Bielefeld University Faculty of Technology

- VANESA -

A bioinformatics software application for the modeling, visualization, analysis, and simulation of biological networks in systems biology applications

PhD Thesis

Doctor of Engineering

Author:

Sebastian Jan Janowski

May 13, 2013

International Graduate Program Bioinformatics of Signaling Networks

PhD Thesis at the International Graduate Program Bioinformatics of Signaling Networks. Submitted to the Faculty of Technology at the Bielefeld University, Germany to obtain the doctorate Doctor of Engineering (Dr.-Ing.).

Title:
VANESA - A bioinformatics software application for the modeling, visualization, analysis, and simulation of biological networks in systems biology applications.

Author: M.Sc. Sebastian Jan Janowski
Completion of Work: May 13, 2013

Supervisors:

Prof. Dr. Ralf Hofestädt
Head of the Department of Bioinformatics and Medical Informatics
Prof. Dr. Christian Kaltschmidt
Head of the Department of Cell Biology
Prof. Dr. Barbara Kaltschmidt
Head of the Department of Neurobiology
Prof. Dr. Jens Stoye
Dean of the Faculty of Technology
Head of the Research Group Genome Informatics

Graduate School Responsible:

Prof. Dr. Karl-Josef Dietz
Head of the Graduate School
Head of the Department of Cellular and Developmental Biology
Dr. Kolja Henckel
Coordinator of the Graduate School

Address of the Responsible Institution:

Bielefeld University
Faculty of Technology
Universitässtraße 25
33615 Bielefeld
Germany

Abstract

This work presents VANESA, a powerful and easy-to-use modeling software. The software application is laid out to support scientists from the natural sciences in the modeling and analysis of biological systems to better understand biological processes. Therefore, it combines different fields of research, such as information fusion, modeling, analysis, simulation, and network visualization, which are some of the most important areas in bioinformatics and systems biology.

Using VANESA, scientists have the possibility to automatically reconstruct important biomedical systems with information from the databases KEGG, MINT, IntAct, HPRD, and BRENDA. Additionally, experimental results can be expanded with database information to better analyze the investigated elements and processes in an overall context. This results in biological models, which enable scientists to focus on complex interactions and/or to investigate the role of individual components and processes within whole biological systems. Furthermore, users have the possibility to use graph theoretical approaches in VANESA to identify regulatory structures and significant actors within the modeled systems. These structures can then be further investigated in the Petri net environment of VANESA for hypothesis generation and *in silico* experiments.

VANESA can be applied in many different life sciences, such as fundamental biology, theoretical biology, systems biology, biotechnology, and medical research, among others. The software application has already been proven useful in several biological and medical application cases [JKT$^+$10, KHA$^+$10, STK$^+$10, JKH$^+$11, PJB$^+$12, PJHB12, KJB$^+$12], in which the provided features were applied to an increasing number of biochemical problems such as signal transduction, cellular rhythms and cell-to-cell communication, among others. Although it is primarily addressed to members of the laboratory, it can be used by any scientist. All interested people, who would like to use VANESA for their own research can download VANESA at http://vanesa.sf.net or start it via Java web start. It is platform-independent and free-of-charge.

List of Contents

List of Figures

List of Tables

Abbreviations

API	Application Programming Interface
ANDvisio	Associative Network Discovery visualization
BioPAX	Biological Pathways Exchange
BRENDA	Braunschweig Enzyme Database
CA	Cellular Automata
CmPI	CELLmicrocosmos 4.2 Pathway Integration
CellML	Cell Markup Language
CSML	CellIllustrator Markup Language
CSV	Comma-Separated Values
CVDs	Cardiovascular Diseases
DAEs	Differential Algebraic Equations
DAWIS-M.D.	Datawarehouse Information System for Metabolic Data
DCM	Dilated Cardiomyopathy
EC	Enzyme Commission
EMBL	European Molecular Biology Laboratory
ETL	Extract, Transform, and Load
GCDML	Genomic Contextual Data Markup Language
GO	Gene Ontology
GraphML	Graph Markup Language
GUI	Graphical User Interface
HPN	Hybrid Petri Net
HPRD	Human Reference Protein Database
HFPNe	Hybrid Functional Petri Nets with extension
IDL	Interface Definition Language
IntAct	Interaction Database
IUBMB	International Union of Biochemistry and Molecular Biology
KEGG	Kyoto Encyclopedia of Genes and Genomes
LIMMA	Linear Models for Microarray Data
L-System	Lindenmayer System
MathML	Mathematical Markup Language

MAGE-ML	Microarray Gene Expression Markup Language
MIF	Molecular Interaction Format
MINT	Molecular Interaction Database
MPDZ	Multiple PDZ Domain Protein
NAR	Nucleic Acid Research
NF-κB	Nuclear Factor 'kappa-light-chain-enhancer' of activated B-cells
OBRC	Online Bioinformatics Resources Collection
ODEs	Ordinary Differential Equations
OMIM	Catalog of Human Genetic and Genomic Disorders
OOM	Object-Oriented Modeling
ORFs	Open Reading Frames
Pathguide	Pathway Resource List
PDB	Protein Data Bank
PDBML	Protein Data Bank Markup Language
PID	Pathway Interaction Database
PubMed	Database of References and Abstracts on Life Sciences and Biomedical Topics
PhyloXML	Phylogenetic Markup Language
PNlib	Petri Net library (for the software application Modelica)
PTMs	Post-Translational Modifications
QS	Quorum Sensing
RT-PCR	Real-Time Polymerase Chain Reaction
SBGN	System Biology Graphical Notations
SBML	Systems Biology Markup Language
SBO	Systems Biology Ontology
SOAP	Simple Object Access Protocol
STRING	Search Tool for the Retrieval of Interacting Genes/Proteins
SVG	Scalable Vector Graphics
UML	Unified Modeling Language
UniProt	Universal Protein Resource
VAML	VANESA Markup Language
VANESA	Visualization and Analysis of Networks in System Biology Applications
xHPNbio	Extended Hybrid Petri Nets for biological applications
XML	Extensible Markup Language
.mo	Modelica Exchange Format

Chapter 1

Introduction

Natural scientists have always endeavored to produce good theoretical models, which they can use for hypothesis testing. Their main goal is to create a complete system that is able to answer fundamental questions and moreover, to imitate cell behavior. But in general, the necessary data and possibilities are missing for such an approach. Over the last decades of biomedical research, it has become apparent that a biological element can never be investigated in isolation, since the degree of regulation covers almost all -omic levels. Cellular life is mostly a network of interacting elements, in which the biological elements, such as DNA, RNA, proteins, and metabolites interact with each other. Overall, scientists from fundamental biology and systems biology experimentally investigate biological systems in a more limited context, focusing on a specific element or regulatory process. This is not surprising, as cellular life is complex and the investigation very time-consuming and experimental analysis quite complicated. Therefore, scientists mostly have only detailed information and broad knowledge about the main interaction partners. But a biological element or process is always a part of a larger machinery or regulatory process. Thus, natural scientists need sophisticated information about the other involved elements and/or processes. And they need sophisticated biological networks presenting the whole context of regulation.

One way to produce further information about many other involved regulatory processes is by performing high-throughput experiments, such as microarrays. With these kinds of experiments a broad spectrum of elements can be investigated. But the results are often not in correlation with each other. Moreover, each of the elements is investigated in isolation reflecting one particular system state. Scientists are faced with linking these results to identify connections, interaction networks, and biological switches that can help in explaining system behavior.

One further possibility for gaining additional knowledge and to link different datasets is by accessing knowledge from biological databases. Biological databases are a great repository storing relevant information. However, this kind of information is distributed over different autonomous

and heterogeneous biological databases, which need to be collected, filtered, cleaned, normalized, and linked in complex and time-consuming processes. Actually, more than 1,380 biological databases covering various areas of biology can be found [GFS12]. Although, data integration tools and data warehouses exist, it is still a challenging task to model biological systems based on this data. In the best case scenario, data is stored in databases with a high curation model, a well-designed interface, and data structure. But in many cases data is only available in flat files, is not normed, and not linked to other data sources or -omic levels.

Finally, scientists have detailed knowledge about the research object on the one hand and furthermore, extensive, but imprecise and not correlated information from high-throughput experiments and database information about the overall context of the system. Overall, this different knowledge has to be incorporated into one model, where unknown elements can be grouped into known biological context, important elements into functional groups, sub-networks and motifs identified and examined. By trying to incorporate all of these details the models become large and complex, which in the most cases overwhelm scientists. Therefore, further filtering and analysis mechanisms are necessary, which limit a model to its most important parts.

Based on these models, it should be possible to simulate the analyzed system to predict cell behavior and to gain new ideas for further experiments and approaches. And this takes scientists to the next challenge. How will it be possible to automatically create such a model in one workflow, consisting of all relevant information and data? Ordinary Differential Equations (ODEs) are one powerful way to model such systems, but this approach needs prior knowledge in mathematics and a complete set of biological data and parameters. Biological models that are adjusted or estimated can be misleading or become meaningless. Therefore, a more intuitive way is necessary, that has both analytic power and a balance of necessary data.

Actually, there are more than 200 bioinformatics solutions in the field of molecular modeling. However, most of them fail in their initial goal or produce results which cannot be used for molecular research. Besides, each of them covers only a limited set of necessary approaches. Some are specialized in biological modeling, others in the analysis of high-throughput experiments, others in the visually exploration of biological data, and still others in network reconstruction. In general, there are only few solutions covering different research fields, such as information fusion, modeling, simulation, and network visualization. And those are focused on a specific kind of biological problem and mainly fail in identifying important regulatory effects on the entire cellular interacting network. A software application, which offers a platform to automatically reconstruct and systematically explore the molecular functionality of a particular biological process, leading to the identification of regulatory modules and networks, is still not available. Thus, a strong need for such a platform that is able to model and simulate changes in cell organization and consequently, discuss fundamental questions, and metabolic or genetic diseases, for example, has emerged.

1.1 Aims and objectives

As motivated by the previous section, the objective of this work is to realize a software application, which assists molecular scientists in the semi-automatic reconstruction, analysis, and simulation of biological systems. Primarily, this framework should be designed for fundamental research in biology and medical investigations, but also can be used in other life sciences, such as theoretical biology, system biology, biotechnology, and medical research. Therefore, scientists working at the bench will have a powerful tool that extends their possibilities in analyzing molecular machineries and expands their view on cellular organization and cell behavior. With mathematical and computational modeling power this integrated framework should present, integrate, and visualize state-of-the-art knowledge. Overall, with three mouse-clicks they should be able to get an overview of the whole molecular context, which they could use for hypothesis generation and testing (see Figure 1.1).

Using this application, what now will be called VANESA (Visualization and Analysis of Networks in System Biology Application), scientists should have the possibility to model biological networks, where they could analyze and simulate biological elements and processes in an overall context. Therefore, biological databases are an important resource in assisting scientists in their research, as they provide important data and knowledge from literature, experiments, and results from several analysis techniques. This knowledge can be used in explaining biological systems and cell behavior, from the genetic level on up to the entire metabolism. This data integration and fusion approach should result in biological models, which enable scientists to focus on complex interactions and/or to investigate the role of individual components (RNA-molecules, genes, proteins) and processes (transcriptions, translations, modifications, phosphorylations) in the reversible or irreversible changes of networks' architecture. These insights into the molecular level should help scientists explore the molecular complexity of a particular disease, developmental processes, and differentiation processes, among others and finally, answer fundamental questions and lead to new ideas for therapeutic strategies, personalized medicine, drug targeting approaches, and biotechnological approaches.

Furthermore, it should be possible to automatically simulate the reconstructed systems in a sophisticated simulation environment, specially designed for biological purposes. The simulation environment should provide a powerful and user-friendly framework for hypothesis generation and testing. This feature should enable users to perform experiments *in silico* in order to determine system dynamics and properties, which later on can be influenced in life-science research to change information and processing flow in the biological system. Using the simulation environment system dynamics and network changes could be simulated and moreover, examined under different circumstances. Simulations should be based on the reconstructed biomedical networks, which are previously enriched with experimentally derived data or kinetic data from biological databases. Finally, mathematical analysis techniques, such as graph theory should highlight the most significant and important actors in the biomedical systems. This could help

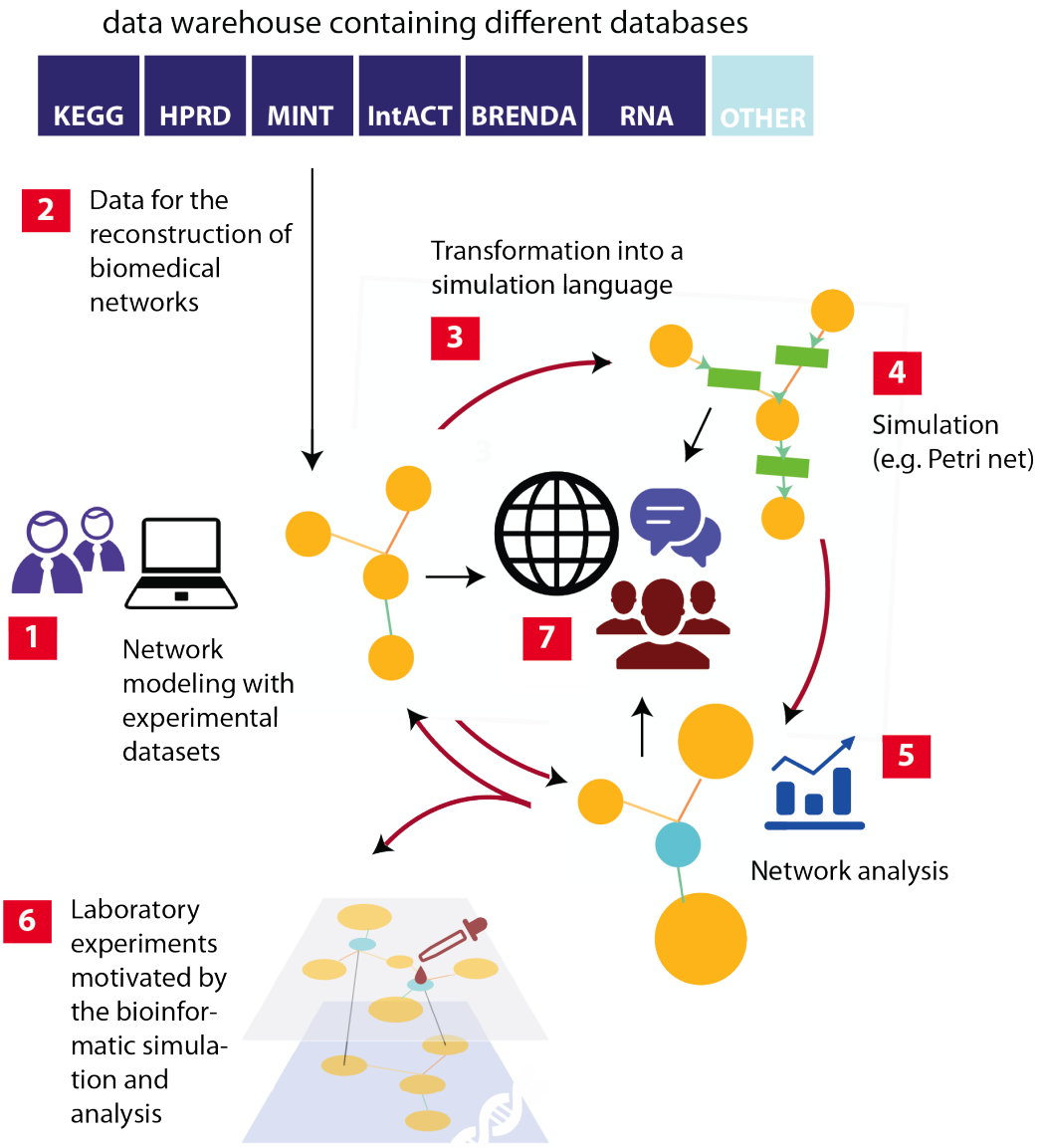


Figure 1.1: An overview of VANESA's aims and objectives. The numbers 1 to 7 represent the stated functionalities on VANESA to model, simulate, analyze, and share biological models. Each number represents a differing bioinformatics approach, which in combination with the other approaches, forms the entire framework and its possibilities. Every modeling approach begins with a basic model, which is either based on experimental datasets or prior knowledge (1). With access to external databases it should be possible to reconstruct molecular networks covering the most important -omic levels (2). Based on the reconstructed models it should be possible to transform each biological model into a simulation language (3), which then can be simulated for hypothesis testing (4). Furthermore, there should be several analysis techniques to analyze simulation results and network structures (5). Based on the bioinformatics results, scientists should get new ideas and suggestions for further laboratory experiments (6). Biological standards should ensure that all model concepts are well-defined and can also be exported, as well as imported into VANESA. Thus, VANESA will enable users to share and evaluate models with other scientists and software applications (7).

in the identification of biological motifs that can be made accessible for fundamental research in biology and biomedicine.

1.2 Structure of the work

In the previous sections it was described what kind of challenges scientists doing molecular research in biology face and how VANESA could assist them in solving the occurring tasks. To realize a software application like VANESA it is necessary to be aware of the complexity of living cells and the various modeling possibilities in bioinformatics. Thus, Chapter 2 begins with the fundamentals of biological cells. The first sections present what exactly characterizes different cell types and how cell behavior can be modeled and analyzed using biological networks. Based on this, different modeling approaches and analysis techniques concerning the best uses for VANESA are discussed and reviewed. Furthermore, one of the main goals of VANESA is to use existing knowledge for network reconstruction. With this in minds, biological databases are highlighted in the next part of the chapter, as these repositories are an important knowledge base for qualitative and quantitative data. In the last part of this chapter different possibilities for the network reconstructions are presented.

Chapter 3 presents work with a concrete relationship to VANESA and its research topic. In the beginning, this chapter focuses on existing software solutions addressed to biological network modeling, analysis, and simulation. Therefore, the most relevant approaches are described and compared regarding their advantages and disadvantages. This review shows the gap in bioinformatics and system modeling and moreover, motivates the realization of VANESA. Further sophisticated bioinformatics approaches from related work are presented, as they serve as powerful features for the modeling, analysis, and simulation of biological systems in VANESA. For the simulation of dynamic systems and the identification of relevant system parameters, Petri net analysis techniques are discussed in more detail. The next part of this chapter discusses special graph theoretical approaches applied on biological networks. It is explained which approaches are best suited for analyzing and determining important topological structures and elements within biological systems. The chapter continues with a discussion on biological databases, covering some of the most important -omic levels, which are an indispensable part of VANESA. The presented databases are briefly described and it is shown how these databases can be accessed with popular database integration solutions. Important standards in systems biology and bioinformatics are presented in the last part of the chapter, as they are used for model consistency and sharing in VANESA.

The focus of Chapter 4 is the software architecture and design of VANESA. The chapter presents the specifications and requirements which were made prior to programming and implementation. This in necessary in order to realize a software solution that is useful, technically sound, and able to help in understanding, reconstructing, analyzing, and simulating entire biological systems.

Therefore, it is shown which requirements are incorporated in VANESA and how the system architecture looks in detail.

Chapter 5 presents the realization of VANESA. It describes how specifications are implemented and algorithms and software components appear in detail. The chapter mainly focuses on the automatic network reconstruction based on database content, the implemented simulation environment in the Petri net language, the realized analysis techniques, network visualization and interaction, and exchange possibilities. The chapter ends with an overview of all important features and the various possibilities of VANESA.

In order to show how useful VANESA can be in life-science research, Chapter 6 presents different application cases from system biology, fundamental research, clinical studies, and biotechnology, in which the software application helped scientists with their molecular research. The chapter introduces different research questions and describes in detail how VANESA was applied and where users where able contribute to new biological findings using this tool.

Finally, this work concludes with Chapter 7, presenting the final results of this work. It summarizes all possibilities and advantages of VANESA and moreover, the benefits scientists have using the software application. In addition, the chapter gives an outlook for further development. As new questions in molecular research will certainly arise in the future, VANESA will be further developed. Therefore, additional bioinformatics approaches are mentioned which can be integrated in the near future. The chapter and work ends with a short and compact discussion as to what makes this software so unique and usable.

Chapter 2

Background

To be able to realize a usable and powerful software application for the reconstruction, modeling, analysis, and simulation of biological systems, each involved scientist needs to be aware of the complexity of cellular life and the possibilities of how to model and analyze it. Bioinformatic modeling, analysis, and simulation are highly interdisciplinary disciplines using techniques and concepts from computer science, statistics, mathematics, chemistry, biology, biochemistry, genetics, and physics, among others. Without knowledge about these research topics it is almost impossible to implement a tool which is able to produce good theoretical models which can be used for hypothesis testing. Therefore, this chapter presents the biological and bioinformatics fundamentals necessary for the realization of VANESA. Each of the sections focuses on a particular topic, briefly discussing the relevant aspects of this work.

The first section focuses on cellular life and its dynamics. It gives an impression of what characterizes a cell and in general, what can be modeled from the biological point of view. In the following Section 2.2, molecular interaction, information flow, metabolism and how many more can be modeled with biological networks, giving scientists the possibility to examine biological relations and processes in detail, are presented. Here, biological networks provide a powerful integrated framework to exhibit, integrate, and visualize knowledge in all levels of detail and thus, are an indispensable part of VANESA. Furthermore, based on the reconstructed networks and models, a large amount of mathematical analysis techniques, derived from graph theory, can be applied as described in Section 2.3. Using graph theory it is possible to examine the topological structure of networks and, for example, determine the most relevant elements within a biological network. This enables scientists to filter and identify important information from large and complex models.

Section 2.4 discusses standards in biology as these are mandatory for understanding, computational usability, and reusability of created models. The following Section 2.5 presents different standard approaches to model and simulate cell behavior. With the aid of computers and mathematical formulas it is possible to formulate and predict the behavior of biological systems

under certain circumstances. Therefore, several modeling paradigms exist, whereby ordinary differential equations are the most common ones. However, other formalisms provide the same power and features and even come with more advantages, such as Petri nets.

To enrich biological models with qualitative and quantitative data, different biological databases can be used as important resources as described in Section 2.6. Biological databases provide data that can help in explaining biological phenomena from the genetic level up to the entire metabolism of a whole organism. Biological databases contain information for almost all -omic levels, based on scientific experiments, published literature, high-throughput experiments, and computational analyses. Database integration even makes it possible to query different databases focusing on different -omic levels simultaneously. Biological information is linked between the data repositories, resulting in one comprehensive view. Furthermore, Section 2.7 shows which possibilities exist to reconstruct biological networks based on database content and the other aforementioned approaches.

Finally, Section 2.8 summarizes all presented approaches and analyzes them with regard to their usability for VANESA. Therefore, it is discussed which approaches are best suited to reach the goal of this work.

2.1 Cellular life

What is cellular life? The simplest answer from the biological point of view is: Anything that contains DNA or RNA [AJL+07], shows self-organization, and has evolved over time as described by Manfred Eigen [Eig71]. Motivated to seek a theory to understand life, many decades ago researchers embarked on the study of biological systems [Sch55, DS99]. Their main goal is not to imitate life but rather to understand the universal logic and properties of living systems. Cellular functions which do not rely on simple enumeration of molecular components and processes, such as transcription, translation, and modifications are carried out constantly. These components never act as one independent element. Thus, present-day cellular biology is challenged to reconstruct coupled dynamical models with many differing elements and strongly interacting systems. Therefore, scientists endeavor to provide a new look at data on the present organisms to validate or reject hypotheses.

The main task for modern biology is to trace phenotypical properties back to specific molecules. Therefore, theoretical models are constructed, consisting of the formation of switching rules that obligate cell features. With modern systems biology and bioinformatics those theoretical models are pictured. Therefore, natural sciences produces a holistic view of different levels of organizations. Using causal relations, theoretical models are constructed using several different switching rules. Through the turning on and off of one or more genes, as controlled by one or more molecules, the properties and dynamics of a cell can change. This can result in different

Property	*E. coli*	Yeast (*S. cerevisae*)	Mammalian (Human Fibroblast)
Cell volume	$\sim 1\ \mu m^3$	$\sim 1{,}000\ \mu m^3$	$\sim 10{,}000\ \mu m^3$
Proteins/cell	$\sim 4 \times 10^6$	$\sim 4 \times 10^9$	$\sim 4 \times 10^{10}$
Mean size of protein	5 nm		
Size of genome	$\sim 4.6 \times 10^6$ bp	$\sim 1.3 \times 10^7$ bp	$\sim 3 \times 10^9$ bp
Genes	$\sim$ 4,500	$\sim$ 6,600	$\sim$ 30,000
Size of regulator binding site	$\sim$ 10 bp	$\sim$ 10 bp	$\sim$ 10 bp
Size of promotor	$\sim$ 100 bp	$\sim$ 1,000 bp	$\sim 10^3$ to 10^4 bp
Size of gene	$\sim$ 1,000 bp	$\sim$ 1,000 bp	$\sim 10^4$ to 10^6 bp (with introns)
Concentration of one protein/cell	$\sim$ 1 nM	$\sim$ 1 pM	$\sim$ 0.1 pM
Ribosomes/cell	$\sim 10^4$	$\sim 10^7$	$\sim 10^8$

Table 2.1: Biological cell characteristics for *E. coli*, Yeast (*S. cerevisae*), and Mammalian (Human Fibroblast) based on [Alo06].

cell behavior, where the concentration of some other molecule is altered, with the effect of turning on or off some other genes [AJL+07, Hol98].

Thus, to model and investigate cellular life, several different key-components of real-life systems have to be considered. The central dogma of molecular biology stated by Francis Crick in 1958 describes the basic information flow in cells with the following sentence: "DNA makes RNA, which in turn makes Proteins" [Cri58, Cri70]. In general, this statement is correct, whereas it is very simplified. Nowadays, natural science has investigated many processes and functions in detail, such as transcription, translation, and post-translational modification, among others, which extend this stated dogma. The investigation of other regulatory processes, such as microRNA fine-regulation are still in their beginning phases.

Table 2.1 gives an example of specific cell type characteristics to show the variety of living organisms [Alo06]. Cell volume, the size of the genome, the promotor size, and many other aspects make a cell unique. Furthermore, transcription time, mRNA lifetime, and cell generation time, among others, differs between different cell types as presented in Table 2.2 [Alo06]. At first glance, it becomes obvious that cellular life is very complex and governed by many different processes and dynamics.

Although, all these presented aspects have to be considered in the modeling of a biological

Property	*E. coli*	Yeast (*S. cerevisae*)	Mammalian (Human Fibroblast)
Diffusion time of protein across cell	$\sim$ 0.1 sec D = 10 μm^2/sec	$\sim$ 10 sec	$\sim$ 100 sec
Diffusion time of small molecule across cell	$\sim$ 0.1 msec D = 1,000 μm^2/sec	$\sim$ 10 msec	$\sim$ 0.1 sec
Time to transcribe a gene	$\sim$ 1 min (80 bp/sec)	$\sim$ 1 min	$\sim$ 30 min (including mRNA processing)
Time to translate a protein	$\sim$ 2 min (40 aa/sec)	$\sim$ 2 min	$\sim$ 30 min (including mRNA nuclear export)
Typical mRNA lifetime	2-5 min	$\sim$ 10 min to over 1 h	$\sim$ 10 min to over 1 h
Cell generation time	$\sim$ 30 min (rich medium) to several hours	$\sim$ 2 h (rich medium) to several hours	$\sim$ 20 h - nondividing
Transition between protein states (active/inactive)	1-100 μs	1-100 μs	1-100 μs
Timescale for equilibrium binding of small molecule to protein (diffusion limited)	$\sim$ 1 ms (1 μM affinity)	$\sim$ 1 sec (1 nM affinity)	$\sim$ 1 sec (1 nM affinity)
Timescale of transcription factor binding to DNA site	$\sim$ 1 sec		
Mutation rate	$\sim 10^{-9}$ /bp/generation	$\sim 10^{-10}$ /bp/generation	$\sim 10^{-10}$ /bp/year

Table 2.2: Biological cell dynamic values for *E. coli*, Yeast (*S. cerevisae*), and Mammalian (Human Fibroblast) based on [Alo06].

system and put into relationship with the biological dogma, it is neither recommended nor practical to model all aspects. Too many unknown parameters will come up, with the danger being that a fitted model will match to nearly anything. Fitted parameters can be even misleading or become meaningless. Furthermore, the larger the model, the longer it will take to determine parameters and to analyze properties of interest. Therefore, each model has to be limited to a practical size and linked to clear scientific questions.

One possibility to limit model size is by using biological networks. These networks can be restricted to only one -omic level, such as metabolomics or proteomics. The main advantage of biological networks is that they can be used to answer scientific questions with the focus on important regulatory elements, rather than building up whole systems.

2.2 Biological networks

Cellular life is mostly a network of interacting elements. To visually represent and analyze the various interactions and relationships, biological systems can be modeled as biological networks, which are based on mathematical graphs (see Definition 1).

Definition 1. *A graph is an ordered pair G = (V, E)*

- *comprising of a set V of vertices and a set E of edges, where each edge is assigned to two (not necessarily disjunct) vertices,*
- *the order of a graph is $|V|$, comprised of the number of vertices,*
- *the size of a graph is $|E|$, comprised of the number of edges,*
- *the degree of a vertex is the number of edges that connect to it and are defined by $N_G(v)$ or $N(v)$.*

The objects, represented by nodes are called "vertices" and the links, represented by directed or undirected arrows, are called "edges". In general, the smallest level of details is the molecular level, describing DNA, RNA, proteins, and metabolites interacting with each other. Thus, nodes can be any kind of biological compounds belonging to such a system. Edges are used to represent biological relations and processes, such as activation, inhibition, and expression, among others. To model all system elements, information flow, and dynamics different biological networks were introduced as described in the following.

- **Transcription networks (or gene regulation networks)**
  Transcriptional networks control the gene expression within cells in time, space, and amplitude [JS11]. Usually these kinds of networks describe how one gene is controlled by the product of another gene. Therefore, the highly interconnected processes are modeled with

a directed graph, in which nodes represent gene, transcription factors, and/or proteins and edges indicate mechanisms, such as transcription, DNA-binding, protein synthesis, degradation, among others. Furthermore, the synthesis of RNA, post-transcriptional events, mRNA turnover, and translation can also be considered. However, as these kinds of networks model a wide range of biological processes, they play a major role in protein-protein interaction networks, signal transduction networks, metabolic networks, and others, which are described in the following.

- **Protein interaction networks**
  In terms of the degree of regulation, it becomes apparent that a protein can never be investigated in isolation. Moreover, it has to be examined in the context of other proteins and their interacting network, in so-called "protein-protein interaction networks". The majority of biological processes within a cell are controlled and mediated by proteins [Hol98, AJL+07]. They interact with other molecules, such as low molecular weight compounds, lipids, and nucleic acids to ensure transcription, translation, splicing, mechanical strength, transport, immunity, signal transduction, growth, development, and many other processes. The types of interactions range from transient interactions, occurring for a limited time, such as they appear in protein kinases, protein phosphates and others, up to static interactions, such as the transfer of biosynthetic intermediates between catalytic sites without the diffusion into the enzyme's surrounding. A further important aspect of protein-protein interaction is the signal transmissions from the external environment to specific locations within the cells.

  However, such protein-protein interaction networks enable the scientist to investigate protein functions, system dynamics and biological mechanisms [DLRF10, Wak05, JS11, PP08, Sut08, Kep07, Zha09]. Reconstructing these kinds of networks, unknown proteins can be grouped into known biological context, important proteins into functional groups, sub-networks and motifs identified and examined in detail. This kind of analysis has become so important and powerful that it already contributes to new therapeutic strategies [KSA08, SWL+05, Sut08].

- **Signal transduction networks**
  Signal transduction networks are of special interest in biological and medical sciences as many diseases are related to disturbances in signaling networks [Alt02]. In general, signal transduction links intracellular processes to the extracellular environment of a cell. The general aim is to model and describe cellular functions in response to external stimuli. Therefore, information transmission is modeled, starting with the binding of extracellular ligands to receptors and resulting in cell response that triggers a cascade of signal transduction reactions. The sequence of reactions involved mainly relies on reversible chemical modifications and complex formations, such as phosphorylation. The final targets of the processes are transcription factors and metabolic enzymes. In summary, signal transduction pathways transform a set of inputs into a set of outputs.

In contrast with other networks, such as protein-protein interaction networks, signaling networks are basically directed. From the topological point of view, the networks involve many different motifs, such as positive and negative feedback loops. One of the most prominent examples is the negative feedback loop of the transcription factor NF-κB [CHL08, Kea06].

- **Metabolic networks**
  Metabolic networks have a fundamental importance in biochemistry and biotechnology, as many scientists modify or alter metabolic networks to produce fine chemicals, antibiotics, industrial enzymes, antibodies, etc. Furthermore, metabolic networks are used in biomedicine enabling a better understanding of metabolic mechanisms and for controlling infections. Therefore, scientists examine differences, synergies, and other interactions between human beings and pathogens. In general, the main goal of metabolic networks is the modeling of cellular processes, such as the up-taking and digesting of substrates from the environment, energy generation, growth, and cell survival, among others. Many of these networks are available online in databases, such as KEGG [KGS$^+$12], EcoCyc [KBMCV$^+$09], and BioCyc [KOMK$^+$05]. The networks refer to metabolites (amino acids, glucose, polysaccharides, glycans, etc.) and their biochemical reactions.

- **Correlation networks**
  Correlation networks represent statistical associations between variables derived from experiments, such as derived from whole genome arrays, mass spectrometry, and enzyme based proteomic experiments, among others [JS11]. The global analysis approach is to give a broad overview of the state of the organism. Due to technological advances in system biology, experimental approaches are able to provide qualitative and quantitative information, which can be used for comprehensive insights into biological systems.

  Usually the resulting datasets are mainly independent variable-unit entries. However, based on the experimentally measured values, correlations can either be determined from the probability point of view or the strength of variable units. The first approach measures if two values have a connection by coincidence or if there seems to be a real link. Therefore, correlation coefficients are calculated expressing the connection probability. The accuracy of this approach mainly depends on the sample size of the experiment. Examining a large number of samples increases the probabilities for finding real connections and moreover, increases the probability of identifying whether weak connections are true. The second approach only considers connection from the strength of variable units, instead of the sampling size. However, an experimental validation based upon the results is the best way to confirm a predicted correlation.

- **Phylogenetic networks**
  Phylogenetic networks describe the evolution and relationship between different organisms. Usually, phylogenetic reconstructions are presented by trees rather than networks,

in which branch points represent the evolutionary separation of two organisms. However, trees do not consider vertical and horizontal gene-transfer events. Thus, phylogenetic networks describe evolutionary processes in more detail. Kunin *et al.* give one prominent example of such a phylogenetic network in their article "The net of life: Reconstructing the microbial phylogenetic network" [KGDO05].

- **Ecological networks**
  Ecological networks typically present food webs. Food webs are limited representations of real ecosystems describing ecological communities focusing on trophic interactions between consumers and resources ("what eats what") [Gra08, DRWM05, Pim02]. In general, two trophic categories exist, called trophic levels. The first ones are the autotrophs, which produce organic matter from inorganic substances. The second level, the heterotrophs, obtains organic matter by feeding on autotrophs and other heterotrophs. It is a unified system of exchange, adopted to analyze interrelationships between community structure, stability, and ecosystem processes.

  The analysis of food webs has shown that the evolution of realistic food web structures can be explained on the basis of simple rules regarding population abundance and species occurrence. For example, ecologists and mathematics have figured out early on, that the structure of food webs consists of non-random properties, such as scaling laws. By examining a predator-prey model (resource-consumer, plant-herbivore, parasite-host), it becomes obvious, that the size of one species is crucial to the stability of the whole system [Hop82].

  However, food webs are an important representation for the prediction of ecological events. They are mainly used to understand biological systems and moreover to protect them from outside influences, such as climate change, foreign wild species, and the narrowing of the habitat.

Summarized, the presented biological networks are able to capture all -omic levels, and furthermore, able to model ecological events and other correlations. With these advantages bioinformatics and systems biology have a set of powerful integrated frameworks to present, integrate, and visualize knowledge. Furthermore, graph theory comes with powerful approaches to analyze those networks as described in the following.

2.3 Graph theory

As mentioned in the previous section, graphs or networks can be used to model many types of biological relations, biological processes, and biological questions. Furthermore, geometry and topology can give important clues about organization and information flow within a system.

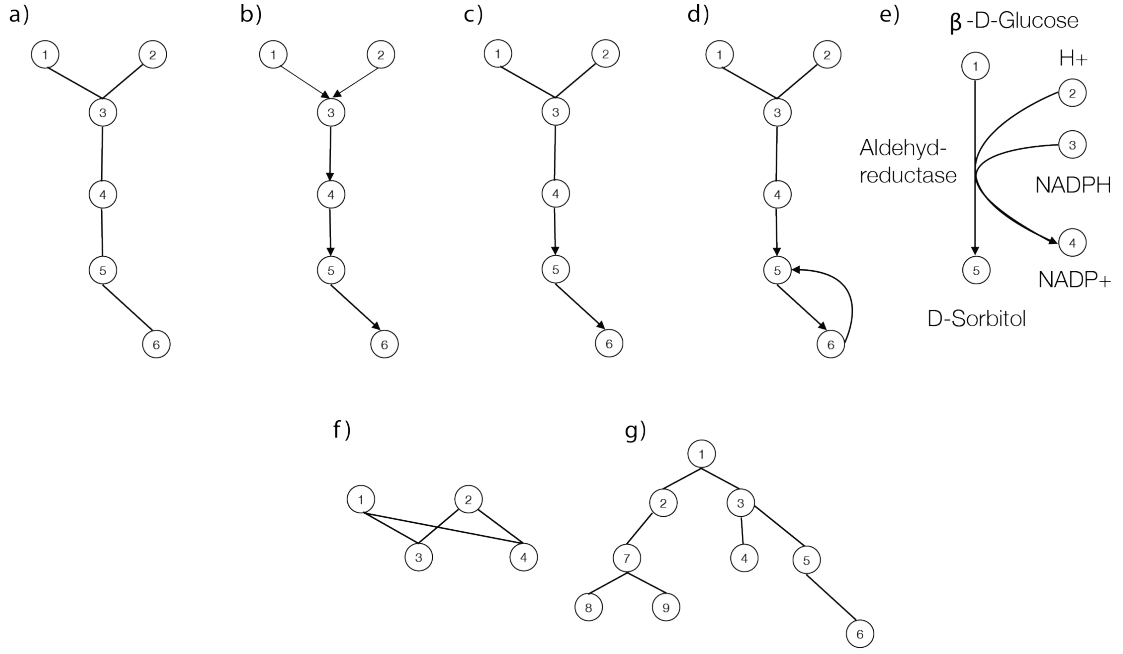


Figure 2.1: Different graph types as they may appear in biological networks: a) undirected, b) directed, c) mixed, d) multi-graph, e) hyper-graph, f) bipartite graph, g) tree.

Graph analysis can determine structural properties of a network. Furthermore, graph theory can analyze vertex degrees, path lengths, diameter, and many other structural properties.

In general, graphs can have different types as presented in Figure 2.1. In a **directed graph** an edge between the vertices u and v is represented by the ordered pair (u, v) [Die00]. Visually the ordered pair represents the direction of the arrowhead. However, there is a big difference between directed and undirected graphs for a given number of vertices. The **amount of directed graphs** $N_{dir}(V)$ with V vertices is much higher than the amount of possible undirected graphs $N_{undir}(V)$ [JS11]:

$$\frac{N_{dir}(V)}{N_{undir}(V)} = 2^{\frac{(V^2-1)}{2}} \tag{2.1}$$

A **mixed graph** has both directed and undirected pairs. In the biological context it can represent protein-protein interaction networks, where some interactions are undirected, such as protein-complex bindings, and some interactions, such as activation, phosphorylation, and other processes are directed. A **multi-graph** contains multiple edges, where two or more edges are incident to the same two vertices. A **hyper-graph** is characterized by more than two elements, which are connected to one interaction. Hyper-graphs are often used to model metabolic networks where several substances are used in one reaction to produce another substance.

A graph is **bipartite** if there is a partition of its vertex set $V = S \cup T$, such that each edge in E has exactly one end-vertex in S and one end-vertex in T. In general, a **tree** is an undirected, connected, acyclic graph, in which any two vertices are connected by one simple path. Vertices

with only one edge are called leaves. All other vertices are inner vertices. However, a tree can also be directed, where the edges are all directed towards a particular vertex, or all directed away from a particular vertex. In case of a **rooted tree**, one vertex is designated the root, in which case the edges have a natural orientation, towards or away from the root. The depth of such a tree is the length of the path from the root to a vertex. The height is the maximal depth [Die00].

A **subgraph** $G' = (V', E')$ of the graph $G = (V, E)$ is a graph where $V' \in V$ and $E' \in E$ [Die00]. The **density** of a graph is given by

$$\frac{2 \mid E \mid}{\mid V \mid (\mid V \mid -1)} \tag{2.2}$$

This definition indicates how dense or connected a graph is determining vertex degrees [PSM+11].

Two graphs G and G' are **isomorphic** $G \simeq G'$, if there exist a bijection $\varphi : V -> V'$ between the vertex sets of G and G', such that any two vertices u and v of G are adjacent in G if and only if (u) and (v) are adjacent in G', based on $xy \in E \Leftrightarrow \varphi(x)\varphi(y) \in E' \; \forall x, y \in V$ [Die00].

Global network properties are topological entities, such as distance, average path length, and diameter. A **path** is a sequence $(v_0, e_1, v_1, e_2, \ldots, v_{k-1}, e_k, v_k)$ of vertices and edges. The **length of a path** is given by its number of edges. The **distance** between two vertices is given by $d_G(u, v)$. A **shortest path** between two vertices is a path with minimal length d_{ij}. The **average path** length is defined by $d = \langle d_{ij} \rangle$. The **diameter** is defined by $d_m = max(d_{ij})$, which represents the maximum path length. The correlation between edges and vertices is given by $\varepsilon(G) := |E|/|V|$ [PSM+11, Die00].

An **Eulerian path** is a path, which contains every edge exactly once. A graph is an **Eulerian graph** if it contains an Eulerian path [Die00]. A path in an undirected graph that visits each vertex exactly once is called a **Hamiltonian path**. A graph that contains a Hamiltonian path is a **Hamilton graph** [Die00].

Going further into detail, vertex degrees and other topological indices are described in the following, which serve as a base for centrality measurements. A **vertex degree** $\delta_G(v) = \delta(v)$ is the number of edges $|E(v)|$ incident to the vertex, with loops counted twice. The **minimum degree** is characterized by $\delta(G) := \min\{d(v) \mid v \in V\}$, the **maximum degree** by $\Delta(G) := \max\{d(v) \mid v \in V\}$, the **average degree** by:

$$d(G) := \sum_{v \in V} \frac{d(v)}{|V|} \tag{2.3}$$

The relation between the degrees is given by $\delta(G) \leq d(G) \leq \Delta(G)$ [JS11, Die00, PSM+11].

The **clustering coefficient**, a basic measurement for the local cohesiveness of a network, measures the probability that two vertices with a common neighbor are connected. In the case of undirected graphs, there exist $E_{max} = k_i(k_i - 1)/2$ possible edges between neighbors. The clustering coefficient C_i of the vertex n_i is then given as the number of edges E_i between the neighbors to the maximal number E_{max} with [JS11]:

$$C_i = \frac{2E_i}{k_i(k_i - 1)} \tag{2.4}$$

The **matching index** quantifies the similarity between two vertices on the number of common neighbors. The index is based on following definition [JS11]:

$$M_{ij} = \frac{\sum \text{common neighbors}}{\sum \text{total number of neighbors}} = \frac{\sum_{k,l}^{N} A_{ik}A_{jl}}{k_i + k_j - \sum_{k,l}^{N} A_{ik}A_{jl}} \tag{2.5}$$

Based on the presented definitions a variety of analysis techniques are possible. The approaches enable structural, as well as individual node analysis. Thus, it is not surprising, that applied to biological networks, it has become an important aspect in system biology, bioinformatics, and theoretical biology [JS11]. Furthermore, it can contribute and increase the power of various modeling approaches, which are discussed in the following.

2.4 Biological standards

Molecular biotechnology, systems biology, bioinformatics, and many other disciplines in biology make it possible to reconstruct and analyze biological systems. More than 300 pathway or molecular interaction-related data resources, visualization, and analysis software tools have been developed[1]. However, the diversity of tools shows several problems in sharing and moving models between each other. An attempt to overcome this problem is the creation of standards [MRC+09, SB08, SHL07].

In an online survey, Klipp *et al.* asked 125 researchers (75% modelers, 4% experimentalists or 21% both) covering various fields, such as modeling of individual pathways, investigation of complex processes, development and application of computational methods, and software development about their opinion on standards [KLH+07]. About 80% of the scientists considered the creation of standards necessary or desirable. This is not surprising, science standards have many advantages as listed in the following:

[1]The number of software applications has been approximated by counting software tools that support the Systems Biology Markup Language (SBML) and the Cell Markup Language (CellML). Software tools are listed at http://www.sbml.org/ and http://www.cellml.org/.

- Model definitions and entities are based on ontologies, defined nomenclature and restrictions. Thus, they become accessible and readable to a wide community.
- Standards improve communication between software tools, free exchange of information, and comparison between different studies, which results in more productive collaborations.
- Complementary resources from multiple simulation/analysis tools can work together, instead of redefining and reconstructing models in each tool.
- Reimplementation of models becomes easier or dispensable, which reduces duplication and redundancy.
- If tools are no longer supported, models developed within the tools can be still used if they are based on standards. Information, knowledge, and research progress is not lost and can be reused.
- Data curation teams can evaluate models without being restricted to a certain tool or formalism.
- In the publication process, any curator can process annotation and normalization before data is published and made available to the scientific community.

Scientists, simultaneously with both tool development and modeling projects, have developed standards to share, evaluate and analyze knowledge and information. Standards are definitions in the form of common, inclusive and computable languages. For the modeling and sharing of biological models main standards exist, such as the Systems Biology Ontology (SBO) [CJK+11], Systems Biology Markup Language (SBML) [FH03, HFS+03], the Cell Markup Language (CellML) [MMR+10], and Biological Pathways Exchange (BioPAX) [DCP+10]. For the graphical representation of biological pathways, languages such as the System Biology Graphical Notations (SBGN) [LNHM+09] have been introduced (see Section 2.5). Model description achieves human and computational usability, reusability, and interoperability when the encoded format is standardized. Models or software tools without standardization are only of limited use, as they do not provide the possibility to share, compare, and/or integrate large amount of systems. Thus, it is important to use common standards as described in the following section.

2.5 System modeling

Modeling biological phenomena with the use of computer applications has become a common task as described in this section. Therefore, different modeling techniques exist to study and analyze the dynamic details of biological systems. In general, biologists are more familiar with

mathematical modeling, whereas computer scientists are accustomed to computational formalism. However, several approaches provide mathematical as well as computational capacities. In order to give an overview of existing modeling languages, the most important techniques in systems biology and biological network modeling are briefly described in the following subsections.

2.5.1 Ordinary differential equations

One of the most powerful techniques in modeling system dynamics are ordinary differential equations (ODEs), which provide a theoretical framework for discrete, continuous, deterministic, and stochastic models. In general, they describe the change rate of variables in the modeled system as a function of time. ODEs have been applied and used in many application cases and proved themselves very useful [vdB11, Alo06, TCN03]. Furthermore, ODEs can be used to model entire systems with given kinetics [GGM+10, DG08]. One common example for modeling gene activation or positive control is the Hill function in which the equilibrium binding of the transcription factor to its site on the promotor is modeled from zero to its maximal saturated level with Definition 2 (see Figure 2.2 for a graphical representation).

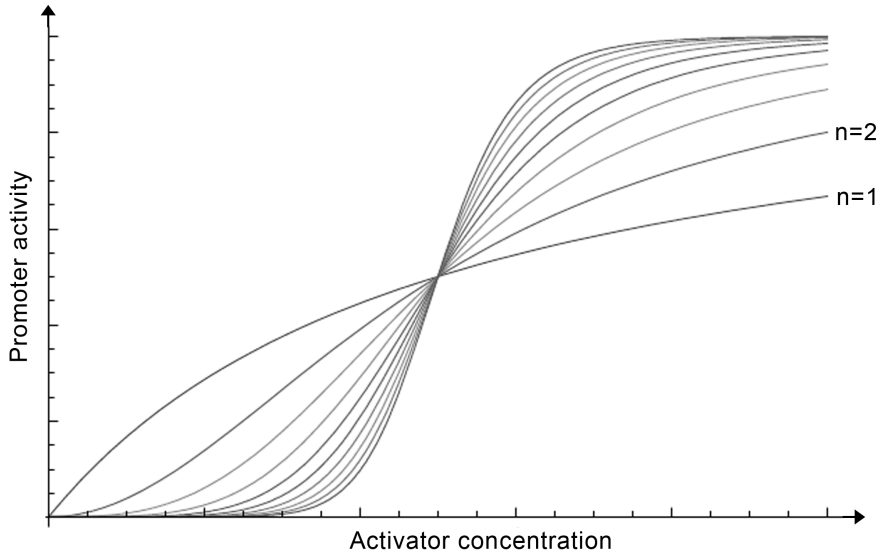


Figure 2.2: Graphical plot of one Hill function with different steepness parameters (n) for the modeling of gene activation and positive control in biology.

