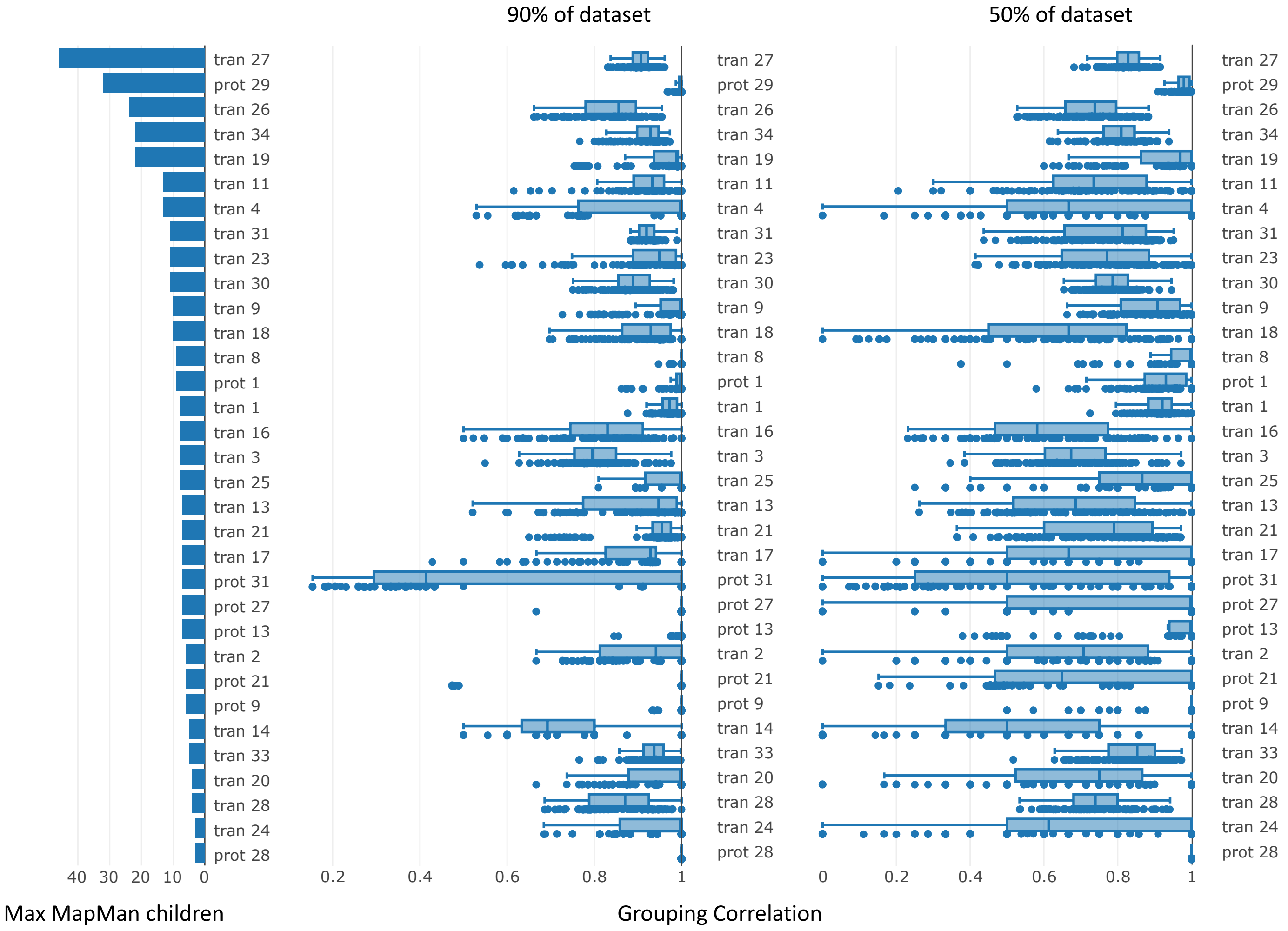


Figure A.6: Robustness of the ST structure against data removal: The left plot shows the size of the corresponding bins from proteome (prot) and transcriptome (tran) datasets. The middle and the right plots show the group correlation between the original ST structure and ST structure of the randomly reduced dataset to 90% and 50% of the original size, correspondingly. 100 repetitions for each bin and each reducing fraction are presented.

Table A.3: Signature Topology Output list for proteome bin 9.