Definition 2. *A Hill function is defined by* $F(X^*)=\frac{\beta X^{*n}}{K^n + X^{*n}}$, *where*

- *K is termed as the activation coefficient,*
- *β the maximal expression level of the promoter,*
- *n the steepness of the input function (the larger n is, the more step-like the curve).*

However, the model reconstruction with ODEs has some major drawbacks when the kinetic system parameters involved are unknown. With increasing network size and complexity it becomes almost impossible to estimate all missing parameters. Due to high-throughput techniques a huge amount of qualitative data is available but the parameter estimation still remains challenging. Furthermore, precise quantitative measurements for parameter estimations are difficult to parametrically explore. A further disadvantage of ODE network modeling and analysis is that ODE-based models do not support any detailed insights into signal and information flow within biological networks. Thus, information flow, biological cascades, and system dependencies cannot be examined in detail.

2.5.2 Object-oriented modeling

Object-Oriented Modeling (OOM) is a paradigm in which a system is primarily modeled with a set of related, interacting objects and the functions and services they provide [RE91]. These objects represent all entities relevant to the application (see Figure 2.3 for an example). Nearly anything can be an object, which is defined as an assembly of classes. A class is a discrete reusable code block that has attributes, takes variables, performs functions, and returns values, among others. In general, objects do not exist in isolation from another. The relationships between the objects represent a wide set of different connections and interactions, for example, how one protein is related to a gene, or how one protein changes the state of another protein by phosphorylation. However, the modeling task is always specified for one specific context, where objects belong to each other and share a set of properties and methods to imitate the real-world system [JGW04, FDM09, DPG93]. Using the standardized Unified Modeling Language (UML) [LS94] the object-oriented models can be made visually accessible through a set of graphic notation techniques.

2.5.3 Rule-based models

Rule-based specifications and formal grammars play an important role in the creation of photorealistic virtual organisms. Particularly plants and scientific models of vegetation structure are modeled with rule-based models [Kur07]. One widely used formalism is the Lindenmayer System (L-System), a parallel rewriting system on strings. Based on an alphabet of symbols, a finite set of rules for string manipulations, a start string called axiom, and a mechanism to visualize data, it is possible to model the morphology of a variety of organisms. With an iterative process, which expands the model with new structures in each time step, growth processes can be modeled and simulated.

For example, having the axiom A and the rules $A \rightarrow B$ (letter A will be transformed into letter B) and the rule $B \rightarrow AB$ (letter B will be transformed into substring AB), a new string is generated in each time step by applying the aforementioned rules. Based on the system

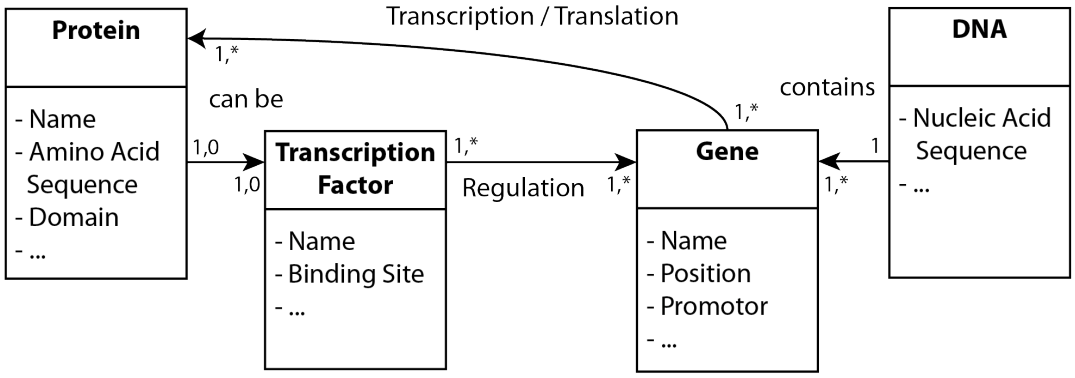


Figure 2.3: An example of an object-oriented model in molecular biology. The model is focused on a mandatory set of properties, whereas a complete model is made up of more attributes and relationships. However, here, a protein can be a transcription factor regulating one or more specific genes. One gene can be even regulated by more than one transcription factor. The genes are derived from the class DNA, which contains a set of genes. Each gene alone or in combination with others can be transcribed and translated into one or more proteins. Each class is characterized by specific attributes, such as binding sites and nucleic acid sites, which are necessary for biological functions and molecular processing.

settings the development sequence for this model is described by: $A \rightarrow B \rightarrow AB \rightarrow BAB \rightarrow ABBAB \rightarrow BABABBAB \rightarrow \ldots$ Finally, the expanded string only needs to be visualized to see developmental growth.

In order to visualize this model, additional geometric rules have to be defined, which reconstruct geometric structures based on the appearance and order of the letters in the development sequence. One of the first examples of branching structures generated by an L-System was given by Prusinkiewicz and Lindenmayer in 1990 [PL90], which is presented in Figure 2.4.

2.5.4 Constraint-based models

Constraint-based models are mainly used for cellular metabolism. The main idea of this approach is to describe detailed dynamic models with a set of constraints which characterize the models possible behaviors. Therefore, stoichiometric, thermodynamic, and enzyme capacity constraints are defined. Instead of single solutions, a set of possible solutions represents different phenotypes which comply with the constraints. Thus, models can comprise thousands of reactions, such as the metabolic reconstruction of the bacterium *Escherichia coli*, where 2,583 constraint reactions were defined [OCN+11]. Furthermore, these models and constraints can be used for other metabolic engineering applications. However, the classical constraint-based models focus at flux balance analysis of metabolic networks [KPE03, Wie01].

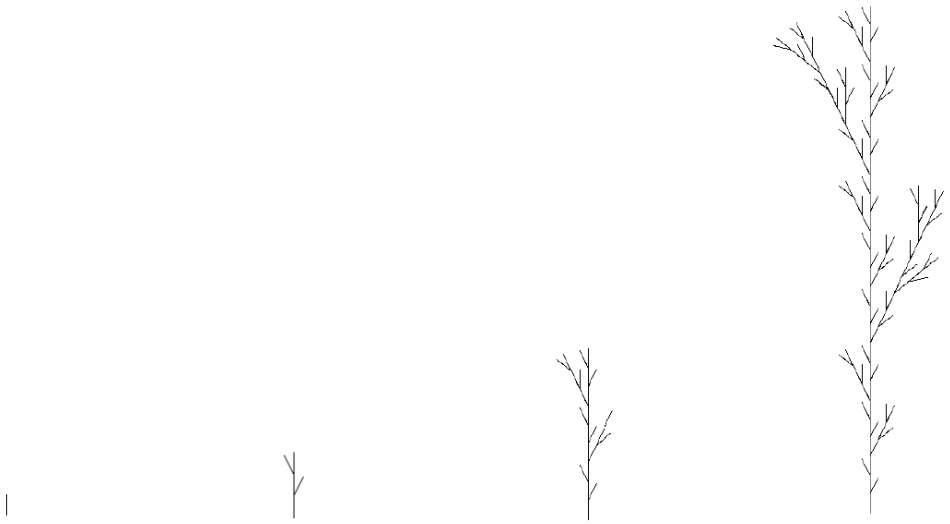


Figure 2.4: A developmental sequence of branching structures generated by an L-System (picture from [PL90]).

2.5.5 Interacting state machines

Interacting state machines are mathematical models for the description of temporal behavior within a system. The model is based on the states of its parts and not on its components. Therefore, hierarchies are expressed by diagram-based formalisms. Each of the parts can be in one of a finite number of states, whereas the machine is in only one state at a given time. However, by initiating a trigger event the machine can change its condition. The main advantage of interacting state machines is that they require little quantitative data, as they model biological behavior in a qualitative way [EHC03, KCH01]. Usually, models described with interacting state machines are used for model checking and interactive execution.

2.5.6 Process algebras

Process algebras are used for the modeling of concurrent systems. The language provides a framework for the high-level description of interactions, communications, and synchronizations using a set of process primitives. Operators are used to combine these primitives. Therefore, this approach provides algebraic laws for the manipulation and analysis of process expressions using equational reasoning. In most of the cases, process algebras are used in signal processing, as presented in the work of Danos and Laneve. The authors introduced a protein algebra to demonstrate how standard biological events can be expressed in simplified signaling pathways [DL04].

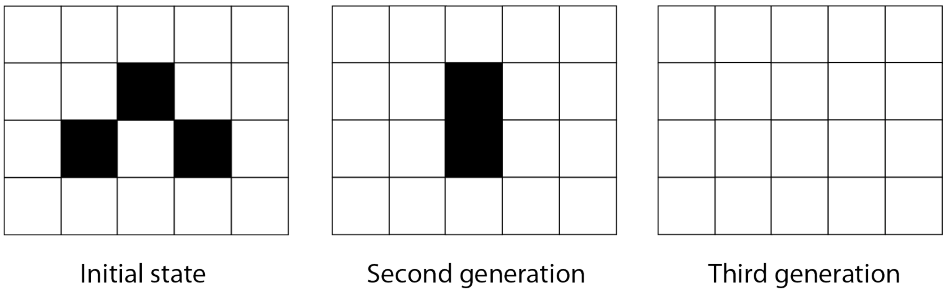


Figure 2.5: An example of a simple cellular automaton with rules and settings of the "Game of Life" approach by John Horton Conway. From left to right: initial state and configuration (generation 1), second generation, and third generation.

2.5.7 Cellular automata

Cellular automata (CA) are used to model and simulate biological self-organization. They use a paradigm of fine-grained, uniform, parallel computation, which were used in many aspects of developmental biology [EE93, WYA+05, WMP99]. With CA whole population dynamics can be simulated in which each individual's fate is dependent on its neighbor's behavior and existence. Therefore, a set of simple rules is defined, that mimics the physical laws of the given system. The evolution of a CA is determined by its initial state, requiring no further input. The simulation is discrete in time, space, state, and once running, evolves with its own given rules.

The most prominent example of a CA is the "Game of Life" devised by the British mathematician John Horton Conway in 1970 [Gar70]. The example is based on a simple deterministic CA consisting of a regular two-dimensional grid of cells, in which each cell has a certain state: Alive or dead. Every cell interacts with its neighbors based on the set of applied rules at each time step (see Figure 2.5).

Following rules are applied to the "Game of Life" to calculate and simulate next generations:

- Any living cell with less than two living neighbors dies because of under-population.
- Any living cell with two or three living neighbors does not change in the next generation.
- Any living cell with more than three living neighbors dies due to overcrowding.
- Any dead cell becomes alive by reproduction, when exactly three neighbors are alive.

Those rules are applied repeatedly to create further generation. Finally after n generations, a picture results, that describes population structure, dynamics, population features, and system robustness, among others.

2.5.8 Agent-based systems

Agent-based systems are similar to the concept of cellular automata, focusing on complex system behavior, structures, and phenomena in dynamics. This approach describes and simulates operations and interactions of autonomous agents in a given space. System operations and interactions are based on simple rules. However, in contrast to CAs, the agents are not placed on a grid or any similar environment. Moreover, the autonomous agents can freely move within the given 2D or 3D space. The most prominent examples are from multi-cellular studies, such as tumor growth studies [ZAD07], morphogenesis [GMTH06], and immune response [LVC+08].

2.5.9 Bayesian networks

A technique for biological network modeling is the so-called "Bayesian networks" theory. Bayesian networks are used for the automatic reconstruction of causal signaling network models from experimentally derived data [LDD+09, Pe'05, SSR+03]. The core of this approach is the notion of conditional independency. This approach calculates probabilistic relationships to estimate which network structures, circuits, and motifs can be derived from given biological data. This results in one or a set of possible directed acyclic graphs that match the experimentally data conditions best. Nodes, which are not connected within the graph, represent variables which are conditionally independent. Nodes that are connected to each other represent strong probabilistic relationships based on experimentally conditions. One example of such an approach is presented in Figure 2.6.

However, the reconstruction of such networks demands a large number of datasets. The greater the network, the larger the necessary experimental datasets must be. Otherwise, probabilistic relationships and independencies cannot be determined.

2.5.10 Boolean networks

In 1969, Boolean networks were introduced by Kauffman to model gene regulatory networks [Kau69]. Here, genes are modeled by Boolean variables which represent their active and inactive states within the model. A Boolean network is a directed graph, where all nodes are equivalent and receive information inputs from their neighbors. Every node can only take two binary values, 0 (OFF) or 1 (ON). These values represent the dynamic activity and behavior of the involved elements. Information flow and statement acting is determined by a logic rule. Therefore, the logical operators *and*, *or*, and *not* are used. If the statement is true, the logical operation results in an ON state, otherwise it remains in the OFF state (an example is given in Figure 2.7).

a)

	Mutation probability	Mutation probability	Mutation probability
Grandfather		1	1
Father			1
Son	0.001	0.025	0.5

b)

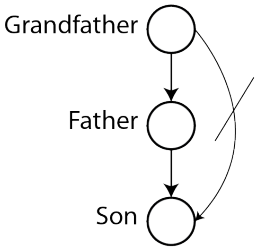
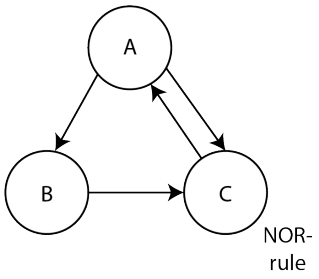


Figure 2.6: This Figure presents a Bayesian network example from classical genetics studying mutations. (a) The probability that the son has a mutation is 0.001. If we know that his grandfather has the same mutation, the probability increases to 0.025. Thus, their genotypes are clearly dependent. But if we also know that his father has the mutation as well, the son's probability increases to 0.5. This additional information indicates that his father, independent of whether his grandfather has or does not have the mutation, only affects the son's probability. Therefore, only one conditionally network can be reconstructed (b), which matches the experimental data. All other possible networks are disregarded.

a)



b)

A	B	C
1	1	0
1	0	0
0	1	0
0	0	1

c)

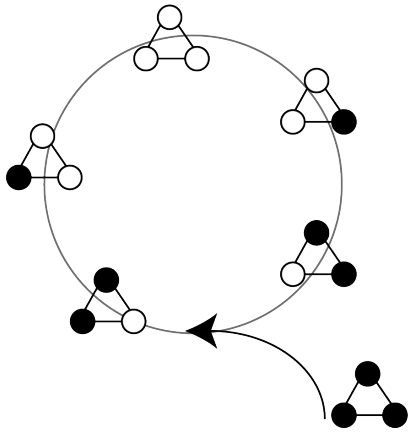


Figure 2.7: The Figure shows a possible Boolean network based on three nodes (a), each having a state 0 (OFF) or 1 (ON). The states for each node is determined by the input of the other nodes. Node 1 and 2 copy their single input, while node 3 performs the Boolean function NOR on its inputs as described in the table (b). The dynamic system is described in (c), where filled nodes are on, lights are off.

The main advantage of this technique is the reduced number of parameters necessary while still capturing network dynamics and producing biologically predictions and insights [Ger04]. However, quantitative measurements cannot be included for precise predictions and analysis.

2.5.11 Boolean formalization

This approach formalizes in Boolean terms genetic situations for the description of complex circuits [Tho73, TD90, BCC+07]. The main goal of this language is to formalize a complex model in a compact and unambiguous way by functions of binary variables. Therefore, three different types are defined and used. The genetic variable describes the gene state, being normal or mutated and the recognition site, being a promoter, operator, terminator, or other. The environment describes temperature and the presence of different substances. Internal variables are used to memorize previous system states at a given time. Associated functions calculate the proceeding periods of the system with regard to the present variables. In order to reduce the algebraic expressions to its simplest form, tabulations of the logic equations as Veitch matrices are used. The Veith matrices give a clear and exhaustive view of all calculated system states and show which states are stable and how the model proceeds from state to state.

2.5.12 Petri net

A Petri net is a mathematical modeling language for the description and analysis of complex and distributed systems. Therefore, it provides an exact mathematical definition of its execution semantics. The language was introduced by Carl Adam Petri in 1962 [Pet62] and constantly developed. Thus, this language comes with a well-developed mathematical theory for process analysis.

Reisig *et al.* presented the first basic definition in their article "A primer in Petri net design" in 1982 [Rei92]. This resulted in the general formalism presented in Definition 3.

Definition 3. *A basic Petri net is defined by the tuple* $PN = (P, T, F, W, m_0)$*, where*

- $P = \{p_1, p_2, \ldots, p_n\}$ *is a finite set of places,*
- $T = \{t_1, t_2, \ldots, t_n\}$ *is a finite set of transitions,*
- P *and* T *are pairwise disjoint,*
- $F \subseteq (P \times T) \cup (T \times P)$ *is a set of arcs from places to transitions and transitions to places, where* $(p_i \rightarrow t_j)$ *denotes the arc from place* p_i *to transition* t_j*, and* $(t_j \rightarrow p_i)$ *the arc from transition* t_j *to place* p_i*,*

- *W is the weight function (W: $F \rightarrow \mathbb{R}$) which assigns every arc a non-negative integer, where (f:$p_i \rightarrow t_j$) denotes the weight of the arc from place p_i to transition t_j,*
- *m_0 is the initial marking $\forall\ p_i \in P$.*

A Petri net is based on a directed bipartite graph, in which the nodes represent transitions and places. Regarding the graphical representation, places are drawn as circles, transitions are drawn as rectangles and arcs are drawn as directed arrows. The directed arcs describe which places are pre- and/or post-conditions for which transitions. Each place can contain tokens, which are drawn as black dots. The start configuration of a Petri net model is described by the state m_0, which assigns tokens to each place. With this graphical notation, processes such as choice, iteration, and concurrent execution can be modeled stepwise and analyzed.

Due to the presented formalism, Petri nets stand out by their balance between modeling power and analyzability in comparison to other modeling techniques. Furthermore, concurrent systems can be automatically determined, although some of the systems are difficult and expensive to determine [PR08]. Thus, the various modeling possibilities and analytic power of the proposed formalism offers a well-developed basis for the description of chemical processes and a mathematical theory for process analysis.

In 1992 Reddy *et al.* proposed a Petri net formalism for biological network modeling in order to represent and analyze metabolic pathways in a qualitative manner [RML92]. Using the biological definition, places represent biological compounds such as metabolites, enzymes, proteins, and cofactors, which are part of biochemical reactions. The directed arcs and transitions are used to represent relations, events, and stoichiometry between the biological compounds. The drawback of the proposed formalism by Reddy *et al.* is that neither complex metabolic processes nor quantitative processes can be modeled. Therefore, Hofestädt *et al.* expanded the existing formalism and introduced a more complex Petri net language [Hof94] to enable quantitative modeling of biochemical networks. This resulted in a formalism that is capable of modeling gene-controlled metabolic networks, cell communication processes, and other signaling processes and regulations (see Figure 2.8).

However, in order to model and analyze kinetic effects, a further formalism was introduced, called functional Petri net (FPN) (see Definition 4). Therefore, Hofestädt and Thelen presented a more detailed formalism [HT98] based on the definition of the self-modifying Petri net by Valk [Val78].

Definition 4. *A functional Petri net is defined by the tuple FPN = (P, T, F, VF, m_0), where*

- *(P,T,F) is a net,*
- *V_F is a mapping, which assigns each f from F a mapping $V_F(f)$,*
- *$V_F(f)$ is an element of: $g(x_1, \ldots, x_n) \mid g: P_N \times \ldots \times P_N \rightarrow \mathbb{N}$, $n \in \mathbb{N}$, and*

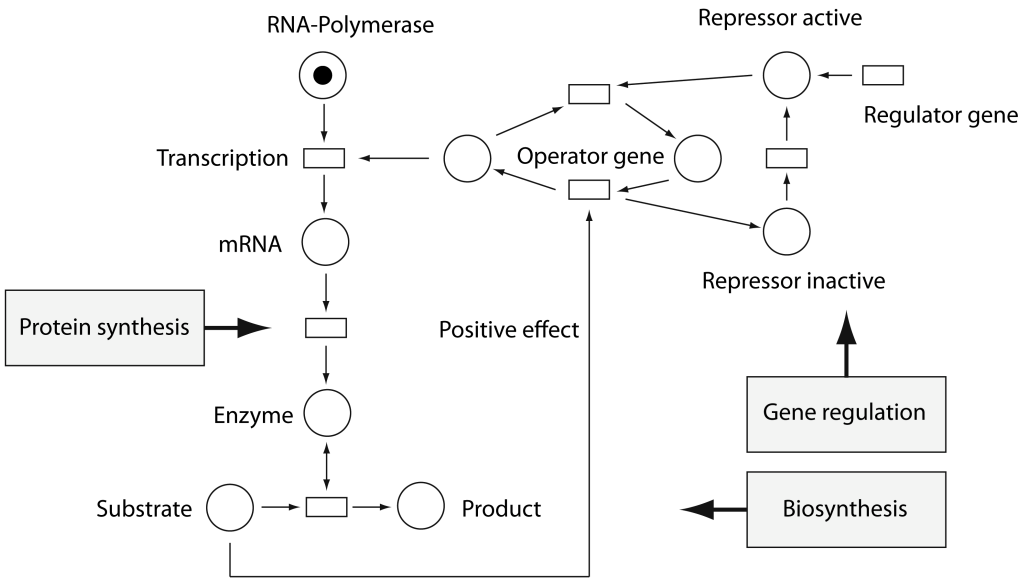


Figure 2.8: This Figure shows the possibility of modeling abstract biological processes with Petri nets. The model is based on gene-controlled biochemical reactions, such as gene regulation, protein synthesis, and others.

- *P_N represents any number $\mathbb{N}$ or the number of tokens represented by the place P_N regarding the actual configuration of FPN.*

The advantage of the FPN is that kinetic effects of biological networks can be analyzed and simulated. Thereby, any qualitative Petri net model can be extended by a functional Petri net. Later this model can be enriched with quantitative experimental data.

The usage of real numbers instead of tokens was not considered in the existing formalisms. Thus, for more realistic simulations of biological networks a further extension of the Petri net classes was necessary. In view of these new demands, Alla and David introduced the Hybrid Petri net (HPN) in 1998 [AD98]. The idea of the HPN is the representation of two kinds of places and transitions that allow calculating discrete and analytical molecular values. Therefore, discrete places (discrete transitions) and continuous places (continuous transitions) are defined. Thus, non-negative real numbers can be used in continuous places, representing the concentration of metabolites and other biological concepts.

Later on, Matsuno *et al.* proposed a Petri net approach for the modeling of gene regulatory networks by discrete and continuous processes with Hybrid Petri nets [MDNM00]. The authors combined the discrete Petri net concept with the continuous one. Their approach uses Petri nets containing discrete places with integer tokens and discrete transitions with time delays as well as continuous places with non-negative real marks and continuous transitions with firing speeds.

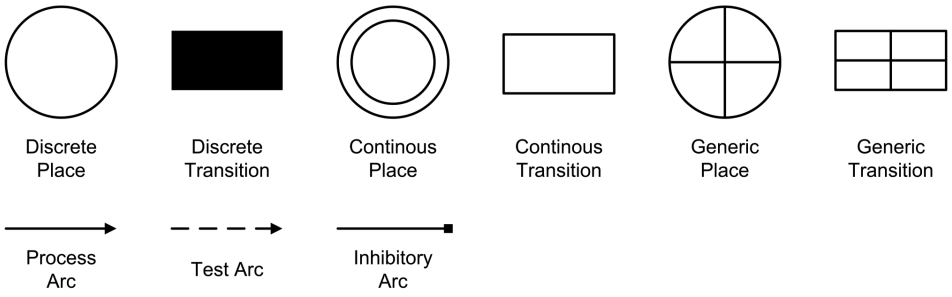


Figure 2.9: Graphical representation of the HFPN formalism. This Petri net formalism introduces two new arcs (test and inhibitor arcs) and combines the discrete Petri net concept with the continuous one to one paradigm.

However, Matsuno *et al.* improved this approach by combining the discrete Petri net concept with the continuous one to a so-called Hybrid Functional Petri Net (HFPN) [MTA+02]. The authors extended the formalism with the feature that arcs as well as the speeds of the transitions are functions depending on the tokens of the places. Furthermore, they extended the HFPN by two specific arcs, called test arcs and inhibitor arcs in order to model inhibition and activation mechanisms in biological processes (see Figure 2.9 for a graphical representation of the formalism). Chen and Hofestädt as well as Doi *et al.* demonstrated the applicability, possibilities, and power of this approach by modeling molecular networks [CH02, DFM+04].

In addition to the aforementioned formalisms, Goss and Peccoud introduced Stochastic Petri nets [GP98]. The main idea behind Stochastic Petri nets is to model the random behavior of molecular reactions, as they have been observed in many experiments with a low concentration of molecular reaction. A stochastic transition does not fire instantaneously but rather with a time delay following an exponential distribution which may depend on the token numbers of the places.

2.5.13 Visual modeling

A further way to model a biological system is by using a standard graphical notation, such as the System Biology Graphical Notations (SBGN) [LNHM+09]. SBGN is a visual language which focuses on the graphical notation of biological networks. It provides a common notation to represent interactions and regulations between molecular species, such as binding, complexation, and protein modification, among others. It consists of three complementary languages: process diagram, entity relationship diagram, and activity flow diagram. Together the different notations enable scientists to represent biological networks in a standard and unambiguous way (see Figure 2.10 for an example).

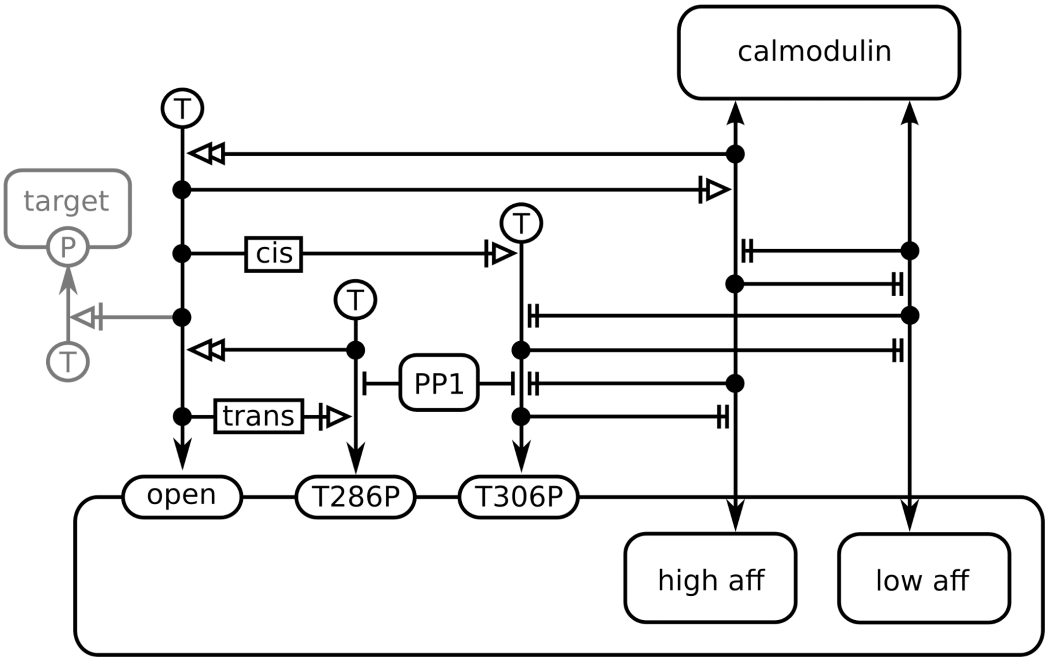


Figure 2.10: SBGN entity relationship diagram representing the effect of calmodulin binding on CaMKII activity, using the nested entities of ER L2 V1 (picture from http://www.sbgn.org/Documents/ER_L1_Examples).

Summary

Each modeling technique comes with specific features and constraints. In order to model and analyze a biological system a powerful theoretical framework is necessary. Thus, visual languages such as SBGN are not suitable for systems biology analysis, as they do not provide any kind of analytical environment. Furthermore, these languages consider only a limited graphical representation of the biological components. Object-oriented models are software-intensive and complex systems. As systems evolve, classes and the function they perform need to be changed more often. This can result in a schema, where complexity continuously grows. Thus, a clean programming, organization, and notation are necessary during model design and software implementation. Furthermore, well-defined interfaces between objects are mandatory to keep the model maintainable. Otherwise, model parameters can become distorted or even incorrect. Ambiguities in data flow can also occur. Therefore, the following review only focuses on modeling techniques that provide sophisticated analysis power and are clean and well-defined in their semantics. To show how often and in which application cases the aforementioned techniques are used, Machado *et al.* summarized literature references, classified by the type of biological process [MCR+11] (see Table 2.3). Boolean formalizations are not considered in this review as this approach is frequently used in system biology and bioinformatics. Furthermore, the same or similar results can be produced with Boolean networks, ODEs, or Petri nets, among others.

	Signaling networks	Gene regulatory networks	Metabolic networks
Boolean networks	+	+ +	
Bayesian networks	+	+ +	
Petri nets	+ +	+	+ +
Process algebras	+ +		
Constraint-based models	+	+	+ +
Differential equations	+ +	+ +	+ +
Rule-based models	+ +		
Interacting state machines	+ +		
Cellular automata	+	+	
Agent-based models	+ +		+

Table 2.3: Overview of the amount of literature references using the presented formalism classified by the type of biological process [MCR+11]. Based on the evaluated information, signaling networks have been modeled and analyzed with all formalisms. Gene regulatory networks and metabolic networks have only been modeled with specific techniques due to their specific system dynamics and topology. However, differential equations, constraint-based models, and Petri nets have been used as universal techniques to examine all of the mentioned networks.

The first thing to point out is that all formalisms have been applied to signaling networks. This it not surprising, as signaling networks have the largest number of features, such as spatial localization, multi-state components, network information flow, and robustness, among others. Therefore, each of the presented formalisms contributes with powerful features. A smaller number of formalisms are applied to metabolic networks. However, this does not indicate that other formalisms are not able to model those systems. Moreover, it seems that Petri nets, process algebras, constraint-based models, and differential equations seem to be powerful enough to consider all aspects of metabolic system dynamics. A further observation indicates that Petri nets, constraint-based models, differential equations, and cellular automata are applied to all kind of biological networks. This makes them potential candidates for whole-cell modeling. The most powerful technique is still differential equations modeling, which is also reflected by the data provided in the table. However, Petri nets are among the formalisms that cover most of the features to model all kinds of biological networks as described in Table 2.4. It is a universal graphical modeling concept for representing processes from different application fields in nearly all degrees of abstraction. Petri nets provide the qualitative modeling approach as well as the quantitative one. Furthermore, qualitative and quantitative formalism can be combined to one

paradigm. The formalism is easy to understand and use.

	Visualization	Topology	Modularity	Hierarchy	Multi-state	Compartments	Spatial	Qualitative	Synchronized	Stochastic	Continuous
Boolean networks	+	+						+	+	e	
Bayesian networks	+	+						+		+	
Petri nets	+	+	+	e	e			+	e	e	e
Process algebras			+	e		e		+		+	
Constraint-based models		+						+			
Differential equations							e			e	+
Rule-based models	+		+		+	+	e	+		+	+
Interacting state machines	+		+	+	+	+				+	
Cellular automata	+				+		+		+	+	
Agent-based models	+				+	+	+			+	

Table 2.4: Overview of implemented features for each modeling formalism based on [MCR+11]: (+) Supported feature; (e) Available through extension. Based on the provided data, the most powerful technique is the Petri net modeling as it includes the advantages and features of all other formalisms.

Once a basic qualitative model is established, it can be successively enriched with quantitative data. Thus, parameter estimations based on experimentally derived data is not implicitly necessary in the network reconstruction process. Furthermore, models can be modeled discretely as well as continuously. It is even possible to integrate ODEs for precise model description. Besides, Petri nets allow hierarchical structuring of models and thus offer the possibility of different detailed views for every observer of the model. Petri net theory provides a variety of established analysis techniques that are well-suited and applicable to biological network modeling. Moreover, database information, as described in the following section, can be used to automatically reconstruct sophisticated network and Petri net models.

2.6 Databases

Databases are an important resource in assisting scientists from natural sciences in their research. They provide knowledge that helps explain biological phenomena from genes up to

the entire metabolism of an organism. They contain life science information collected from scientific experiments, published literature, high-throughput experiment technology, and computational analyses. Using a database for information storage, users can benefit from reduced data redundancy, improved data security, improved data access, greater data integrity, increased consistency, and many other aspects [EN10].

The first biological database emerged in 1965 when Margaret Dayhoff published the Atlas of Protein Sequence and Structure [SD10]. In the 70s the first protein structure database, called Protein Data Bank (PDB) was founded [BKW+77, Mey97, BHNM07]. A few years, later in 1981, the first repository for nucleotide sequences was established called European Molecular Biology Laboratory (EMBL) [HC86, CAB+09] and one year later the GenBank [BFG+85, BKMC+12]. Since then, more and more biological databases have developed. The 19th annual database issue of Nucleic Acid Research (NAR) now lists more than 1,380 databases in molecular biology [GFS12]. The Pathway Resource List (Pathguide) [BCS06], a meta-database with an overview of more than 325 biological pathway related resources, with more than 100 databases focused on protein-protein interaction, is an additional important resource for biological databases. To make it easier for researchers to quickly find relevant information about useful molecular resources, tools and databases, community-curated databases with content and links to other biological databases were established. Some of the most important are MetaBase [BCP+12], Online Bioinformatics Resources Collection (OBRC) [CCB+07], BioDBCore [Bat10], and the Bioinformatics Links Directory [FBM+05, BYYO11]. Currently, more than 1,800 entries are listed in MetaBase, each describing different biological databases. BioDBCore gives a brief description of the core attributes of biological databases, whereas OBRC contains annotations and links for more than 1,700 bioinformatic databases and software tools. The Bioinformatics Links Directory curates links to software tools and databases. Using these resources, users have the possibility to contribute, update, and maintain database content.

A further resource for information is a so-called "wiki". A wiki is community website which allows its users to add, modify, comment, or delete content. Thus, content can be created and reviewed collaboratively. The most popular wiki is the online encyclopedia Wikipedia[2]. Finn *et al.* [FGB12] say that wikis are undoubtedly changing the way biological databases operate, since a database is no longer closed, moreover it can be commented, updated and improved by users; "A static resource is becoming dynamic." On the other hand new problems arise, such as the curation, participation and responsibility of researchers. At any rate, a growing number of databases using wiki systems and technologies suggest a new type of database storage, access, and editing.

In general, biological knowledge is distributed amongst many different general and specialized categories, covering all -omic levels. Concerning the NAR, the main categories for biological databases are:

[2] www.wikipedia.org

- Nucleotide sequence databases
- RNA sequence databases
- Protein sequence databases
- Structure databases
- Genomics databases (non-vertebrate)
- Metabolic and signaling pathways
- Human and other vertebrate genomes
- Human genes and diseases
- Microarray and other gene expression databases
- Proteomics resources
- Other molecular biology databases
- Organelle databases
- Plant databases
- Immunological databases
- Cell biology databases

To give insight into the growth of biological databases, Bolser *et al.* [BCP+12] examined the database for biomedical publications PubMed [WCE+04] for entries publishing new database information. They queried the database for unique publications containing the word "database" in the title. In 1980 only 2 publications were listed, whereas the number of database publications increased to nearly 1,200 in 2010 (see Figure 2.11). Querying the NAR for listed databases gives further insight into the growth of databases (see Figure 2.12).

However, having access to such a large amount of databases results in certain challenges. New high-throughput methods in biology generate more datasets than ever before. An ever increasing number of publicly available databases that analyze, integrate, and summarize leads to several problems. No single database has stored sufficient information about one specific topic or the resources to capture and organize all the published information. Moreover, researchers are challenged to query multiple databases to interrogate the largest possible dataset. Even on a specific biological or biomedical topic it might become a challenging and time-consuming task. Besides, a small number of databases are independently funded and pursue their goals in isolation. This especially applies to databases with a very specific topic. Another problem

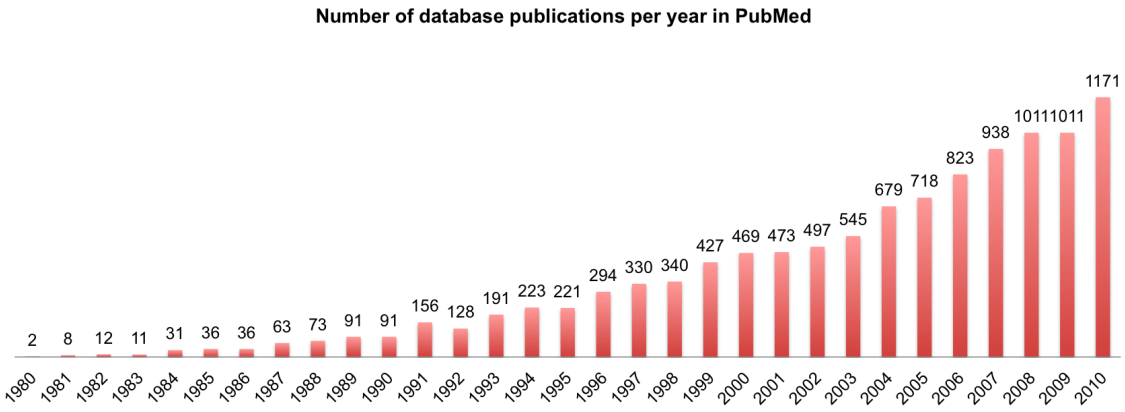


Figure 2.11: The growth of database publications per year. Each bar shows the number of research articles with the keyword "database" appearing in the article title in the given year. This data is provided by Bolser *et al.* [BCP+12] who counted indexed articles in PubMed [WCE+04].

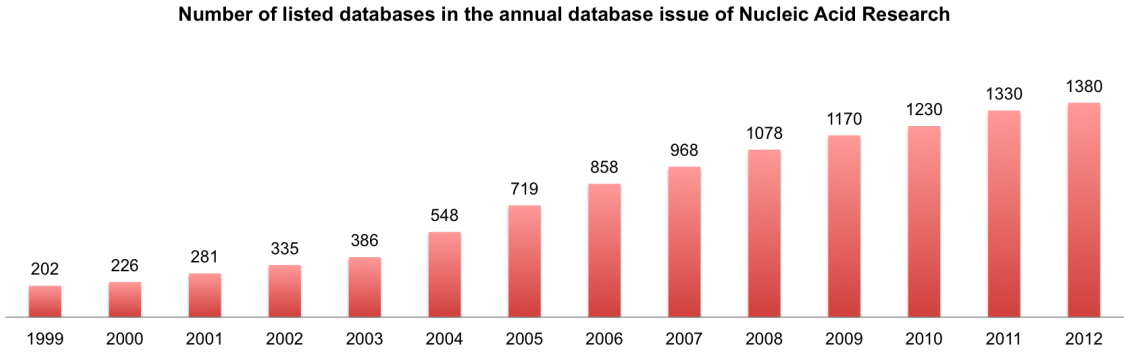


Figure 2.12: The number of listed databases in the annual issues of NAR. Each bar shows the number of databases in the given year.

occurs when resources only exist for short time due to the end of funding. Afterwards they are neither supported nor is database content up to date.

Database content is a further criterion for quality. To be confident about data, it should be collected from well-curated resources, experiments, literature, and evaluated analysis methods. In the best case scenario, a senior curator additionally curates provided information. Thus, noisy data, redundant information from overlapping sets of experiments or literature, different publication structures, definitions, and identifiers would not become a hurdle in obtaining information of interest. Ironically, sometimes new resources are created to cope with the perceived problems or omissions of existing databases instead of improving them.

In summary, biological databases are a great resource for research and almost all of them are free and available via internet. However, it is very difficult to judge the strengths, weaknesses, or status of the available databases. To cope with the variety and amount of available data, the heterogeneity of the data in different sources, and the autonomy and different capabilities of the sources, the topics biological data integration and data warehousing have cropped up and became a major focus of bioinformatics as described in the next section.

2.6.1 Data integration

The increasing number of databases and their high heterogeneity makes it difficult for scientists to find and work with the provided knowledge in a global -omics context [LC03, SKSB00, KHH11]. Biological data should enable scientists to perform comparative analyses and modeling in disciplines such as genomics, proteomics, and metabolomics. For example, a scientist might need to link gene expression data to protein-protein interactions, or sequences to disease and molecular structures. Some databases share links to other resources, but some share only few or no links to other databases, particularly to databases covering other -omic levels. Looking at the complexity of data, the way it is published and made available, users are challenged to process each data source as to how it fits into their overall research.

Biological data presents numerous challenges from the lack of standardized data to data inconsistencies resulting from experimental data variations and annotations [TB11]. Missing standards and consensus for basic biological terms also cause semantic heterogeneity. Furthermore, data quality differs from database to database depending on the curation model. Similar data can be overlapping in the databases and moreover, have different meaning, interpretation, types of terms and concepts. Also database access and structure might be a problem, since databases operate autonomously and free in their internal design and publication structure. Databases might be only available through a specific interface and service, where each database has to be queried independently.

To adopt to the aforementioned problems, computer scientists have come up with a technology called data integration. In recent years many systems for data integration were developed in

bioinformatics to cope with heterogeneous, autonomous, and distributed data sources. Data integration is one of the most important tasks in bioinformatics [KHH11], integrating various data sources into one local repository. In order to migrate data from one database into another, adapted extract, transform, and load functions are used (ETL). Using ETL functions, data is read from one database, converted from its previous form into the intended, cleaned, filtered, annotated, and linked to other database content and finally loaded into the new database or data warehouse.

In general, two data integration approaches exist, namely the materialized integration, where all data is stored in one central repository and virtual integration, where the integration of data is temporal and ensued with the data query. Having all information stored in one local repository, users are able to query just one, rather then different and heterogeneous databases for the information of interest. This results in a repository, called a data warehouse. Data warehouses are one of the widely used architectures of materialized integration [EN10]. Therefore, a brief overview of existing data warehouses is given in Section 3.5.

2.7 Network reconstruction

A biological network, as described in Section 2.2 consists of a set of different biological elements being in interaction with each other. Such a network can be reconstructed by hand, with experimental data, information from literature, and/or database knowledge. In the first case, users need to put all involved elements into relation and draw the resulting models as a graph. They have several possibilities to model the system. They can use directed, undirected, mixed, or other graphs as presented in Section 2.3. Furthermore, they can use a standard graphical notation, such as SBGN for the visual modeling as presented in Section 2.5.13.

In terms of a network reconstruction with experimental data correlation, networks have to be reconstructed as described in Section 2.2. Therefore, a well-established modeling and analysis technique is necessary. One possible approach are Bayesian networks as described in Section 2.5.9. Bayesian networks offer one way to automatically reconstruct signaling networks from experimentally derived data. The only disadvantage of this approach is the necessary input data. To be able to produce unambiguous results a huge set of experimental data is mandatory.

A further way to reconstruct biological networks is by using text mining approaches [KAAEV05, HKA+05]. Text mining is equivalent to text analytics, with the goal of turning text into data for further analysis. This approach can be used, for example, to find interaction partners for a gene by analyzing a set of publications. The collected data is then modeled as a graph. In general, this technique is based on statistical pattern learning. The main disadvantage of this approach is still the interpretation of the input text. In many cases relations are identified which are positive-false or false-positive. Although the analysis and results are becoming better and better, the resulting networks need to be evaluated by an expert.

A more reliable way to reconstruct biological networks is by querying biological databases. Therefore, more than 1,300 different biological databases exist that can be accessed as described in Section 2.6. Using complex queries, data transformations, and data integration techniques rudimentary data, such as genes and proteins can be linked with each other. Many databases provide links between the different biological compounds. If such a link does not exist, it is even possible to establish connections by mining genomic databases. Hence, several attempts have been made to reconstruct metabolic pathways via genome sequence comparison [MK96, BOGK98]. Such attempts have a certain limit, as the results do not reflect all involved molecular functions. Due to cellular functions, such as translation, transcription, post-modification, and many more processes with genome sequence comparison and analysis it is often not possible to predict direct correlations and further regulatory or metabolic processes.

However, several databases do exist, which contain more detailed information about metabolic pathways, such as the KEGG database [KGS+12]. The information about the networks can be accessed via Internet or by parsing provided flat-files. The disadvantage with online access is that the elements cannot be analyzed and combined with other -omic level data and experimental datasets. Therefore, flat-files have to processed, filtered, normalized, and integrated into one model. Actually, the KEGG database consists of more than 121 tables, where at least 23 tables are necessary to reconstruct the backbone of a biological network. The other tables store further information, such as diseases, drugs, taxonomies, and much more (see Figure 2.13 for a simplified scheme of the KEGG database structure). With access to that data it is possible to reconstruct metabolic networks as they are presented by KEGG and to analyze the biological elements in detail or overall context.

2.8 Discussion

The previous sections gave an impression of how complex cellular life is and presented which possibilities exist to reconstruct, model, and analyze cell behavior. Based on the logical analysis of the available information and approaches, now it is discussed which of the approaches are necessary to make VANESA a useful software.

Cellular life is very complex and governed by thousands of macroscopic functions being constantly carried out. To produce good theoretical models which can be used for hypothesis testing, the models need to be manageable. This can only be achieved by reducing a biological system to the known and essential parts, which are necessary to answer the underlying research questions. By trying to model a complete system, regardless of the lack of data and parameters, it is very likely that the modeled systems can be misleading. Therefore, VANESA needs to have a clear focus rather than model all levels of biological details.

One of the best ways to start modeling a biological system is by using biological networks. A small network consisting of known and already analyzed elements can be the initial point for the

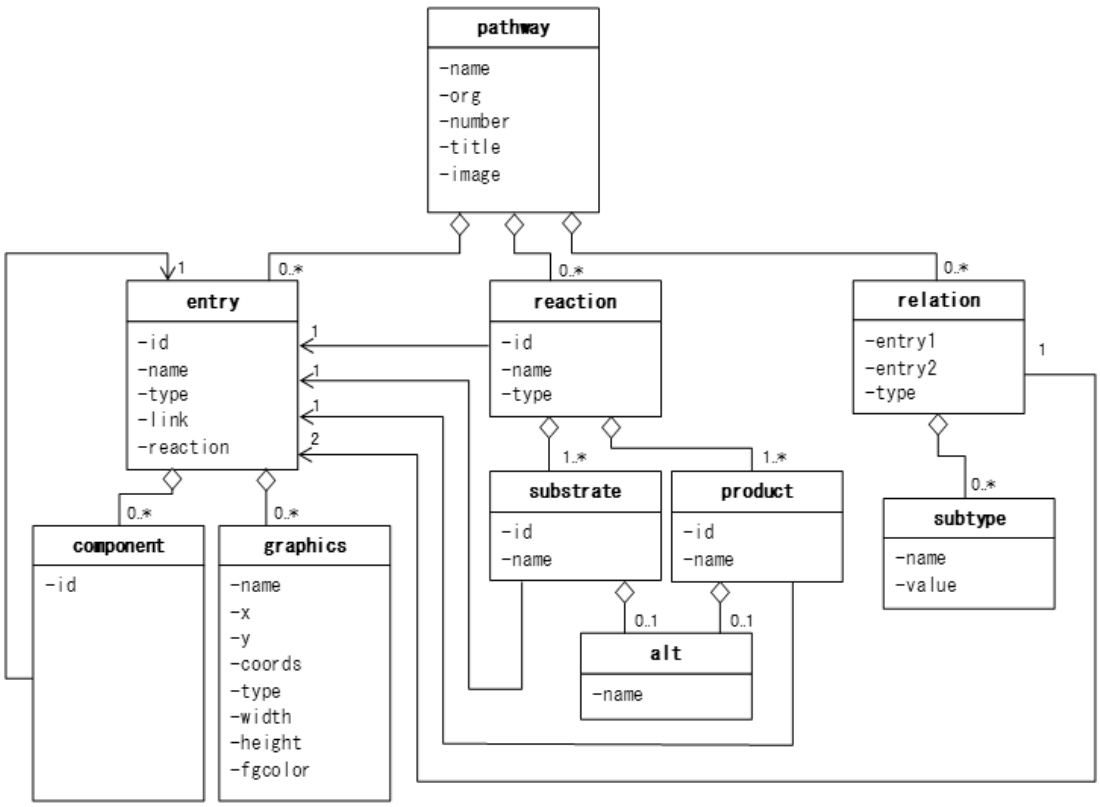


Figure 2.13: Simplified scheme of the KEGG database structure. The pathway element is the root element of the biological network, consisting of a list of entry, relation, and reaction elements. Theses entities specify the graph information. Additional elements specify more detailed information about the biological compounds, relations, and reactions within the model (picture from http://www.kegg.jp/kegg/xml/docs/).

reconstruction of a more significant system. Therefore, there are different biological networks which can be used as powerful integrated frameworks to present, integrate, and visualize knowledge. As these networks are intuitive and easy to extend in knowledge, any scientist can work with them. With biological networks different -omic levels can be modeled, describing elements such as genes, RNAs, proteins, and metabolites being in interactions and relationships with each other. Moreover, biological databases can be used to reconstruct or enrich those networks with relevant information and new data. Kinetics and other information can be queried to model a system in a more precise way. With database integration modules it is even possible to query multiple databases with one view instead of consulting each database separately. Besides, data integration tools filter, normalize, and link heterogeneous data from different distributed data sources. Concerning the great benefit of the access to such a data repository, any kind of modeling software solution should have the possibility to access such a repository.