Group Label	Group Size	Step Gain from root pro element	Cluster Density	Elements
9.1.1.p13	1	5.89	0	Cre13.g571150.t1.1
9.1.2.p0	1	11.19	0	Cre12.g496750.t1.1
9.1.2.p20	1	9.39	0	Cre16.g679500.t1.1
9.2.2	1	12.64	0	Cre05.g232200.t1.1
9.9.p2	1	11.64	0	Cre17.g721300.t1.1
9.1.1.5	2	8.95	1.33	Cre12.g516450.t1.1; Cre09.g415850.t1.1
9.1.2.mix|p11|p7	2	5.59	0.93	Cre01.g061100.t1.1; Cre03.g178250.t1.1

9.7	3	6.36	2.39	Cre13.g567600.t1.1;
				Cre16.g691850.t1.1;
				Cre06.g304350.t1.1

9.9.mix|p14|p16|p6|p15	4	4.43	1.22	Cre16.g680000.t1.1;
				Cre13.g581600.t1.1;
				Cre02.g116750.t1.1;
				Cre17.g698000.t1.1

9.mix|5|6	5	4.29	1.27	Cre03.g156950.t1.1;
				Cre06.g262700.t1.1;
				Cre15.g638500.t1.1;
				Cre11.g468950.t1.1;
				Cre12.g522600.t1.1

Table A.4: Signature Topology Output list for transcriptome bin 9.

Group Label	Group Size	Step Gainfrom root pro Element	Cluster Density	Elements
9.1.1.p2	1	13.96	0	Cre02.g088000
9.1.1.p39	1	17.14	0	Cre08.g378550
9.1.1.p48	1	17.18	0	Cre02.g076750
9.1.2.p17	1	14.56	0	Cre16.g679500
9.1.2.p88	1	18.2	0	Cre16.g681700
9.2.2.p35	1	16.45	0	Cre27.g775750
9.2.2.p62	1	18.63	0	Cre01.g072650
9.2.2.p89	1	16.1	0	Cre05.g232200
9.2.3	1	14.24	0	Cre16.g671000
9.7.p99	1	18.07	0	Cre04.g228450
9.8.p54	1	13.53	0	Cre06.g278750
9.8.p64	1	15.28	0	Cre06.g257550
9.8.p66	1	13.08	0	Cre15.g641200
9.9.p45	1	13.49	0	Cre18.g743700
9.99.p36	1	15.12	0	Cre18.g747950
9.1.1.mix|p53|p84	2	13.51	2.5	Cre18.g745500; Cre12.g555150
9.2.2.mix|p101|p31	2	16.12	0.79	Cre33.g782800; Cre27.g775700
9.5.mix|p50|p41	2	17.17	1.96	Cre01.g052050; Cre24.g768900
9.7.mix|p79|p91	2	16.08	2.21	Cre14.g617200; Cre14.g617151
9.9.mix|p5|p43	2	15.17	1.56	Cre17.g726250; Cre10.g419050
9.1.2.mix|p103|p7|p63	3	16.67	2.31	Cre14.g617800; Cre02.g125850; Cre06.g278550
9.3	3	16.4	2.5	Cre16.g687950; Cre02.g094300; Cre27.g775600
9.99.mix|p13|p57|p82	3	16.02	2.35	Cre10.g429800; Cre12.g559950; Cre03.g154850
9.1.2.mix|p87|p90|p102|p98	4	12.64	2.04	Cre15.g636400; Cre03.g178250; Cre22.g763400; Cre12.g523500
9.5.mix|p56|p23|p71|p11|p14	5	10.26	2.02	Cre06.g262700; Cre12.g523850; Cre03.g156950; Cre01.g051900; Cre11.g468950
9.7.mix|p80|p72|p34|p76|p58|p77	6	12.64	2.34	Cre02.g082700; Cre16.g690200; Cre19.g757300; Cre17.g732850; Cre01.g055550; Cre14.g617150
9.mix|4|6	7	11.68	3.21	Cre09.g395950; Cre03.g169550; Cre12.g525700; Cre13.g575000; Cre12.g522600; Cre16.g690050; Cre15.g638500
9.7.mix|p68|p1|p86|p47|p19|p74|p93|p59|p95	9	9.81	2.36	Cre12.g537450; Cre06.g304350; Cre01.g049500; Cre03.g157700; Cre03.g154350; Cre04.g221700; Cre16.g691850; Cre13.g567600; Cre10.g418600
9.1.1.mix|5|p9|p8|p10|p24|p37|p75|p22|p67	11	9.41	2.34	Cre12.g516450; Cre06.g293850; Cre09.g415850; Cre03.g204650; Cre12.g555250; Cre05.g240800; Cre02.g140350; Cre13.g568800; Cre16.g664600; Cre07.g333900; Cre13.g571150
9.9.mix|p46|p12|p33|p40|p28|p73|p27|p78|p44|p6|p69|p97|p100	13	7.68	1.32	Cre09.g415550; Cre10.g420700; Cre07.g340350; Cre16.g680000; Cre01.g018800; Cre15.g635850; Cre07.g338050; Cre09.g395350; Cre02.g116750; Cre17.g721300; Cre17.g698000; Cre09.g402300; Cre13.g581600
9.1.2.mix|p96|p16|p18|p29|p30|p38|p42|p51|p94|p0|p15|p20|p81|p85|p92	15	8.54	2.9	Cre09.g405850; Cre08.g378050; Cre07.g327400; Cre12.g511200; Cre06.g267200; Cre12.g535950; Cre13.g597700; Cre02.g100200; Cre10.g450400; Cre12.g492300; Cre01.g061100; Cre10.g422600; Cre12.g496750; Cre08.g378900; Cre10.g434450

Table A.5: Signature Topology Output list for transcriptome bin 29.