A further advantage of biological networks is that a wide range of graphical theoretical analysis techniques can be applied on reconstructed models. Graph theory can give important clues about topological network properties, such as the identification of the most important nodes within a system, or average path lengths between different elements in a biological model. This is important in as much as biological networks can become large and complex. Scientists need a tool which assists them in identifying relevant information. Therefore, VANESA should also support such mathematical analysis.

When it comes to simulating cell behavior scientist often speak about ODE modeling. Indeed, it is one of the most powerful approaches, but needs prior knowledge in mathematics and a complete set of biological data and parameters. These are high requirements for a modeling approach when scientists try to reconstruct and understand system behavior or unknown regulatory processes. Thus, a more intuitive approach is necessary in VANESA, which can be used in the beginning without biological data and is still able to imitate and predict cell behavior. Therefore, Petri nets can be used for the description, simulation, and analysis of complex and distributed systems. Petri nets cover most of the needed features for network modeling and provide qualitative as well as quantitative modeling features. Furthermore, it is possible to integrate ODEs for precise model descriptions.

Putting all this approaches into one framework, VANESA becomes a powerful modeling, simulation, and analysis software that can be used to discuss and answer complex biological questions. As this chapter discussed the basics of system modeling the following chapter will go into more detail and present, which detailed bioinformatics approaches and data sources are available and can be used by VANESA. In addition, it lists the most relevant software solutions in the same field of studies with its advantages and disadvantages and shows why VANESA is so important.

Chapter 3

Related work

This chapter presents and discusses important and relevant work in terms of VANESA[1]. Therefore, the first section refers to software applications, which are able to model and simulate biological systems. Based on the discussion of these tools, the need and motivation for VANESA is given. Discussed are the state-of-the-art applications CellDesigner, CellIllustrator, Cytoscape, E-Cell, Gepasi, JDesigner, PNlib, and Snoopy. The following sections present further analysis approaches from related work, which are very valuable and therefore, should also be integrated in VANESA. Section 3.2 describes sophisticated analysis methods for Petri nets that can be used in VANESA for the calculation of possible system states, among others. Important graph theoretical approaches specially applied to biological networks are discussed in Section 3.3. The section focuses on centrality measurement techniques and explains how it can be used for network analysis. Section 3.4 lists existing biological databases, which can be used to enrich biological models with important information and also used for the reconstruction of biological networks. Databases discussed are KEGG, STRING, BioCarta, PID, HPRD, IntAct, MINT, ENZYME, and BRENDA. The important data integration tools and approaches, BioWarehouse, ONDEX, BioDWH, and DAWIS-M.D. are presented in Section 3.5. These are discussed as possible approaches and repositories for VANESA in order to access the aforementioned biological databases. Standard exchange formats are discussed in Section 3.6, as these are necessary to share and evaluate in VANESA reconstructed systems within different software applications. Finally, the last section discusses which approaches VANESA should make use of.

3.1 Competitive bioinformatics software applications

In general, more than 1,700 bioinformatic databases and software tools exist, as mentioned in Section 2.6. Using these resources, scientists have the possibility to model biological systems in

[1]Presented data and information is based on mentioned publications and provided statistics from August 2012.

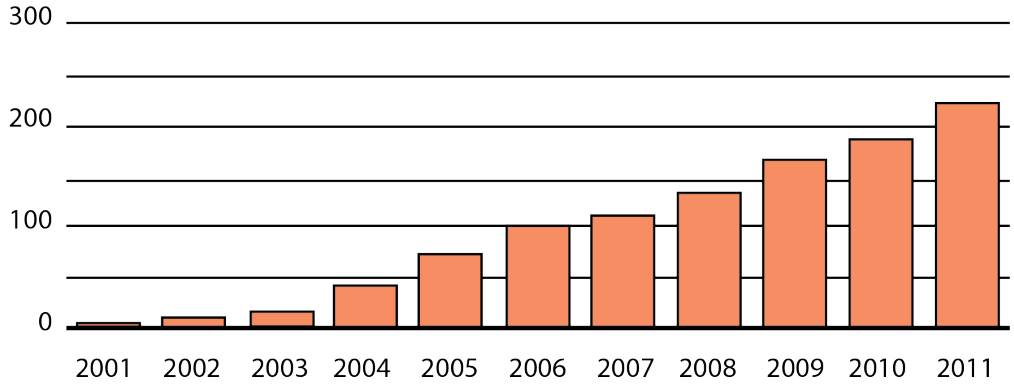


Figure 3.1: Number of software applications providing SBML functionality from 2001 to 2011 (picture from http://sbml.org/).

many different ways and furthermore, enrich any kind of biological system with relevant biomedical knowledge. In 2011, the SBML website[2] listed more than 200 software tools which provide biological modeling based on the Systems Biology Markup Language (SBML) [FH03, HFS+03]. (see Figure 3.1). However, the number of available tools is constantly increasing. In order to narrow computational tools, only the best suited applications for the modeling, visualization, analysis, and simulation of biological networks are considered, which are also supported and state-of-the-art. Therefore, Copeland *et al.* highlighted a small, representative portion of available tools from each -omic area [CBC+12]. Still, this review lists more than 30 tools specialized in biological modeling. For the following discussion only those tools which are able to model, reconstruct, visualize, and simulate biological systems in one single comprehensive framework are taken into account. Discussed in alphabetic order are the state-of-the-art applications CellDesigner, CellIllustrator, Cytoscape, E-Cell, Gepasi, JDesigner, PNlib, and Snoopy. For each of the software applications a short summary is given presenting the most important features of the tool. Finally, the last part of this section discusses the advantages and disadvantages of each of these applications.

[2] http://sbml.org/

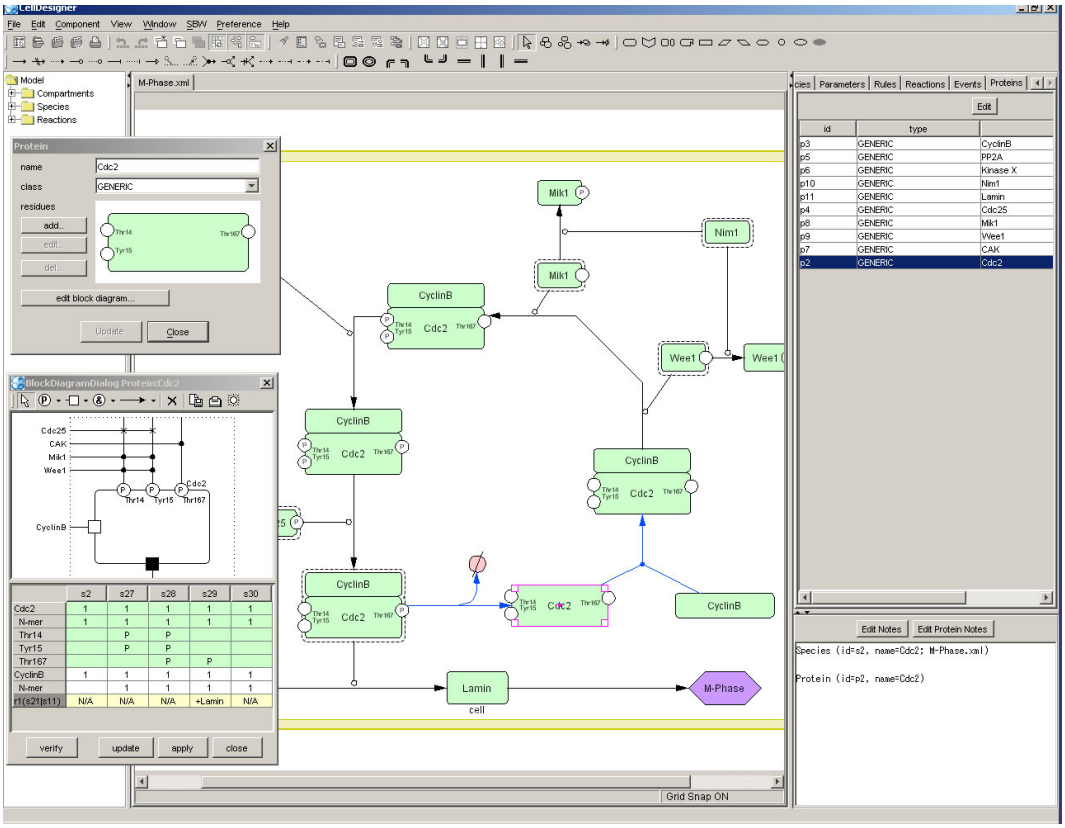


Figure 3.2: A screenshot of CellDesigner modeling a biochemical reaction (picture from http://www.systems-biology.org).

CellDesigner

CellDesigner is a structured diagram editor for drawing gene-regulatory and biochemical networks (see Figure 3.2). It was developed by the Systems Biology Institute (SBI) in Tokyo, Japan [FMKT03]. The core members of this software application are Akira Funahashi, Hiroaki Kitano, and Akiya Jouraku. The main goal of this application is to visually represent biochemical reactions in a comprehensive graphical notation such as SBGN (Systems Biology Graphical Notation) [LNHM+09]. Besides, in the new version it enables users to connect from species name or ID to the databases Saccharomyces Genome Database [CHA+12], iHOP (Information Hyperlinked over Proteins) [HV04], and the Genome Network Platform (http://genomenetwork.nig.ac.jp). Furthermore, it is possible to get basic information about a biological element from PubMed [WCE+04] or Entrez Gene, the search engine from NCBI (http://www.ncbi.nlm.nih.gov). To assist users in the simulation, CellDesigner is able to connect to the SBML ODE Solver [MFM+06] and Copasi, a biochemical network simulator [HSG+06]. Simulations can be set up in a control panel, where users are able to adjust system amounts and parameters. CellDesigner is free-of-charge and available at http://www.celldesigner.org in Version 4.2 running under Windows and Linux.

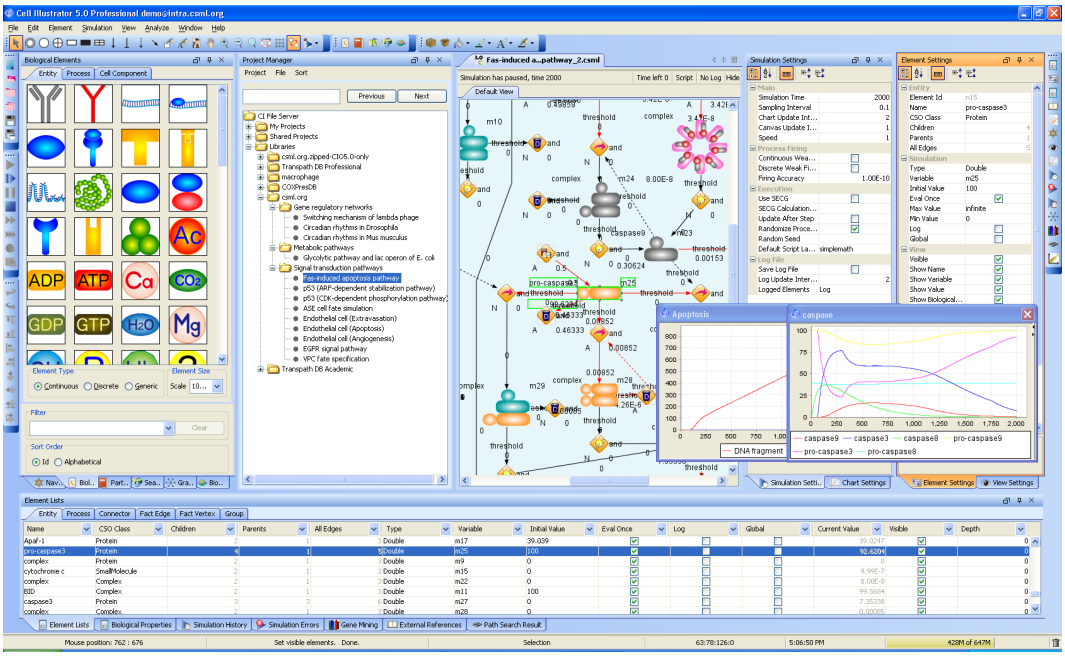


Figure 3.3: A screenshot of CellIllustrator 5.0 visualizing simulation progress and results of the apoptosis pathway (picture from http://www.cellillustrator.com/).

CellIllustrator

The software application CellIllustrator [NSJ+10] is a software platform for systems biology that uses the concept of the Petri net language for the modeling and simulating of biological networks. The first version of CellIllustrator was published as Genomic Object Net [MDDM00] in 2000 under Matsuno *et al.* at the Faculty of Science, Yamaguchi University, Japan. The software application employs the concept of a Hybrid Petri net as the modeling and simulation method. To handle any type of objects, the existing paradigm has been extended to Hybrid Functional Petri nets with extension (HFPNe). This paradigm is more suitable for biological network modeling and simulation, since HFPNe can handle discrete and continuous events simultaneously. Any kind of function can be assigned to delay, weight and speed parameters of these elements. Additionally, ordinary differential equations can be modeled and integrated into a subset of HFPNe.

Furthermore, CellIllustrator is able to import pathways or single reactions from the Transpath database [KPV+06]. To import networks from other tools, SBML, CellML, and BioPAX data exchange formats are supported. In addition, CellIllustrator has its own format called CellIllustrator Markup Language (CSML). Simulation results can be visualized either in 2D or 3D plots in an all-in-one-window environment (see Figure 3.3). To make the network visualization more legible graph grid layout algorithms are implemented. The latest version of CellIllustrator is Version 5.0, which is commercially as online version available at http://www.cellillustrator.com.

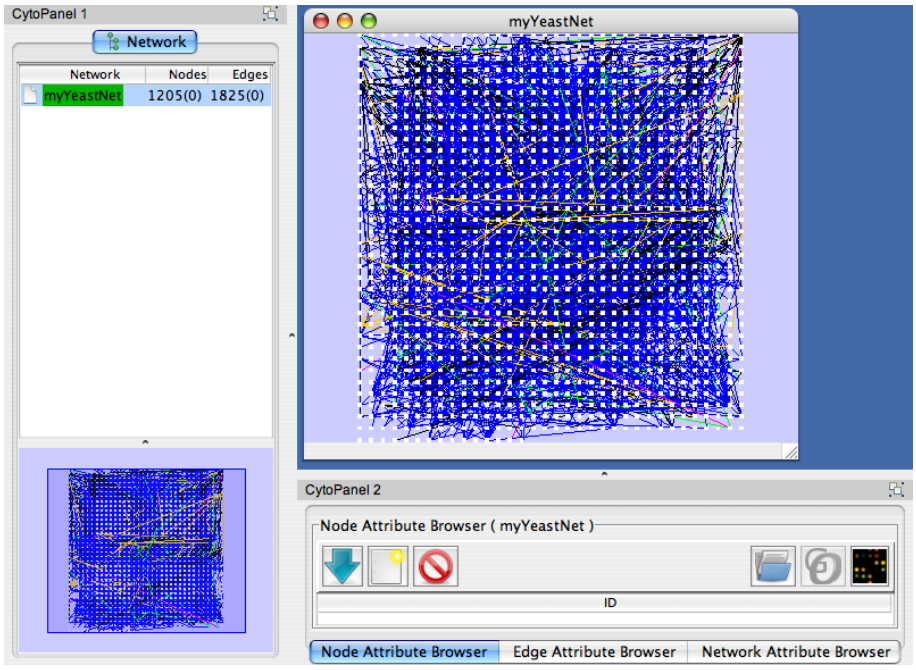


Figure 3.4: A screenshot of the Cytoscape plugin BioNetBuilder reconstructing a *Saccharomyces cerevisiae* protein-protein interaction network (picture from http://err.bio.nyu.edu/cytoscape/bionetbuilder/tutorial/).

Cytoscape

Cytoscape is an open source bioinformatics software platform for data integration and visualization [SOR+11]. The first version of Cytoscape was published by Shannon *et al* from the Institute for Systems Biology, Seattle, Washington [SAO+03]. Nowadays, it is supported and funded by many different institutions, particularly by Agilent Technologies, University of Toronto, Institute Pasteur, Memorial Sloan-Kettering Cancer Center, Institute for Systems Biology, and the University of California San Diego. Primarily, Cytoscape enables users to visualize molecular interaction networks and biological pathways and integrate these with any type of attribute data, such as gene expression profiles. Furthermore, Cytoscape supports standard network and annotation files such as BioPAX [DCP+10], SBML, and others. Additional features are available as plugins, which are developed by third parties focusing on network and molecular profiling analyses, new layouts, additional file format support, scripting, and connection with databases. For network reconstruction there is the plugin BioNetBuilder [ACDL+07] (see Figure 3.4), which uses the databases KEGG [KGS+12], HPRD [KPGK+09], BioGrid [BSR+08], and GO [BDD+12], among others for its modeling. Furthermore, simulation plugins exist, such as the SimBoolNet [ZZP+09], for the simulation of Boolean networks or FERN for the stochastic simulation and evaluation of reaction networks [EFZ08]. Most of the plugins are available free-of-charge. Cytoscape uses an open API based on JAVA technology and version 2.8.3 is available at http://www.cytoscape.org.

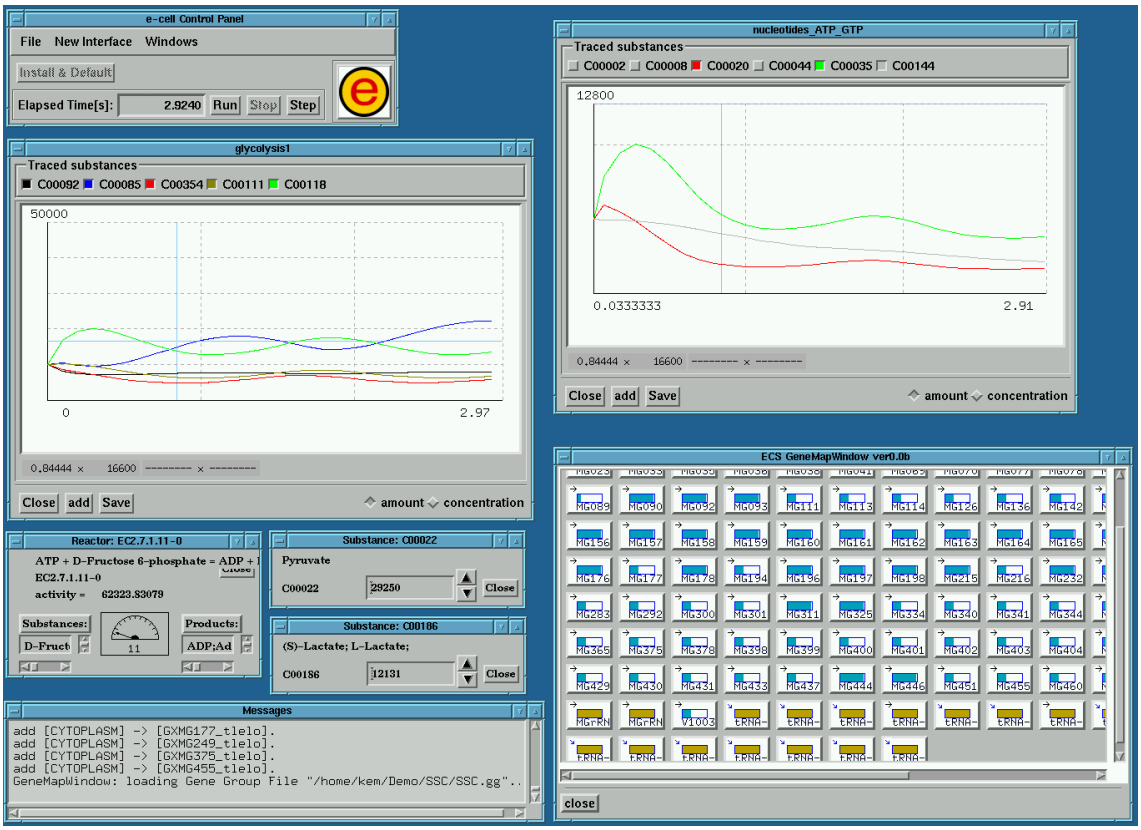


Figure 3.5: A screenshot of the E-Cell simulation environment (picture from [TKHT04]).

E-Cell

The E-Cell project [THT+99] is an international research project aimed at modeling and reconstructing biological phenomena *in silico*. The main goal of this software application is to develop a dynamical cell with all its functions. It has been developed by Hashimoto *et al.* at the Institute for Advanced Biosciences, Keio University, Yokohama, Japan. The software platform allows precise whole cell simulations with object-oriented modeling. Therefore, numerical integration methods are encapsulated into biologically related object classes. Virtually any integration algorithm can be used for simulation [TKHT04]. Thus, users have the possibility to define functions of proteins, protein-protein interactions, protein-DNA interactions, regulation of gene expressions, and other cellular cell processes with a set of functions rules. Therefore, hundreds of reaction rules are provided and available for simulation progress (see Figure 3.5). E-Cell Version 3 is freely available at http://www.e-cell.org and runs on several different platforms such as Microsoft Windows and Linux.

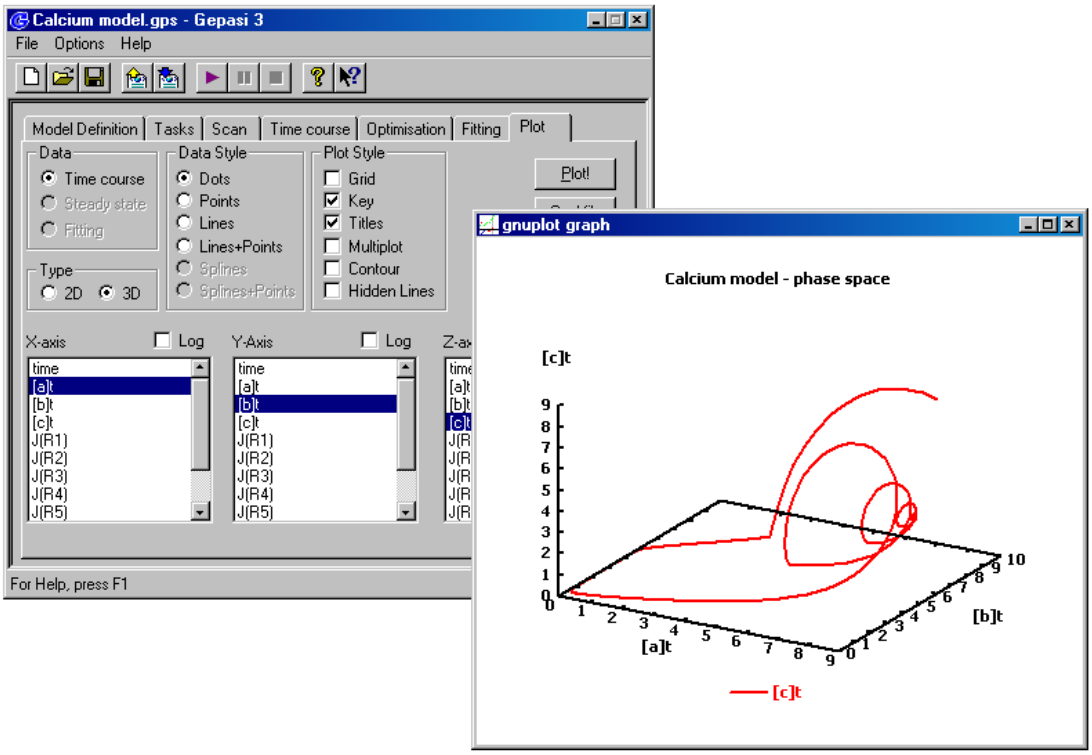


Figure 3.6: A screenshot of the Gepasi simulation environment (picture from http://www.gepasi.org/gep3plot.png).

Gepasi

Gepasi is a software application for the modeling and simulating of biochemical systems [Men97, Men93]. It has been developed by Pedro Mendes at the Department of Biological Sciences, University of Wales, Aberystwyth, UK. Gepasi uses mathematical formulas to transform biochemical properties into kinetic models. It provides a number of tools to fit data, to optimize any function of the model, to perform metabolic control analysis and linear stability analysis. Sophisticated numerical algorithms realize simulation processes and analysis tasks. The simulation results can be plotted in 2D and 3D (see Figure 3.6). Furthermore, the software application supports SBML 1.0 import and export. The latest version of Gepasi is 3.30 and freely available at http://www.gepasi.org. It only runs using Microsoft Windows.

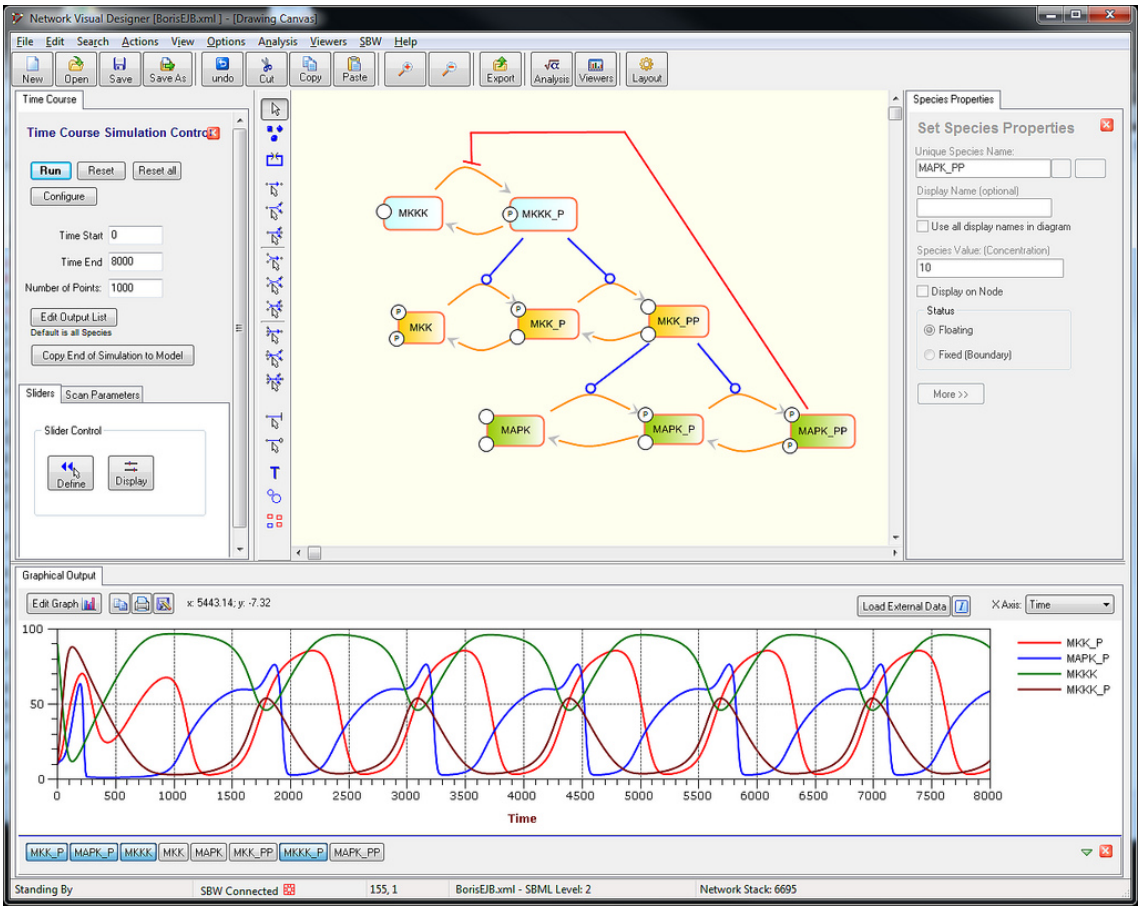


Figure 3.7: A screenshot of JDesigner simulating the phosphorylation of a protein-protein interaction network (picture from http://sbw.kgi.edu/software/jdesigner.htm)

JDesigner

JDesigner is a software application that enables users to draw a biochemical network, which can be exported to SBML for further processing [SHF+02]. The development of JDesigner was supported by the California Institute of Technology, Pasadena, California and more recently by the KECK Institute of applied sciences, Claremont, California USA. JDesigner represents networks by using one notation for chemical species, which can be decorated with visual cues (see Figure 3.7). This is also possible for reactions. Although it is a network design tool it also supports simulations. It has the ability to use JARNAC as a simulation server via the Systems Biology Workbench (SBW) [SHF+02] which is an open source framework connecting heterogeneous software applications. JDesigner is an open source project distributed under the LGPL license and available at http://sbw.kgi.edu/software/jdesigner.htm.

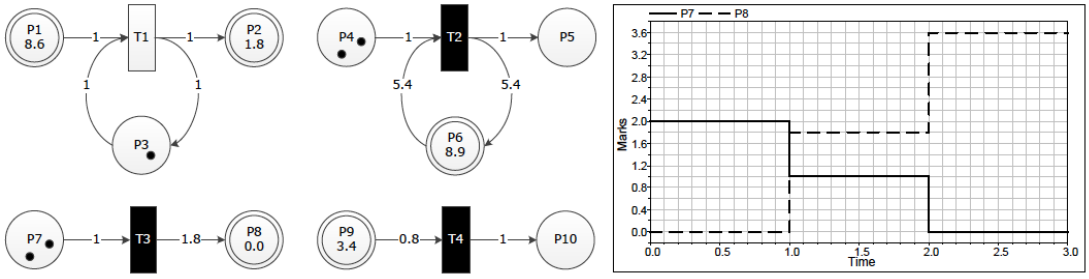


Figure 3.8: A screenshot of results of the PNlib in Dymola (Modelica) visualizing and simulating a basic Petri net model (picture from [PJHB12]).

PNlib

The PNlib is the powerful new state-of-the-art Petri net simulation library [PB11]. Proß *et al.* have developed the PNlib library using the Modelica language [Ass05] at the Department of Engineering and Mathematics, University of Applied Sciences, Bielefeld, Germany. Modelica was developed and promoted by the Modelica Association since 1996 for modeling, simulation, and programming. Primarily it is focused on physical and technical systems and processes. Now, Modelica, embedding the PNlib, provides the possibility to simulate biological systems (see Figure 3.8).

The PNlib is based on the Extended Hybrid Petri Nets for biological applications (xHPNbio) formalism [PJB+12]. The mathematical modeling concept xHPNbio was specially developed for scientists, based on the demands of biological processes. The focus of this formalism is the processing of experimental data to gain usable new insights about biological systems. The xHPNbio elements are modeled object-oriented by discrete, algebraic, and differential equations in the Modelica language. In order to achieve reliable simulation results, simulations can be performed with different solver settings in Modelica. The mathematical modeling concept xHPNbio allows users to model and simulate many kinds of processes: business processes, production processes, logistic processes, work flows, traffic flows, data flows, multi-processor systems, communication protocols, and functional principals. Hierarchical modeling, hybrid simulation, and animations are also featured. All this is possible due its universal and generic design. The PNlib for Modelica is only available for the commercial interpreter of Modelica, called ”Dymola” [AB13]. The developers of the PNlib are presently working on a module for the free interpreter of Modelica, called ”openModelica” [FAL+05].

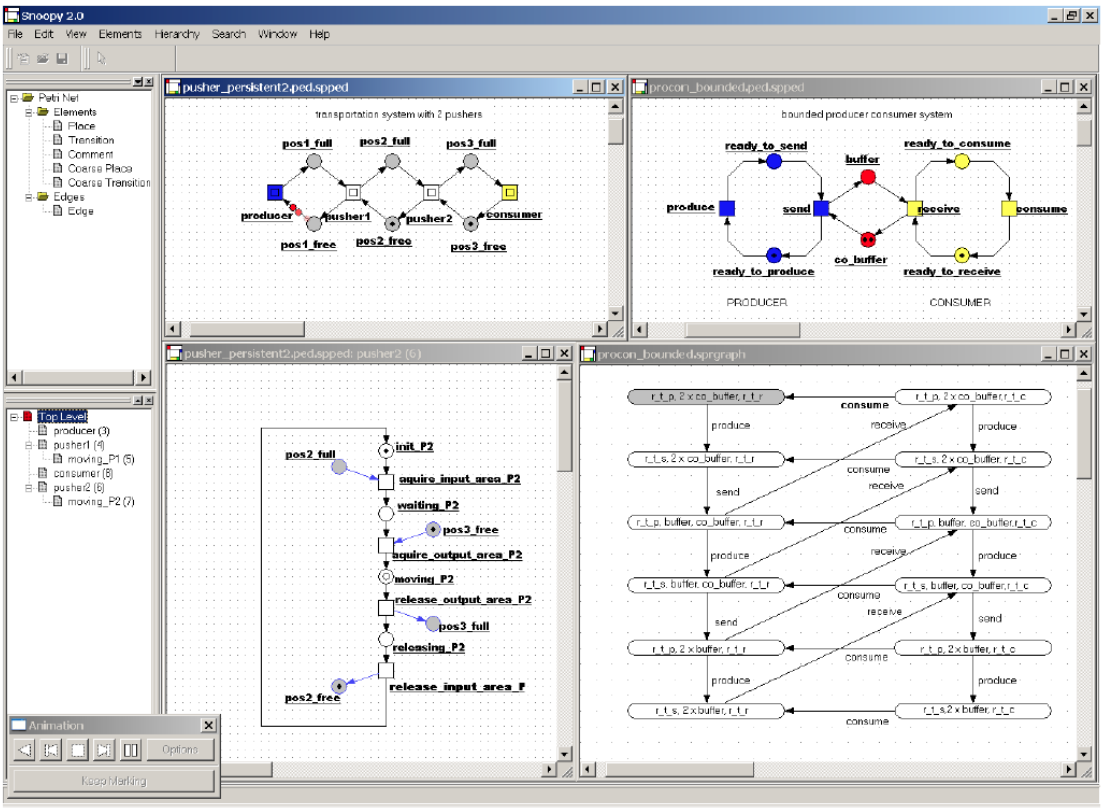


Figure 3.9: A screenshot of Snoopy visualizing simulation progress and results.

Snoopy

Snoopy [HRSR08, RMH10] is a unifying Petri net framework to investigate biomolecular networks. It has been designed and implemented by Heiner *et al.* at the Brandenburg University of Technology at Cottbus, Germany. The simulation environment comprises a family of related Petri net classes, such as time Petri nets, stochastic Petri nets, continuous Petri nets, hybrid Petri nets, colored Petri nets, and extended Petri nets, among others. The mentioned classes enhance standard Petri nets in various ways to meet the demands of biological scientists. For example, the extended Petri nets are characterized by read arcs, inhibitor arcs, equal arcs, and reset arcs. Using these formalisms, scientists are able to reconstruct and simulate any kind of dynamic network. Larger networks can be hierarchically structured. If further demands on the supported Petri nets should arise, the software application can be extended by new properties and even by new Petri net classes. This is possible due to the generic data structure of the software application. Furthermore, users are able to move between the qualitative, stochastic, and continuous modeling paradigms. However, this transformation from one paradigm into another is not possible without information loss.

Simulation results are visualized within a built-in animation environment (see Figure 3.9). To be able to share results with other scientists and software applications, Snoopy offers SBML

support with both import and export functions. Snoopy is available for all major operating systems, such as Windows, Linux, and Mac OS-X. It is available free-of-charge at http://www-dssz.informatik.tu-cottbus.de/snoopy.html.

Summary

In order to compare the aforementioned software applications, each of the tools was examined in terms of graphical modeling usability, possibility to automatically reconstruct biological networks based on database information, network analysis (graph theory, mathematical analysis, Petri net analysis, etc.), network visualization and interaction, and the possibility to simulate biological systems (see Table 3.1). Although all tools are quite strong in their main application field they have certain disadvantages or are missing certain approaches for the entire spectrum of biological modeling.

	Network modeling	Network reconstruction	Network analysis	Network visualization	Simulation
E-Cell	+	m	o	o	++
Gepasi	-	m	o	m	+
Cytoscape	++	p -	p ++	++	p -
CellIllustrator	++	+	-	o	-
Snoopy	o	m	+	o	++
PNlib	o	m	o	o	++
CellDesigner	++	-	o	++	p o
JDesigner	+	m	m	+	p o

Table 3.1: Comparison of existing software applications concerning necessary features for the modeling and analysis of biological systems. Legend: ++ (strong), + (good), o (sufficient), - (weak), m (missing), p (only available as plugin).

The software applications E-Cell and Gepasi are well-suited for precise process simulations and even whole cell simulations. Therefore, they provide a powerful framework, in which users can set up equations and specify system parameters. However, programming skills are essential for these tools and users need to have prior knowledge about differential equations. Database access to biological databases is not provided and the analysis is only possible with the specified equations and resulting simulations.

Cytoscape offers a strong platform for visualization. Besides, Cytoscape offers a well-designed plugin structure for the integration of new program modules. For practically every aspect of

system modeling and simulation a third-party plugin exists. However, many of these plugins show disadvantages in their possibilities or are no longer available in the new version of Cytoscape. The BioNetBuilder, one of the strongest plugins for automatic network reconstruction, is able to access some important biological databases but fails in reconstructing sophisticated networks. The reconstructed networks are a collection of unfiltered data resulting in furballs in most cases. Besides, many different and cumbersome steps govern network reconstruction. For the simulation, several different ODE solvers are available, such as FERN, for example. However, the quality of the results is not always comprehensible and it is difficult to use the results for the identification of network motifs, regulatory switches, and so on. So far, a plugin for Petri net analysis is not provided.

The JDesigner is a good modeling and visualization tool, but without the possibility of accessing life-science databases for the reconstruction of biological models. However, the simulation can be performed with the external tools JARNAC or the Systems Biology Workbench. Therefore, the models can be exported and further investigated based on the results of the provided ODE solver.

The CellDesigner is strong in its possibility to draw and model biological systems but weak in network reconstruction and analysis. Although the software application is able to access some important databases, the tool only enables users to enrich model elements with given database information. Based on this information, users can manually extend their networks step by step. Simulations can be performed using external tools such as Copasi and other ODE solvers.

By contrast, CellIllustrator offers an easy-to-use interface, which enables drawing, modeling, analyzing, and simulating complex biological processes and systems based on extended hybrid functional Petri nets (HFPNe). However, the weakness of CellIllustrator is the simulation itself. There is no information about how the Petri nets and the corresponding processes are defined and simulated. It is not known how conflicts in Petri nets are resolved, how the hybrid simulation is performed, and which integrators are used. Due to its evolutionary design and many changes, the core has become opaque over the last few years. Further down the line, there is no possibility to adapt solver settings to achieve reliable simulation results.

An alternative to CellIllustrator is Snoopy. Using Snoopy, Petri nets can be modeled time-free (qualitative model) or its behavior can be associated with time (quantitative model) such as stochastic, continuous, and hybrid Petri nets. Furthermore, users are able to work with different classes of Petri nets by converting them into each other. But this is not possible without information loss. A further feature of the application is that large systems can be hierarchically structured to manage complex networks. However, the software application has some drawbacks. For example, a continuous Petri net is interpreted as a graphical representation of a system of ordinary differential equations. Hence, the general Petri net property of non-negative marks cannot be held during simulation. Furthermore, conflict situations in hybrid Petri nets can occur such as negative markings. Moreover, places cannot be provided with

capacities and no functions can be assigned to arcs in hybrid Petri nets, which is essential for biological modeling and simulation. The modeling possibilities are not intuitive and network reconstruction not provided.

The PNlib for Modelica is another way to model and simulate biological systems using Petri nets. It is based on the xHPNbio formalism, which has been specially developed for biological applications. Due to this formalism, new classes of biological Petri nets can be object-oriented modeled. Furthermore, Modelica allows users to choose between several solver settings to perform hybrid Petri net simulations. The drawback of this approach is that it is only available as an add-on for the commercial version of Modelica. Furthermore, network reconstruction possibilities are not provided and the modeling and visualization features are weak.

In summary, no existing application is able to model, visualize, analyze, and simulate a biological model with sophisticated methods. Users are faced with using many different approaches and tools in combination to cover all important aspects in dynamic cell modeling. Furthermore, users need prior knowledge in mathematics and a good background in computer science. Another drawback of all the presented tools is the access to biological databases and moreover, the possibility to reconstruct biological systems using the provided information. None of the tools was able to convince or at least to produce biological networks suitable for biological analysis. This is because of the missing link to some important databases or the produced results, which were not specific enough, mainly resulting in furballs. The existing knowledge from existing databases could not be employed in a usable way. The simulation is another drawback of some of the tools. If simulation techniques are provided, they are mainly based on mathematical approaches. This requires mathematical knowledge and moreover, a set of biological data and parameters that can be used for simulations. Therefore, Petri nets are more suitable, as they can simulate biological networks in a qualitative and quantitative manner. However, CellIllustrator failed in producing comprehensible results and Snoopy showed other disadvantages. The PNlib is convincing but comes with weak modeling possibilities and visualization. Furthermore, it is not connected to any kind of biological database. Finally, a strong need for a software application exists, which provides strong modeling features where models can be reconstructed or enriched with biological database information, then analyzed in different ways, and finally simulated in a qualitative and quantitative manner.

3.2 Petri net analysis

In terms of analytic power, Petri net concepts provide sophisticated approaches for the analysis of biological systems [GBSH+08, LSG+06, SHK06, ZOS02]. Petri net analysis can be divided into two topics, namely dynamic analysis and static analysis. Dynamic analysis is performed by calculating the whole or partial space state of the model, such as done with the liveness and reversibility analysis. Static analysis is performed without reconstructing or calculating the state

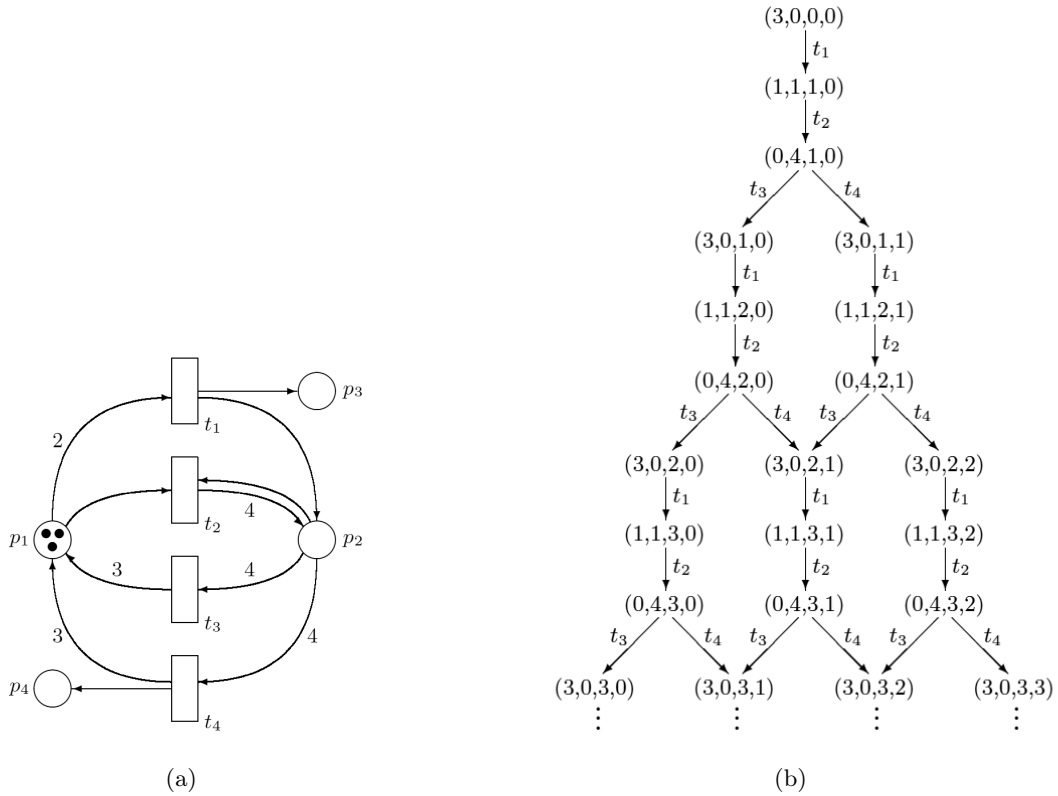
space, as done with the boundedness property analysis. However, liveness, reversibility, and boundedness are the three major Petri net behavioral properties and described as follows:

- *Liveness of a Petri net*: In an infinite net behavior, with a sufficient amount of input compounds, the network will never stop working. It is assumed that each transition will stay alive in the whole state space. Independent of past events, it will always be enabled. An interruption in signal flow indicates a modeling error.
- *Reversibility of a Petri net*: A system is reversible if the initial marking of the model can be reached again.
- *Boundedness of a Petri net*: This property indicates if the system accumulates unlimited tokens in one place. If the maximal number of tokes for each place is limited, a Petri net is bounded.

One way to perform dynamic analysis is by constructing a dedicated graph, the so-called "reachability graph" or "covering graph", which is finite for bounded Petri nets. A reachability graph represents all possible markings of a model by considering all possible transition firings (see Figure 3.10). Based on this graph it can be determined if a given Petri net can reach a given system state by finding a set of necessary conditions. Therefore, the graph is walked through until a given Petri net property is found, such as a requested marking. However, the construction of such a reachability graph can be exponential in time and space. Furthermore, the reachability graph has several other problems such as the Finite Reachability Tree Problem (FRTP), the Finite Reachability Set Problem (FRSP), the Quasi-Liveness Problem (QLP) or the equivalent problem called the Coverability Problem (CP) and the Regularity Problem (RP) [VJ85, KM69]. This motivated another approach called "minimal coverability graph" [Fin93].

The basic idea of a coverability graph is to reduce markings. Therefore, markings are covered, which represent a system state that only differs in the accumulation of tokens. System dynamics and properties remain similar. One example for such a coverability graph is given in Figure 3.11. If transition t_3 is active, this normally results in the marking {3,0,1,0} as presented in the reachability graph in Figure 3.10 (b). A new marking m_j can be covered, when the graph from the root to the actual state contains a marking m_i that has similar properties. Therefore, the tokens in the places of the new marking need to have at least the same number of tokens as in the previous markings and furthermore, one place needs to have more tokens. For example, as the marking {3,0,1,0} contains in place 3, more tokens than the marking {3,0,0,0}, place 3 in the new marking can be covered with ω. This indicates, that the system state does not really change with the new marking. Only the number of tokens in place 3 increase.

The system's invariant properties are an example of static analysis. Therefore, model validation on structural properties can be performed with t-invariants, and respectively, p-invariants as counterparts. These invariants correspond to sub-networks which describe basic system behavior. Periodic system behavior can be determined by a t-invariant. It is calculated which



(a) (b)

Figure 3.10: (a) Discrete Petri net with an initial marking in place 1 [PW08]. (b) Corresponding reachability graph that lists all possible markings of the Petri net presented in (a) [PW08]. Edges show how each marking can be reached. Each edge represents an active transition firing.

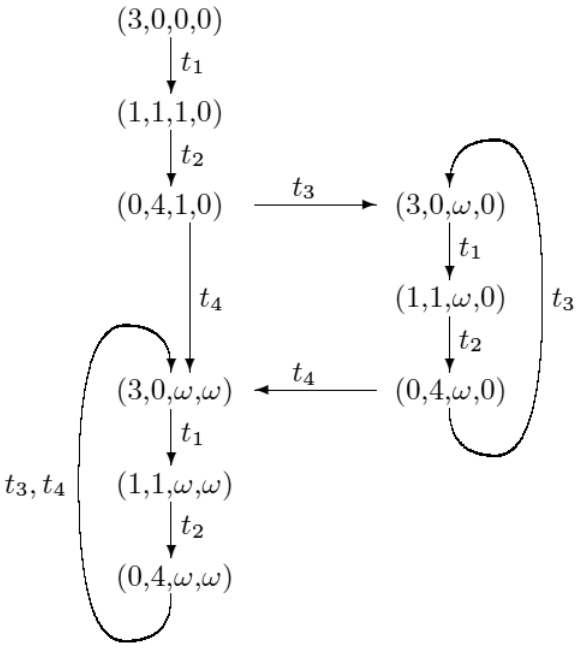


Figure 3.11: A minimal coverability graph for the Petri net presented in Figure 3.10 (a) [PW08].

and how often transitions have to fire until reaching the initial marking, (if possible) without reconstructing or calculating state space.

In summary, the aforementioned analysis approaches can be used for practical problems in several ways. System models can be validated by determining if they are able to reach a certain state that is experimentally derived. If this is not possible, the model has to be revalidated. On the other hand, Petri net models can calculate a set of transition firing steps to reach a certain behavior in real life systems. Thus, starting conditions can be optimized or influenced to obtain effects faster or more efficiently.

3.3 Centrality measurements

Network centralities are a common method to determine important elements within a system. In the social sciences it is a common task to model relationships with graphs and based on that, to identify people that are more influential than others. Similar questions can also be asked of biological networks.

A **centrality** is defined by the function $\mathcal{C} : V \mapsto \mathbb{R}$ on a directed or undirected graph $G = (V, E)$, which assigns a real number to every vertex. If one vertex is more central than another one, then $\mathcal{C}(v_1) > \mathcal{C}(v_2)$ is given [KLP+05]. However, centrality measurements are only comparable inside the same network and some measurements can only be applied on connected networks.

One of the first centrality measurements is the **degree centrality**, defined by:

$$\mathcal{C}_{deg}(v) := |e|e \in E \wedge v \in e| \tag{3.1}$$

This measurement counts the number of edges connected to a vertex. In several studies, this measurement was used to identify essential elements within a biological network. A study on *Saccharomyces cerevisiae* revealed that proteins with a high degree centrality are more essential in comparison to others [JMBO01]. Other studies described similar findings with degree centralities as described by Hahn *et al.* [HK05].

The **average neighbor degree** is defined by [JS11]:

$$k_{i,nn} = \frac{1}{k_i} \sum_{j=1}^{N_v} A_{ij} k_j \tag{3.2}$$

for each vertex n_i over all vertices N. A is the adjacency matrix of the graph G.

Further centrality measurements are stated on network paths. They give information about the importance of certain paths by using information about path length. The first presented measurement is called eccentricity centrality. For every vertex it determines the maximum distance to all other vertices. The vertex with the shortest paths to all other vertices is the vertex with the highest eccentricity value. Formally, the **eccentricity centrality** is defined as [HH95]:

$$\mathcal{C}_{ecc}(v_1) := \frac{1}{\max\{dist(v_1, v_2) : v_2 \in V\}} \tag{3.3}$$

The second important centrality measurement is the **closeness centrality**, which assigns a vertex v a high value if the shortest path distances for all other vertices to v is minimized. Formally, it is defined as [Sab66]:

$$\mathcal{C}_{clo}(v_1) := \frac{1}{\sum_{v_2 \in V} dist(v_1, v_2)} \tag{3.4}$$

The **shortest path betweenness centrality** measures the ability to monitor communication between other vertices. These vertices, which are on the shortest paths between all other vertices, are the most relevant ones. Let $\sigma_{v_1 v_2}$ be the number of shortest paths between v_1 and v_2, whereas more than one shortest path can exist. $\sigma_{v_1 v_2}(w)$ denotes the number of shortest paths, including w as an interior vertex which is neither start nor end vertex of the paths. The communication rate is given by:

$$\delta_{v_1 v_2}(w) := \frac{\sigma_{v_1 v_2}(w)}{\sigma_{v_1 v_2}} \tag{3.5}$$

If no shortest path between v_1 and v_2 exists, then $\delta_{v_1 v_2}(w) := 0$. With these definitions the shortest path betweenness centrality can be defined as [Fre77]:

$$\mathcal{C}_{spb}(w) := \sum_{v_1 \in V \wedge v_1 \neq w} \sum_{v_2 \in V \wedge v_2 \neq w} \delta_{v_1 v_2}(w) \tag{3.6}$$

A further centrality measurement is based on the eigenvector. It is used on strongly connected graphs such as protein-protein interaction networks, to determine essential elements within a network. The **eigenvector centrality** is the eigenvector C_{eiv} of the largest eigenvalue λ_{max} in absolute value of the equation system $\lambda C_{eiv} = AC_{eiv}$, where A is the adjacency matrix of the graph G [Bon72].

In summary, all presented measurements are able to identify important elements within a graph. However, without a clear scientific question the presented approaches can be misleading. Furthermore, scientists need to have in mind that a large set of graphs can share the same graph topological values [Lew12]. In general, the number of possible graphs for a given node size is very large as presented in Table 3.2 [SS67]. Based on the non-isomorphic graphs, it was examined how many graphs share the same graph topology during preliminary work. The Figures 3.12, 3.13, and 3.14 present the distribution of graphs with same topological values.

Nodes	Number of connected isomorphic graphs	Number of connected non-isomorphic graphs
3	8	2
4	64	6
5	1,024	21
6	32,768	112
7	2,097,152	853
8	268,435,456	11,117
9	68,719,476,736	261,080
10	35,184,372,088,832	11,716,571

Table 3.2: For a given network size many different graphs can be reconstructed, where the difference between isomorphic and non-isomorphic graphs is significant.

Inferentially, thousands of different graphs share the same topological values. And having in mind that the discussed and examined graphs in biology have, in most cases, more than 30 nodes, the number of different graphs with the same topological values increases dramatically. Thus, graph theory has to be very carefully considered and only applied when it is linked to a specific scientific question.

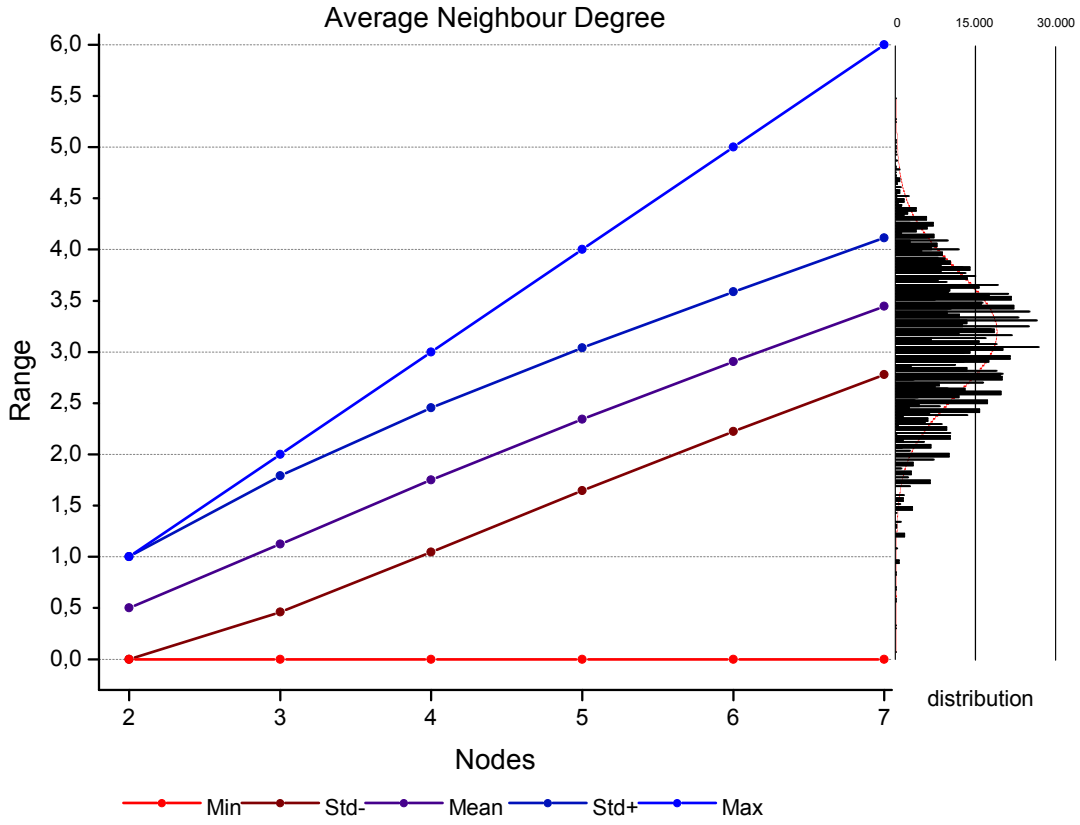


Figure 3.12: The analysis of the distribution of graphs with the same average neighbor degree resembles a Gaussian curve, where thousands of different networks share the same average neighbor degree. The conclusion is that one specific average neighbor degree cannot characterize a unique network type [Lew12].

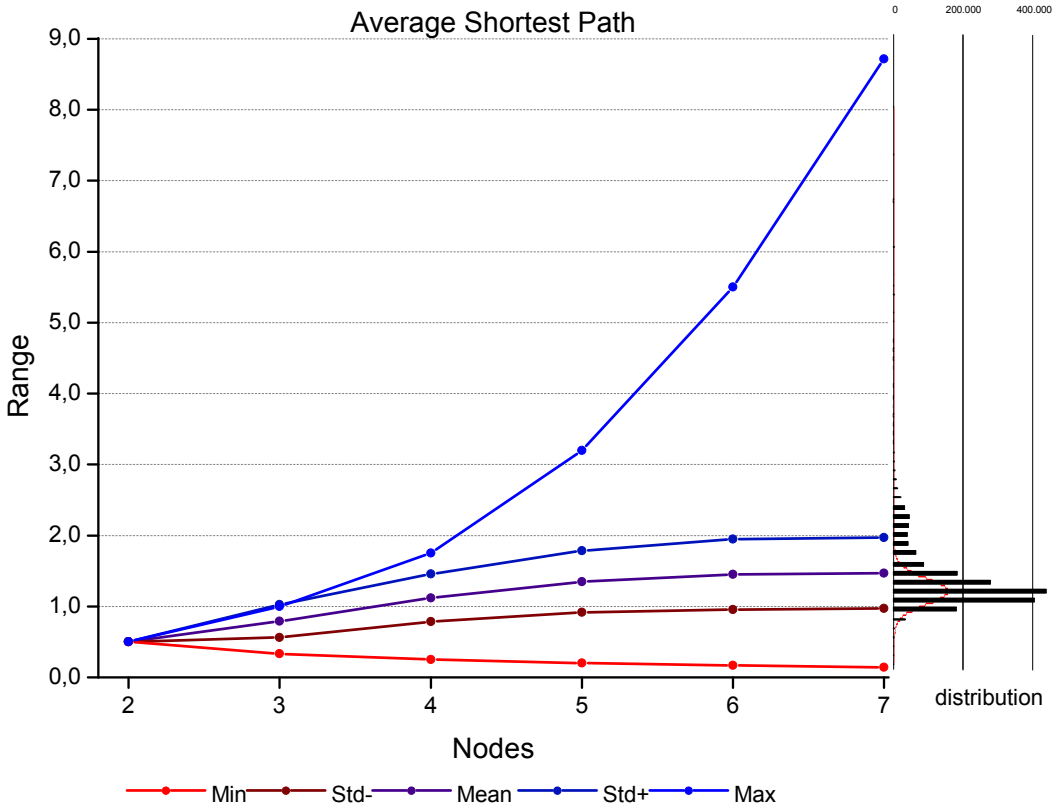


Figure 3.13: The analysis of the distribution of graphs with the same shortest path degree shows that most of the networks have a shortest path degree value between 0.6 and 2. Exceptions with higher values are rare. Thus, it is not possible to draw back any information on network structures based mainly on shortest path degree values between 0.6 and 2 [Lew12].

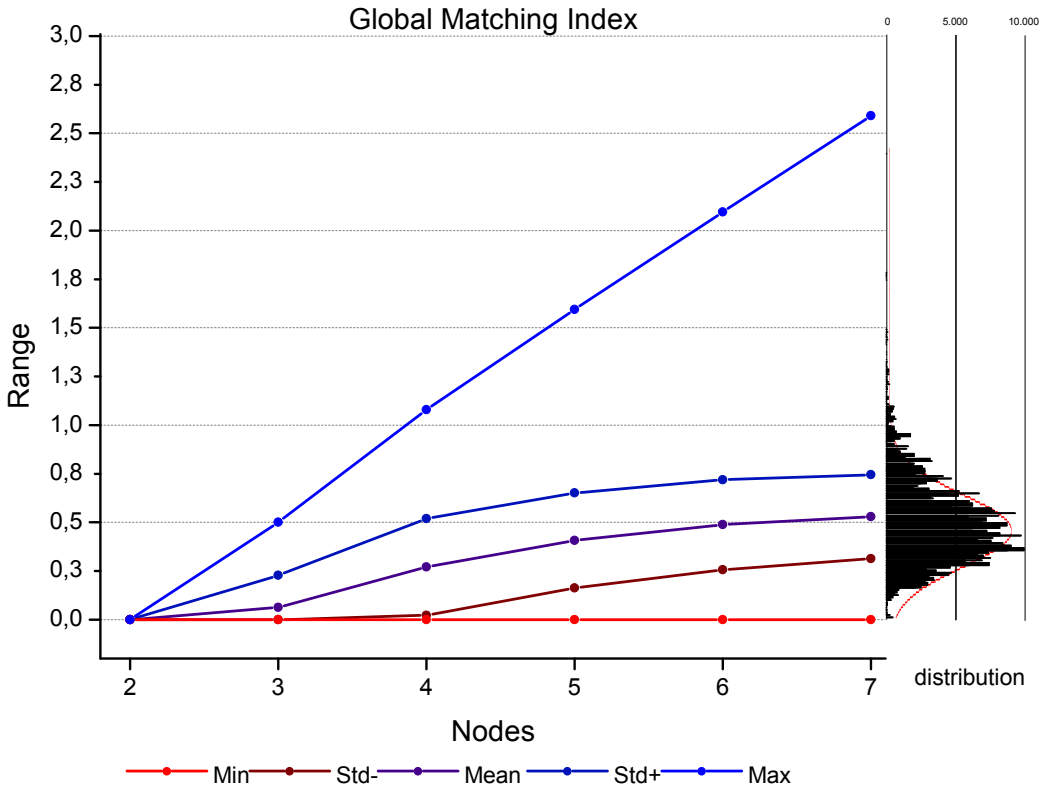


Figure 3.14: The analysis of the distribution of graphs with the same matching index resembles a Gaussian curve, mainly covering the values between 0.1 and 1.1. Most of the networks are located in this area and share the same value [Lew12].

3.4 Biological databases

As described in Section 2.6, more than 1,380 biological databases exist, covering various areas of molecular biology. Due to the large number of databases, it is not possible to mention them one by one. Therefore, only the most important and well-known databases in the field of network modeling which are suitable for this project are described in the following. Categories and subcategories are taken from Nucleic Acid Research (NAR) [GFS12].

- **Kyoto Encyclopedia of Genes and Genomes (KEGG)**
  Category: *Genomics databases (non-vertebrate)*,
  Subcategory: *General genomics databases*
  Category: *Metabolic and signaling pathways*, Subcategory: *Metabolic pathways*
  For the computational analysis of molecular interaction networks with target molecules, metabolizing enzymes, drugs, and chemical structure transformation networks within cells, the KEGG database is of great value [KGS+12]. KEGG serves as an information source on genomes, enzymatic pathways, and biological chemicals to reconstruct biological systems for modeling and browsing biological data. Furthermore, KEGG links its datasets to other important databases such as Universal Protein Resource (UniProt) [WAB+06], Catalog of Human Genetic and Genomic Disorders (OMIM) [ABSH09], and Gene Ontology (GO) [BDD+12]. To continue, the KEGG resource contains data of molecular systems for normal and pertubated molecular states. Especially the pertubated networks are important as they can be used as a reference for disease and drug targeting, for example by integrating experimental datasets. KEGG has been developed over the last 16 years and significantly expanded. The most important part of KEGG are the pathway maps. These maps are manually curated by capturing and organizing experimental information in computable form.

- **Search Tool for the Retrieval of Interacting Genes/Proteins (STRING)**
  Category: *Genomics databases (non-vertebrate)*,
  Subcategory: *General genomics databases*
  The String database [SFK+11] provides uniquely comprehensive coverage and easy access to both experimental as well as predicted interaction information. The interactions include direct (physical) and indirect (functional) associations. Data is derived from genomic context, high-throughput experiments, co-expression, and previous knowledge from sources such as PubMed [WCE+04], a database for scientific publications on life sciences and biomedical topics. STRING quantitatively integrates interaction data from these repositories for a large number of organisms, and transfers information between these organisms where applicable. Interactions in STRING are provided with a confidence score, accessory information such as protein domains, and 3D structures. The database currently covers more than 5 million proteins from more than 1,100 organisms. Furthermore, the

database provides an interactive network viewer, which can be used for the visualization of the protein-protein interactions.

- **BioCarta**
  Category: *Metabolic and signaling pathways*, Subcategory: *Metabolic pathways*
  Category: *Metabolic and signaling pathways*, Subcategory: *Protein-protein interactions*
  BioCarta is a dynamic community-fed forum for information exchange and collaboration between researchers, educators, and students, integrating proteomic information from the scientific community (http://www.biocarta.com/). It contains classical pathways as well as current suggestions for new pathways of metabolic and signaling pathways. The forum is focused on interaction maps in humans, both in the healthy and pertubated state. Actually, more than 350 pathways are available to the community. The maps depict molecular relationships from areas of active research such as genomics and proteomics. With dynamic graphical models users are able to observe and analyze how genes and proteins interact in a complete map. It also catalogs and summarizes important resources providing information for over 120,000 genes from multiple species.

- **Pathway Interaction Database (PID)**
  Category: *Metabolic and signaling pathways*, Subcategory: *Signaling pathways*
  The Pathway Interaction Database [SAK+09] is a curated collection of information about known biomolecular interactions and key cellular processes assembled into signaling pathways. It contains 136 human pathways with 9,215 interactions, curated by NCI-Nature and 322 human pathways with 7,575 interactions imported from BioCarta and Reactome. However, it does not include interaction data deriving from high-throughput protein-protein interaction experiments. The database recognizes several kinds of events such as transcription, translation, translocation, reactions, protein-protein interactions, modification, and black-box processes whose internal composition is not provided. Primarily, the database serves as a research tool for the cancer research community and other researchers interested in cellular pathways such as neuroscientists, developmental biologists and immunologists.

- **Human Reference Protein Database (HPRD)**
  Category: *Human and other vertebrate genomes*,
  Subcategory: *Human Open Reading Frames (ORFs)*
  HPRD [KPGK+09] is a database of curated proteomic information pertaining to human proteins and a very important knowledge base for genomic and proteomic research. The provided data is experimentally derived, based on mass spectrometry, protein-microarray, protein-protein interaction, Post-Translational Modifications (PTMs), and tissue expression. Overall, HPRD lists 30,047 protein entries and 39,194 protein-protein interactions. Furthermore, 22,490 subcellular localization and 470 domains are documented. Information about protein expression (112,158 entries) and PTMs are also given (93,710 entries).

- **Interaction Database (IntAct)**
  Category: *Metabolic and signaling pathways*, Subcategory: *Protein-protein interactions*
  A further resource for protein-protein interaction data is the IntAct database [KAB+12]. IntAct provides data either curated from literature or from raw data deposits. Primarily, it consists of protein-protein interaction data, where each entry is reviewed by a senior curator. It also captures the protein-small molecule, protein-nucleic acid, and protein-gene loci interactions. It contains approximately 293,000 binary interactions from more than 5,009 scientific publications and 15,000 experiments, referencing 62,000 proteins, 144 small molecules, and 233 genes.

- **Molecular Interaction Database (MINT)**
  Category: *Metabolic and signaling pathways*, Subcategory: *Protein-protein interactions*
  The Molecular Interaction Database [LBP+12] contains approximately 235,000 interactions from over 4,800 publications. MINT is not specialized in a specific organism. Moreover, it contains interactions from more than 30 different species. Nevertheless, it provides 28,283 interactions for *Homo sapiens*, 4,808 interactions for *Mus musculus*, and 2,804 entries for *Rattus norvegicus*, which are all of great value. Furthermore, data from other species can be used for cross-species-analysis.

- **ENZYME**
  Category: *Metabolic and signaling pathways*,
  Subcategory: *Enzymes and enzyme nomenclature*
  The ENZYME [Bai00] database is a repository of information relative to the nomenclature of enzymes. It is primarily based on the recommendations of the International Union of Biochemistry and Molecular Biology (IUBMB). The database describes each type of characterized enzyme for which an Enzyme Commission (EC) number has been provided. The ENZYME database is an indispensable resource for the development of enzyme databases and metabolic pathways. Furthermore, it is helpful in the development of computer programs for the reconstruction and analysis of biological pathways. It contains more than 4,851 active entries on enzyme information.

- **Braunschweig Enzyme Database (BRENDA)**
  Category: *Metabolic and signaling pathways*,
  Subcategory: *Enzymes and enzyme nomenclature*
  BRENDA [SGC+11] is the main repository for manually annotated enzyme functional and property data for the scientific community. The database is characterized by high scientific knowledge that is mainly manually extracted from primary literature. It covers information on function, structure, occurrence, preparation, and application of all enzyme classes that have been classified by the International Union of Biochemistry and Molecular Biology (IUBMB), as well as properties of mutants and engineered variants.

In summary, for each -omic level related to this research appropriate databases exist. Scientists are not limited to a particular database, moreover, they can choose between different data sources and information. However, there are no general reasons why one database is better than the other, since all databases contain important and relevant information. For example, BRENDA and ENZYME have a similar focus, curation model and reputation. The choice of a particular database always depends on the project requirement.

However, this section has shown how the mentioned databases can contribute to the overall research question of this project and what kind of information can be retrieved. It becomes obvious that only one single database is not able to explain all biological processes in an entire biological system. Thus, several databases have to be linked to each other in order to support scientists in modeling and explaining biological phenomena and systems.

3.5 Data integration approaches

There are several data warehouses that contain important life science databases and information. Some of the most popular data warehouses in the field of bioinformatics are the BioWarehouse, ONDEX, BioDWH, and DAWIS-M.D. In the following, a brief description of the aforementioned systems is given:

- **BioWarehouse**
  The BioWarehouse [LPW+06] is an open-source software environment for integrating a set of biological databases into a single repository for data management, mining, and exploration. It supports loader programs that translate the flat file representation of a source database into the warehouse schema. Each loader is implemented for a particular data source and applies a degree of semantic normalization to the respective source data, decreasing semantic heterogeneity. Following databases and formats can be loaded: BioCyc [KOMK+05], BioPax [DCP+10], ChIP-Chip data (meta data, gene expression data, transcription factors, antibodies), CMR [PUD+01], eco2dbase [VSC+92], Enzyme [Bai00], GenBank [BKMC+12], Gene Ontology [BDD+12], KEGG [KGS+12], MAGE-ML [SMS+02], MetaCyc [CAD+12], NCBI Taxonomy [Kar00], and the UniProt resource [AJMO+12], containing SwissProt and TrEMBL. Users are able to install the software on the local computer or use the publicly available version of BioWarehouse called PublicHouse[3].

- **ONDEX**
  The ONDEX framework [KBT+06] is a freely available software system that combines semantic database integration and text mining with methods for graph-based analysis.

[3] http://biowarehouse.ai.sri.com/PublicHouseOverview.html

Based on an integrated ontology in the ONDEX back-end and front-end, all biological databases and their content can be represented as graphs, where nodes and edges have a distinct set of concepts and relations. A concept can be a gene, an enzyme, a transcription factor, a pathway or any other biological element that is linked by a relation to another element. Both concepts and relations have properties and optional characteristics that are derived from database content. Following parsers for databases and tools are provided: AraCyc [MZR03], AtRegNet [PJS+06], BioCyc [KOMK+05], BioGRID [BSR+08], BRENDA [SGC+11], Cytoscape [SOR+11], EcoCyc [KBMCV+09], GOA [DHAF+12], Gramene [LJH+08], Grassius [YNF+09], KEGG [KGS+12], Medline [WCE+04], [CAD+12], O-GlycBase [GBR+99], OMIM [ABSH09], PDB [WIN+05], Pfam [PCE+12], SGD [CHA+12], TAIR [RBB+03], TIGR [QLH+00], Transfac [MKMF+06], Transpath [KPV+06], UniProt [WAB+06], and WordNet[Mil98]. Furthermore, each concept and relation can be enriched with literature information from PubMed [WCE+04], derived by text mining algorithms. To make the data and graphs accessible to users, visualization and analysis algorithms are provided. In addition, ONDEX provides microarray and statistical analysis.

- **BioDWH**
  The BioDWH [TKKH08, KHH11] is a powerful data integration framework which loads some of the most important biological databases into one data warehouse. External databases can be integrated into the system by using specific data parsers. With an object-relational mapping (ORM), database entities can be represented as objects in a relational database system, which makes it easy to access their properties and relationships. Actually, parsers for following databases are provided: BRENDA [SGC+11], EMBLBank [CAB+09], ENZYME [Bai00], EPD [SPPB06], Gene Ontology [BDD+12], HPRD [KPGK+09], IntAct [KAB+12], iProClass [HBCW03], JASPAR [PCTK+10], KEGG [KGS+12], MINT [LBP+12], OMIM [ABSH09], Reactome [MGG+09], SCOP [AHC+08], Transfac [MKMF+06], Transpath [KPV+06] and UniProt [WAB+06].

- **DAWIS-M.D.**
  The data warehouse information system DAWIS-M.D. (Data Warehouse Information System for Metabolic Data) was created based on the powerful data integration framework BioWDH. DAWIS-M.D. is a platform-independent web application that provides an integrated view of comprehensive biomedical knowledge from integrated data sources [HKJ+11]. Furthermore, DAWIS-M.D. is able to divide the provided database content into meaningful domains such as compound, disease, drug, enzyme, gene, gene ontology, genome, glycan, pathway, protein, reaction, reactant pair, and transcription factor, and moreover, to identify the relationship between the domains. This makes it easy for scientists to find information of interest and to understand complex biological mechanisms and interactions.

In summary, several well-suited approaches exist, which VANESA can use. Table 3.3 summarizes all important features and serves as a base for discussion. An important aspect of data warehouses are the technological features such as user model, level transparency, and update mechanisms, among others. Data updates within the BioDWH integration tool are realized automatically, whereas the other mentioned solutions have to be updated manually. This is a main advantage in terms of data relevance. Furthermore, no critical expertise is necessary and all databases can be queried. The BioWarehouse enables users to develop a user-specific data warehouse. Ondex's objective is the integration of (un)structured sources.

3.6 Standard exchange formats

This section gives insight into currently available and popular standards such as the Systems Biology Ontology (SBO) and Extensible Markup Language (XML) -based [BPSM+08] standards for the exchange of pathway data within systems biology. Here, only XML-based formats are considered, since it is used as universal language in data exchange. McEntire *et al.* [MKA+00] and Achard *et al.* [AVB01] have shown in their studies that this language is very flexible and simple to use and therefore, a powerful standard in bioinformatics and systems biology in comparison to Comma Separated Values (CSV), Excel, and other file formats. More than 85 standards can be found within systems biology [SHL07]. In the following, some of the most important XML-based standards are described, with information on concept and addressee.

- **Systems Biology Ontology (SBO)**
  The SBO ontology [CJK+11] is a well-defined logic about biological terms, including single identifiers for each distinct entity, allowing clear reference and identification. Furthermore, it is augmented with terminological knowledge such as synonyms, abbreviations and acronyms. The terminology is also used to specify the type of the components being represented in a model and their role in systems biology descriptions. Thus, the ontology allows unambiguous and explicit understanding of the meaning of the involved components in a system and moreover, enables mapping between elements of different models encoded in this format.

  The ontology is a well-defined logic about biological terms, including a single identifier for each distinct entity, allowing clear reference and identification. It is composed of seven vocabulary branches: systems description parameter, participant role, modeling framework, mathematical expression, occurring entity representation, physical entity representation, and metadata representation. The terminology is also used to specify the type of components represented in a model and their role in systems biology descriptions. Thus, the ontology allows unambiguous and explicit understanding of the meaning of the involved components in a system and moreover, enables mapping between elements of different models encoded in this format.

	BioWarehouse	ONDEX	BioDWH	DAWIS-M.D.
Level of transparency	Sources specified by user	Sources selected by system	Sources specified by user	Sources hard-wired by system
Data model	Structured relational	Structured relational	Structured relational	Structured relational
User model	Expertise in query language	No critical expertise	No critical expertise	No critical expertise
Aim of integration	Query oriented	Portal, browsing-based	Query oriented	Portal, browsing-based
Objective	Development of user-specific data warehouses	Integration of (un)structured sources	Development of user-specific data warehouses	Comprehensive view of biomedical knowledge from different data sources
Graphical user interface	Command line	Java-based application	Java-based application	Web application
Updates	Manually	Manually	Automatically	Manually
Supporting database technologies	MySQL, Oracle	PostgreSQL	ORM on MySQL	ORM on MySQL

Table 3.3: Comparison of the data integration systems and data warehouses BioWarehouse, ONDEX, BioDWH, and DAWIS-M.D. in reference to their technical features.

- **Biological Pathways Exchange (BioPAX)**
  BioPAX is a standard language to represent biological pathways at the molecular and cellular level [DCP+10]. The main goal of BioPAX is the exchange of information between several pathway databases such as Reactome [MGG+09] and BioCyc [KOMK+05]. It was introduced through a community process to make complete representation of basic cellular processes substantially easier to collect, to index, to interpret, and to share. BioPAX covers concepts such as metabolic and signaling pathways, gene regulatory networks, and genetic and molecular interactions. Therefore, it has a structure for substances, interactions, pathways, and links to organisms and experiments. The language is distributed as an ontology definition with associated documentation and a validator for checking. Therefore, the BioPAX community cooperates with the SBML and CellML mathematical modeling language communities. For better accessing and manipulating data in the BioPax format, a house-implemented java library called "Paxtool" is available. BioPax Level 3 is currently available at http://www.biopax.org.

- **BioXSD**
  BioXSD is common exchange format for basic bioinformatics data [KPJ+10]. Using this format it should be possible to establish a common web service for the exchange of data for bioinformaticians in the World Wide Web. This format should fill gaps between specialized XML formats such as Systems Biology Markup Language (SBML) [FH03, HFS+03], Microarray Gene Expression Markup Language (MAGE-ML) [SMS+02], Genomic Contextual Data Markup Language (GCDML) [KGM+08], Protein Data Bank Markup Language (PDBML) [WIN+05], Molecular Interaction Format (MIF) [HMPB+04], and Phylogenetic Markup Language (PhyloXML) [HZ09]. Therefore, BioXSD defines data formats such as, biological sequences, sequence alignments, sequence annotation and references to data, resources, and vocabularies in a variety of possibilities. BioXSD serves as a canonical data model and is available at http://bioxsd.org as Version 1.1.

- **Cell Markup Language (CellML)**
  CellML [CLN+03, MMR+10] is a language for representing mathematical models. Using Differential Algebraic Equations (DAEs) any cellular model can be represented in CellML. In addition, CellML represents entities using a component-based approach, where relationships between components are represented by connections. The developers have implemented an Application Programming Interface (API) for working with CellML models and files. Thus, software developers do not need to reinvent the same functionality each time they develop a new tool. The API enables users to retrieve information, to manipulate, and to extend a model. The API interfaces are designed to be independent in any programming language, platform, or vendor, and are expressed in the Interface Definition Language (IDL). At the present time, CellML is available at http://www.cellml.org in Version 1.1.

- **Mathematical Markup Language (MathML)**
  MathML is a low-level specification for describing mathematics [San03, ABC+10]. It is used wherever mathematics needs to be handled by software, such as mathematical expressions in web pages and workflows in science and technology. Actually, MathML is available at http://www.w3.org/Math/ as Version 3.

- **Protein Data Bank Markup Language (PDBML)**
  The PDB is the single worldwide repository for macromolecular structure data [BWF+00]. For more than 30 years, the data resources has used a column-oriented format to store and share archival entries [WIN+05]. Facing more and more complex data for macromolecular structures, the used data format constrained several limitations such as internal structure and the organization of records. Therefore, a new XML based data format, called PDBML has been introduced [WIN+05]. It builds the content of the PDB exchange dictionary and can be used as a specific exchange medium for detailed molecular protein structures, such as data derived from experimental crystallography. PDBML is currently available at http://pdbml.pdb.org as version 3.3 to all users.

- **Systems Biology Markup Language (SBML)**
  SBML is an exchange format for representing biochemical reaction networks [FH03, HFS+03]. Using SBML, users are able to describe models in many areas of computational biology, including cell signaling pathways, metabolic pathways, gene regulation, and others. Therefore, SBML has the structure, ontology, and links, for pathways and interactions. To enable mathematical descriptions, the SBML Level 2 uses MathML for more complex mathematical formulas. This extends the features of SBML and also results in a greater compatibility with CellML. Furthermore, it provides the possibility to specify delay functions and define discrete events that can occur at specified transitions in a certain state in biological models. In order to help users to read, write, manipulate, translate, and validate SBML files and data streams, the LibSBML API is available in different common programming languages, such as JAVA, C, C++, and others. Presently, SBML Level 2 is available at http://sbml.org/Software/libSBML and SBML Level 3 is being developed.

In summary, it is recommended that any kind of biological modeling software should use standards. One of the main standards for the modeling of biological systems is the Systems Biology Ontology. Using this standard ensures the usability, reusability, and interoperability of biological models. Furthermore, data exchange standards can easily access models encoded in this format. For instance, SBML, MathML, and CellML support SBO definitions, which makes it easy to translate any kind of SBO model into such an exchange format. However, there is a significant difference in the scope of the mentioned standard exchange formats. By studying the most important formats and considering recommendations from literature [SHL07, SB08], SBML and CellML are proposed as a means for the exchange of biochemical reaction networks and models between different software tools. They provide an ontology and structure

that can even be used for simulations. They also provide constructs that are similar to the object models used in packages specialized for simulating and analyzing biochemical networks. CellML and SBML, embedding MathML, provide users with the possibility for the representation of whole models in differential algebraic expressions. Besides, SBML and CellML have an API, which allows reading, writing and manipulating models in an easy manner. Furthermore, SBML and CellML have much in common, since the development of both standards takes place cooperatively. Formats such as PDBML only focus on particular substances. Thus, they are not appropriate for network models. This also applies to MathML, which only provides basic mathematics. Furthermore, BioXSD and BioPax exist and can be used as data standards. However, BioXSD is focused on data that is not supported by the main formats and thus, very specialized and not capable of representing entire biological systems. BioPax is only focused on pathway maps, which can be shared between databases and tools. SBML and CellML can support dynamic systems in ways not possible for BioPax.

3.7 Discussion

Hundreds of modeling and simulation tools exist, as described in Section 3.1, but none are able to reconstruct a biological model with database information, which can be visualized, analyzed, and automatically simulated in a qualitative and quantitative manner. However, Chapter 1 introduced the work of natural scientists and showed that they need a software application which is able to reconstruct models based on selected biological databases. Furthermore, they need a possibility for automatically simulating the models with an intuitive formalism that works both with and without biological data.

With references to the here presented approaches, if a contiguous simulation based on database reconstructed networks is possible, the simulation processing is performed in external tools or using a third-party ODE solver. To perform a successful simulation, prior knowledge in mathematics and a complete set of biological data is necessary. As an alternative, Petri net simulation tools can be used. Although different Petri net simulation tools exist, they have certain disadvantages. The discussion in Section 3.1 revealed that the presented tools have some inconsistencies in their execution semantics and/or are not comprehensible in simulation results. In conclusion, users do not have a tool for biological modeling which enables them to start modeling a biological system with advanced modeling techniques which can be automatically extended or enriched with biological knowledge from important databases, and then analyzed and simulated. This has strongly motivated the realization of VANESA.

In Chapter 2 it was discussed on which basic bioinformatics approaches VANESA should be based. This chapter presented further approaches, which can be used in VANESA to make it even more powerful. Therefore, Petri net analysis techniques and graph theoretical approaches have to also be considered in the realization of VANESA. With reachability graphs or covering

graphs, dynamic analysis could be performed to determine different system states. Invariant properties could give clues for static analysis. In addition, centrality measurement would highlight important elements within a biological network and point out structures relevant for analysis.

In terms of an automatic network reconstruction, there are several well-suited databases, which can be used as valuable data repositories. Furthermore, these databases are accessible in well-curated data warehouses, which facilitate the access. The last section focused on data exchange formats and showed that several possibilities exist to exchange models between different software applications. In the following, the next chapter discusses which and how these approaches should be realized in VANESA.

Chapter 4

Design and system architecture

The main goal of this work is the realization of a framework which can be used for modeling, visualization, analysis, and simulation of biological networks in the natural sciences. Therefore, important aspects about VANESA were discussed in the Sections 2.8 and 3.7. The present chapter focuses on concrete requirements and design principles for the system architecture of VANESA.

The first section begins with the backgrounds of VANESA, as VANESA reuses certain well-established programming modules from the modeling software Network editor, which was realized in preliminary work. Furthermore, the established requirements on VANESA are presented in the following, which define the functionalities that should be offered. Therefore, the following aspects are discussed: Network reconstruction, database access, Petri net simulation, Petri net analysis, network analysis based on graph theory, system standards and data exchange possibilities, user interface, and network interaction design. Section 4.2 presents the resulting system architecture. It is demonstrated how all bioinformatics approaches are realized in VANESA and how they are interconnected with each other. The last section, 4.3, summarizes the presented results and discusses the advantages of the design and architecture.

4.1 Design requirements

Section 3.1 presented several software applications which are able to model, analyze, and simulate biological networks in one framework and workflow. However, as reviewed in Section 3.7, most of the tools are failing in their initial goal or are not appropriate for the research presented here. Therefore, the new software environment VANESA should be realized, with which users can model, visualize, analyze, and simulate biological processes and systems. As a base for VANESA the software application Network Editor was used (see Figure 4.1), which was realized in preliminary in-house work [Jan08]. The initial goal of this application was to test

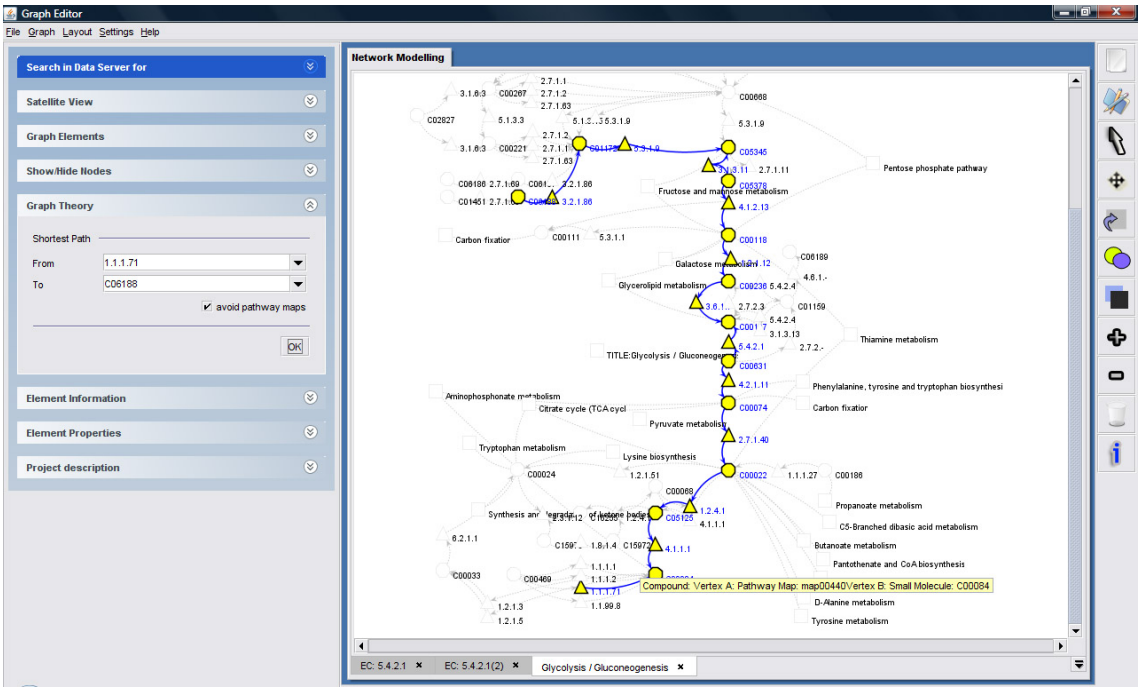


Figure 4.1: Screenshot of the analysis of the Glycolysis KEGG pathway within the software application Network Editor. The focus of the analysis is the shortest path between the alcohol dehydrogenase (EC: 1.1.1.71) and the phospho-beta-glucosidase (EC: 3.2.1.86)

a network based approach to model biological systems. As the resulting application was very promising and helpful in molecular research, it was decided to reuse aspects of this software application. Thus, instead of reinventing the wheel, VANESA could make use of some of the programming modules of the Network Editor.

As the name already suggests, the Network Editor was primarily designed to create and edit networks. It provided a well-designed user interface, in which the intuitive interaction with graphs is possible. This concept of the visual interaction should be also incorporated into VANESA. Furthermore, the Network Editor is able to reconstruct simple networks with data from the databases KEGG and BRENDA. Although the data structure of the databases changed and the Network Editor is no longer able to access up-to-date data, this basic principle should be also integrated into VANESA. Furthermore, the Network Editor provided two basic analysis methods. The first method focused on the computation of the shortest path from one node to another and the second function enabled the highlighting of equal nodes in different models. Although, this comparison was very simple, as it only highlighted elements with the same biological name, users liked it. Thus, it was decided to reuse these functions and implement further graph theory approaches in VANESA. Nevertheless, in order to realize a software application that can meet the initial goal of this work and is able to compete with other state-of-the-art tools, VANESA needs much more useful approaches. Therefore, the following section presents all important stated requirements placed on VANESA in order to become a meaningful and

useful tool.

4.1.1 Network reconstruction

Database information can be used in various of ways, for example to get information about a certain biological element or to unravel metabolic regulation, as presented in Section 2.6. Furthermore, biological databases can be used as useful data sources for the reconstruction and analysis of biological networks as described in Section 2.7. Therefore, VANESA needs access to external databases and other data resources that allow users to interpret and analyze their own data in the context of existing knowledge. Hence, following requirements on the database choice are established:

1. All databases should be free-of-charge and accessible by using a Simple Object Access Protocol (SOAP) or an Application Programming Interface (API).
2. All databases should use the same terms, identifiers, and publication structures as cited in literature.
3. Provided datasets must be up to date and should not overlap.
4. The selected databases should be well-curated.
5. Only databases which can be used for the reconstruction of biological networks should be integrated.
6. The integrated databases should be focusing on metabolic pathways, signaling pathways, and protein-protein interaction networks.
7. At least one of the selected databases should focus on genomic information to gather detailed insights into regulatory processes.
8. A further database should provide information on protein-protein interactions and complexes.
9. Furthermore, one database should provide detailed information on proteins and open reading frames (ORFs).

Section 3.4 presented several different databases and sources which fulfill these requirements. However, to limit selection, only those databases and sources which have a very good curation model, a good reputation among scientists, and from personal experience, are best processed in terms of data structure and provided ontology are considered for integration. This is why the databases KEGG, BRENDA, IntAct, MINT, and HPRD were selected in the first place. These databases provide sophisticated life sciences information collected from scientific experiments, published literature, high-throughput experiment technologies, and experimental analyses. However, for projects specific to other research applications or -omic levels it should be

still possible to integrate additional databases into VANESA. This should be possible through an easy-to-use programming interface.

After a detailed peer review on existing databases, it was necessary to identify a data warehouse containing the aforementioned selected biological databases with the following steps, instead of querying each database one by one. Section 3.5 presented several sophisticated approaches, whereas the BioDWH / DAWIS-M.D. data warehouses meet all requirements for the presently discussed application. Both systems do not need critical experience and distinguish themselves with a comprehensive view of biological knowledge from different data sources. Furthermore, several other databases are provided that can also be accessed. Thus, further additional requirements are established in VANESA:

10. Users should be able to reconstruct biological networks by using information from the databases KEGG, BRENDA, IntAct, MINT, and HPRD.

11. Computer scientists should be able to integrate new databases with an easy-to-use interface.

12. It should be possible to query each integrated database separately or in combination with each other.

13. Users should be able to use Boolean operators, such as AND, NOT, and OR to combine different search terms with each other.

14. Each database should have a graphical user form in which users can type in their search terms.

4.1.2 Simulation

As discussed in Section 2.8, using Petri nets is one of the most powerful techniques to model and simulate cell behavior. With Petri nets, users are able to qualitatively as well as quantitatively reconstruct models. Furthermore, ODEs can be incorporated into hybrid Petri nets when kinetics are available. This approach is well-suited for users who are not familiar with modeling techniques. No mathematical knowledge is needed prior modeling and a Petri net model can be continuously extended with new knowledge and data, without changing the initial structures and parameters. Thus, a model can grow over years and become more complete and meaningful. Furthermore, sophisticated analysis techniques are available which can be applied to calculate possible system states, among others. Section 3.2 described some of the possible approaches which have already helped in answering essential questions in biology and other research fields. In order to provide users with all these possibilities, VANESA should also be able to simulate biological models in the Petri net language. Therefore, following requirements are described:

1. Qualitative, stochastic, continuous, hybrid, and functional Petri net modeling possibilities have to be provided.
2. New graph classes for the xHPNbio library have to be provided.
3. It should be possible to integrate external libraries such as the PNlib, for simulation processing.
4. There should be a possibility for specifying systems of ordinary differential equations (ODEs) for continuous Petri Nets within the graphical user interface (GUI).
5. Manually created networks and database-reconstructed networks should be easily transformed into the Petri nets language.
6. Simulation results should be animated within the GUI, whereby simulation processing should be performed in the background.
7. Simulation results should be available as tables and also visualized in diagrams, showing the evolution of the token numbers on selected places or the firing times of selected transitions over time.
8. Animations of the simulation results should be triggered manually or be performed in the automatic mode.
9. In terms of analytic power, the following Petri net concepts for model validation and analysis should be provided:
    - It should be possible to check the liveness of a Petri net.
    - Reversibility of a Petri net should also be examined.
    - Users should be able to determine a model for p-invariants and t-invariants.
    - Covering graphs should give hints about possible system states.

4.1.3 Network analysis

Graph theory, especially centrality measurement, is a powerful mathematical approach to analyze organization and information flow within a biological network. Based on graph geometry and topology, different scientific questions can be discussed as presented in Section 2.3 and 3.3. Providing scientists with the possibility to apply graph theory as network analysis on biological networks, they can discuss many important questions, such as “Which element is the most important one?”, “How dense is the biological network?”, “How many interactions need to get passed in order to get from element A to element B?”, “Which motifs occur most in the network?”, and “How does the reconstructed real-world model differ from randomly generated

networks?". Furthermore, in the view of network complexity and size, this kind of network analysis can be very useful. Because in our day and age biological networks can become very large, a sophisticated method is necessary to reduce a model to its most significant elements. Graph theory is one possible way for that. Depending on the scientific question, the analysis can help in identifying those elements which are necessary for a better understanding of the research question. Based on these results, the model can be better analyzed and reduced in complexity. Hence, graph theory as a network analysis approach should also be an important part of VANESA. Therefore, the following requirements are indicated:

1. An adjacency matrix for each biological network should be provided on which the calculation is performed. This ensures fast and efficient processing.

2. Users should have the possibility to calculate the following graph theoretical approaches:

   - Minimum degree, maximum degree, average degree, average neighbor degree, and the distribution of different vertex degrees within a network.

   - Minimum, maximum, and average shortest paths within a network.

   - Graph density, matching index, and clustering coefficient.

3. Furthermore, users should be able to compare biological networks with each other to identify similarities and differences.

4. For theoretical biology it should be possible to reconstruct networks based on random graphs, such as Eulerian and Hamilton graphs.