Due to the table size, it is attached as a separate file ST_outputList_tran29.txt

Declaration

I, Nathan Mikhaylenko, declare that the PhD thesis entitled “Signature Topology: functional analysis of -omics data” is no more than 100,000 words in length including quotes and exclusive of tables, figures, appendices, bibliography, references and footnotes. This thesis contains no material that has been submitted previously, in whole or in part, for the award of any other academic degree or diploma. Except where otherwise indicated, this thesis is entirely my own work.

Acknowledgments

I would like to thank my professor Timo Mühlhaus for supplying valuable ideas and giving me motivation, challenge and support. I am glad to be a part of Biotech faculty and will remember our Christmas parties. I thank my colleagues from Computational System Biology team for friendly working space and company. Especially I am grateful to Sabrina Gödel and Patrik Blume for emotional support and great time together on our side of the room.

I thank my friends and relatives for being there for me and believing in me through all these years.

I am endlessly grateful to my wife for patience and just right combination of quiet support and fierce encouragements that brought me to the end of this long and successful journey.

Curriculum vitae

Nathan Mikhaylenko

Bio-Informatiker

nathan.mikhay@gmail.com

Ausbildung

10/2015 – 08/2022

Technische Universität Kaiserslautern, Deutschland
PhD in Computational System Biology
- Omics Data Analysis
- Plant Genomics
- Deep Neural Network

10/2014 – 09/2015

Technische Universität Kaiserslautern, Deutschland
Master in Molekular- und Zellbiologie
- VP in Neurophysiologie
- VP in Hefe Zellbiologie

07/2007 – 06/2012

Lomonosov Moscow State University, Russland
Dipl.-Biologe (Schwerpunkt: Zoologie)
- Genome-Labor-Techniken
- Computational Analysis von Audiodaten

Erfahrung

02/2020 – 05/2023

Wissenschaftlicher Mitarbeiter: Bioinformatiker,
Institute für Medizininformatik und Biometrie (IMB) TU Dresden, Dresden
- Analysis der onkologischen Omics-Daten
- Unterstützung individueller Medizin

12/2014 – 12/2015

HiWi-Mitarbeiter,
BioTech-Lehrstuhl an der TU Kaiserslautern
HiWi in Systembiologie mit Kenntnissen in:
- Omics Data Analysis
- Dynamical Programming
- Ontologies Development

05/2012 – 07/2014

Wissenschaftlicher Mitarbeiter: Biomathematiker,
Institute for System Biology, Moskau, Russland
- Modellierung von biochemischen Systemen zur Analyse und Vorhersage von biochemischen Eigenschaften neuer pharmakologischer Agenten
- Differentialgleichung-basierte Modellierung
- Biochemische- und Pharmakokinetik

10/2010 – 05/2012

Museumsprogrammleiter,
Tsaritsyno Museum-Reservat, Moskau, Russland
- Vorbereitung und Durchführung von ökologischen Programmen im Park und im botanischen Glashaus
- Interaktive Vorträge

Fähigkeiten

Sprachkenntnisse

- ◦ Deutsch (C1)
- ◦ Englisch (C2)
- ◦ Russisch (Muttersprache)

Kommunikative Fähigkeiten

- ◦ Mündliche Präsentation
- ◦ Poster Session
- ◦ Team-Arbeit

Programmiersprachen

- ◦ F#
- ◦ R
- ◦ C#
- ◦ SQL
- ◦ Python
- ◦ Matlab

Software

- ◦ Visual Studio
- ◦ SQLite
- ◦ LaTeX
- ◦ GitHub
- ◦ GePhi
- ◦ Linux

Wissenschaftliche Teilnahme

09/2022	***Wissenschaftliche Publikation,*** **PLOS One, journal** Computational gene expression analysis reveals distinct molecular subgroups of T-cell prolymphocytic leukemia
07/2021	***Konferenz Poster,*** **ISMB/ECCB, virtual event** Gene expression profiling reveals distinct molecular subgroups of T-cell prolymphocytic leukemia
07/2017	***Konferenz Poster,*** **iSCB, Prag** Dynamic ontology analysis: functional-based analysis of stress reaction

Soziales Engagement

02/2018 – 04/2018	***Studenteninitiative für Kinder Kaiserslautern e.V.,*** **Kottenschule, Kaiserslautern** Unterstützen der Grundschüler bei Hausaufgaben
06/2017	***Explore Science,*** **Louisenpark, Mannheim** Standbetreuung für Heilbronn Museum Experimenta
09/2008 – 05/2013	***Karate Trainer,*** **Schule Nr 510, Moskau, Russland** Karate Unterricht für Schulkinder