5. A comparison between reconstructed biological networks and random artificial networks should also be possible, in order to identify structures and motifs that only appear in real-world systems.

6. Results should be visually accessible in the visualization pane, and if possible, animated in an interactive way.

4.1.4 System standards and data exchange possibilities

In order to assign meaning to model constituents, a controlled vocabulary of relationships should be assigned to VANESA. Therefore, semantic standards should provide a unified and common definition for all words, phrases, and vocabulary used to describe a particular data type or subject area. In terms of biological network modeling and analysis, models should be based on the Systems Biology Ontology (SBO). In terms of data exchange, the Systems Biology Markup Language (SBML) and Cell Markup Language (CellML) are the most important formats within systems biology and bioinformatics, as described in Section 3.6. SBML is capable of describing

the significant elements of biochemical reaction networks and besides, it supports an API, which allows reading, writing, and manipulating models in an easy way.

CellML is focused on describing whole biological models with differential algebraic equations (DAEs). Since VANESA is not meant to reconstruct models only based on DAEs, this exchange format is not considered. However, to give users the possibility to perform basic mathematics on the exported networks, MathML should be provided. Using this format, users are able to analyze reconstructed systems in research areas, such as science, business and economics. In order to allow simulations within other software applications, such as CellIllustrator and Dymola, additional export formats have to be considered. Therefore, the export formats CellIllustrator Markup Language (CSML) and Modelica Exchange Format (.mo) should be considered. For software applications that need a general biological network representation only based on nodes and edges, a simple .txt format has to be made available. In addition, VANESA should have its own Extensible Markup Language (XML) based format for the digital storage and exchange of models and results, called VANESA Markup Language (VAML). The realization of this format is motivated by the fact that other export files are not able to capture all necessary network properties for reconstructed models and Petri nets. Furthermore, a simple .txt data file import for experimental data should be provided, which enables users to map experimental data on an existing model. In summary, this results in the following requirements for system standards and data exchange possibilities in VANESA:

1. Models should be based on the Systems Biology Ontology (SBO).

2. SBML as standard exchange format for the import and export of models should be supported.

3. Users should have the possibility to export networks in the MathML language.

4. In order to perform model simulations in external applications, such as CellIllustrator and Dymola, the export formats CellIllustrator Markup Language (CSML) and Modelica Exchange Format (.mo) should be available.

5. A simple .txt format, including basic network structures, should also be available to export networks to other software applications, such as the mathematical application MATLAB[1].

6. A simple .txt import format should be provided to map experimental data on exiting networks.

7. The VANESA Markup Language (VAML) format has to be defined and realized in order to capture all network properties and dynamics, including specific project settings and parameters.

[1]MATLAB is a high-level language and interactive environment for numerical computation, visualization, and programming. It is commercially available at http://www.mathworks.de/products/matlab/.

4.1.5 User interface and interaction design

For the network modeling, analysis, visualization, and simulation, users need to have cognitive support in providing both detailed information of the currently most relevant objects, as well as giving the user an idea of its entire context. Therefore, VANESA needs to be truly interactive to give insight and organize results and ideas. It should be easy to use and helpful to users. Within a short time, users should be able to learn how to work with the software application and be able to accomplish a certain task. Everything should be immediately accessible without taking long or roundabout routes. Thus, the following requirements for user interface and interaction with VANESA are established:

User interface:

1. A biologically sophisticated graphical user interface (GUI) should be available to users, with which scientists should be able to intuitively model and simulate complex dynamic interactions and processes in one active window.

2. The GUI should adapt automatically to the selected network class, whether it is a biological network or a Petri net.

3. The GUI has to be truly interactive, making users aware of further possibilities and restraints.

4. A comprehensive graphical network representation of biological research data is needed.

5. Information has to be visualized in a clear and understandable manner to meet the purposes of underlying research activities, that is, to quickly understand the information and to show the matching objects in response to a query.

6. Everything should be reachable within a maximum of three mouse-clicks.

7. Users should be supported by navigational guidance.

8. In order to have a clearly defined and recognizable structure of the GUI, VANESA should be divided into three panels:

    - The main panel in the middle of the software application, where models are visualized and accessible to users.

    - The toolbar on the right-hand side of the software application, where regularly used approaches are available to users.

    - An option pane on left-hand side of the software application, where each bioinformatics approach should have its own graphical panel enabling user to perform an analysis.

Interaction design:

9. Scientists should have the possibility to visually draw and edit any kind of biological model or Petri net.
10. It should be possible to create a model with simple "drag and drop" functions or the provided network reconstruction functions.
11. During the modeling process, users should be supported in naming network elements based on biological standards.
12. It should be possible to interactively select, transform, and analyze network elements.
13. Users should have the possibility to edit, compare, manipulate, transform, and zoom into parts of biological networks.
14. A logical or semantic zoom for network modeling has to be supported.
15. Filtering should enable users to reduce network complexity.
16. Interactions with the models should allow the placement of limits.
17. Significant objects should be highlighted.
18. Graph layouts should visualize networks in an appropriate way.

4.2 System architecture

The requirements used for VANESA are vast and complex but finally, they present a well-established guide for a powerful framework. Based on the aforementioned guidelines, a system architecture for VANESA which is able to offer all required features (see Figure 4.2), was elaborated. Each of the bioinformatics approaches is an individual module in VANESA, which gets a certain input and produces a specific output. In order to reach the main goals of VANESA, namely the reconstruction, analysis, visualization, and simulation of biological networks, all modules are interconnected. The implementation of these modules and the overall framework is realized in the programming language JAVA, as this language is platform independent, well-known, and easily understandable by any computer scientist. In the following, each of the modules presented in Figure 4.2 is briefly described in the overall context to give an impression of the workflow in VANESA.

Network modeling and reconstruction
A modeling process begins with a clear scientific question stated by the user (1). Either the user models the biological system by hand with knowledge from literature and/or own studies, or he formulates a query to automatically reconstruct a system with information derived from

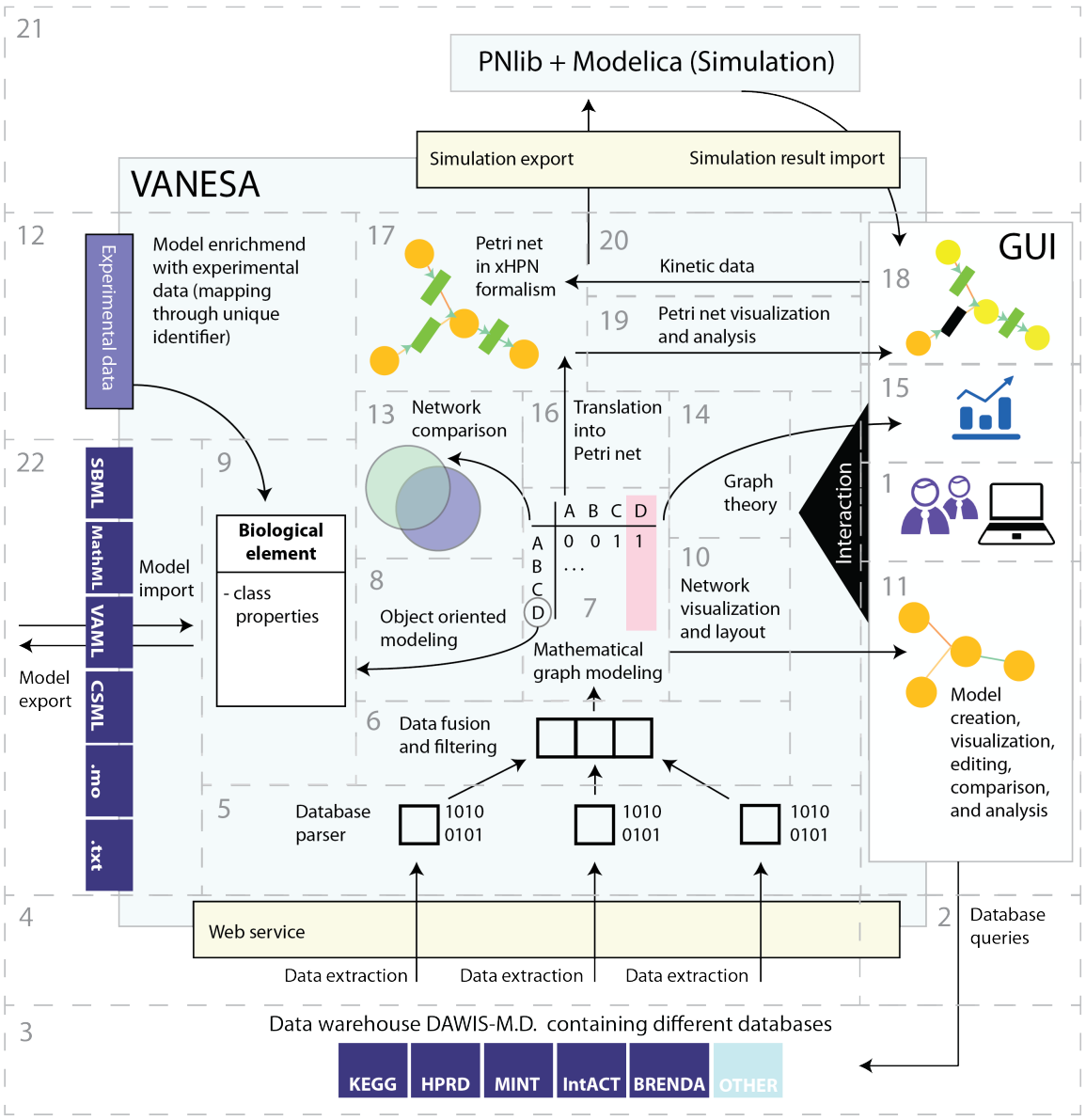


Figure 4.2: Overview of the system architecture of VANESA. The numbers represent the different modules providing the stated functionalities. Each number represents a different bioinformatics module in VANESA, which in combination with the other approaches, forms the entire framework and its possibilities as described in Section 4.2.

the integrated databases (2). For database access, VANESA offers a form where scientists can query each database for relevant information. Database information is gained by accessing the data warehouse DAWIS-M.D., which contains the selected databases KEGG, HPRD, IntAct, BRENDA, and Mint (3). Besides, the data warehouse contains even more databases covering other -omic level, which can be accessed by need. Therefore, VANESA must be simply extended by a new database parser. Access to the databases is realized by an asynchronous web service, which automatically connects to the data repository via internet (4). The web service extracts the user-specific data and sends it to VANESA, where it is parsed and normalized for further processing (5). In the next step, the queried data is filtered with regard to usability and fused into one data structure (6). In the first place, a mathematical network model which reflects the structure and topology of the reconstructed network model is constructed (7). Using object-oriented modeling (8), the elements of the adjacency matrix are expanded to java classes, storing model significant data and database knowledge (9). Before visualizing the reconstructed model, the network is automatically layouted with the appropriate layout algorithms (10) and then presented in the graphical user interface (11). In the graphical user interface, the user has the possibility to examine, discuss, edit, extend, and reduce the model.

Network analysis

For further analysis, VANESA offers even more bioinformatics approaches. On the one hand, users are able to map results from laboratory experiments on an existing network by using a text import function (12). These results are linked to the network model and made visually accessible. Therefore, each of the results is stored in the corresponding network element class (9), where it can be accessed for further analysis. Furthermore, users are able to compare different models with each other to analyze similarities and differences in network structure, system regulations, and dynamics (13). Additionally, graph theoretical analyses can be applied to the networks to identify relevant elements and structures within them (14). To make calculated and predicted results more intuitive and understandable, the calculations of these approaches are directly applied and dynamically visualized on the networks. This is realized using an animation algorithm, which users can interactively influence and control (15). For example, if one element within the network is more important concerning graph theory than others, the element is visually enlarged, whereas the other elements become smaller in an animation sequence.

Simulation

For the simulation processing, users are able to transform a biological model into a Petri net (16). VANESA can automatically translate the network structure into the xHPN formalism (17), which then, can be examined and edited in the visualization pane of VANESA (18). Furthermore, users have the possibility to directly model a system using the Petri net language without first reconstructing a model. However, based on the Petri net, users are able to check the liveness, reversibility, t-invariants, p-invariants, and possible system states before simulating it (19). This can help in building and setting up meaningful models, which can be used for hypothesis testing. However, if kinetic data is available it can be incorporated into the Petri

net by placing ODEs or system parameters, such as capacities and thresholds on the places and transitions (20). Finally, simulations are performed using the integrated PNlib[2], which runs invisibly in the background (21). Once the simulation processing is finished, the results are automatically transformed back to VANESA and made visually accessible with charts and network animations (18).

Exchange
Biological standards should ensure that all model concepts are well-defined and can also be exported and imported into VANESA (21). Thus, VANESA enables users to share and evaluate models with any other software application supporting the export files SBML, MathML, CellML, VAML, and/or the network text export file.

4.3 Discussion

Although many different requirements have been integrated into VANESA, it was possible to implement a software application that is already technically sound and useful. The main requirements were made prior to programming, ensuring that all necessary features are taken into account for the modeling, analysis, visualization, and simulation of biological systems. The initial concept for VANESA is derived from the preliminary work performed on the software application Network Editor. However, in order to create a powerful new framework which is able to support scientists in their research, the different requirements were clustered into functional groups and realized as individual programming modules in VANESA. Each of the modules has its own properties and defined data inputs and outputs. These modules interact with each other through programming interfaces. Thus, it is possible to combine different bioinformatics approaches as in a building-block system. Due to VANESAs generic design, it is even possible to extend the presented architecture with new programming interfaces, databases, exchange formats, and/or analysis techniques. Hence, it can be adapted to meet new challenges and further research questions in the natural sciences. In order to give an overview of how the realization of the provided features looks in detail, the following chapter presents the most important implementations in VANESA.

[2]Although, the PNlib is best suited because of its possibilities and features, it is only available in the commercial product Dymola. However, the authors are working on an alternative version, the prototype, of which, is already available and accessible. One main advantage of the PNlib is that once integrated or linked, it should work in both versions, the commercial and the open source version.

Chapter 5

Implementation

This chapter deals with the technical realization of the design concepts and requirements mentioned in Chapter 4. Therefore, the most important implementations are presented and put into relationship with the overall system architecture. The first section discusses the implemented data structure, which is used to model and simulate biomedical systems in VANESA, followed by Section 5.2, which describes the network reconstruction. It is explained which databases are integrated in the software framework and how networks are automatically reconstructed, based on the provided database information. The chapter continues with Section 5.3, which demonstrates the Petri net simulation processing using the xHPN formalism. Implemented Petri net analysis techniques are discussed in Section 5.4. In addition, VANESA provides graph theoretical approaches as described in Section 5.5. Here, the main implementations are listed to give an impression of the various possibilities. The next section, 5.6, introduces the network comparison techniques used to identify similarities and differences between a set of networks. Section 5.7 focuses on network visualization and interaction. It is discussed, how VANESA facilitates users in their visual analysis. Section 5.8 deals with biological standards and the provided exchange formats. Standards and exchange formats are very important, as they guarantee that the reconstructed models can be made accessible and shared with other software applications in the field of biomedical network modeling. The chapter ends with a summary in Section 5.9, listing the various features of VANESA.

5.1 Data model

In order to model a biomedical network enriched with database information and experimental findings, a new extended data structure for the backend had to be realized in VANESA. Therefore, an integrated data structure consisting of a 9-tuple has been defined and implemented (see Definition 5).

Definition 5. *The integrated data structure O is a 9-tuple = (C, R, CV, CT, RT, P, cv, ct, rt) that consists of:*

- *C(O) = {c_1, c_2, ... , c_n} is a finite non-empty set of discrete concepts such as: metabolites, enzymes, substances, substrates, products, signals, genes, proteins, cells, complexes, activators, and inhibitors, among others.*
- *R(O) = {r_1, r_2, ... , r_n} is a finite non-empty set of discrete relations such as activation, inhibition, expression, repression, state change, binding / association, dissociation, phosphorylation, dephosphorylation, glycosylation, ubiquitination, and methylation, among others.*
- *CV(O) is a finite non-empty set of vocabularies inspired by the Systems Biology Ontology,*
- *CT(O) is a tree consisting of concept classes,*
- *RT(O) is a tree consisting of relation classes,*
- *C(O), R(O), CT(O) and RT(O) are pairwise disjoint,*
- *P(O) is a finite non-empty set of additional properties consisting of:*
  - *CPN(O) is a finite non-empty set of concept names and definitions,*
  - *RPN(O) is a finite non-empty set of relation names and definitions,*
  - *CPE(O) is a finite non-empty set of experimental results,*
  - *CPD(O) is a finite non-empty set of database properties for concepts,*
  - *RPD(O) is a finite non-empty set of database properties for relations,*
  - *CPS(O) is a finite non-empty set of molecular structure properties for concepts,*
- *functions that assign vocabularies, concept classes and relation classes are:*
  - *cv: $C(O) \cup R(O) \rightarrow CV(O)$,*
  - *ct: $C(O) \rightarrow CT(O)$,*
  - *cv: $R(O) \rightarrow RT(O)$,*
- *functions that optionally link additional properties to concepts or relations are:*
  - *def $C(O) \rightarrow DEF(O)$,*
  - *def $R(O) \rightarrow DEF(O)$,*
  - *cpn $\rightarrow \{(cpn_1 \times ... \times cpn_n) \mid cpn_j \in CPN(O)\}$,*

- $rpn \rightarrow \{(rpn_1 \times ... \times rpn_n) \mid rpn_j \in RPN(O)\}$,
- $cpe \rightarrow \{(cpe_1 \times ... \times cpe_n) \mid cpe_j \in CPE(O)\}$,
- $cpd \rightarrow \{(cpd_1 \times ... \times cpd_n) \mid cpd_j \in CPD(O)\}$,
- $rpd \rightarrow \{(rpd_1 \times ... \times rpd_n) \mid rpd_j \in RPD(O)\}$,
- $cps \rightarrow \{(cps_1 \times ... \times cps_n) \mid cps_j \in CPS(O)\}$.

In simple terms, the backend structure is a computational representation of a mathematical graph, in which concepts are the nodes and relations are the edges. Each concept represents a real world entity, with specific properties and characteristics. Relations are used to represent how the concepts are related to each other.

The Network Editor was based on the graph representation of the JUNG[1] library. However, for the extended data structure and new analysis approaches, a new graph structure had to be implemented. Therefore, VANESA represents a biological network as an adjacency matrix with links to the data structure. The reason for this choice is that a node-node adjacency matrix is the most basic form of representing network topology. It is very convenient to work with, easily accessible for many graph algorithms, and facilitates important graph manipulation and analysis operations [CSLR01]. Furthermore, the node-node adjacency matrix is very suitable for dense graphs, as they appear in biological terms nowadays.

The front-end of VANESA is a visible graph. This graph has been modified and extended to meet the new integrated ontology. For simple network modeling approaches, Definition 6 has been defined and implemented.

Definition 6. *The front-end is a visible graph G described with the 10-tuple G(O, CO, SO, L, color, shape, size, x, y, visibility) that consists of:*

- *an integrated data structure O,*
- *CO(G) is a finite non-empty set of colors,*
- *SO(G) is a finite non-empty set of shapes,*
- *L(G) is a finite non-empty set of graph layouts,*
- *the functions color, shape, size, visibility, x and y (coordinates) which affect the way concepts and relations are visualized in the front-end:*
  - *color:* $C(O) \cup R(O) \rightarrow CO(G)$,
  - *shape:* $C(O) \cup R(O) \rightarrow SO(G)$,

[1] http://jung.sourceforge.net/

- *size: C(O)* $\rightarrow \mathbb{R}$,
- *x: C(O)* $\rightarrow \mathbb{R}$,
- *y: C(O)* $\rightarrow \mathbb{R}$,
- *visibility: C(O)* $\cup$ *R(O)* $\rightarrow$ *{true, false}*.

For the simulation of a biological network, the integrated data structure and visible graph has to be transformed into the Petri net language. As described in Section 3.1, Proß *et al.* have defined the Extended Hybrid Petri Nets for biological applications (xHPNbio) formalism, a powerful mathematical modeling concept properly adapted to the demands of biological processes. During the transformation, each node in a biological network is replaced by a place and an edge connecting two elements which is replaced by a transition, according to Definition 7.

Definition 7. *For simulation processes the xHPN formalism is used with the 18-tuple (PD, PC, TD, TS, TC, F, G, T, I, R, f,* c_l, c_u, *d, h, v, s,* m_0*) [PJHB12]*

- $PD = \{pd_1, pd_2, \ldots, pd_{pd}\}$ *is a finite set of discrete places,*
- $PC = \{pc_1, pc_2, \ldots, pc_{pc}\}$ *is a finite set of continuous places,*
- $TD = \{td_1, td_2, \ldots, td_{td}\}$ *is a finite set of discrete transitions,*
- $TS = \{ts_1, ts_2, \ldots, ts_{ts}\}$ *is a finite set of stochastic transitions,*
- $TC = \{tc_1, tc_2, \ldots, tc_{tc}\}$ *is a finite set of continuous transitions,*
- *PD, PC, TD, TS, and TC are pairwise disjoint,*
- $F \subseteq (PD \times TD \cup PD \times TS \cup PD \times TC \cup PC \times TC \cup PC \times TD \cup PC \times TS)$ *is a set of arcs from places to transitions, where* $(p_i \rightarrow t_j)$ *denotes the arc from place* p_i *to transition* t_j,
- $G \subseteq (TD \times PD \cup TD \times PC \cup TS \times PD \cup TS \times PC \cup TC \times PC \cup TC \times PD)$ *is a set of arcs from transitions to places, where* $(t_j \rightarrow p_i)$ *denotes the arc from transition* t_j *to place* p_i,
- $T \subseteq (PD \times TD \cup PD \times TS \cup PD \times TC \cup PC \times TC \cup PC \times TD \cup PC \times TS)$ *is a set of test arcs,*
- $I \subseteq (PD \times TD \cup PD \times TS \cup PD \times TC \cup PC \times TC \cup PC \times TD \cup PC \times TS)$ *is a set of inhibitor arcs,*
- $R \subseteq (PD \times TD \cup PD \times TS \cup PD \times TC \cup PC \times TC \cup PC \times TD \cup PC \times TS)$ *is a set of read arcs,*

- *F, G, T, I, and R are pairwise disjoint,*
- *f: $(F \cup G \cup T \cup I, m) \rightarrow \mathbb{R}_{\geq 0}$ is an arc weight function which assigns a non-negative integer to every arc connected to a discrete place. All others are assigned a non-negative real number depending on a concrete marking m , where $(f{:}p_i \rightarrow t_j)$ denotes the weight of the arc from place p_i to transition t_j,*
- *if $p_i \in PD$, $t_j \in TC$ then $(p_i \rightarrow t_j) \in F$ if only if $(t_j \rightarrow p_i) \in G$ and $(f{:}p_i \rightarrow t_j) = (f{:}t_j \rightarrow p_i)$,*
- *c_l: $\{PD \rightarrow \mathbb{N}_0, PC \rightarrow \mathbb{R}_{\geq 0}\}$ are the minimum capacities of the places,*
- *c_u: $\{PD \rightarrow \mathbb{N}_0, PC \rightarrow \mathbb{R}_{\geq 0}\}$ are the maximum capacities of the places,*
- *d: $TD \rightarrow \mathbb{R}_{\geq 0}$ is a delay function which assigns a positive, real-valued delay to every discrete transition,*
- *h: $(TS, m) \rightarrow \mathbb{R}_{\geq 0}$ is a hazard function which assigns a positive, real-valued random delay depending on a concrete marking m to every stochastic transition,*
- *v: $(TC,m) \rightarrow \mathbb{R}_{\geq 0}$ is a maximum speed function which assigns a positive, real-valued maximum speed depending on a concrete marking m to every continuous transition,*
- *s: $(TD \times TS \times TC, mv) \rightarrow \{true, false\}$ is a condition function which assigns a condition depending on all possible model variables (mv) to every transition, e.g., time,*
- *m_0: $\{PD \rightarrow \mathbb{N}_0, PC \rightarrow \mathbb{R}_{\geq 0}\}$ is the initial marking which must satisfy the condition $c_l(p_i) \leq m_0(p_i) \leq c_u(p_i) \; \forall \; p_i \in (PD \cup PC)$.*

From the biological point of view, the formalisms presented in Definition 5 and 7 have the following meanings: Places are biological compounds such as: metabolites, enzymes, substances, substrates, products, signals, genes, proteins, cells, complexes, activators, inhibitors, repressors, promoters, transcription factors, and RNAs, among others. Transitions are biological processes such as: biochemical reactions, metabolic reactions, interactions, regulatory reactions, signal transduction reactions, chemical reactions, binding, and phosphorylation, among others. The marking of a Petri net describes biological concentrations such as the amount of molecules or cells. Additionally, every place can be assigned with minimum and maximum capacities. Normal arcs are used to connect biological compounds and processes with each other. Furthermore, test arcs are used to describe activation processes such as the transcription process, activation in gene regulation, enzyme activity, and activation mechanisms, among others. Inhibitor arcs describe inhibition mechanisms such as the repression of gene regulation, among others. Read arcs describe catalytic processes. Additionally, each arc can be weighted with biological coefficients to include stoichiometric coefficients and yield coefficients in the model. Furthermore, each biological process can have a delay. In order to model and simulate random duration of biological processes, hazard functions can be assigned. For example, such a hazard function can be used

to model and simulate stochastic kinetics. Maximum speeds of biological processes, such as kinetics effects/laws can also be assigned.

Due to the used data structure, a variety of networks can be reconstructed, analyzed, simulated, and visualized. As a result of the generic design, the software application is not only limited to biological networks. Because of the strict separation of internal data structure and graphical representation it is straightforward to extend VANESA by new graph and network classes. The generic design allows software programmers to easily extend existing components by introducing new concepts applying the re-use and specialization of already existing ones.

5.2 Network reconstruction

To assist scientists in describing a biological system with all its possible reactions, transformations, and modifications, an advanced database consulting module has been implemented in VANESA. The data extraction, transformation and loading was a major concern for the effectiveness in conveying information since the amount of datasets has become enormous and multi-dimensional. Therefore, a new ETL (Extraction, Loading, and Transformation) module for several databases has been implemented. Now eleven different databases from DAWIS-M.D. covering almost all -omic levels can be accessed (see Figure 5.1). Most of the biological processes within a cell such as enzymatic reaction, protein-protein interaction, metabolic and signaling pathways, among others, can be modeled and analyzed.

One possibility to connect to the integrated databases is by using a local database access. VANESA offers the functionality to directly connect to a mySQL server and the necessary databases. Users only need to set up the connection parameters such as host, database name, username, and password in the settings panel of VANESA. More convenient is the access via the implemented web service, which is realized by an asynchronous Axis2 web service technology [JA11]. Using this web service it is possible to consult the BioDWH and DAWIS-M.D. to gather biological and medical information. Queries can be sent simultaneously without loss of performance and connection dropouts. Thus, it is possible to query the data warehouse for large and complex datasets. VANESA uses MySQL [SZT12] standards and optimization techniques to ensure efficient data exchange.

Based on the web service, any kind of network can be loaded. Due to the asynchronous web technique, communication deadlocks from client to server are excluded. Thus, networks with increased size can be efficiently loaded. Actually, the manageable size of a network is 1,500 nodes with about 6,000 edges. Bigger networks result in the performance reduction of VANESA due to the used graph library JUNG 1.7. Using the database search panel, users are able to search in the integrated databases, such as KEGG, BRENDA, Mint, IntAct, and HPRD. Therefore, users have several possibilities to formulate their queries. They are able to search for all kinds

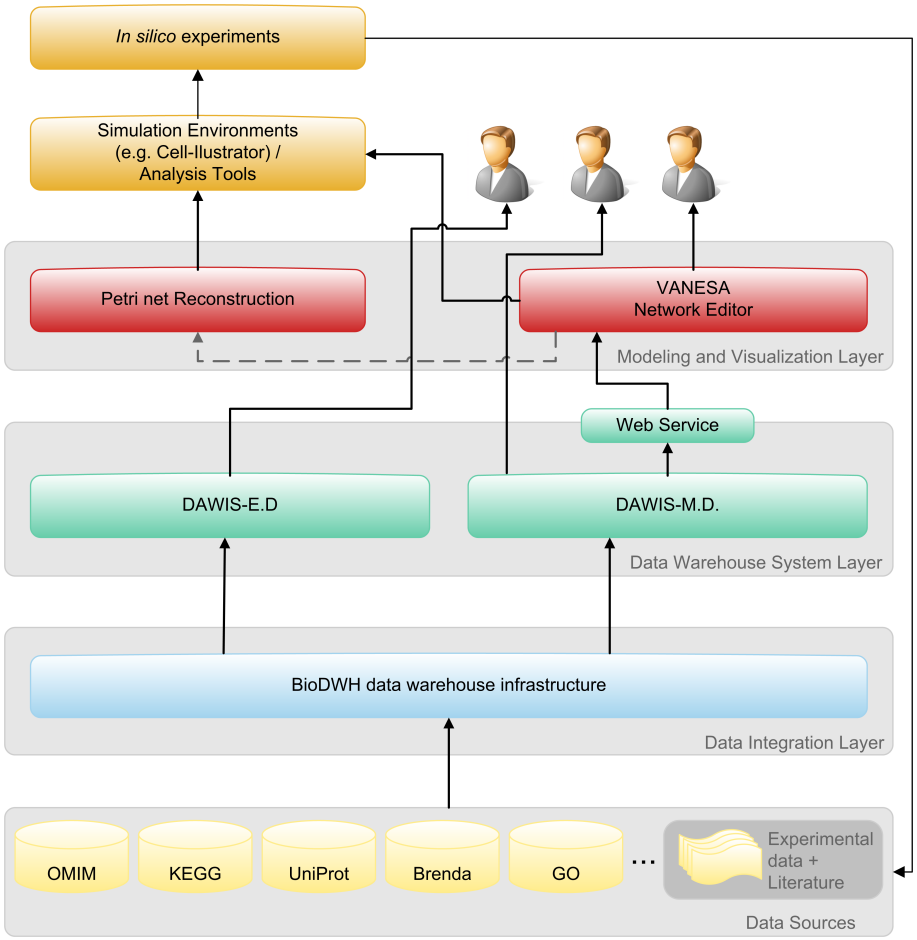


Figure 5.1: This picture presents the overall data integration and consulting architecture of VANESA. Biological data is extracted, transformed, and loaded from different data sources such as KEGG, UniProt, BRENDA, OMIM, and GO, among others, into the BioDWH, which is then accessible to DAWIS-M.D. for metabolic data and DAWIS-E.D for experimental data. Using an Axis 2 web service VANESA is able to query DAWIS-M.D. in order to reconstruct, model, and visualize biological networks. Based on these networks, an automatic Petri net reconstruction and simulation can be performed for *in silico* experiments.

of biological elements matching or partially matching a given biological definition, name, or identifier. Database content can be queried for all sources or targets. Optionally, users are able to select whether or not a database query is supposed to be organism specific. Boolean operators such as *And*/*Or*/*Not* provide an additional way to formulate specific and complex queries for an advanced search within the linked databases. Using the different search forms, it is possible to reconstruct networks based on KEGG pathways, enzymes, proteins, genes, and compounds, among others. The tabs in the database panel divide the different databases from each other. Each tab represents a database module, which is linked to specific algorithms to query and reconstruct biomedical networks from DAWIS-M.D.

However, information stored in databases is distributed over many tables. In order to recon-

struct a biological network, links, and connections from one biological compound to another have to be established. This process is done piece by piece until completing a certain pathway map or reaching a given network size. Primarily, the KEGG database is used to reconstruct metabolic networks in VANESA. Therefore, a set of algorithms is implemented, which is based on the KEGG Markup language. Although KGML is an exchange format, it describes the relation between the different objects within the database in a generalized way. With the KGML specification it is possible to query the relational data schema efficiently and finally, to reconstruct metabolic networks as they are presented by KEGG. In order to access more detailed information, additional tables from KEGG have to be queried. Actually, KEGG consists of 121 different tables. Twenty-three tables are necessary to reconstruct a basic network without any additional data, such as information about protein-interactions, disease/drugs, and involved chemical substances, among others. In general, for each database a query class exists, where the specific queries are stored that are used to gather the necessary information. If a database scheme changes, only these queries have to be modified in VANESA. This ensures that the computer scientist only needs a short time to adapt to the new changes. Two basic examples of such queries are presented in the following:

```
1) public static final String getKEGGpathwayByName = "SELECT pathway_name,
title, org,number,image,link FROM kegg_pathway p where pathway_name = ?;";

2) public static final String getKEGGentriesByPathwayName = "SELECT k.entry_id,
k.link, k.type, n.ec, g.background, g.foreground, g.graphicsName,
g.graphicsType, g.x, g.y FROM dawis_md.kegg_entry k
left outer join dawis_md.kegg_entry_name n
on k.entry_id=n.entry_ID Inner join dawis_md.kegg_graphics g
on k.entry_id=g.entry_ID where k.pathway_name=?
and n.pathway_name=? and g.pathway_name=?
Order by k.entry_id;";
```

Here, the questions marks are automatically replaced by the pathway name typed in by the user in the database search form, which enables VANESA to consult the database to get all KEGG pathways and entries which match the search term. Using SQL joins on specific identifiers such as gene names, it is even possible to connect different databases with each other. Another advantage is that queries can be optimized by using advanced SQL statements. Thus, no intervention in the programming code is necessary. Once the data from the database is received, the information is processed.

The BRENDA database can also be used to reconstruct metabolic networks. For each queried enzyme, substrate, or product, a reaction list is created containing all involved biological elements such as inhibitors, cofactors, etc. This list is further converted into a connection matrix substrate-product, which can be directly interpreted for analysis and visualization. Information on reversibility and the type of connection is also considered in this matrix. Therefore, the

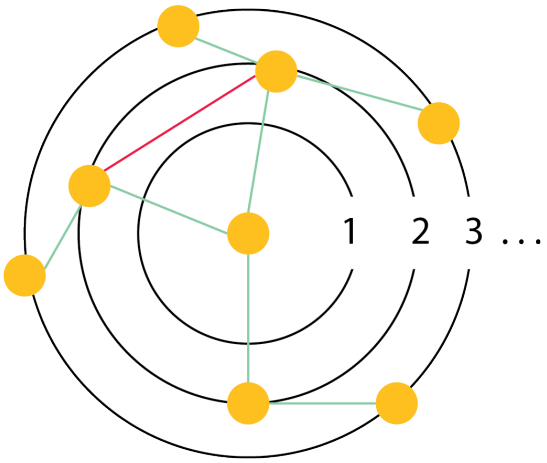


Figure 5.2: This picture presents how networks are reconstructed in VANESA. In the initial phase, the queried research object is placed as a preliminary root (Level 1). In the following step, selected databases are queried in DAWIS-M.D. for elements, which interact with this node. If interactions exist, new nodes are added to the graph and interconnected with it (Level 2). In addition, algorithms in VANESA check by querying the databases if the other nodes can be interconnected. These processes are then repeated in an iterative way until reaching a certain network size specified by the user (Level 3).

matrix needs to be further processed to remove connection via currency metabolites. Currency metabolites are mainly used as carriers for transferring electrons and other functional groups such as ATP, H_2O, CO_2, and others. Nevertheless, currency metabolites are not shown or even considered in metabolic pathways, since structure analysis with connections through currency metabolites produces meaningless results.

One structure analysis approach which break down with currency metabolites is the calculation of the shortest path length from one element to another. For example, in terms of biochemistry and KEGG information, the path length from glucose to pyruvate should be nine. If currency metabolites, such as ATP and ADP are also considered, the path length becomes two, since the first reaction uses glucoses and produces ADP while the last reaction consumes ADP and produces pyruvate. Beside the fact that structural analysis can become biologically meaningless, the network size might increase with currency metabolites, as large number of edges have to be added. Therefore, the covering of such elements is essential to draw meaningful conclusions from a graph analysis.

In order to address this problem, top-ranked metabolites can be excluded by the user in VANESA. Based on their connection degree, VANESA calculates their ranking and provides users with the possibility of disregarding all or only selected metabolites identified as currency metabolites. All calculations can be performed organism specific, to narrow search and analysis. Furthermore, the protein-protein interaction databases Mint, IntAct, and HPRD can be

used for the reconstruction of biological networks. The network reconstruction algorithm also works with a connection matrix, whereby connections and complexes between proteins are determined. Thus, users have the possibility to reconstruct sophisticated networks with a chosen network size, that specifies up to which network degree the model should be reconstructed (only first interaction partners, first interaction partners and their neighbors, neighbors of neighbors, and so on - see Figure 5.2). Additionally, binary interactions and complex interactions can be included and excluded in the model.

5.3 Petri net simulation processing

For the simulation of biological processes, VANESA makes use of the xHPNbio formalism. With a biologically sophisticated graphical user interface, network models can be reconstructed and automatically translated into the language of the xHPNbio paradigm. Due to VANESAs generic design and strict separation of internal data structure and graphical representation, it is possible to convert the integrated data structure for network representation (see Definition 5) into the xHPNbio formalism (see Definition 7) that can be then, further simulated within the PNlib.

A digital communication bridge, invisibly running in the background, realizes the communication between VANESA and the PNlib. Once a Petri net model with initial markings and arc weights is reconstructed in VANESA, a script automatically translates the ready Petri net model into the appropriate .mo data exchange format and starts Modelica (Dymola) and the corresponding PNlib in the background for simulation as presented in Figure 5.3. An example of such a .mo data exchange format is presented in the following. It shows an extract of the simulation parameters for the transcription-regulated *lac*-operon system of the bacterium *Escherichia coli* as presented in Figure 5.4.

```
1 model simulation
2       PNlib.IA inhibitorArc1;
3       PNlib.IA inhibitorArc2;
4       PNlib.PC P1003(nIn=1,nOut=0,startMarks=0.0,minMarks=-1.0,maxMarks=1.0E9);
5       PNlib.TC T1079(nIn=1,nOut=1,maximumSpeed=1.0,arcWeightIn={P1008.t*0.1},
6       arcWeightOut={(4*40*P1008.t^3*20^4)/((20^4+P1008.t^4)2)});
7       arcWeightOut={1});
8       ...
9 equation
10      connect(P1044.outTransition[1],T1088.inPlaces[1]);
11      connect(P1008.outTransition[1],T1079.inPlaces[1]);
12      connect(inhibitorArc1.outTransition,T1096.inPlaces[2]);
13      ...
14 end simulation;
```

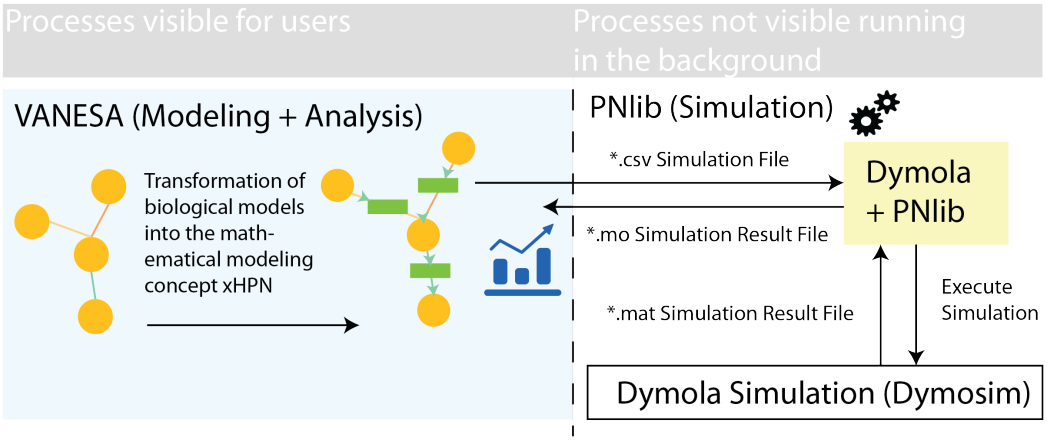


Figure 5.3: VANESA enables Petri net simulations using the features of Dymola and the PNlib. Therefore, VANESA automatically transforms its models into the xHPNbio formalism and exports it into the simulation environment. Simulation processing is performed automatically and is not visible in the background. As soon as the processing is finished, results are automatically sent back to VANESA, where they are visualized with plots and special animation techniques.

Format tags, such as PNlib.PC (continuous place), PNlib.IA (inhibitor arc), and PNlib.TC (continuous transition) with its markings and weights build up the backbone for simulation processing. Using these connection tags the different xHPNbio Petri net elements are connected to each other.

Due to multi-threading, VANESA is able to detect simulation results as soon as they are available. When simulation processing is finished, the simulation results are automatically loaded into VANESA. The data and simulation exchange is realized by a .csv file, which lists progress steps and information. Finally, results are matched on the network and made visible. With the supported xHPNbio paradigm, sophisticated simulations can be performed using qualitative, stochastic, continuous, hybrid, and functional modeling features. Furthermore, ODEs can be used for continuous Petri net models. Thus, users are able to check behavioral and structural properties. Simulation results are available as tables or as charts, and can be exported in JPEG files.

Simulation results can also be animated within the graphical user interface. Therefore, a new approach has been implemented which represents the token numbers on selected places over time (see Figure 5.5). Simulations can be triggered manually or be animated within the active window. The animation is interactive and can be performed for each time interval. During the animation, the nodes change their size and color depending on the amount of tokens. If the amount of tokens increases, the node gets bigger and is colored in red. If the amount of tokens decreases, the place gets smaller and the node is colored in blue. Thus, users are able to intuitively recognize system state changes and information flow within the reconstructed models.

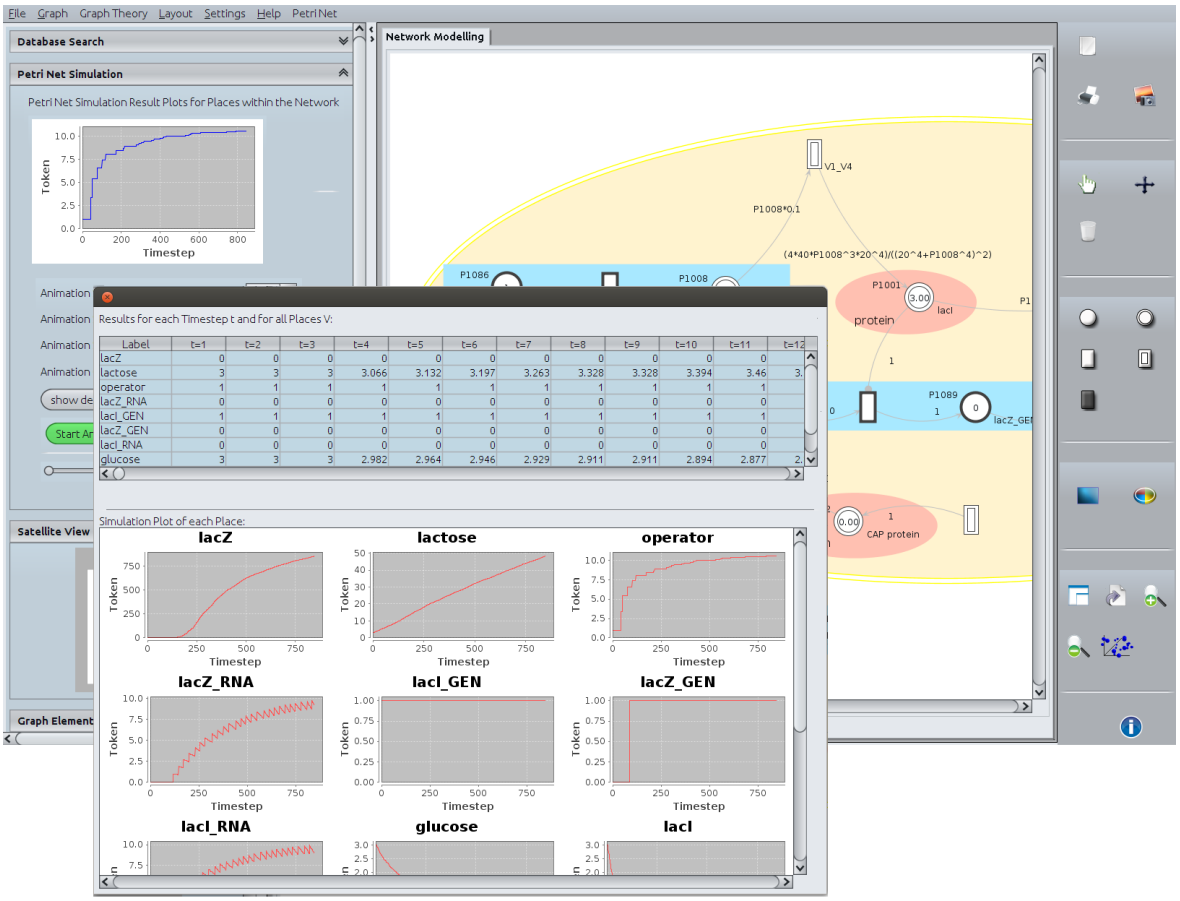


Figure 5.4: Simulation results of the transcription-regulated *lac*-operon system of the bacterium *Escherichia coli* within VANESA. The presented example simulates the cell behavior of the bacterium in response to decreasing glucose and increasing lactose in the cell environment. The charts show the cell dynamics of involved biological elements such as lactose, glucose, and the *lacZ* gene, among others.

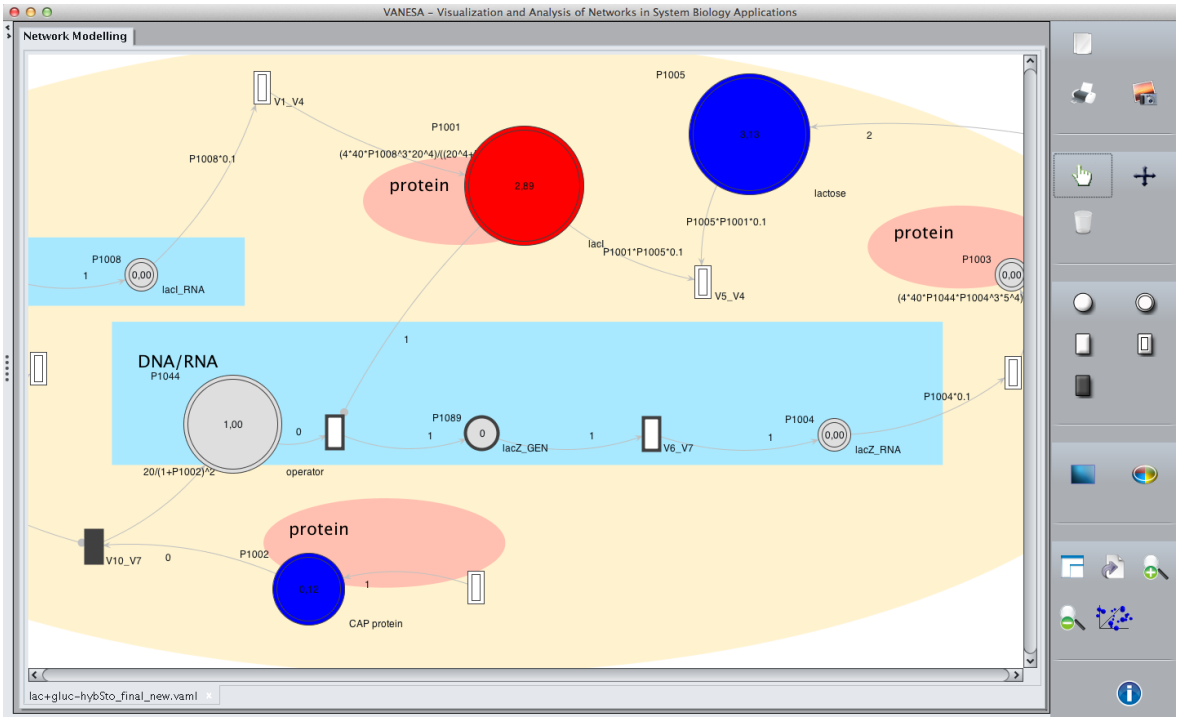


Figure 5.5: The figure shows a screenshot of an animated Petri net simulation of the transcription-regulated *lac*-operon system of the bacterium *Escherichia coli* within VANESA. It demonstrates decreasing glucose and increasing lactose in the cell environment. The model is created with discrete (single-edged circles) and continuous places (double-edged circles), and discrete (single-edged rectangles), continuous (double-edged rectangles), and stochastic transitions (black rectangles). Places and transitions are interconnected with each other using normal arcs (edges with arrowheads) or inhibitory arcs (edges with circles). Using ODEs, the rate change of variables in the modeled system is described. The numbers within the places represent the current token numbers. The node size highlights the amount of tokens compared to the other places. The node color indicates whether the token number decreases (blue), increases (red), or remains the same (grey) in the next simulation step.

Petri net validation is also provided, to prevent modeling errors such as connection from place-to-place, among others. This validation is based on the data structure presented in Definition 7. During model creation, algorithms constantly check to see if the Petri net is valid in accordance with the aforementioned data structure. Additionally, a general syntax formula validation algorithm checks for the correctness of the mathematical equations placed on the continuous or hybrid Petri nets. If they are not correct, users are made aware of the specific modeling error. Furthermore, users have the possibility to move between the different classes of modeling and simulation concepts, since the graphical user interface adapts automatically to the net class in the active window.

In addition to the possibility to simulate networks with Modelica in VANESA, an additional export function for the software application CellIllustrator is implemented. It is possible to transform any kind of reconstructed network model into the CSML export file, which then can be loaded in CellIllustrator to simulate and analyze system dynamics. Therefore, an appropriate CSML XML file is automatically constructed in VANESA, in which the java objects are mapped on the specific CSML element and attribute tags. This is done in accordance to the Cell System Markup Language Specification version 1.9[2]. The CellIllustrator export functions supports discrete, continuous, and hybrid Petri net models.

5.4 Petri net analysis

In order to perform Petri net model validation and analysis, several techniques had to be implemented. To provide additional insights into network behavior and furthermore, to detect system inconsistencies, mathematical invariant concepts are now provided. Therefore, the t-Invariant vector I_T regarding to Definition 8 [PW08] is calculated in VANESA:

Definition 8. *A t-Invariant is a vector* I_T *if*
$W^+ \cdot I_T = W^- \cdot I_T \iff W^+ \cdot I_T - W^- \cdot I_T = 0 \iff (W^+ - W^-) \cdot I_T = 0 \iff W^T \cdot I_T = 0$
with $I_T \in Z^T \geq 0$ *where the transitions of a Petri net are described as a pair of* $|P| \times |T|$ *matrices with*

- W^-, *defined by* $\forall$ *s,t :* W^- *[p,t] = f (p,t)*
- W^+, *defined by* $\forall$ *s,t :* W^+ *[p,t] = f (t,p)*
- W^T = W^+ - W^-

The t-invariant is a vector defining a multiset of transitions. In general, the vector has two biological interpretations. On the one hand, a t-invariant represents a multiset of transitions which reproduce a given marking by their ordered firing. This can contribute to a deeper

[2] http://www.csml.org/download/CSML_1.9_Specification.pdf

understanding of the systems behavior. Additionally, the t-invariant can be used to examine relative firing rates of transitions. Thus, transitions firing rates occurring permanently and concurrently can be detected. This leads to better insights into the system's steady state behavior and activity level.

A further invariant concept is the p-invariant. In the context of networks, p-invariants represent a set of places, over which the weighted sum of tokens is constant, independent of the firing events. This technically indicates token-preserving, which reflects compound preservations in a system and also corresponds to active/inactive states of a given compound. Therefore, any two reachable markings m_1, m_2 have to hold the equation: $x \times m_1 = x \times m_2$. To test p-invariants, the vector I_P is calculated regarding to Definition 9 [PW08] in VANESA:

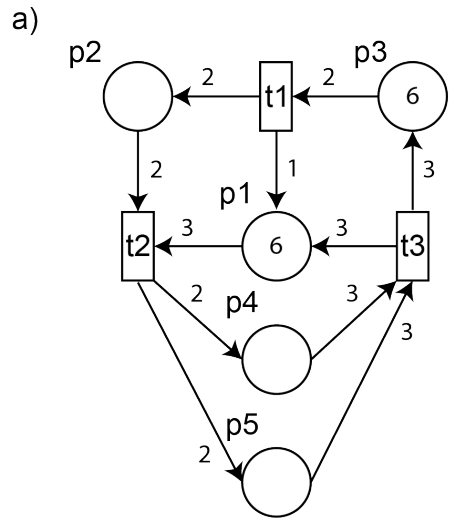
Definition 9. *A p-Invariant is a vector I_P if*
$W^+ \cdot I_P^T = W^- \cdot I_P^T \iff W^+ \cdot I_P^T - W^- \cdot I_P^T = 0 \iff (W^+ - W^-) \cdot I_P^T = 0 \iff W^T \cdot I_P^T = 0$
with $I_P \in Z^S \geq 0$ where the transitions of a Petri net are described as a pair of $|P| \times |T|$ matrices with

- *W^-, defined by $\forall$ s,t : W^- [p,t] = f (p,t)*
- *W^+, defined by $\forall$ s,t : W^+ [p,t] = f (t,p)*
- *W^T = W^+ - W^-*

An example of t- and p-invariant calculations, based on the Petri net presented in Figure 5.6, is given in the following. The t-invariant is derived from the incidence matrix from VANESA and results in the linear equation presented in Example 1. In summary, for the equation only the solution {3,3,2} exists. The solution can be interpreted as a firing sequence to reach the initial marking. Transition 1 has to fire three times, transition 2 has to fire three times, and transition 3, twice in order to reach the initial marking. For the linear equation presented in Example 2, four solutions for the p-invariants exist. The four invariants are {P1,P3,P5}, {P2,P3,P5}, {P1,P3,P4}, and {P2,P3,P4} with the weight given by the positive integer number in the solution vector.

Example 1. *The following linear equation has to be solved to calculate the t-invariant for the presented Petri net in Figure 5.6. The solution for this linear equation is (3,3,2).*

$$\begin{array}{rrrrrcl} x_1 & +2x_2 & -2x_3 & & & = & 0 \\ -3x_1 & -2x_2 & & +2x_4 & +2x_5 & = & 0 \\ 3x_1 & & +3x_3 & -3x_4 & -3x_5 & = & 0 \end{array}$$



b)

	t1	t2	t3
P1	+1	-3	+3
P2	+2	-2	0
P3	-2	0	+3
P4	0	+2	-3
P5	0	+2	-3

Figure 5.6: a) Example of a weighted discrete Petri net with initial marking. b) Corresponding incidence (stoichiometric) matrix which describes how many tokens each place receives and how many tokens are taken by which transition. Using this incidence matrix, new markings can be calculated.

Example 2. *The following linear equation has to be solved to calculate the p-invariant for the presented Petri net in Figure 5.6. The solutions for this linear equation are (2,0,1,0,3), (0,1,1,0,1), (2,0,1,3,0), and (0,1,1,1,0).*

$$\begin{array}{rrrcl} y_1 & -3y_2 & +3y_3 & = & 0 \\ 2y_1 & -2y_2 & & = & 0 \\ -2y_1 & & +3y_3 & = & 0 \\ & +2y_2 & -3y_3 & = & 0 \\ & +2y_2 & -3y_3 & = & 0 \end{array}$$

In addition to the t- and p-invariant calculations, a reachability and/or covering graph for a given Petri net can be constructed in VANESA. With such graphs, users have the possibility to visually examine system conditions and Petri net properties. As already mentioned in Section 3.2, the time and space complexity for a covering graph is $(2^{O(\sqrt{n})})$. Having this in mind, users have the possibility to first check to see if it is even possible to reach a selected marking m_i. This can be done according to Definition 10 (sufficient criterion).

Definition 10. *m_i can be reached from the initial marking m_0 by firing $i \in \mathbb{N}_0$ transitions if*

- *I_P is an invariant,*
- *and $I_P^T \cdot m_0 = I_P^T \cdot m_i$.*

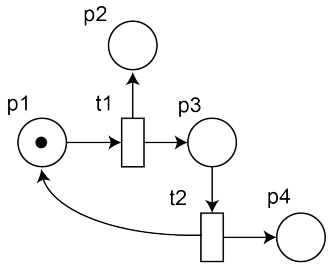


Figure 5.7: A basic discrete Petri net with an initial marking in place one (p1).

However, in order to construct a reachability or coverability graph, all possible firing events from marking m_0 to m_i have to be calculated. Therefore, for each firing event all transitions have to be checked if they are active. This is decided in accordance with Definition 11.

Definition 11. *A transition can fire when following basic rules apply:*

- *A transition t is enabled in a marking m, written as* $m\,[t\rangle$*, if* $\forall p \in {}^\bullet t : f(p,t) \leq m(p)$,
- *A transition t, which is enabled in m, may fire. When t in m fires, a new marking* m_i *is reached, written as* $m[t\rangle m_i$*, with* $\forall p \in P : m_i(p) = m(p) - f(p,t) + f(t,p)$,
- *the firing itself is timeless and atomic.*

If at least one transition is able to fire, a firing sequence is calculated (see Definition 12). Therefore, the forward matrix W^-, backward matrix W^+, and transition matrix W^T is determined and calculated as described in the Example 3 for the presented Petri net in Figure 5.7.

Definition 12. *A firing sequence is defined by*

- $m_i = \{m \mid \exists w : m_i = m_0 + W^T \times o(w) \wedge$ *is a firing sequence of N*$\}$,
- W^-, *defined by* $\forall$ *s,t :* W^- *[p,t] = f (p,t),*
- W^+, *defined by* $\forall$ *s,t :* W^+ *[p,t] = f (t,p),*
- $W^T = W^+$ - W^-.

Example 3. *A firing sequence consisting of the firing state of transition one results in the following transition matrices and firing equation:*

$$W^- = \begin{bmatrix} & t_1 & t_2 \\ p_1 & 1 & 0 \\ p_2 & 0 & 0 \\ p_3 & 0 & 1 \\ p_4 & 0 & 0 \end{bmatrix} W^+ = \begin{bmatrix} & t_1 & t_2 \\ p_1 & 0 & 1 \\ p_2 & 1 & 0 \\ p_3 & 1 & 0 \\ p_4 & 0 & 1 \end{bmatrix} W^T = \begin{bmatrix} & t_1 & t_2 \\ p_1 & -1 & 1 \\ p_2 & 1 & 0 \\ p_3 & 1 & -1 \\ p_4 & 0 & 1 \end{bmatrix}$$

$$m_i = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \end{pmatrix} + \begin{pmatrix} -1 & 1 \\ 1 & 0 \\ 1 & -1 \\ 0 & 1 \end{pmatrix} \times \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 1 \\ 0 \end{pmatrix}$$

Based on Definition 10, the Algorithm 1 was implemented in order to check if a marking m_i can be reached or covered [Bri11].

Algorithm 1 Check if marking m_i from m_0 can be reached or covered

Input: matrix W^T, initial marking m_0, system state m_1, (possible) p-Invariant I_P^T
Output: Boolean: If m_i can be reached or covered from m_0

```
1: if W^T · I_P^T = 0 then
2:    // p-invariant = true
3:    if I_P^T · m_0 = I_P^T · m_1 then
4:       // sufficient criterion met
5:       cov ← createCoverabilityGraph
6:       if m_1 ∈ cov then
7:          return m_1 is reachable
8:       if m_1 ∉ cov and ∃ m_i ∈ cov with m_i > m_1 then
9:          return is coverable // m_1 is covered by m_i
10:      return neither reachable, nor coverable
11:   else
12:      return not reachable
13: else
14:   return not reachable
```

Algorithm 2 reconstructs the coverability graph [Bri11]. Input is the root node which describes the initial marking m_0. Based on this node, all other possible markings are iteratively calculated. For each marking, it is determined which transitions are able to fire (line 2-4). For each active transition, a new marking is calculated and represented as node (line 5-6). If the

marking does not exceed the specified maximum capacity of the place (line 7), processing is continued. Otherwise, the iteration for the actual marking is stopped. In the next step, it is checked if the calculated marking is already existing in the coverability graph (line 8-11). If the coverability graph contains such a marking, an edge is drawn to that state and the iteration is stopped. If not, all nodes from the root to the processed node are checked for markings that might cover it (see line 12-14). Further on, the algorithm looks up if the marking for the actual processed place is really higher than the ones listed in the path. If it is higher, the marking for this place is replaced by an ω (see line 17) which means that the marking for this place is constantly increasing and furthermore, that it can cover all other possible markings. If such a covered marking already exists within the coverability graph (see line 19), an edge to that covering node is drawn, as long as it is not the same node as the actual processed node (see line 20-21). Otherwise, a new node is added to the coverability graph, which later is iteratively processed.

The algorithm for the reachability graph is similar, with the difference that all possible markings are calculated and interconnected. With a simple graphical form, users have the possibility to check if a marking m_i can be reached or covered. In the first place, a yes-or-no answer is given. Regardless of the result, users have the possibility to construct and visualize a reachability or coverability graph. Thus, it is possible to examine how a given marking can be reached and furthermore, which other markings are possible (see Figure 5.8 for an example).

5.5 Graph theoretical analysis

As described in Section 2.3 and 3.3, graph theory can identify several important structures in networks. Important actors within a network can be determined, as well as paths highlighted, which play an important role regarding centrality measurement. In order to support users with the possibility to apply centrality measurements in VANESA, several new algorithms were implemented, which are applied on the adjacency matrix of the corresponding networks.

In summary, users have the possibility to calculate the amount and distribution of different vertex degrees (see Algorithm 3), the largest (see Algorithm 4), the smallest (see Algorithm 5), and the average vertex degree within a network (see Algorithm 6), as well as the average neighbor degree (see Algorithm 7). In addition, graph density (see Algorithm 8), centralization (see Algorithm 9), global matching index (see Algorithm 10), and the clustering coefficient (see Algorithm 11) can be determined. Furthermore, the shortest and maximum paths, as well as the average length of all shortest and maximum paths can be identified and highlighted.

Based on these algorithms, biomedical networks can be analyzed in different sophisticated ways, as presented in Figure 5.9. Especially in high and dense graphs which suffer from visual orientation, important elements can be made visually accessible such as biological hubs. Therefore,

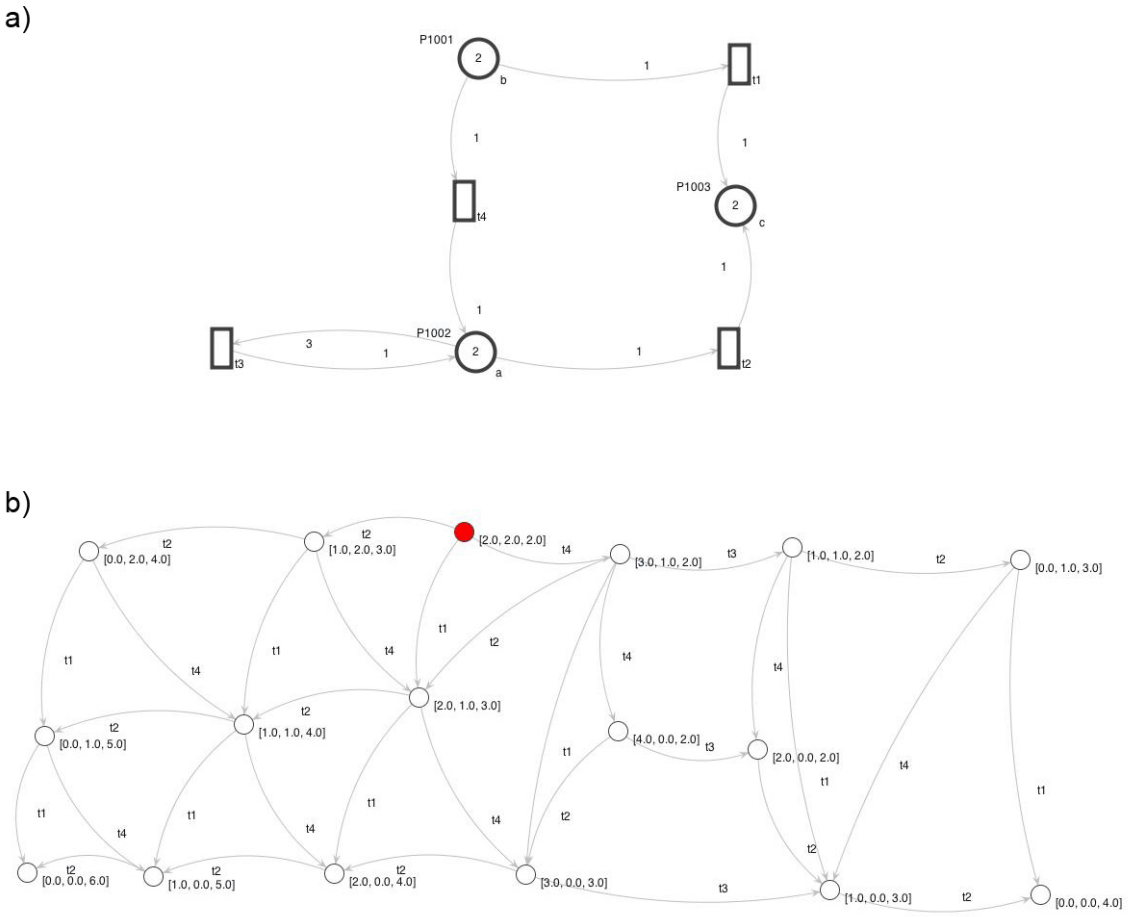


Figure 5.8: a) A basic discrete Petri net with an initial marking in VANESA. b) The corresponding reachability graph for the given Petri net in VANESA. The red node represents the root m_0 and all other nodes the possible markings m_i. The edges indicate which transition has to fire in order to reach the next marking m_i.

Algorithm 2 Construction of the coverability graph

Input: matrix W^T, matrix W^-, transitions T, places P, upper bound for places $\text{cap}(p)$ for all $p \in P$, initial marking $m_0 = root$, graph $G = (V, E)$ with $V = \{root\}$ and $E = \{\}$
Output: covering graph G

```
1: function computeNode(node) // ( node_p ∀ p ∈ P)
2: for all t_j ∈ T do
3:   col ← W^-(·, j)
4:   if node ≥ col then
5:     n ← node
6:     n ← n + W^T(·, j)
7:     if n_p ≤ cap(p)   ∀ p ∈ P  then
8:       if n = k with k ∈ V then
9:         E ← E ∪ {(node, k)}
10:        found ← true
11:      else
12:        L ← parents(node)
13:        for all ℓ ∈ L do
14:          if n > ℓ then
15:            for all p ∈ P do
16:              if n_p > ℓ_p and cap(p) = ∞ then
17:                n_p ← ω
18:            if n = k with k ∈ V then
19:              found ← true
20:              if n ≠ node then
21:                E ← E ∪ {(node, k)}
22:            if not found then
23:              V ← V ∪ {n}
24:              E ← E ∪ {(node, n)}
25:              computeNode(n)
26:              found ← true
27:        if not found then
28:          V ← V ∪ {n}
29:          E ← E ∪ {(node, n)}
30:          computeNode(n)
```

an animation algorithm has been implemented which interactively highlights centrality measurement results. Using a slide control, users have the possibility to dynamically visualize the biological elements in order of their importance. In the presented example, the vertices with the most incident edges are highlighted and furthermore, vertices with the same node degree are similarly colored. Although the network is strongly connected, the most important actors are visible. In a normal network visualization approach, this information would be lost, resulting in a meaningless furball.

Furthermore, it is possible to compare a set of different networks with each other, based on the aforementioned centrality measurements [Lew12] (see Figure 5.10). With a parallel coordinate plot, network properties can be visualized and examined in the overall context. Thus, network

Algorithm 3 Count different vertex degrees: $\mathcal{O}(n^2)$

Input: Adjacency matrix
Output: Number of different vertex degrees

```
1: vertexDegrees = []
2: for each row in adjacency matrix do
3:    N(v) = count entries in actual row
4:    if N(v) not in vertexDegrees then
5:       vertexDegrees.add(N(v))
6:    else
7:       continue
8: return vertexDegrees.size()
```

Algorithm 4 Largest vertex degree: $\mathcal{O}(n^2)$

Input: Adjacency matrix
Output: (Int) Largest vertex degree

```
1: vertexDegrees = []
2: for each row in adjacency matrix do
3:    N(v) = count entries in actual row
4:    if N(v) <= vertexDegree then
5:       continue
6:    else
7:       vertexDegree = N(v)
8: return vertexDegree
```

Algorithm 5 Smallest vertex degree: $\mathcal{O}(n^2)$

Input: Adjacency matrix
Output: (Int) Smallest vertex degree

```
1: int vertexDegree
2: for each row in adjacency matrix do
3:    N(v) = count entries in actual row
4:    if N(v) >= vertexDegree then
5:       continue
6:    else
7:       vertexDegree = N(v)
8: return vertexDegree
```

Algorithm 6 Average vertex degree: $\mathcal{O}(n^2)$

Input: Adjacency matrix
Output: (Double) Average vertex degree

```
1: averagegVertexDegree = 0
2: for each row in adjacency matrix do
3:    N(v) = count entries in actual row
4:    averagegVertexDegree += N(v)
```

5: **return** $\dfrac{averageVertexDegree}{|V|}$

Algorithm 7 Average neighbor degree: $\mathcal{O}(n^2)$

Input: Adjacency matrix, |V|
Output: Average neighbor degree

```
1: nodecount = const.value
2: nodedegree = [] //Array with size of nodecount
3: for each row in adjacency matrix do
4:    degree = count of ones in current row
5:    nodedegree[rownumber] = degree
6: for i = 0 to nodecount−1 do
7:    for j = 0 to nodecount−1 do
8:       if adjacencymatrix[i][j] then
9:          neighbordegree+=nodedegree[j]
10: avgnbdegree = neighbordegrees/nodecount
```

Algorithm 8 Graph density: $\mathcal{O}(1)$

Input: |V|, |E|
Output: (Double) density

1: density = $\frac{2 \cdot |E|}{|V| \cdot (|V|-1)}$

Algorithm 9 Graph centralization: $\mathcal{O}(1)$

Input: |V|, max $N(v)$, Graph density
Output: centralization

1: centralization = $\frac{|V|}{|V|-2} \cdot \left(\frac{maxN(v)}{|V|-1} - Graphdensity \right)$

Algorithm 10 Global *matching index* : $\mathcal{O}(n^2(n-1))$

Input: Adjacency matrix, |V|
Output: Global matching index

```
1: seti =[], setj =[]
2: similars = 0, allnodes = 0, paircounter = 0
3: matchingindex = 0
4: for each nodepair p_ij do
5:    seti = getneighbors(i)
6:    setj = getneighbors(j)
7:    similars = count(seti union setj)
8:    allnodes = seti.size+setj.size - 2·similars
9:    if allnodes > 0 then
10:      matchingindex+=similars/allnodes
11:   clear(sets)
12:   similars = 0
13:   paircounter++
14: matchingindex/=paircounter
```

Algorithm 11 Global *Clustering*-coefficient: $\mathcal{O}(1)$

Input: Adjacency matrix, |V|, |E|
Output: Clustering coefficient

1: clusteringcoefficient = $\frac{2 \cdot |E|}{|V| \cdot (|V|-1)}$

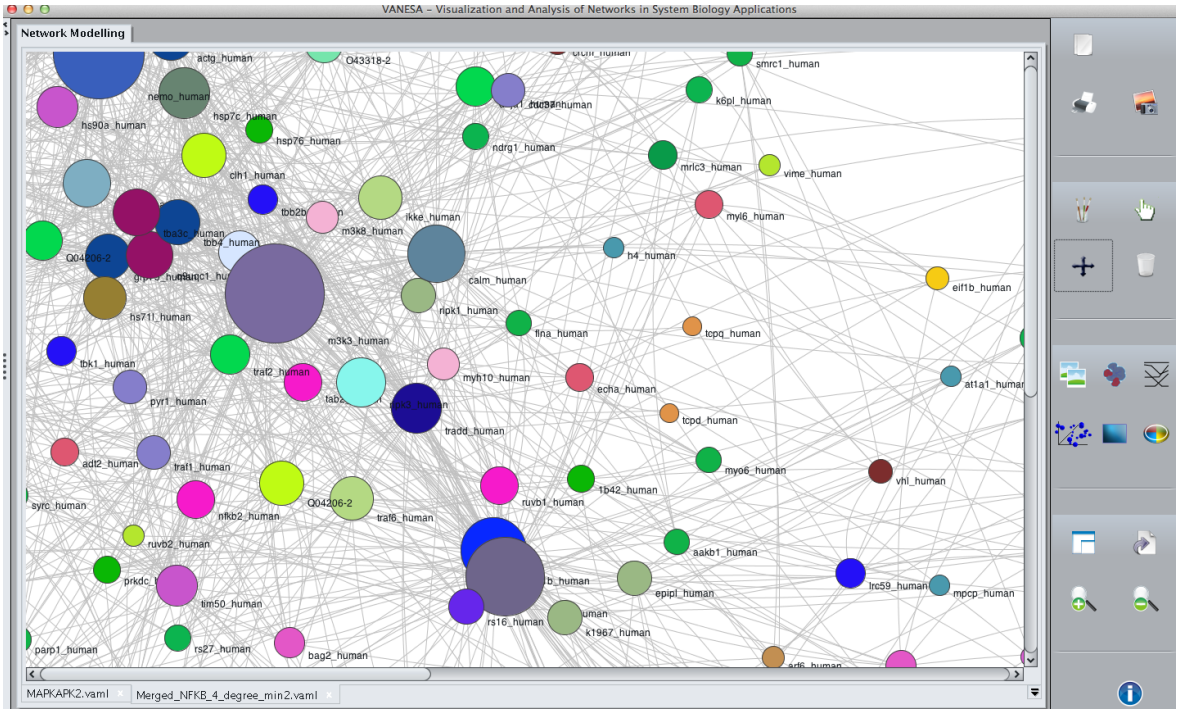


Figure 5.9: Biological hub detection measurement in a biological protein-protein interaction network in VANESA. Nodes with the most incident edges are highlighted. Nodes with the same vertex degree are colored in the same way.

properties, which only occur in special kinds of biological networks, such as a high graph density value in protein-protein interaction networks, can be easily identified. With this information, networks and motifs can be classified and characterized.

Additionally, all reconstructed networks can be compared with randomly generated networks. Therefore, VANESA offers the possibility to generate random, regular, bipartite, connected, and Hamilton graphs with a given node size. The graphs can be directed or undirected, as well as weighted. Therefore, the Barabási-Albert (BA) model is implemented, among others, which generates random scale-free networks [AB02]. With these networks, statistically meaningful comparative analysis can be performed, which is especially useful in theoretical biology.

5.6 Network comparison

To determine differences and similarities between a given set of networks, new graph comparison techniques have been implemented in VANESA. One of the realized techniques is the so-called "heat-graph" approach.

A heat graph is a graphical representation of a set of different networks, where the individual nodes are color-coded in accordance with their frequency of occurrence in the set of networks.

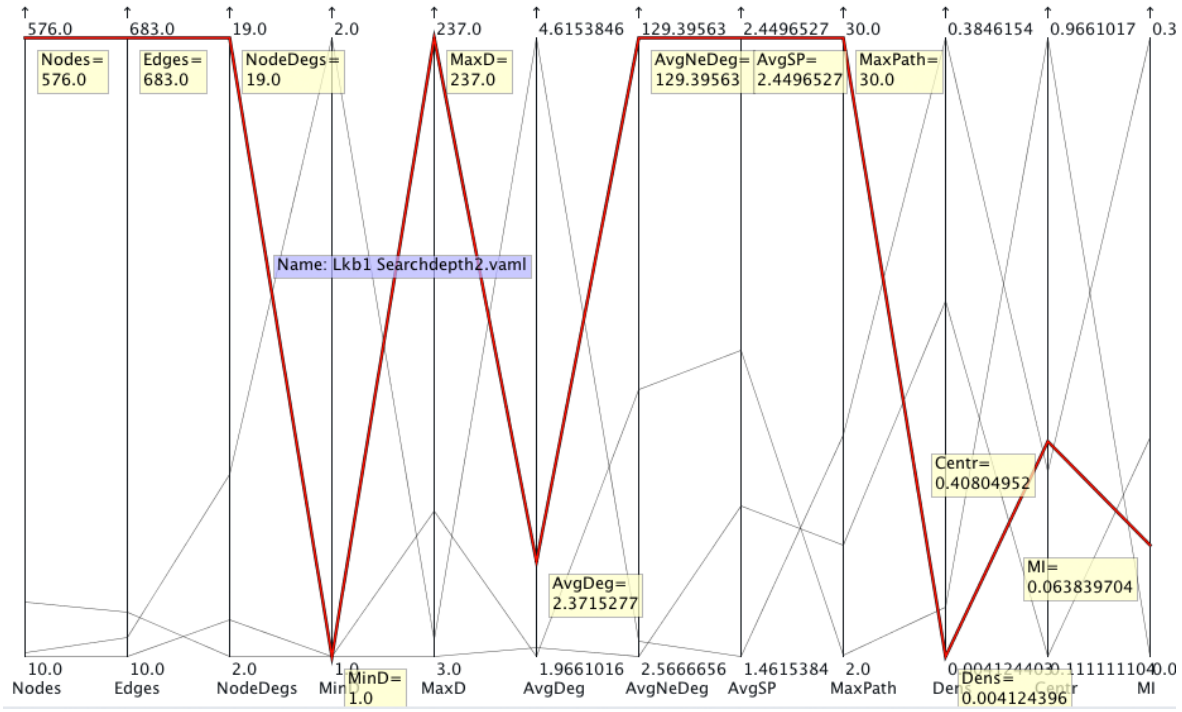


Figure 5.10: Comparison of a protein-protein interaction network with randomly generated networks, based on following centrality measurements: largest, smallest and average vertex degree, average neighbor degree, graph density, centralization, global matching index, and clustering coefficient, average shortest path, and maximum path length. Results are visualized in a parallel coordinate plot, which intuitively shows that a protein-protein interaction network differs in centrality values, such as average neighbor degree and average shortest path.

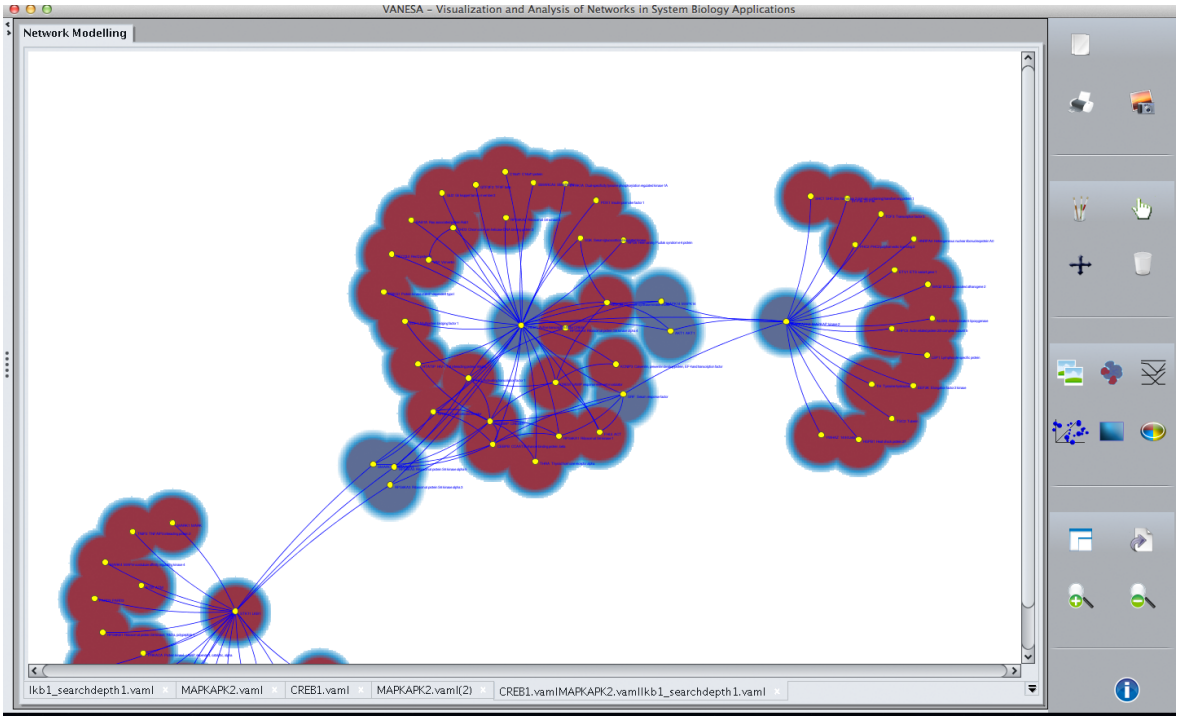


Figure 5.11: Heat-graph result for 4 biological networks. It is clearly visible that the backbone of the merged graph is constructed of proteins that appear in all networks (blue circle), whereas specialized proteins (red circles) influence the information and processing flow within the network.

The more often a certain node appears across the different networks, the more important it is in its function, and thus, color-coded by a large blue circle in the heat-graph approach. The less it appears within the set of networks, the more specialized it is in its function and thus, color-coded by small red circles (see Figure 5.11). However, colors, shapes, and outward appearance can be adapted by users.

The heat-graph is constructed by following four steps:

1. Consider a set S of graphs $G_1 \dots G_n$ containing equal subgraphs or nodes. Based on S create one merged graph G_m, consisting of all graphs in the set S (see Algorithm 5.6),
2. compute a vector where graph topological similarities over the set S are represented,
3. layout the merged graph,
4. paint the heat-graph using custom color mapping.

The comparison vector is constructed by identifying equal or similar topological structures in S. If topological similarities exist such as similar subgraphs, information about these structures is stored in a matrix/vector product, which is described in Definition 13.

Algorithm 12 Merge a set of graphs

Input: Set S consisting of graphs $G_1 \dots G_n$
Output: Merged graph: G_m

```
1: Initialize G_m = ∅
2: for G_i ∈ S do
3:    for v ∈ V_{G_i} do
4:       if v ∉ G_m then
5:          G_m = G_m ∪ v
6:    for e ∈ E_{G_i} do
7:       if e ∉ E_{G_m} then
8:          E_{G_i} = E_{G_i} ∪ e
```

Definition 13. *A comparison vector V is a vector containing elements v_1 to v_i where:*

$$v_x := \sum_{e \in G_m} \begin{cases} 1 & e = x \\ 0 & e \neq x \end{cases}$$

Finally, the heat graph is visualized. For each node in G_m a specific corona is painted. The color is defined by the function: $c(x, y) = max(c(x, y), DCF(v, x, v))$, where DCF is the distance correction function $DCF(v, x, y) = hammingwindow(x - v_x + R + y - v_y + R, 2R)$ with R = *circle radius defined by the user*. The distance function is used to create smooth transitions between overlapping coronas. Briefly, the hamming window function calculates a shape similar to that of a cosine wave for two overlapping circles. In the last step, users can choose color mapping that highlights results in different ways.

In addition to the heat-graph approach, VANESA offers the possibility to visualize overlapping biological networks in a 2.5D space. In this restricted three-dimensional representation each network is separately visualized. The common sub graphs are visualized on parallel two-dimensional planes (see Figure 5.12 and 5.13 for two examples). The method is similar to the heat-graph approach, although it uses other drawing aesthetics. In the heat-graph approach, overlapping parts are highlighted with different colors in one merged graph. In this approach, the intersections are visualized in the middle plane of all other networks. The advantage of this method is that by visual analysis, connections between different networks can be highlighted and simultaneously exposed in their differences. This approach is especially well-suited for large networks as it reduces the size of all networks to a subset of relevant overlapping subgraphs. Furthermore, users can visualize each network in 3D space, where they can navigate through the network, rotate it, and center as they wish. In addition, it is also possible to compare two networks with each other in 2D space, such as presented in Figure 5.14.

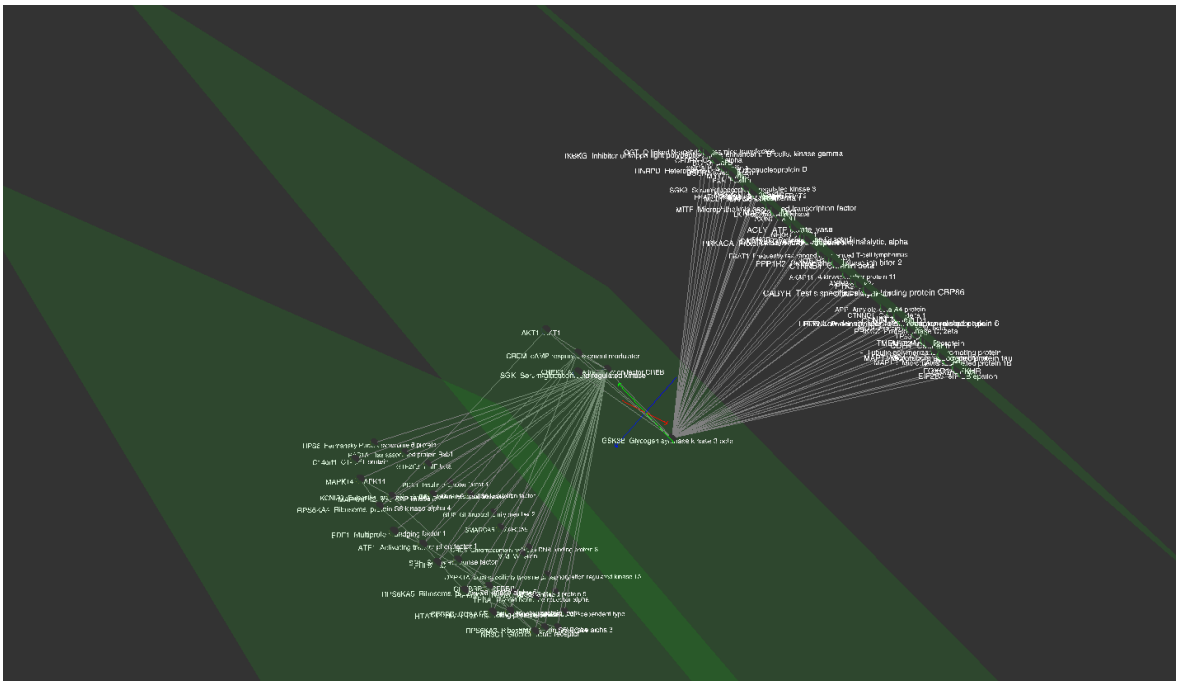


Figure 5.12: 2.5D comparison function in VANESA, where overlapping biological networks are visualized in a restricted three-dimensional space.

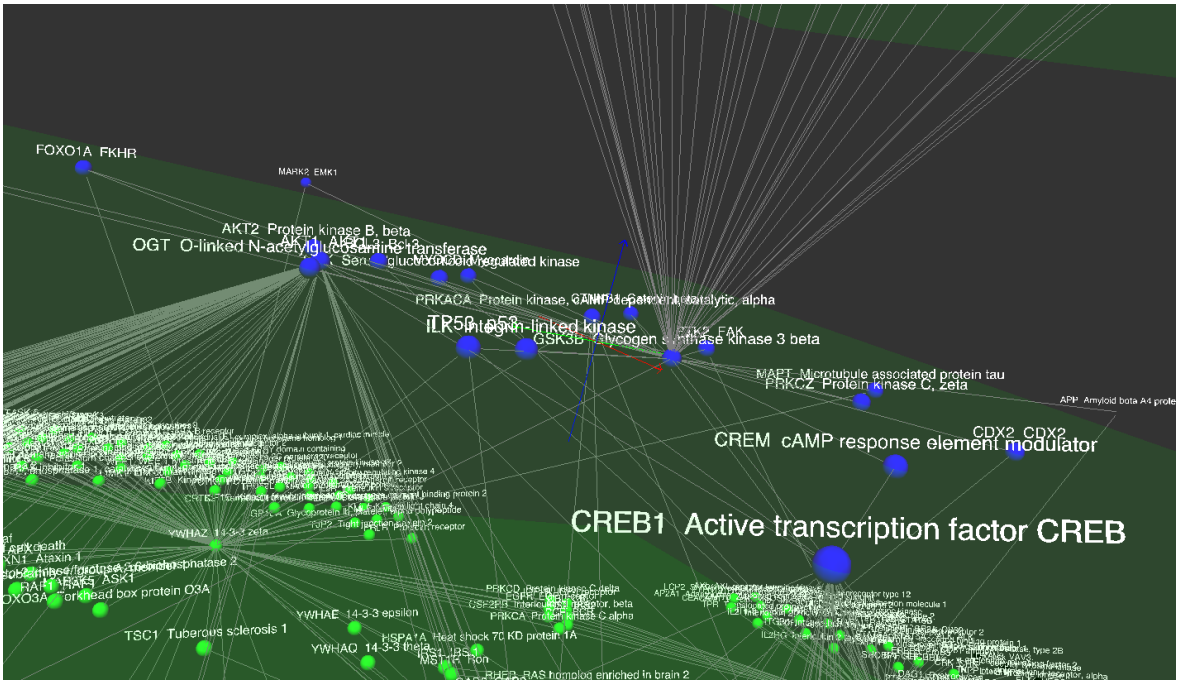


Figure 5.13: 2.5D comparison function in VANESA. A zoom-in of the middle plane, where overlapping parts of a set of protein-protein interaction networks are visualized.

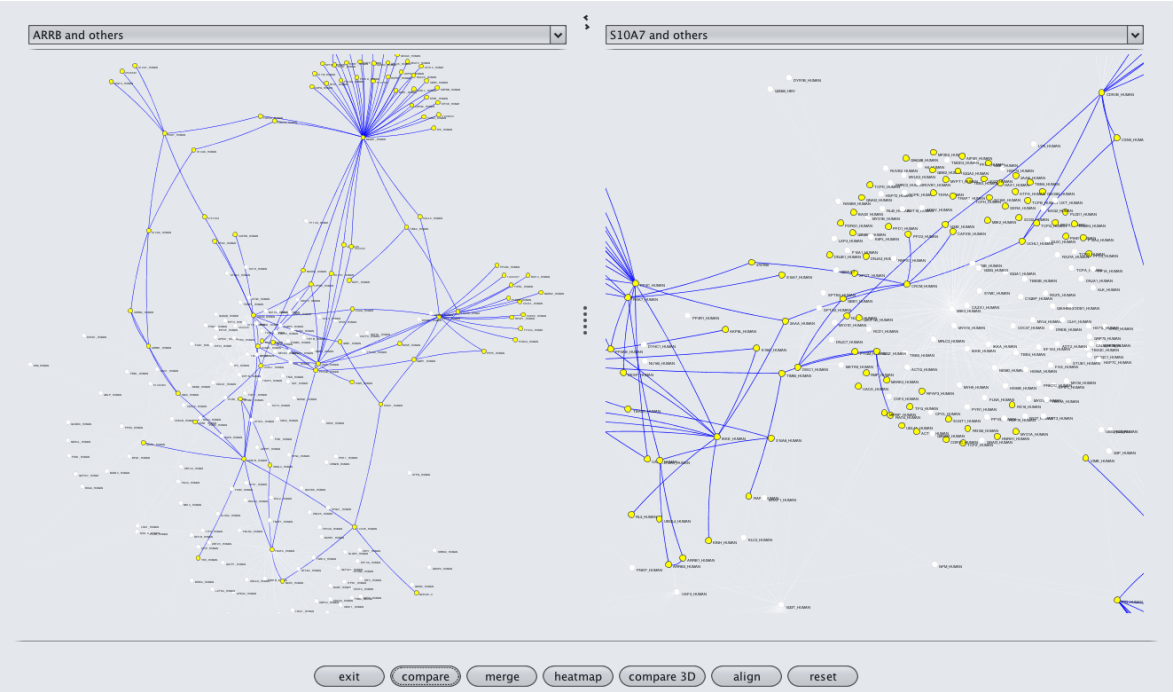


Figure 5.14: The figure shows two different regulatory networks being compared in terms of similarities and network structures within VANESA. Biological elements, which occur in both networks, are colored yellow. Regulatory processes taking place in both systems are highlighted with blue edges. Based on the results of the network comparison functions, scientists are able to focus on specific structures and elements, which they could visually examine within the visualization pane of VANESA.

5.7 Network visualization and interaction

One of the most important parts in biological network modeling is the representation of knowledge, data, and results. The aim of VANESA is to support biological scientists working at the bench. Data in different kinds of formats has to be transformed into pictures that can be interpreted by human beings. Solving a problem means not only to calculating results. It means representing results so that the solution is transparent and understandable. Results need to be rearranged to lead insights into a collection of data. The resulting representation needs to be simple in order to efficiently encode data.

In order to develop a simple and efficient user interface for VANESA, scientist's experiences and interactions were studied. The focus of the user interface design and human-system interaction is the reconstruction and analysis of biological networks at hand, without drawing unnecessary attention to the operational process. Therefore, a balance between technical functionality and visual elements had to be found to create a graphical user interface that is operational, usable, and adaptable to changing user needs. As a guideline for the design process, the concepts defined in the EN ISO 9241[3] standard for ergonomics of human-system interaction were used, particularly the sub-sections 10-14 for the design of dialogues between humans and information systems. Additionally to the EN ISO 9241 standard, there were many meaningful discussions with scientists from the laboratory to discern what users aim to do with biological networks and how the system should coincide with given research activities. A lot of prototyping and usability testing was necessary to get a good understanding of scientist's needs.

To see how visually presented information may affect people's understanding, special attention was paid to how researchers perceive, think about, and interact with biological graphs. The complexity of data can make analysis a challenging task. Scientific visualization must assist researchers in data analysis and speed up progress in having visual access to large quantities of data. Visualization can provide valuable assistance for data analysis and decision-making tasks [Fry08, TM04].

Therefore, experiences have been gained on eye-tracking in graph layout visualization, visual acuity, surrounding items, and color scales. These factors were studied as a basis for graph layout design (see Figure 5.15) to determine what catches the user's attention. Neuromorphic models were used to identify elements of a visual scene that are likely to attract attention. The iLab Neuromorphic Vision toolkit[4] was used for this purpose, which is based on the theory of computational modeling of visual attention [IK01]. As model input, different kinds of biological networks, as well as different networks sizes were tested. The neuromorphic model studies provide valuable insight into when, why, and whether specific visualization techniques provide effective cognitive support.

[3] http://www.iso.org
[4] http://ilab.usc.edu/toolkit/home.shtml

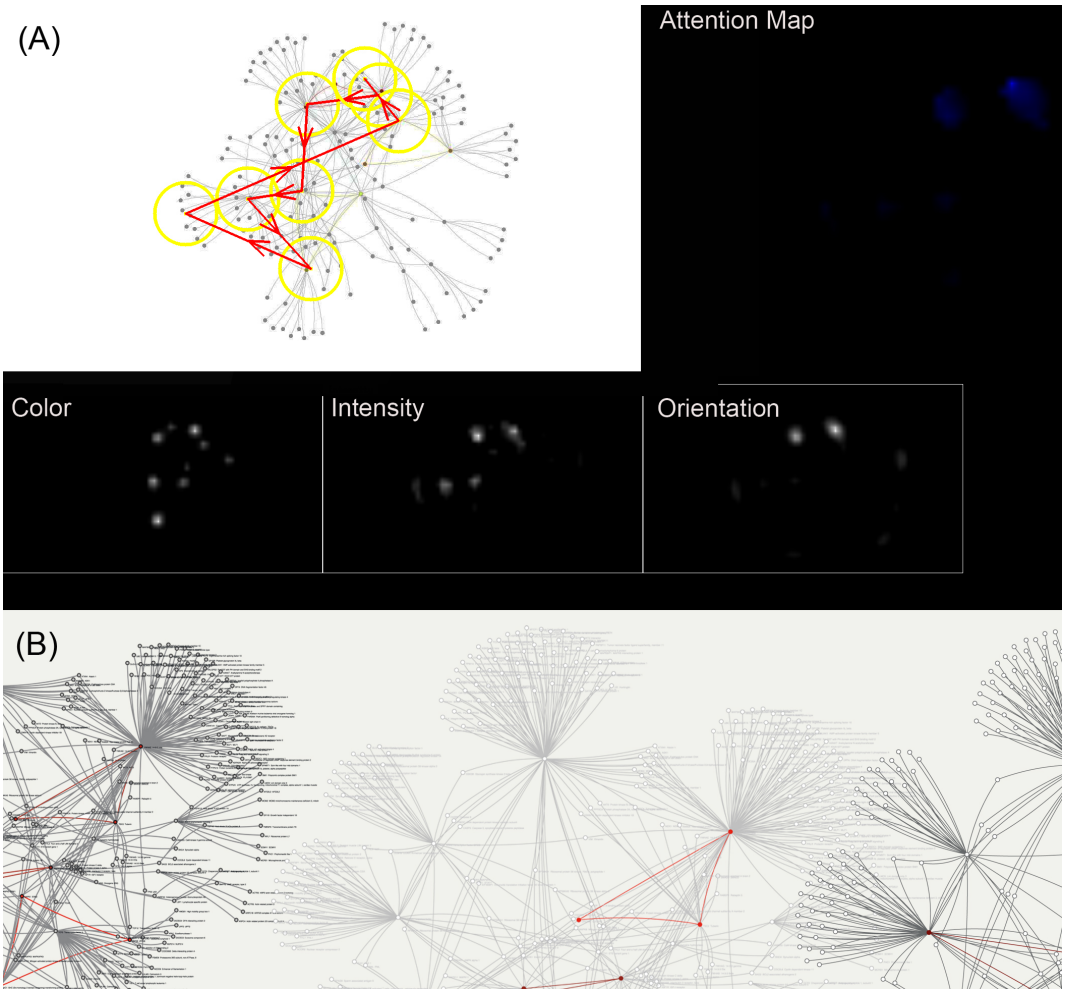


Figure 5.15: This figure shows one computational neuromorphic model study for the information visualization design of VANESA (A). During the study, the effect of color, intensity, and orientation on a middle-sized protein-protein interaction network was analyzed (B). The results have been used to provide valuable insights into why, when, and whether specific visualization techniques provide effective cognitive support for human observers. The aim was to find visualization techniques that assist researchers with cognitive support in providing information of the most relevant objects within biological networks. The starting point was the investigation of how to visually guide users in detecting the most important network motifs and elements within a graph.

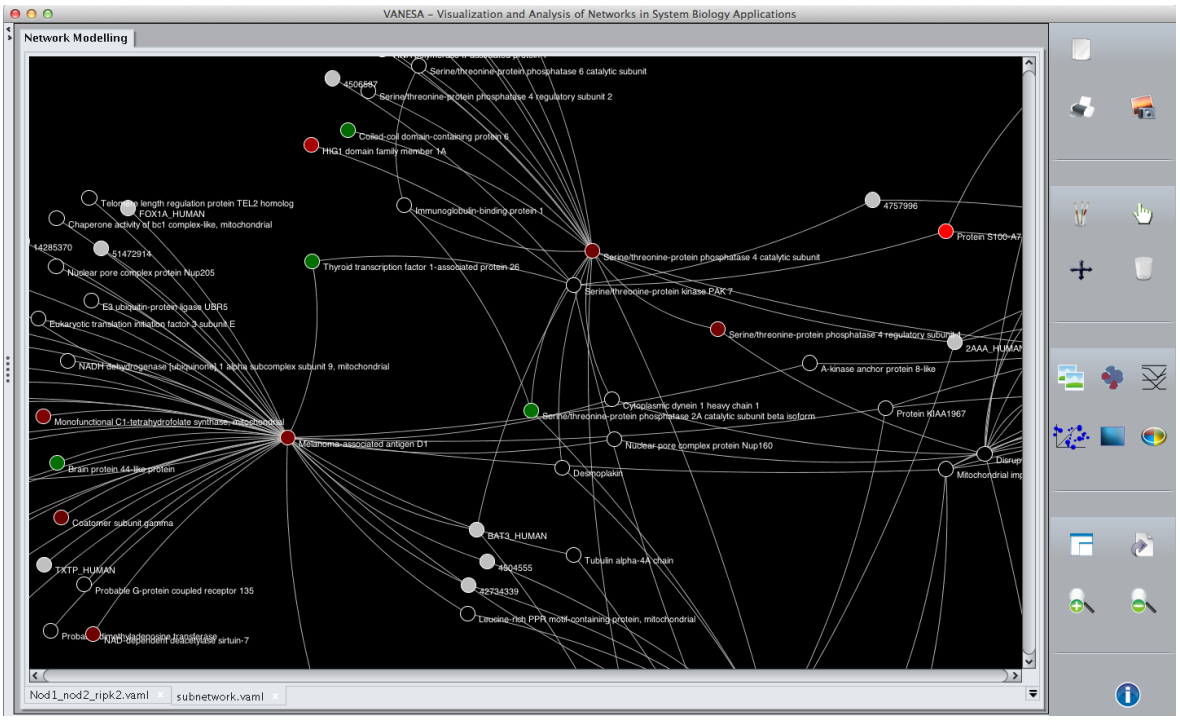


Figure 5.16: This figure shows how an inverted background and foreground increases contrast and the visual appearance of a network model. Users are able to focus better on network structures and model elements.

It was discovered that users get tired of examining a biological network after a certain time. Their eyes become stressed and their concentration decreases because of the size and complexity of data. The contrast between white background and black network visualizations particularly makes it difficult to focus on certain elements. A white background can become very aggressive after several minutes. Therefore, a function has been implemented which inverts background and foreground. Using a black background and grey network representation, users feel more comfortable and are able to identify structures and important elements much easier (see Figure 5.16). Besides, they spent more time on analysis and are more active in the interactive exploration of the system. Shades, contrasts, transparency, size, and network layout also make information more visually accessible. Especially network visualization is of great importance. Although the JUNG library offers several layouts, they are not appropriate for arranging large and complex networks. Therefore, an external java library was integrated, which includes a spring-embedded layout called GEM Layout [FLM94]. This network layout makes it possible to minimize edge crossing, edge breaks, considers node distance, min/max size of nodes, edge length, and tries to build up symmetry. This layout is always applied when users automatically reconstruct biological networks from database content.

Furthermore, VANESA supports zoom-in and zoom-out functions to aid in examining certain details. Rotate and stretch functions make the network interaction even more convenient. As biological networks can be large in dimension and size, an additional visualization viewer acts

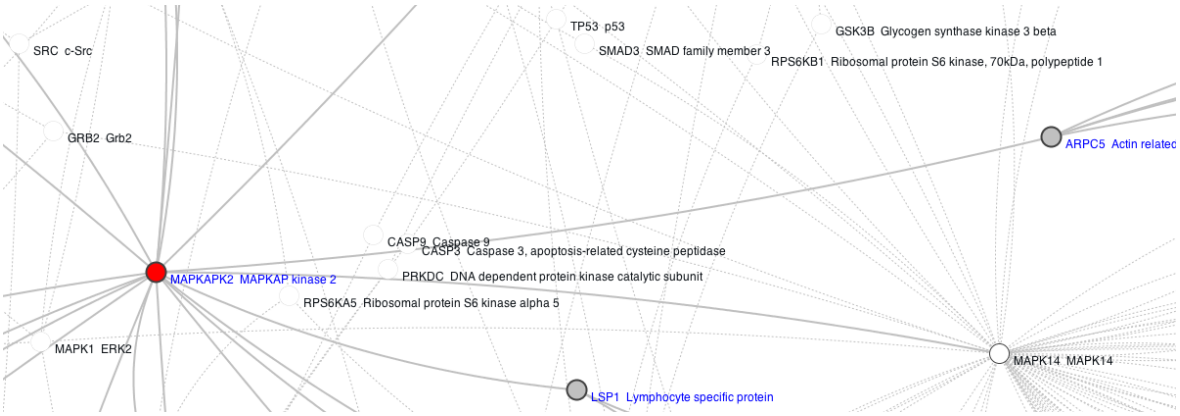


Figure 5.17: Selected elements within a biological network are highlighted in VANESA. This enables users to focus on important elements, rather than being overwhelmed with the whole network complexity.

as a satellite view, which helps users orient themselves in the model. In the satellite view, the full graph is always visible and all mouse actions affect the network in the main panel. A rectangular shape in the satellite view shows the visible bounds of the main panel. To make the interaction more intuitive, a search form enables users to find network elements. Therefore, an additional panel lists all network elements within the network. The elements are ordered by name and can be selected with a mouse click. Using this list, users can intuitively search for elements of interest and automatically center them in an animated way in the main panel. As a reconstructed biological model can consist of many different elements from various -omic levels, different elements can be displayed or hidden in the network representation in order to increase the clarity of the network representation. Thus, users have the possibility to focus on particular elements before going into the overall context of the model (see Figure 5.17). Furthermore, users can adopt the network representation. Each biological element has its own characteristics. VANESA offers a specialized form for each of the biological compounds, in which users can specify identifier, name, description, synonyms, and structural information, among others. Node color and shape can be also changed in order to highlight selected elements within the network. It is even possible to draw an entire cell and to place the network elements on drawn compartments, such as cell membrane, nucleus, and others.

5.8 Data exchange

For the exchange of models between different software tools, a script has been implemented, which converts the data structure presented in Definition 5 into the respective format of the selected exchange formats (see Section 3.6). For each format, the concepts, relations, and additional attributes of the models are analyzed, transformed, annotated, and validated, to ensure well-defined and correct results. In the example of the Systems Biology Markup

Language (SBML), the models are passed and checked within the SBML provided library called LibSBML. This ensures maximal compatibility and error checking, which guarantees valid models that can be exported, as well as imported. The same method applies to the Mathematical Markup Language (MathML). For the CellIllustrator Markup Language (CSML) and Modelica Exchange Format (.mo) adapted scripts have been implemented, since these formats do not support libraries for the creation and manipulation of biological models. Here, the ontology presented in Definition 7 is converted into the specific structure of the formats. The .txt file export is only based on the adjacency matrix. The export is realized by writing the matrix entries in the form of node-to-node relations.

In addition to the previous mentioned data exchange formats, users have the possibility to export their created models as JPEG or Scalable Vector Graphics (SVG) file formats. Therefore, new algorithms have been implemented to export the visualized network models as digital pictures. JPEGs can be used for basic graphic representations, as they are well-suited for displaying the overall model. However, in order to represent all details of a network model, line-drawing, fonts, and iconic graphics need sharp contrasts, without causing noticeable artifacts. This is also possible with JPEG files but increases the size of the file exponentially, especially when it comes to visualizing dense graphs. Furthermore, JPEGs are not well-suited to files that undergo multiple edits. For that reason, a SVG file format export has been implemented. Using the SVG export in VANESA, models can be represented in any size, as the network representations are saved as vector graphics. Thus, scientists can represent their results as a whole or as any picture cutout in high resolution. Moreover, the SVG pictures can be easily edited, manipulated, shifted, and cropped, with any imaging software application. This is important, since scientists need the possibility to visually highlight, combine, or annotate certain details in their models before publishing them.

In order to import biological data that can be mapped on an existing network, an easy-to-use import function has been realized. It is possible to map microarray results on a database created network as shown in the cholesteatoma application case (Figure 6.1 on page 127). Therefore, the text file has to be constructed with a list of nodes and list of edges. After each node additional parameters can be specified, such as description and experimental value. After each edge, which is specified by a from-to-to relation, the direction can be optionally specified. One example for such a file is shown in the following:

```
#Nodes
BIRC2#          Apoptosis inhibitor 1;                      1.0
LRRC1#          Leucine rich repeat containing 1;           0.67
...
#Edges
BIRC2;TRAF2;    FALSE
RIPK2;CASP8;    FALSE
...
```

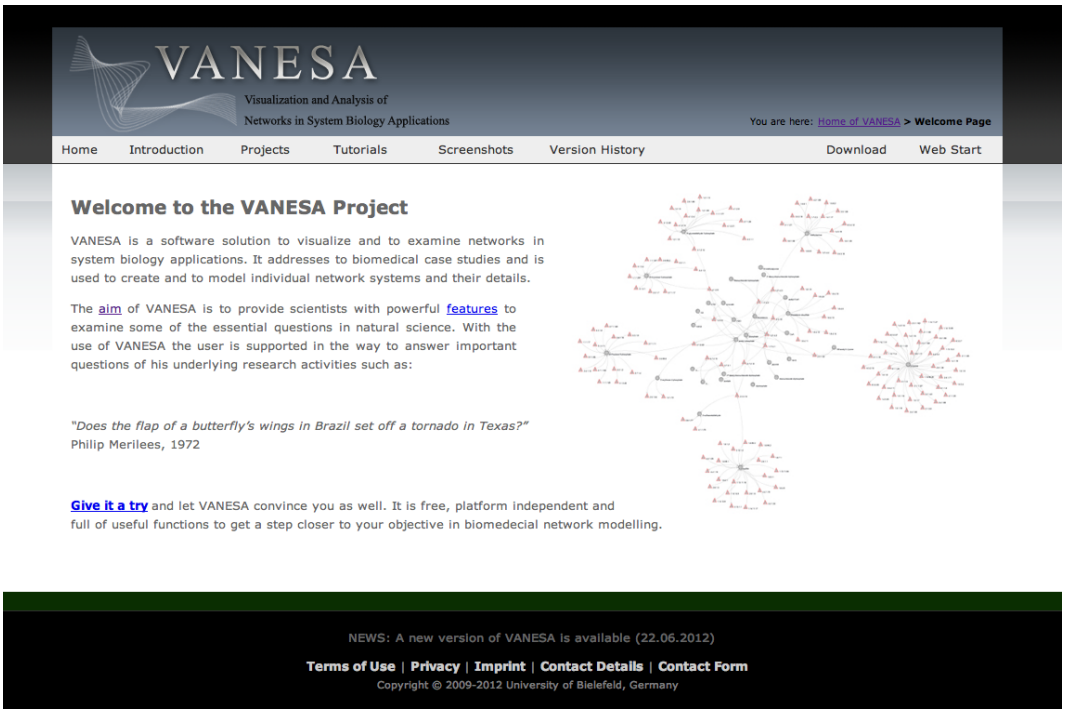


Figure 5.18: This picture shows the webpage of VANESA. At www.vanesa.sf.net all users can instantly start VANESA via web start or find information on the aims, the software design, features, as well as ongoing projects and cooperations.

Based on this data structure, VANESA is able to read in the information and reconstruct the network with the corresponding experimental values. Therefore, a color gradient is used to intuitively present the experimental results. The gradient goes from bright green (small value) to black (value between 0.8 and 1.2) up to bright red (high value) for microarray mapping.

5.9 Feature Summary

As presented in the aforementioned sections, VANESA offers a wide range of useful functions. All presented features together create a powerful framework which is able to reach the initial goal of VANESA, namely to reconstruct, model, and simulate biomedical networks in a new sophisticated network based approach.

In order to make VANESA available to all users, a web page has been created in preliminary work [Jan09], describing the software application and moreover, providing a web start possibility to instantly start VANESA. Thus, VANESA is available to all users at www.vanesa.sf.net (see Figure 5.18). The webpage provides information on the aims, the software design, features, as well as ongoing projects and cooperations. A tutorial and screenshots provide an introduction into the software application. The version history keeps user up-to-date about new software releases. To the best of our knowledge, the terms of use and privacy have been written with

the attempt to comply with all legal obligations. However, VANESA is free-of-charge for any academic or private use. Commercial use is restricted and subject to a royalty obligation. In addition to the previously mentioned information, a link to a specially created SourceForge [5] project is provided to download the latest programming code of VANESA. The main advantage of SourceForge is that other developers can join the programming of VANESA and so doing, help in improving and extending the software application.

To remind all reader of the various features provided in VANESA, the following listing summarizes all functions, grouped into topic and research areas:

- **Life-science database consulting:**
  - Access to the high quality data warehouse DAWIS-M.D.
  - Automatic reconstruction of protein-protein interaction networks and signaling networks from the databases HPRD, MINT, and IntAct.
  - Automatic reconstruction of metabolic networks from the BRENDA database.
  - Automatic reconstruction of metabolic and gene-regulatory pathways from the KEGG database.
  - Sophisticated search form for each database, in which users are able to use Boolean operators in their queries to narrow search.
  - Possibility to load whole pathway maps or only networks with a limited size.
  - Possibility of neglecting currency metabolites in the automatic network reconstruction.
  - Organism specific search.
- **Petri net simulations:**
  - Petri net simulation in the xHPNbio formalism in the PNlib of Modelica.
  - Petri net simulation in the software application CEllIllustrator.
  - Possibility to model qualitative, stochastic, continuous, hybrid, and functional Petri nets.
  - ODEs can be integrated into continuous Petri net models.
  - Automatic translation of biological networks into the Petri net language.
  - Basic Petri net validation checker to prevent model errors.

[5] SourceForge is a centralized location for software developers to control and manage free and open source software development.

  - Automatic simulation within one active window.
  - Token animations and charts to present simulation results.
  - Export of simulation charts as tables and/or jpeg-files.

- **Petri net analysis:**
  - Basic t-invariant calculation.
  - Basic p-invariant calculation.
  - Construction of a coverability graph.
  - Construction of a reachability graph.

- **Graph theory and centrality measurement:**
  - Hub detection based on node degree and average neighbor degree.
  - Shortest path calculation and visualization.
  - Possibility to determine the largest, smallest and average vertex degree within a network, as well as the average neighbor degree.
  - Possibility to calculate global network properties, such as graph density, centralization, global matching index, and clustering coefficients.
  - Calculation of all shortest and maximum paths, as well as the average lengths.
  - Possibility to compare a set of different networks within a parallel coordinate plot with regard to the aforementioned centrality measurements.
  - Possibility to generate random, regular, bipartite, connected, and Hamilton graphs.
  - Animation of centrality measurements.

- **Network comparison based on topological similarities and biological identifier:**
  - Comparison of a set of networks based on the heat-graph approach.
  - Comparison of a set of networks in 2.5D space.
  - Comparison of two networks in 2D space.

- **Network interaction and visualization:**
  - Network representation in 2D space.
  - Network representation in 3D space.

  - Possibility to merge two graphs with each other.
  - Different graph layouts, such as the spring-embedded GEM layout.
  - Sophisticated network editing and manipulation functions.
  - Users are able to invert foreground and background.
  - Zoom-in / Zoom-out functions.
  - Rotate and stretch functions.
  - Copy and paste functions.

- **Data exchange formats:**
  - SBML export and import support.
  - MathML export.
  - CellML export.
  - Basic txt-network export and import support.
  - Experimental data import and mapping by a txt-network file.
  - JPEG and SVG export.

- **Graphical user interface:**
  - Satellite view to get an orientation in large networks.
  - Possibility to hide certain biological elements.
  - Possibility to search for graph elements, which can be centered in an animated way.
  - Possibility to divide biological networks into biological compartments, such as nucleus, mitochondrion, among others.
  - Possibility to change color and shape of biological objects.
  - Auto-suggest function for the naming of biomedical elements.

Chapter 6

Application cases

VANESA provides a new state-of-the-art framework that can be used in a wide area of applications. To show how powerful this framework is, this chapter presents some selected application cases in which the software application was used as a valuable tool for the reconstruction, analysis, and prediction of biological systems. The first two sections deal with the use of VANESA in clinical studies based on the human experimental series. The first section presents how scientists from medicine and fundamental research used VANESA to reconstruct protein-protein and gene regulation networks to examine cholesteatoma-related genes. In addition, it is shown how scientists used the graph theoretical framework to identify the most important actors within the biological networks, which were further investigated in experimental studies. The second section deals with the investigation of the dilated cardiomyopathy disease. Here, VANESA was used to reconstruct signaling-pathways having a major role in cardiovascular diseases, among others. The aim was to identify important regulatory elements and structures in perturbed cardiovascular diseases pathways, which can be further investigated in the laboratory.

Since VANESA is not only limited to clinical studies, it can also be used in answering fundamental questions in molecular biology. Section 6.3 shows how the NF-κB system was reconstructed and simulated using VANESA. The aim was to model and simulate NF-κB dynamics as they appear in stem cells from mice. It is shown, how life-science database information and data collected from literature were used to reconstruct the dynamic system and how it was automatically simulated using hybrid Petri nets in VANESA. Scientists from neurobiology and cell-biology performed the system reconstruction, whereas computer scientists realized the hybrid Petri net modeling. This demonstrates a further powerful feature of VANESA, namely, the possibility to cooperatively work on one model with scientists from different research fields. Furthermore, VANESA was used in biotechnological research applications as described in Section 6.4. In the modeling of pathogen bacterial cell-to-cell communication, VANESA demonstrated its ability to model and simulate population dynamics. The models were reconstructed with information from several different life-science databases and exported to the Petri net simulation software

CellIllustrator for simulation. The last section summarizes the presented results and highlights the importance and usability of VANESA.

6.1 Identification of novel cholesteatoma-related genes

Cholesteatoma is a potentially life-threatening middle-ear disease [ST07]. First reports were presented in 1972 by Lim and Saunders [LS72], who performed a detailed histology of cholesteatoma. Since then, it is investigated in detail and characterized as a gradually expanding destructive epithelial lesion, which can cause extensive local tissue destruction in the temporal bone [HdHTDG07, SGTS11]. First of all, clinical symptoms are conductive hearing loss and later on sensorineural hearing loss, vertigo or facial palsy, and infection through the tegmen of the middle-ear. This infection can in fact, result in meningitis or an intracranial abscess. In 2010, Nunes *et al.* reported that the annual incidence of cholesteatoma revolves in approximately 3 out of 100,000 cases in children and 9 out of 100,000 cases in adults [NdBC+10].

In order to identify novel cholesteatoma-related genes, scientists from medicine, molecular biology, neurobiology, and bioinformatics started to investigate the middle-ear disease in detailed experimental molecular studies, in which VANESA was used as the major bioinformatics tool for the biological system modeling and analysis [KJB+12]. The aim was to identify and reconstruct protein-protein interaction networks which might describe the transition of a healthy system into an altered one that causes the development of cholesteatoma. So far, the complex interaction networks for cholesteatoma are unknown. The approaches which show small regulatory motifs are unable to help in answering fundamental questions. Moreover, information about this disease is distributed over many different databases and other repositories, where data is not interconnected. Here, VANESA was used to automatically collect knowledge from different important databases to reconstruct a sophisticated biological model which can be enriched with experimental findings and analyzed with graph theoretical approaches, for further experimental investigations.

At the onset of this study, differentially expressed genes in human cholesteatoma in comparison to healthy external auditory canal skin were investigated. Therefore, whole human genome microarrays, containing 19,596 human genes, were used to identify significant differentially expressed genes. This approach has already proved very useful as reported by Kwon *et al.* [KKKJ06] and others [YKW+06, EPN00]. The microarray analysis was performed with an in-house R-statistic-software-based analysis pipeline, which includes some of the most important Bioconductor[1] software packages. Primarily, the Linear Models for Microarray Data (LIMMA) package [SGC+05] was used to normalize and identify the most significantly expressed genes. Therefore, a background correction was performed [RSO+07], followed by a Loess-normalization within the arrays and across the different samples [SS03]. In the last step, linear models were

[1] www.bioconductor.org

Up-regulated processes	Involved genes	Down-regulated processes	Involved genes
Signal transduction	162	Signal transduction	72
Cell communication	156	Cell communication	69
Protein metabolism	96	Cell growth and/or maintenance	44
Cell growth and/or maintenance	70	Energy pathways	25
Immune response	66	Metabolism	25
Energy pathways	65	Immune response	13
Apoptosis	6	Carbohydrate metabolism	2
Anti-apoptosis	5	Cell adhesion	2
Inflammatory response	5	Apoptosis	1
Pyrimidine salvage	3	Cell differentiation	1

Table 6.1: Some of the most important affected processes in cholesteatoma based on expression values derived from analyzed whole human genome microarrays.

applied to perform hypothesis tests such as (log2) fold changes, standard errors, t-statistics and p-values. Using these methods and statistics, the most significant differentially expressed genes within seven selected human samples were identified, resulting in 766 up-regulated and 369 down-regulated genes in cholesteatoma. Table 6.1 lists some of the most important regulated functions with the number of identified regulated genes, which were are at least twice as high as in the cholesteatoma samples.

Based on the list of up- and down- regulated genes, the aim was to identify motifs and regulatory structures that turn a healthy biochemical system into a cholesteatoma. Therefore, VANESA was used to reconstruct the underlying biological networks. The initial point for the network reconstruction was a list of about 20 hand-selected genes showing a strong differentially expression pattern in the cholesteatoma. For each of the selected genes, protein-protein interaction and signaling networks were reconstructed with information derived from the databases IntAct [KAB+12], HPRD [KPGK+09], and Mint [LBP+12]. This resulted in a set of biological networks containing the direct interaction partners and other nearby biological elements. In general, each of the networks is constructed of 15 up to 200 biological elements. Based on these networks, medical scientists started to compare and investigate the different reconstructed networks in VANESA to identify significant regulatory motifs and structures. Therefore, the network comparison function of VANESA was used, which highlighted similarities and differences between the analyzed networks. Having a notion about the relevant elements and structures,

the networks were reduced to the relevant parts and merged into a global network, containing the significant structures and elements of the initial networks.

In the next step the microarray results were mapped on the analyzed and filtered networks using the microarray-fold-change import function in VANESA. Each of the nodes within the networks was colored with regard to its fold-change, showing the biological regulatory effect in the system. The graph theoretical environment with its hub detection and other centrality measurements highlighted the most significant elements. One example of such a network is presented in Figure 6.1. This signaling network represents the S100 interaction network. Proteins are the nodes and the edges are protein activation/inactivations, such as phosphorylation and dephosphorylation across a set of proteins. The colors of the nodes represent the microarray expression levels. The network shows the correlation of the up-regulated S100A7, S100A8, and S100A9 genes, as well as the ILK, IκB, USF2, and ARRB2 genes. In addition, three other signaling networks were reconstructed revealing regulatory motifs for cytokeratin and different matrix metalloproteinases, among others.

Due to these reconstructed and analyzed models, scientists were able to identify genes potentially involved in cholesteatoma development, and furthermore, regulatory motifs and structures which were so far unknown. They investigated all elements within the network showing a high fold-change which were closely connected in the reconstructed signaling pathways. This step was performed in the network visualization pane of VANESA, where the scientist could visually and interactive examine the reconstructed systems. After selecting the most promising motifs, selected genes within these regulatory structures were further analyzed with the Real-Time Polymerase Chain Reaction (RT-PCR) to prove that the reconstructed regulatory networks were correct. Therefore, the following genes, which revealed a much higher expression in cholesteatoma and are part of signaling modules, were investigated: genes involved in metabolism activity, such as matrix metalloproteinases, PI3, SERPINB3, and SERPINB4, genes involved in cell growth and/or maintenance activity, such as SPP1, KRT6B, PRPH, SPRR1B, and LAMC2, genes involved in signal transduction such as LCN2, GJB2, and CEACAM6, and genes involved in cell communication processes such as CDH19 and genes belonging to the S100 family. In addition, relevant genes involved in the regulation of nucleobase, nucleoside, nucleotide and nucleic acid metabolism such as TFAP2B, ID4, and PAX3, genes responsible for immune response such as SP5, FGFBP2, and CXCL1, and seven other apoptose and anti-apoptose relevant genes were examined. In summary, in almost all cases PCR proved the identified regulatory motifs and models reconstructed in VANESA correct (see Figure 6.2).

The study revealed even more potentially involved genes in the cholesteatoma development. After analyzing and putting all results into relationship, it was possible to demonstrate that the expression profile of cholesteatoma is similar to a metastatic tumor and chronically inflamed tissue. Furthermore, the reconstructed biological networks, which include cholesteatoma-regulated transcripts are a valuable new framework for drug-targeting and therapy-development. These regulatory networks enriched with experimental findings are, to the best of our knowledge, the

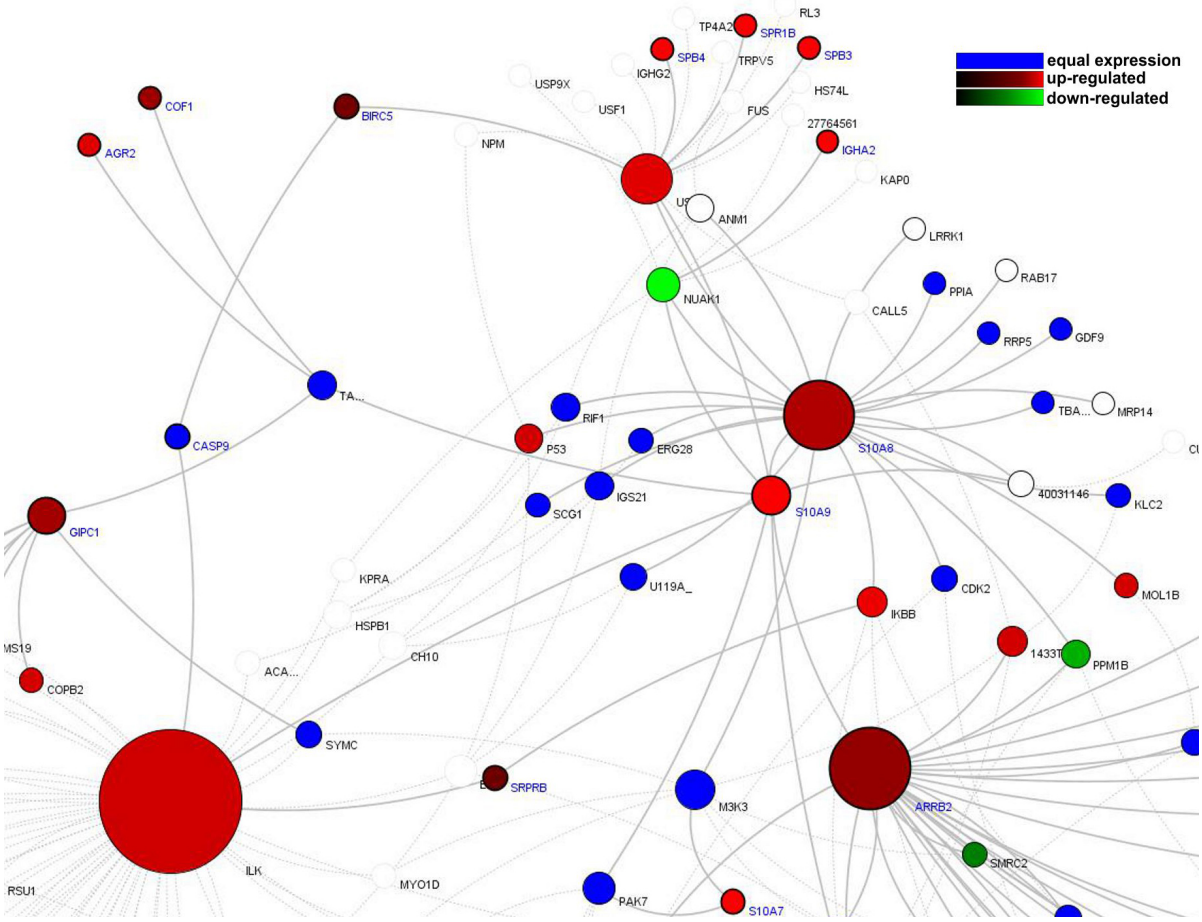


Figure 6.1: Reconstructed S100 protein-protein interaction network with its regulatory motifs in VANESA. The network shows an up-regulation of S100A7, S100A8 and S100A9 as well as ILK, IκB, USF2, and ARRB2. Centrality measurement highlights the most prominent actors within the network (the larger the circles, the more important/prominent the elements). Colors represent the expression values of the mapped microarray experiments (red= up-regulated, green= down-regulated, blue= equal expression, white= not investigated in microarray experiments).

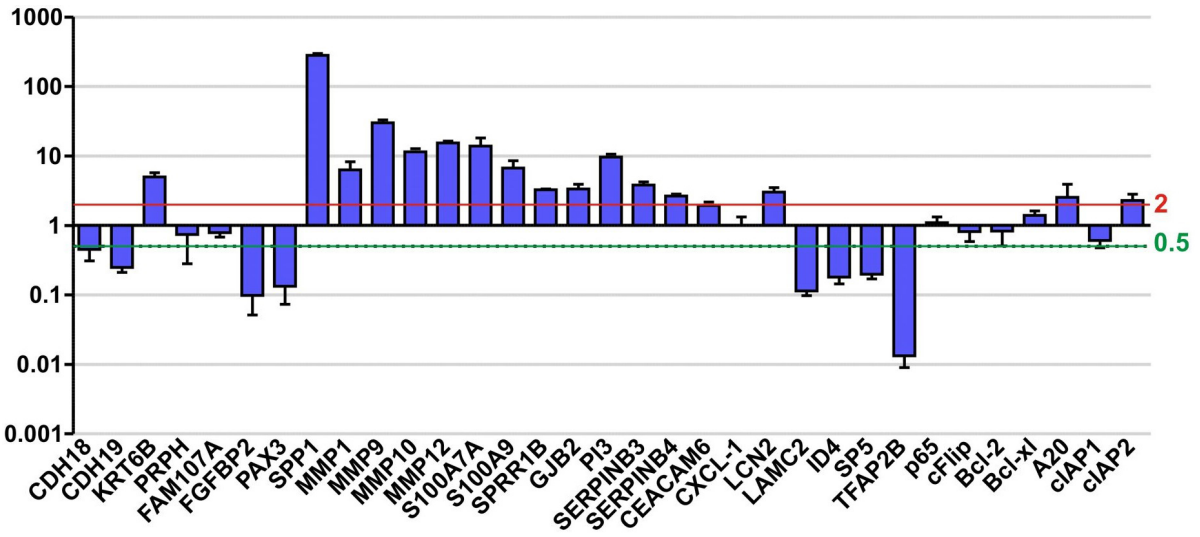


Figure 6.2: RT-PCR analysis of potential transcripts that might be involved in major processes in cholesteatoma. The figure shows the expression of different genes in external auditory canal skin and cholesteatoma. In summary, metalloproteinases (MMPs) and their substrates are highly up-regulated in cholesteatoma; transcripts that encode e.g. for tumor suppressors are down-regulated.

first ones, which can help scientists from medicine and biology to identify molecular switches turning a healthy system into a unhealthy one. Further biological analyses on these networks are ongoing and already show new findings, which are under experimental investigations.

6.2 Investigation on the dilated cardiomyopathy disease

Most industrial countries face high and increasing rates of Cardiovascular Diseases (CVDs), that by now are some of the leading causes of death. A detailed study in 2006 showed that CVDs, as an underlying cause of death, has accounted for 34,3% of all deaths in the United States (data is provided by the American Heart Association: Heart Disease and Stroke Statistics - 2010 Update[2]). CVDs and all other heart diseases and failures accounted for about 56% of all deaths in 2006. Still, the mortality statistics remain similar and therefore, a lot of attention has been focused on the metabolic aspects of CVDs related pathways. The aim is to discover new CVDs specific molecular targets to promote the investigation of protein's functional roles in their specific biomedical pathway.

Based on a project founded by the European Union [CA08], scientists from biology and bioinformatics used VANESA to reconstruct associative metabolic and protein-protein interaction networks derived from misleading proteins in an experimental CVDs sample case [KHA+10]. The research object was a dilated cardiomyopathy (DCM) case of a female patient with renal

[2]http://www.americanheart.org/presenter.jhtml?identifier=3000090

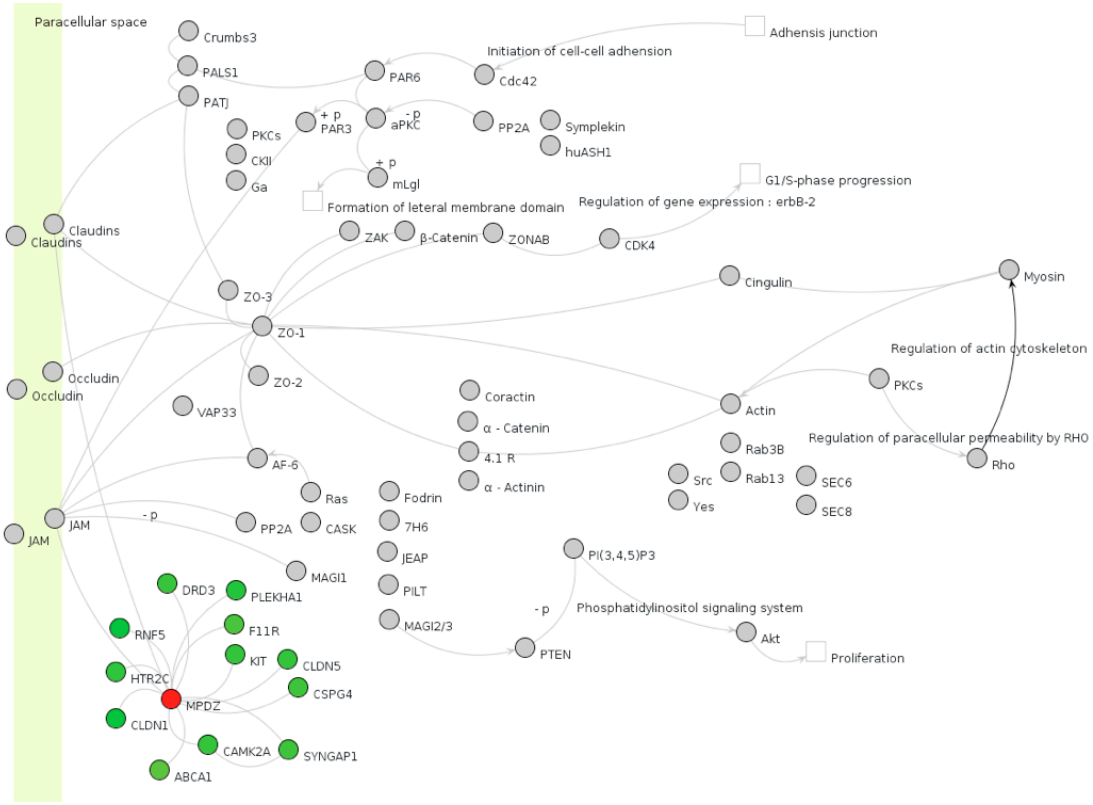


Figure 6.3: A reconstructed MPDZ signaling pathway from VANESA, based on database information from KEGG and HPRD. Green colored nodes represent important regulatory elements having a major role in cardiovascular diseases.

insufficiency. DCM is a condition in which the heart becomes weakened and enlarged, and cannot pump blood efficiently. In order to understand the biomedical system and to find new starting points for experiments and therapies, metabolic pathways, enriched with information about protein-protein interaction, should be modeled.

From a set of experimentally relevant identified proteins, interaction networks had to be reconstructed within VANESA. One of the experimentally investigated proteins was MPDZ, a holding protein showing a large diversity of interacting proteins [USFL98]. Using VANESA, the biochemical environment of MPDZ was investigated in detail by reconstructing metabolic and protein-protein interaction networks based on data from the databases KEGG [KGS+12] and HPRD [KPGK+09] (see Figure 6.3). The main goal was to identify involved metabolic pathways and furthermore, proteins related to the DCM case that interact with MPDZ. In the first case, VANESA was used to find and reconstruct all KEGG pathways containing the MPDZ protein. These networks were then automatically expanded with information from the database HPRD. By analyzing these networks, molecular scientists were able to identify twelve proteins which are major regulation elements in the human tight junction signaling pathway and related to the DCM case. All-in-all, it was possible to link the following pathways with MPDZ and the DCM case: dilated cardiomyopathy with renal insufficiency, diabetes type 2,

Common perturbed pathways	Protein names	Swissprot code
Tyrosin metabolism (hsa00350)	Catechol-O-methyltransferase	P21964
Basal transcription factor (hsa03022)	General transcription factor II, General transcription factor IIB	O15359, Q00403
Protein export (hsa03060)	Signal recognition particle 54kDa	P13624
MAPK signaling pathway (hsa04010)	Arrestin beta 1	P49407
Cell cycle (hsa04110)	Cyclin A1, MCM6 minichromosome maintenance deficient 6, Cyclin-dependent kinase 7, Cyclin-dependent kinase inhibitor 1C	P20248, Q14566, P50613, P49918
Ubiquitin mediated proteolysis (hsa4120)	Ubiquitin-conjugating enzyme E2I (UBC9 homolog, yeast)	P50550
Focal adhesion (hsa04510)	Caveolin 1, Caveolae protein, 22kDa	Q03135
Antigen processing and presentation (hsa04612)	Heat shock 90kDa protein 1, Alpha calnexin	P07900, P27824
Small cell lung cancer (hsa05222)	TNF receptor-associated factor 4	Q14848
Porphyrin and chlorophyll metabolism (hsa00860)	Heme oxygenase (decycling) 1	P09601
Neuroactive ligand receptor interaction (hsa04080)	Nuclear receptor subfamily 3, group C, member 1 (glucocorticoid receptor)	P04150
Insulin signaling pathway (hsa04910)	Flotillin 2	Q14254

Table 6.2: Summary of proteins identified in the reconstructed networks from VANESA which have an important role in perturbed CVDs pathways.

and pulmonary disease, among other various perturbed pathways as listed in Table 6.2. Thus, scientists from biology and medicine were provided with predicted gene-controlled and protein-protein interaction processes for the discovery of novel biomarkers and unknown therapeutic targets. Furthermore, the research progress allowed the investigation of the biological functionality of DCM related genes, among others.

In further bioinformatic studies, VANESA, the 3D visualization tool CELLmicrocosmos 4.2 Pathway Integration (CmPI) [SKS+10], and the text-mining tool Associative Network Discovery visualization (ANDvisio) [DIKI10], were used to reconstruct an abstract cell environment for the investigated CVDs systems [STK+10]. Related to the real localization of the signaling pathways within a true cell it was aimed at gaining insights into the functional as well as spatial interrelationships of MPDZ (see Figure 6.4). Therefore, experimentally gained knowledge was used to reconstruct a virtual cell environment in the tool CmPI based on the reconstructed

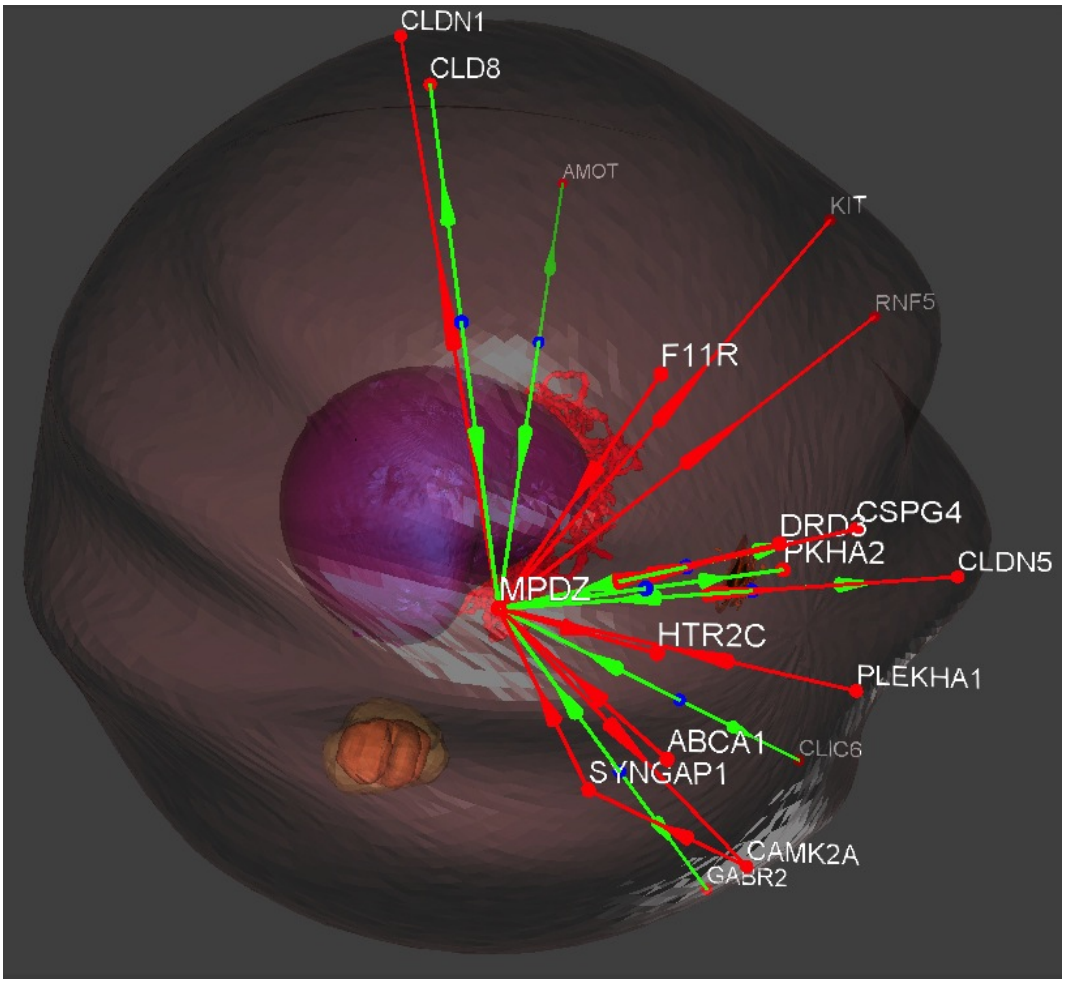


Figure 6.4: The MPDZ signaling pathway visualized in an abstract virtual cell environment in the software application CmPI, based on information derived from VANESA and the text-mining tool ANDvisio. Proteins are placed due to their real cell location and information flow within a cell.

networks from VANESA and localization information from the text-mining tool ANDVisio. This approach demonstrates that with experimentally derived data, data mining, text mining, and data fusion, information flow and metabolic processing can be reconstructed in 3D space. This extends the usual approaches in 2D network modeling, as new insights can be gained on cell localization.

6.3 Modeling the NF-κB system

The Nuclear Factor 'kappa-light-chain-enhancer' of activated B-cells (NF-κB) transcription factor is one of the most investigated transcription factors in humans and animals. NF-κB is involved in anti-apoptotic signaling, neuroprotection, learning and memory, cancer, and innate

immunity. The NF-κB pathway is a ubiquitous stress -response that activates the NF-κB family of transcription factors [WD10]. Antigen receptors, receptors of the innate immune system, and certain intracellular stressors are potent activators of this pathway. The transcriptional program that is activated is both anti-apoptotic and highly pro-inflammatory. Any compromise in engagement of the pathway results in immune-deficiency, whereas constitutive activation generates a sustained inflammatory response that may promote malignancy. Furthermore, experimental approaches have demonstrated diverse functions of the NF-κB transcription factor in the nervous system. For example, inhibition of neuronal NF-κB by super-repressor IκB resulted in the loss of neuroprotection and defects in learning and memory [KK09]. Thus NF-κB is the subject of much active research and of great clinical significance.

In order to understand the regulatory mechanism and the dynamics of this transcription factor, scientists from cell biology, neurobiology, and bioinformatics used VANESA to reconstruct and simulate the NF-κB system to understand cell signaling flow. With access to the databases KEGG [KGS+12], HPRD [KPGK+09], IntAct [KAB+12], and MINT [LBP+12], an automatic reconstruction of the corresponding protein-protein interaction and signaling networks was realized in VANESA. Therefore, scientists queried each of the aforementioned databases with a list of known involved genes. This list was derived from several microarray experiments, analyzing genes from cortex and hippocampus of wild-type and transgenic mice. The analysis of these microarrays, similar to the work presented in Section 6.1, was performed with an in-house R-solution. The laboratory experiments focused on 7,000 genes playing an important role in neuroprotection, learning, and memory, among others.

Using VANESA, its was possible to reconstruct a biological model containing the selected genes organized in protein-protein interaction and signaling networks to gain further meaningful data. This resulted in a combination of tightly interlinked complex systems at various levels of magnitude. The main resulting network contains 778 nodes and 1,868 edges, which clearly shows how many different elements are involved in the direct NF-κB regulation system. Further studies performed in-house, with a genome-wide transcription-factor binding site search, revealed even more elements regulated by NF-κB. About 9 % of all genes in the human genome reveal a potential NF-κB binding site (see Figure 6.5). Thus, it is not surprising that computational and experimental results have yielded datasets of increased size and complexity, which produced large dense graphs.

With regard to the visualization of the resulting networks, VANESA and the software application GI-EB [NHE12], a framework for edge bundling integrating topology, geometry, and importance, were linked with each other to help and assist scientists in visually exploring the system. This software linking resulted in a new way to analyze large scale networks that combines data integration and modeling methods with centrality measurement techniques [JKH+11]. This approach uses edge bundling methods to reduce visual clutters and shows high-level edge patterns, and important system structures (see Figure 6.6). Furthermore, centrality measurement as individual-level network analysis was performed in GI-EB and in VANESA. As centrality

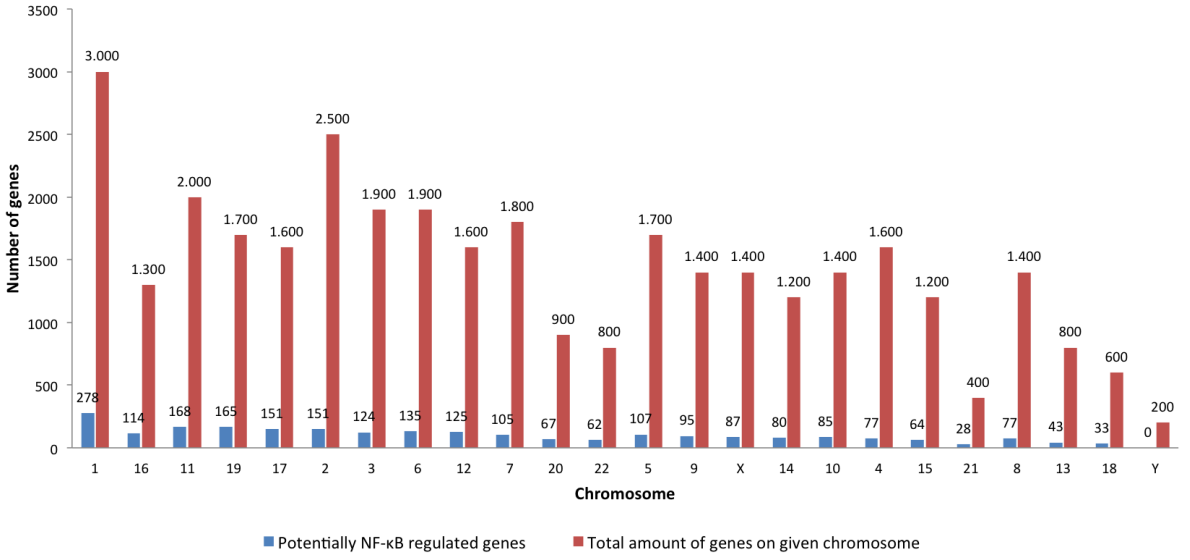


Figure 6.5: Using an in-house bioinformatics transcription factor binding search software, a genome wide analysis for genes potentially regulated by NF-κB was performed. The analysis revealed that each chromosome contains genes with a potentially NF-κB binding site, except for chromosome Y. In summary, 2,421 potentially NF-κB regulated genes were identified in the human. Each chromosome shows an affinity for 5.5% to 9.27% of the total amount of genes to be regulated by the transcription.

measurement a combination of different measures such as degree, stress, betweenness, and closeness was applied on the networks. This approach has already proven useful in other application cases [YBL06] and also in this study, important elements within the dense networks have been indicated. The visualization and analysis technique produced pictures that clearly show the most significant topological structures of the input network. Significant regulatory elements and structures are visually accessible, indicating common biological processes. Thus, it was possible for molecular scientists from biology to focus on regulatory elements and structures that have not yet been found and were unknown so far. This motivated new experiments and further investigations.

However, in order to understand system dynamics, important paths and network structures were further examined in the Petri net environment of VANESA. Therefore, the initial network was reduced to the most important elements and automatically translated into the Petri net language in VANESA (see Figure 6.7). Using hybrid Petri nets and the model parameter presented in Table 6.3 and 6.4, the simulation was performed. For each place in the Petri net, specific ODEs for the functional processing were applied, mainly based on the mathematical Hill function. For places where dynamics are not known, discrete instead of continuous places and transitions were used.

The simulation results from VANESA reflect the importance of elements such as TNF and other cytokines and chemokine as major input signals (see Figure 6.8). Further on, proteins

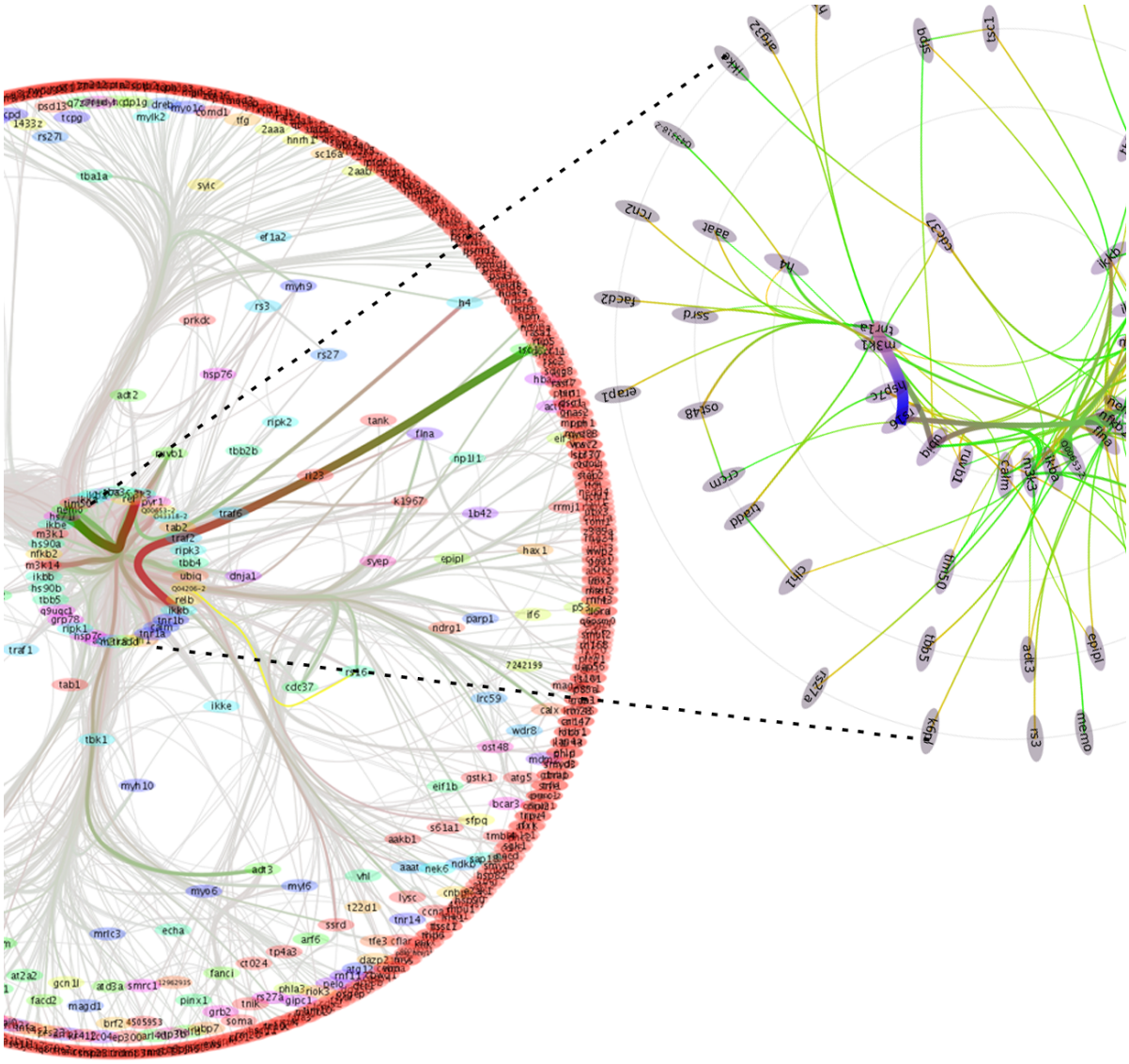


Figure 6.6: A visualization of a reconstructed NF-κB protein-protein interaction and signaling network from VANESA and the software application GI-EB. The network consists of 778 nodes and 1,868 edges and represents the first direct interaction partners of the transcription factor. The most important paths and structures are highlighted with edge bundling and centrality measurement methods. Biological elements are placed on concentric circles according to the centrality measurements. Nodes which are more important than others are placed in the inner circles, less important nodes are placed outside the main core of the visualized network.

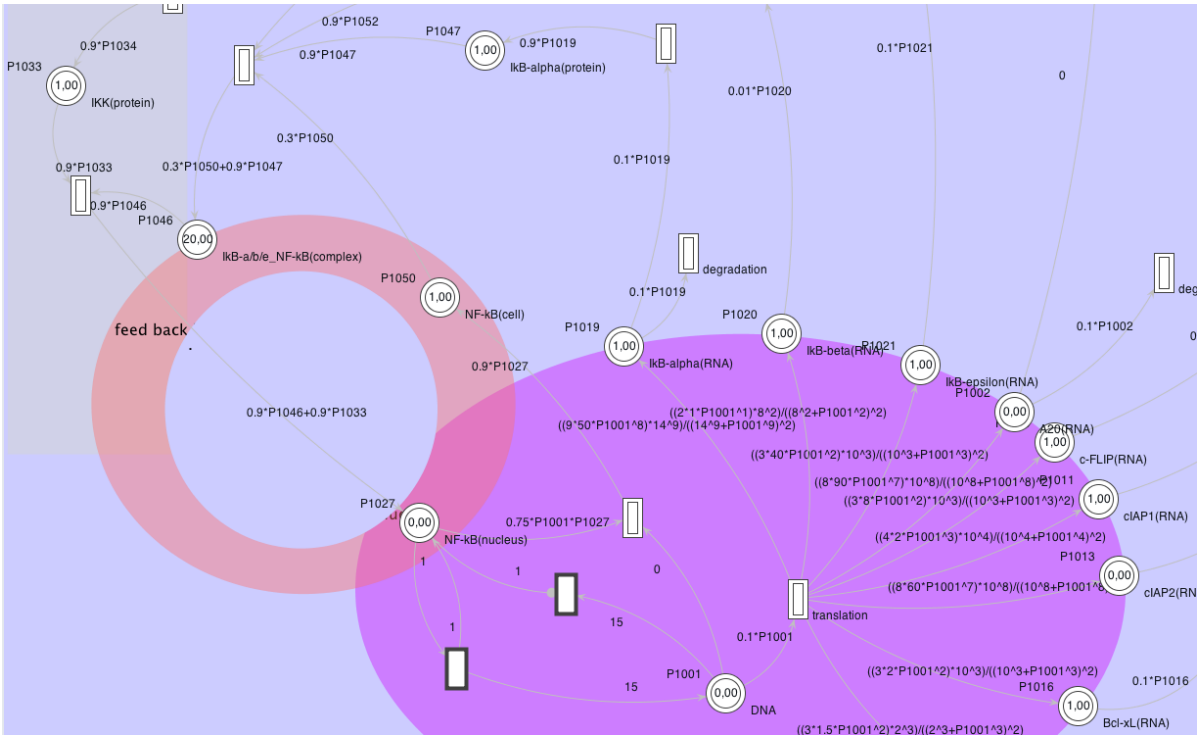


Figure 6.7: A reconstructed NF-κB hybrid Petri net model in VANESA. The figure shows the negative feed-back loop of NF-κB, which results in oscillating levels of NF-κB activity within the nucleus. There, it turns on the expression of specific genes that have a DNA-binding site for NF-κB. Based on the laboratory investigated gene mRNA level and additional experimental data, the translation of the corresponding proteins was simulated, which then, influence the NF-κB transcription factor in a so-called feed-back loop.

Property	IκB-α	IκB-β	IκB-ϵ
Cytoplasmic IκB association (μM^{-1} min^{-1})	0.3×10^2	0.3×10^2	0.3×10^2
Cytoplasmic IκB dissociation (min^{-1})	6×10^{-5}	6×10^{-5}	6×10^{-5}
Nuclear IκB association (μM^{-1} min^{-1})	0.3×10^2	0.3×10^2	0.3×10^2
Nuclear IκB dissociation (min^{-1})	6×10^{-5}	6×10^{-5}	6×10^{-5}
IκB- NF-κB + IKK association (μM^{-1} min^{-1})	11.1	2.88	4.2
IκB- NF-κB + IKK dissociation (min^{-1})	7.5×10^{-2}	10.5×10^{-2}	10.5×10^{-2}
IκB- IKK + NF-κB association (μM^{-1} min^{-1})	0.3×10^2	0.3×10^2	0.3×10^2
IκB- IKK + NF-κB dissociation (min^{-1})	6×10^{-5}	6×10^{-5}	6×10^{-5}
IKK-mediated IκB degradation (min^{-1})	3.6×10^{-1}	1.2×10^{-1}	1.8×10^{-1}
Cytoplasmic bound IκB degradation (min^{-1})	6×10^{-5}	6×10^{-5}	6×10^{-5}
IκB- NF-κB export (min^{-1})	8.28×10^{-1}	4.14×10^{-1}	4.14×10^{-1}
Inducible IκB transcription (μM^{-1} min^{-1})	1.386		

Table 6.3: The table shows NF-κB dimer model parameters from literature [WBH05, Kea06, BKK+07] which were partially used to simulate the NF-κB system in VANESA. Therefore, values for the negative feed-back loop of NF-κB were investigated such as cytoplasmic association times, nuclear association times, dissociation times, degradation times, transcription times, and export times, among others. Primarily, the table focuses on the IκB-α, IκB-β, and IκB-ϵ proteins, which inhibit NF-κB in its functions.

Property	IκB-α	IκB-β	IκB-ϵ
Constitutive RNA synthesis (min^{-1})	1.85×10^{-4}	4.27×10^{-5}	4.27×10^{-5}
RNA degradation (min^{-1})	3.36×10^{-2}	1.68×10^{-2}	1.18×10^{-2}
Hill coefficient	2		
Cytoplasmic protein synthesis (min^{-1})	24.48×10^{-2}	24.48×10^{-2}	24.48×10^{-2}
Cytoplasmic protein degradation (min^{-1})	1.2×10^{-1}	1.8×10^{-1}	1.8×10^{-1}
Nuclear import (min^{-1})	1.8×10^{-2}	1.8×10^{-2}	1.8×10^{-2}
Nuclear export (min^{-1})	1.2×10^{-2}	1.2×10^{-2}	1.2×10^{-2}
Cytoplasmic IKK association (μM^{-1} min^{-1})	1.35	0.36	0.54
IKK-mediated IκB degradation (min^{-1})	7.5×10^{-2}	10.5×10^{-2}	10.5×10^{-2}
IKK-bound IκB degradation (min^{-1})	1.8×10^{-3}	0.6×10^{-3}	1.2×10^{-3}

Table 6.4: The table shows additional NF-κB dimer model parameters from literature [WBH05, Kea06, BKK+07] which were partially used to simulate the NF-κB system in VANESA. The table focuses on RNA synthesis, the optimal hill coefficient for ODE modeling, protein synthesis and degradation time, nuclear export and import parameters, as well as IKK regulation times, among others.

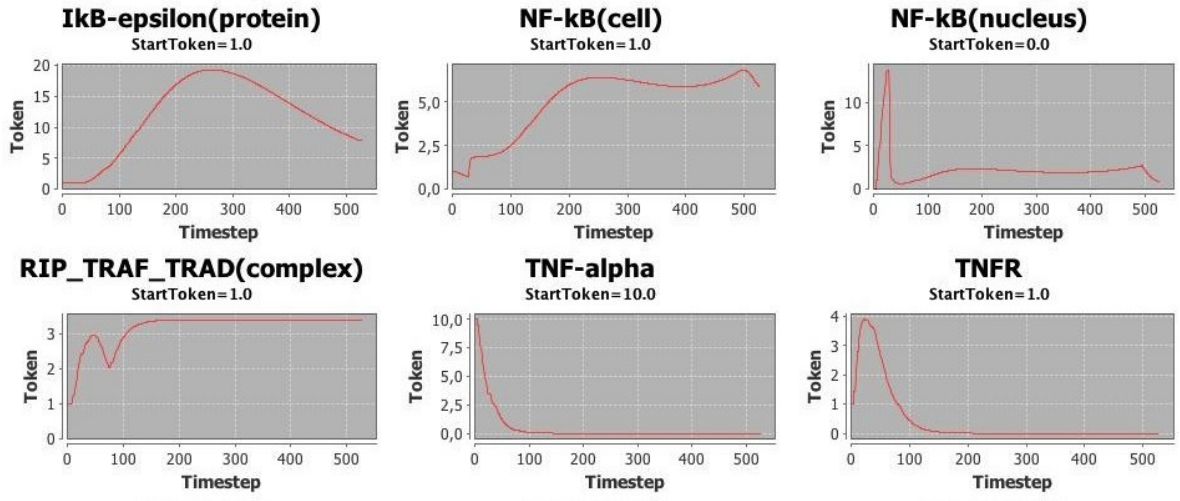


Figure 6.8: The figure shows one part of the NF-κB simulation results performed in VANESA. The charts show the relative mRNA expression values of selected elements of the dynamic system on a time-line of 500 minutes. The focus of these charts are the oscillating levels of NF-κB activity within the nucleus with regard to a TNF-α stimulus, showing different dynamics than the ones known from literature. This has led to new hypotheses.

that directly influence the translocation of the NF-κB transcription factor, such as IKK and the IκB- isoforms clearly stand out and demonstrate their regulatory effect on the network. Having a closer look at the simulation results, the functional sense of the NF-κB - system becomes more accessible. In addition to the most important biological elements of the known system such as TNFR-α, NF-κB, IKK, IκB-α, IκB-β, IκB-ϵ, it was possible to identify new possible regulatory structures and potential proteins that might play a further crucial role within the system. One example points out the role of the specific interacting heat-shock protein HSC70 with p65. Moreover, the simulation reveals different oscillating levels of NF-κB activity within the nucleus with regard to a TNF-α stimulus. This is very interesting as all other published models show different cell dynamics. This led to new hypotheses and motivated further experiments.

6.4 Modeling cell-to-cell communication

Due to the complexity of pathway interactions and large numbers of components involved in cell proliferation, cell differentiation, signal transduction, cellular rhythms, and cell-to-cell communication, it is quite difficult to intuitively understand the behavior of cellular networks. Particularly, the understanding of the molecular mechanism of cell-to-cell communication is fundamental for system biology. Cell-to-cell communication or Quorum Sensing (QS), the use of small molecule signals to coordinate complex patterns of behavior in bacteria, has been the focus of many reports over the past decade and became a major objective in bioinformatics. In general, bacterial cells are able to adapt their behavior to the environment and its conditions to

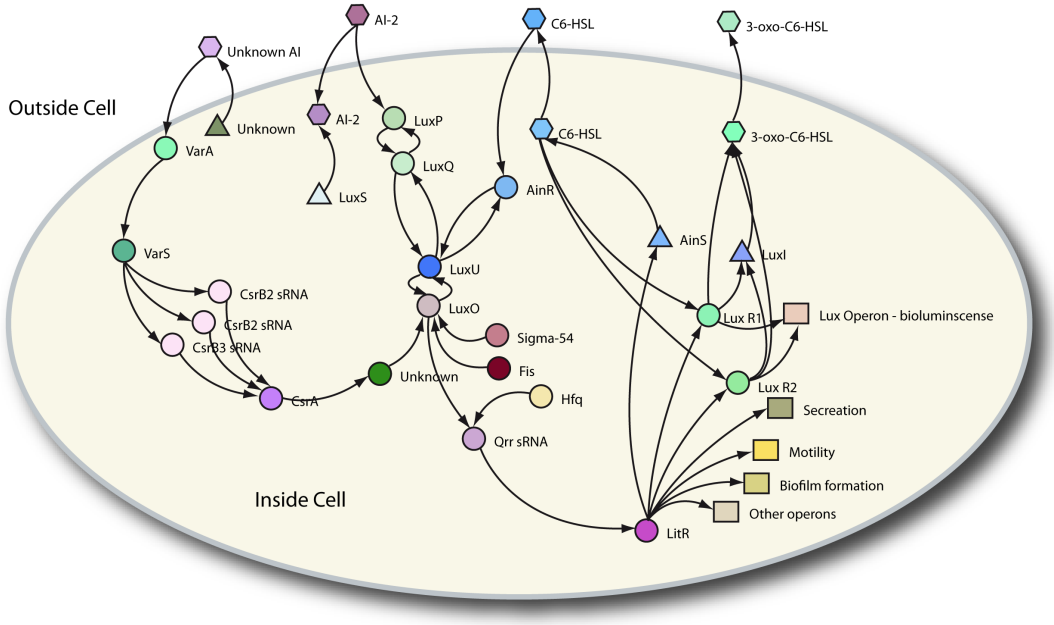


Figure 6.9: The figure presents the reconstructed QS system for the bacteria *Aliivibrio salmonicida* from VANESA. The model is based on information from eleven different life-science databases showing the intracellular network. The main goal of this model is to describe and investigate molecular switches causing gene activation of bioluminescence and other processes.

control and limit activities within their community [Sch01, RS06]. Therefore, bacteria use QS to coordinate their gene expression according to the local density of their population. However, until now, many details of the intracellular molecular machinery that are responsible for the complex collective behavior of cellular and multicellular populations are unknown.

Therefore, *vibrio* infections of the organism *Aliivibrio salmonicida* were examined in detail. *Aliivibrio salmonicida* is a moderate halophilic and psychrophilic bacterium infecting fish populations. The virulence factors make the bacterium one of the major agents of cold-water *vibriosis*. In most of the cases, infections due to *Aliivibrio salmonicida* result in tissue degradation, haemolysis, and sepsis *in vivo* [TN88]. Thus, it is not surprising that QS regulated virulence factor production is of great importance as fish infection with *vibrio* bacteria is one of the major bacterial threats in marine aquaculture [EHG89, TIS04]. In general, the QS system is comparable to an 'on' and 'off' system under the influence of molecular noise [GTL06]. The cell population uses small, freely diffusible signaling molecules such as autoinducers or pheromones for communication processes. Due to specific receptors, external signal molecules can be recognized, which results in the transcription of certain regulatory genes. At low population density, the concentration of autoinducers in the environment is almost zero and the intracellular network remains in the 'off' state. When cell density increases, the concentration of autoinducers increases. Once a certain autoinducers threshold is reached, the QS network becomes active and turns on the expression of the phenotype specific genes. The switch from the

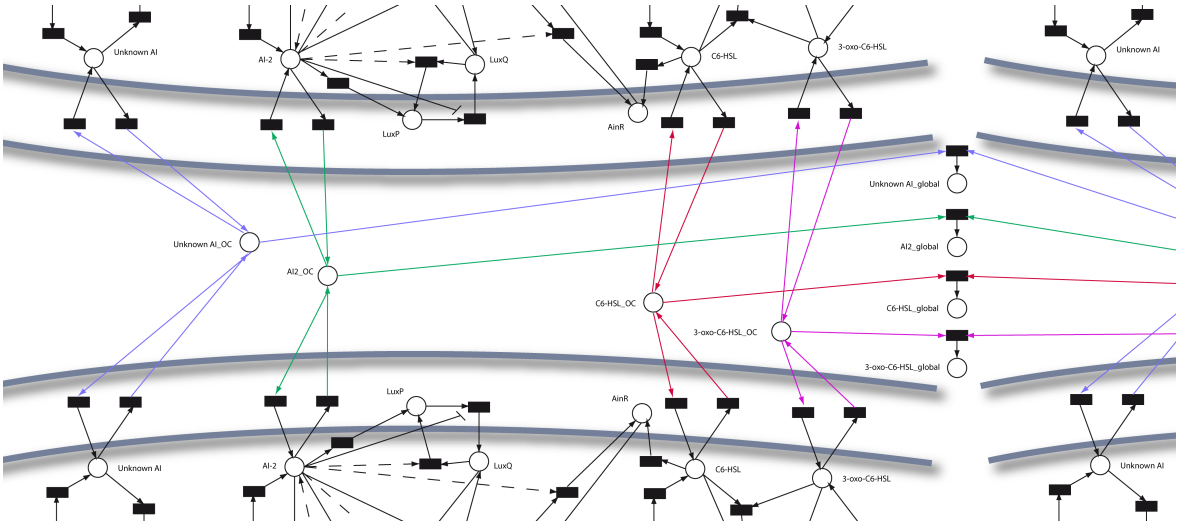


Figure 6.10: An overview of a reconstructed QS system consisting of four modeled *Aliivibrio salmonicida* cells based on the Petri net language. The modeled cells perform cell-to-cell communication by exchanging small molecule signals, such as C6-HSL, 3-oxo-C6-HSL, and other auto inducers to coordinate complex patterns of behavior.

steady-state network into the QS network is a so-called positive feed-back loop, which results in motility, biofilm formation, virulence factor production, and secretion or other processes, to achieve advantages in environmental adaptation and survival.

Motivated by the goal of having a better understanding of QS processes, VANESA was used by scientists from biology to reconstruct cell-to-cell and cell differentiation models [JKT+10]. In order to understand the QS system, the signaling networks of *Aliivibrio salmonicida* under high and low population density were modeled. As the starting point for the reconstruction, only a few proteins were known, which were derived from experimental datasets [HLH+08]. Based on these proteins, VANESA was used to reconstruct signaling networks based on database information from the eleven different life science databases UniProt [AJMO+12], KEGG [KGS+12], OMIM [ABSH09], GO [BDD+12], ENZYME [Bai00], BRENDA [SGC+11], PDB [WIN+05], MINT [LBP+12], SCOP [AHC+08], EMBL-Bank [CAB+09], and Pub-Chem [WXS+12]. As VANESA only queries the databases KEGG, HPRD, Mint, and Intact, parts of the aforementioned databases had to be integrated into the software application for this research application. The signaling networks were created step-by-step, adding all relevant biological elements from the aforementioned data sources. In the final step, all networks were merged in VANESA, resulting in one system (see Figure 6.9). Thus, it was possible to consider important regulatory elements such as transcription factors and sRNAs, among others.

These networks were the base for further simulation processing. Therefore, the reconstructed networks were automatically transformed into the Petri net language in VANESA, to simulate cell-to-cell communication structures for hypothesis generation and testing. The models were exported from VANESA into CellIllustrator, where they were further investigated by experts

from biology. In this application case, the simulation was performed in the software application CellIllustrator [NSJ+10], in order to combine the reconstructed model with other already existing models. Furthermore, the different models were combined into a system of interacting cells (see Figure 6.10) imitating cell communication.

These studies subsequently revealed a much more complex system for the intracellular network switch into the 'on' state. The different gene activation mechanisms by LuxR1/LuxR2 and LitR were investigated in detail, as well as secretion, mobility, biofilm formation, and other operons. New intracellular circuitry of signal transduction and gene expression were detected and successfully simulated. It was shown how the concentration of the autoinducer C6-HSL inside and outside the cell influences changing speed parameters and how cell behavior varies. Based on this new QS model, catalytic efficiency was calculated to influence the signal molecules, the diffusion speed of the medium, and finally, to find new ways to block virulence factors.

6.5 Summary

VANESA, with is various functionalities, has proved very useful in a wide area of application cases. Using VANESA, scientists have been able to model and simulate intracellular molecular machineries from a variety of research cases. Using this tool they were able to extend and deepen their knowledge and moreover, were motivated to perform further experiments based on the resulting molecular insights of VANESA. The here presented application cases already showed some valuable results. In a clinical trail, the middle-ear disease cholesteatoma was examined. Therefore, signaling networks for the development of the cholesteatoma were reconstructed in VANESA, which turn healthy external auditory-canal skin into deregulated cells. The reconstructed networks formed a valuable framework to gain better insight into regulatory mechanisms and possible drug-targeting approaches. More than twenty important gene motifs and regulatory structures were identified that are related to metastatic tumors and chronic inflamed tissue development. Furthermore, VANESA proved very useful during the investigation of cardiovascular diseases. Therefore, associated deregulated pathways with regard to dilated cardiomyopathy were examined regarding to their major regulation elements. More than twelve proteins, which have an important role in common pertubated pathways were identified with VANESA. This helped scientists from medicine and biology in finding novel biomarkers. Besides, it was demonstrated how VANESA can work cooperatively with other software tools, such as the 3D cell visualization tool CmPI and the text mining tool ANDvisio. With this approach, an abstract virtual 3D cell environment was reconstructed to examine information flow and biological processing in the cell environment.

As presented, VANESA is also able to contribute to fundamental research in molecular biology. It was used by scientists from molecular biology to reconstruct the negative feed-back loop

for the transcription factor NF-κB. Based on the reconstructed networks, centrality measurements pointed out new regulatory motifs which were successfully simulated within the Petri net environment of VANESA. This resulted in new insight into the molecular level of regulation. Moreover, VANESA played an important role in economical studies as well. In order to understand the spread of pathogen bacteria in fish populations, cell-to-cell communication processes were modeled and simulated with VANESA. Here, it was possible to identify the molecular switches, which are involved in turning on virulence factor production.

In summary, VANESA proved very useful as a valuable tool in molecular research. Any scientist can work with this tool. It can greatly help scientists in modeling and simulating the role of individual components and processes in the reversible or irreversible changes of network architecture. This can contribute to new biological findings, and possible medical and biotechnological strategies. And in application cases in which additional data is necessary, the software application can be extended by new data repositories and analytic approaches. Furthermore, VANESA can be integrated into a pipeline of other tools to perform even more sophisticated investigations. This makes VANESA really unique in the wide area of bio-research modeling tools.

Chapter 7

Summary

During the analysis of a biological element or process, scientists are faced with collecting, analyzing, and modeling a huge amount of data in order to create a meaningful model for hypothesis testing. Therefore, the aim of this work was to realize a software application which assists scientists in the reconstruction, analysis, and simulation of biological systems. In general, detailed information about the research object is available, but more extensive information about the global context and underlying system is missing. Moreover, this information is necessary, as a biological object or process never acts as an independent unit. It is a part of a bigger machinery or regulatory process which needs to be mapped out in order to understand dynamic cell system behavior. Although information can be found stored in different databases, data is distributed over many heterogeneous data sources. This data needs to be filtered in terms of content and quality, and moreover, normalized and linked in complex and time-consuming processes. Motivated by this problem, VANESA has been implemented. It is a powerful and easy-to-use modeling software for the automatic reconstruction and analysis of biological networks based on life-science database information. Using VANESA, scientists are able to model any kind of biological processes and systems as biological networks. VANESA combines different fields of research such as information fusion, modeling, Petri net simulation, and network visualization, which are some of the most important areas in bioinformatics and systems biology.

Based on a simple list of research objects such as investigated proteins or genes, biological models can be automatically reconstructed using VANESA. Therefore, VANESA accesses the data warehouse DAWIS-M.D. and extracts information from the life science databases KEGG, HPRD, IntAct, and MINT to produce a complete interaction network system covering the most important -omic levels. Data is automatically gathered, normalized, and linked into one model in VANESA. Using the functionality of the data warehouse system DAWIS-M.D., it is even possible to link different biological domains such as genes, enzymes, proteins, and compounds with each other, which then, can be represented as biomedical networks within the software application. Each database can be queried one-by-one or in combination. With the

provided information, signaling networks, protein-protein interaction networks, metabolic pathways, regulatory networks, disease/drug networks, and chemical networks can be automatically reconstructed. This results in biological models, which enable scientists to focus on complex interactions and/or to investigate the role of individual components and processes within entire biological systems.

As database-generated networks can be large and complex, VANESA also provides a graph theoretical network analysis environment, with which the relative importance and significance of elements in a biological network can be determined. Biological elements having an important degree distribution can be easily identified and linked to network functionality and information flow. Further centrality analysis ranks vertices and edges to identify network motifs, which directly or indirectly regulate system behavior. This results in pictures that clearly show the most significant skeletal structures of the input network. Thus, users can interactively explore the networks and moreover, filter important information from the mass of data. In addition, it is possible to compare biological networks with random artificial networks to identify structures and motifs that only appear in real-world systems. This makes VANESA also a useful tool for theoretical biology.

Simulation is another strength of VANESA. For the simulation of biological processes, users do not need data for kinetics or knowledge about mathematical differentiation equations and programming. VANESA provides a biologically sophisticated graphical user interface in which biological models can be modeled and simulated using the PNlib in Modelica. Simulations can be performed using qualitative, stochastic, continuous, hybrid, and functional Petri nets of the xHPN paradigm. Due to VANESA's generic design and strict separation of internal data structure and graphical representation, it is possible to automatically convert the integrated ontology for network representation into the xHPN formalism. Thus, basic networks and reconstructed networks can be easily transformed and simulated in the Petri net language. In addition to the possibility to perform simulation processing using the PNlib, VANESA can export models into CellIllustrator, one of the best-known simulations environments. Petri net analysis techniques, which allow checking the liveness, boundedness, and reversibility of Petri nets, are also supported in VANESA.

Another important aspect of VANESA is the graphical user interface, network visualization, and human-system-interaction which has been specially designed for the needs of scientists in bioresearch fields. The elements are designed to be truly interactive to assist users in their research. Everything is reachable within a maximum of three mouse-clicks. With a biologically sophisticated graphical user interface, users are able to intuitively model and simulate complex dynamic interactions and processes in one active window. Users can automatically reconstruct biological networks or draw any kind of biological model by hand. Furthermore, they have the possibility to edit, compare, manipulate, transform, and zoom into parts of the biological networks. Information is always visualized in a clear and understandable manner. This helps in better understanding and identifying relevant objects, processes, and motifs.

Graph layouts visualize networks in an appropriate way. This prevents data from being visualized as an impenetrable furball. Filtering algorithms and graph theory enables users to reduce network complexity to focus on significant objects. Since VANESA also enable users to perform Petri net simulations, the software application automatically adapts to the selected class of graphs. In the Petri net view, users are able to edit discrete Petri nets as well as specify systems of ordinary differential equations (ODEs) for continuous Petri Nets. Simulation results can be animated or visualized as diagrams or made accessible in tables, which then can be used in other programs, such as Excel.

Analyzing, discussing, and sharing results with VANESA make this software even more powerful. Scientists from different research areas can cooperatively work on one existing model. One case for usage would be that scientists from biology reconstruct a biological model, whereas scientists from bioinformatics or mathematics perform the simulation processing. However, with the possibility of sharing results with some of the most important common software-independent standard data formats, such as SBML, MathML, and an additional easy-to-read text data network exchange format, users can incorporate their thoughts and hypotheses in existing models worldwide.

7.1 Future perspectives

As new questions from the natural science arise, new approaches and data sources will be necessary for the future. Therefore, the development of VANESA will be continued. Due to its design, it is easy to extend VANESA and integrate new data sources and/or algorithms. Any scientist from computer science, with a little bit of background knowledge, can do this. However, the following list shows some of the next developments in VANESA.

- **Additional databases**
  A consideration is being made to introduce and integrate a new microRNA database into DAWIS-M.D., which can be made accessible for VANESA. Based on information from the biological databases TarBase [SCH05], miRBase [KGJ10], experimental datasets, as well as predicted bioinformatics data, a new level of fine-regulation should be linked to the already accessible -omic levels in VANESA. The structural database will be added for RNA, where 3D structures can be well approximated by 2D structures. With the calculated structures, direct functions for interaction with other RNAs, proteins, genes, and metabolites will be determined. In addition to the RNA database, the MetaCyc database [CAD+12] should be integrated. The MetaCyc database contains more than 1,790 pathways from more than 2,216 different organisms. Information stored in this database is non-redundant, experimentally elucidated, and curated from scientific experimental literature. This new repository should be a further important resource for metabolic pathways. In addition, it is intended to integrate a transcription factor database, such as Transfac

[MKMF+06], Transpath [KPV+06], and JASPAR [PCTK+10]. Additionally, the OMIM database [ABSH09] should be used, containing important information about known diseases and dysregulations in cells.

- **Global network properties**
  Global network properties should be investigated more intensively. The statistical testing of network properties is probably the most crucial and widely underestimated aspect of complex network analysis. Actually, several graph theoretical approaches are implemented in VANESA. However, it is planned to implement new bioinformatics approaches for high-level statistical testing, such as flow-based centralities [Kos11]. Using these measurements, it would be possible to find specific correlations between network flow structures, biological hubs, and global network properties.

- **Petri net simulation**
  Another important aspect of future development should be the automatic Petri net simulation of reconstructed networks in VANESA. The goal is to provide an open-source Petri net simulation tool based on OpenModellica [FAL+05]. However, some essential features in OpenModelica are not yet supported but necessary for the simulation of biological networks. So far, only Dymola and CellIllustrator [NSJ+10] can be used for simulation processing in VANESA. Being successful in implementing the necessary features in OpenModellica, VANESA would become one of the most powerful Petri net simulation software.

- **Visualization**
  Next to the aforementioned developments, added improvements should be made in information visualization. The aim is to simplify the automatic creation and modification of networks, as well as to investigate new graph-layout visualization techniques. The interest in information visualization has emerged from the problem to understand huge and complex networks as they appear nowadays. Information visualization is a critical component in scientific research and too often unattended. And still, large datasets and the different kinds of datasets are big challenges. Therefore, a focus should be made on the creation of new approaches for conveying abstract information to see, explore, and understand large amounts of information in intuitive ways. One example for such an approach is the combination of edge bundling techniques with centrality measurements [JKH+11], as already applied in this work. Using such techniques, scientists working at the bench would have the possibility to interactively and intuitive access their datasets, which in terms, would lead to a high acceptance of the software application and result in new findings in complex datasets.

By realizing the aforementioned approaches, VANESA will be able to assist scientists in answering new questions which are in the future. Especially the microRNA regulation and modeling is of great interest and importance, as the investigation of this -omic level is still in its beginning

stages. Having access to a microRNA database it should be possible to develop a new kind of network analysis and theory. In combination with other life-science databases, more and detailed knowledge and information could be made available for network reconstruction and knowledge seeking. This would give scientists the possibility to go further into the details of molecular cell biology and open the door to new therapeutical approaches and network pharmacology. Chapter 6, where the investigation of the cholesteatoma is described, already showed the potential with the existing features in VANESA. This can be extended to another level. However, as the integration of new databases will create more complex networks, the work on the visualization is mandatory. Further work should be performed on the Petri net simulation. With the existing features and the recommended approaches, VANESA will become even more powerful and will stay up-to-date and be useful to researchers from different fields of studies in the next level of molecular research.

7.2 Discussion

VANESA, with its provided features has already proved very useful in a wide range of application cases. Scientists from biology, biotechnology, medicine, bioinformatics, and other disciplines used the software application as a valuable part in their molecular research. First application cases have already led to new biological findings and motivated further laboratory experiments. Primarily, VANESA is intended for biological scientists working at the bench but any scientist can use it due to its intuitive design. VANESA is unique in its ability to reconstruct, visualize, analyze, and simulate biological systems. With just three mouse-clicks and some information about the research object, users have the possibility to model and analyze entire molecular interaction systems in the form of biological networks. Instead of collecting, transforming, normalizing, and linking information from distributed and heterogeneous data sources, VANESA enables users to gather useful information from some of the most important life-science database in one workflow. This gives scientists the possibility to analyze their research objects in a global as well as detailed context, considering the most important interacting elements and -omic levels, such as genomics, transcriptomics, metabolomics, and proteomics. During research studies these reconstructed models can be then extended by new elements, enriched with experimental findings, and simulated to imitate cell behavior and cell changes. This can truly help scientists in their aim of understanding molecular dynamics and information flow. All in all, with its features, good balance of computational power, and farsightedness, the software application can be a really helpful tool in the natural sciences and furthermore, has the potential to be placed in the same row with some of the most important software applications.

Bibliography

[AB02] R. Albert and A. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74(1):47–75, 2002.

[AB13] Dassault Systems AB. Dymola: Dynamic Modeling Laboratory. *Dymola Release notes - www.3ds.com*, 2013.

[ABC+10] R. Ausbrooks, S. Buswell, D. Carlisle, G. Chavchanidze, S. Dalmas, S. Devitt, A. Diaz, S. Dooley, R. Hunter, P. Ion, M. Kohlhase, A. Lazrek, P. Libbrecht, B. Miller, R. Miner, C. Rowley, M. Sargent, B. Smith, N. Soiffer, R. Sutor, and S. Watt. Mathematical Markup Language (MathML) Version 3.0. *W3C recommendations - www.w3.org*, pages 115–121, October 2010.

[ABSH09] J. Amberger, C. A. Bocchini, A. F. Scott, and A. Hamosh. McKusick's Online Mendelian Inheritance in Man (OMIM). *Nucleic Acids Research*, 37(Database Issue):793–796, January 2009.

[ACDL+07] I. Avila-Campillo, K. Drew, J. Lin, D. J. Reiss, and R. Bonneau. BioNetBuilder: automatic integration of biological networks. *Bioinformatics*, 23(3):392–393, February 2007.

[AD98] H. Alla and R. David. Continuous and hybrid Petri nets. *Journal of Circuits Systems and Computers*, 8(1):159–188, 1998.

[AHC+08] A. Andreeva, D. Howorth, J. M. Chandonia, S. E. Brenner, T. J. Hubbard, C. Chothia, and A. G. Murzin. Data growth and its impact on the SCOP database: new developments. *Nucleic Acids Research*, 36(Database Issue):419–425, January 2008.

[AJL+07] B. Alberts, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Molecular Biology of the Cell*. Garland Science, 5th edition, November 2007.

[AJMO+12] R. Apweiler, M. Jesus Martin, C. O'onovan, M. Magrane, Y. Alam-Faruque, R. Antunes, E. Barrera Casanova, B. Bely, M. Bingley, L. Bower, B. Bursteinas, W. Mun Chan, G. Chavali, A. Da Silva, E. Dimmer, R. Eberhardt, F. Fazzini, A. Fedotov, J. Garavelli, L. G. Castro, M. Gardner, R. Hieta, R. Huntley, J. Jacobsen, D. Legge, W. Liu, J. Luo, S. Orchard, S. Patient, K. Pichler, D. Poggioli, N. Pontikos, S. Pundir, S. Rosanoff, T. Sawford, H. Sehra, E. Turner, T. Wardell, X. Watkins, M. Corbett, M. Donnelly, P. van Rensburg, M. Goujon, H. McWilliam, R. Lopez, I. Xenarios, L. Bougueleret, A. Bridge, S. Poux,

N. Redaschi, G. Argoud-Puy, A. Auchincloss, K. Axelsen, D. Baratin, M. C. Blatter, B. Boeckmann, J. Bolleman, L. Bollondi, E. Boutet, S. Braconi Quintaje, L. Breuza, E. deCastro, L. Cerutti, E. Coudert, B. Cuche, I. Cusin, M. Doche, D. Dornevil, S. Duvaud, A. Estreicher, L. Famiglietti, M. Feuermann, S. Gehant, S. Ferro, E. Gasteiger, V. Gerritsen, A. Gos, N. Gruaz-Gumowski, U. Hinz, C. Hulo, N. Hulo, J. James, S. Jimenez, F. Jungo, T. Kappler, G. Keller, V. Lara, P. Lemercier, D. Lieberherr, X. Martin, P. Masson, M. Moinat, A. Morgat, S. Paesano, I. Pedruzzi, S. Pilbout, M. Pozzato, M. Pruess, C. Rivoire, B. Roechert, M. Schneider, C. Sigrist, K. Sonesson, S. Staehli, E. Stanley, A. Stutz, S. Sundaram, M. Tognolli, L. Verbregue, A. L. Veuthey, C. H. Wu, C. N. Arighi, L. Arminski, W. C. Barker, C. Chen, Y. Chen, P. Dubey, H. Huang, A. Kukreja, K. Laiho, R. Mazumder, P. McGarvey, D. A. Natale, T. G. Natarajan, N. V. Roberts, B. E. Suzek, C. Vinayaka, Q. Wang, Y. Wang, L. S. Yeh, and J. Zhang. Reorganizing the protein space at the Universal Protein Resource (UniProt). *Nucleic Acids Research*, 40(Database Issue):71–75, January 2012.

[Alo06] U. Alon. *An Introduction to Systems Biology: Design Principles of Biological Circuits*. Simplicity in Biology. Chapman and Hall/CRC, 1st edition, July 2006.

[Alt02] A Altman. *Signal transduction pathways in autoimmunity*, volume 5. Karger, 2002.

[Ass05] Modelica Association. Modelica - A unified object-oriented language for physical systems modeling. *Language Specification*, 2:7–11, 2005.

[AVB01] F. Achard, G. Vaysseix, and E. Barillot. XML, bioinformatics and data integration. *Bioinformatics*, 17(2):115–125, February 2001.

[Bai00] A. Bairoch. The ENZYME database in 2000. *Nucleic Acids Research*, 28(1):304–305, January 2000.

[Bat10] A. Bateman. Curators of the world unite: the International Society of Biocuration. *Bioinformatics*, 26(8):991, April 2010.

[BCC+07] G. Bernot, F. Cassez, J. P. Comet, F. Delaplace, C. Müller, and O. Roux. Semantics of Biological Regulatory Networks. *Electronic Notes in Theoretical Computer Science*, 180(3):3–14, 2007.

[BCP+12] D. M. Bolser, P. Y. Chibon, N. Palopoli, S. Gong, D. Jacob, V. D. Del Angel, D. Swan, S. Bassi, V. Gonzalez, P. Suravajhala, S. Hwang, P. Romano, R. Edwards, B. Bishop, J. Eargle, T. Shtatland, N. J. Provart, D. Clements, D. P. Renfro, D. Bhak, and J. Bhak. MetaBase–the wiki-database of biological databases. *Nucleic Acids Research*, 40(Database Issue):1250–1254, January 2012.

[BCS06] G. D. Bader, M. P. Cary, and C. Sander. Pathguide: a pathway resource list. *Nucleic Acids Research*, 34(Database Issue):504–506, January 2006.

[BDD+12] J. A. Blake, M. Dolan, H. Drabkin, D. P. Hill, L. Ni, D. Sitnikov, S. Burgess, T. Buza, C. Gresham, F. McCarthy, L. Pillai, H. Wang, S. Carbon, S. E. Lewis, C. J. Mungall, P. Gaudet, R. L. Chisholm, P. Fey, W. A. Kibbe,

S. Basu, D. A. Siegele, B. K. McIntosh, D. P. Renfro, A. E. Zweifel, J. C. Hu, N. H. Brown, S. Tweedie, Y. Alam-Faruque, R. Apweiler, A. Auchinchloss, K. Axelsen, G. Argoud-Puy, B. Bely, M. Blatter, L. Bougueleret, E. Boutet, S. Branconi, L. Breuza, A. Bridge, P. Browne, W. M. Chan, E. Coudert, I. Cusin, E. Dimmer, P. Duek-Roggli, R. Eberhardt, A. Estreicher, L. Famiglietti, S. Ferro-Rojas, M. Feuermann, M. Gardner, A. Gos, N. Gruaz-Gumowski, U. Hinz, C. Hulo, R. Huntley, J. James, S. Jimenez, F. Jungo, G. Keller, K. Laiho, D. Legge, P. Lemercier, D. Lieberherr, M. Magrane, M. J. Martin, P. Masson, M. Moinat, C. O'Donovan, I. Pedruzzi, K. Pichler, D. Poggioli, P. Porras Millan, S. Poux, C. Rivoire, B. Roechert, T. Sawford, M. Schneider, H. Sehra, E. Stanley, A. Stutz, S. Sundaram, M. Tognolli, I. Xenarios, R. Foulger, J. Lomax, P. Roncaglia, E. Camon, V. K. Khodiyar, R. C. Lovering, P. J. Talmud, M. Chibucos, M. Gwinn Giglio, K. Dolinski, S. Heinicke, M. S. Livstone, R. Stephan, M. A. Harris, S. G. Oliver, K. Rutherford, V. Wood, J. Bahler, A. Lock, P. J. Kersey, M. D. McDowall, D. M. Staines, M. Dwinell, M. Shimoyama, S. Laulederkind, T. Hayman, S. Wang, V. Petri, T. Lowry, P. D'Eustachio, L. Matthews, C. D. Amundsen, R. Balakrishnan, G. Binkley, J. M. Cherry, K. R. Christie, M. C. Costanzo, S. S. Dwight, S. R. Engel, D. G. Fisk, J. E. Hirschman, B. C. Hitz, E. L. Hong, K. Karra, C. J. Krieger, S. R. Miyasato, R. S. Nash, J. Park, M. S. Skrzypek, S. Weng, E. D. Wong, T. Z. Berardini, D. Li, E. Huala, D. Slonim, H. Wick, P. Thomas, J. Chan, R. Kishore, P. Sternberg, K. Van Auken, D. Howe, and M. Westerfield. The Gene Ontology: enhancements for 2011. *Nucleic Acids Research*, 40(Database Issue):559–564, January 2012.

[BFG+85] C. Burks, J. W. Fickett, W. B. Goad, M. Kanehisa, F. I. Lewitter, W. P. Rindone, C. D. Swindell, C. S. Tung, and H. S. Bilofsky. The GenBank nucleic acid sequence database. *Computer Applications in the Biosciences*, 1(4):225–233, December 1985.

[BHNM07] H. Berman, K. Henrick, H. Nakamura, and J. L. Markley. The worldwide Protein Data Bank (wwPDB): ensuring a single, uniform archive of PDB data. *Nucleic Acids Research*, 35(Database Issue):301–303, January 2007.

[BKK+07] S. Basak, H. Kim, J. D. Kearns, V. Tergaonkar, E. O'Dea, S. L. Werner, C. A. Benedict, C. F. Ware, G. Ghosh, I. M. Verma, and A. Hoffmann. A fourth IkappaB protein within the NF-kappaB signaling module. *Cell*, 128(2):369–381, January 2007.

[BKMC+12] D. A. Benson, I. Karsch-Mizrachi, K. Clark, D. J. Lipman, J. Ostell, and E. W. Sayers. GenBank. *Nucleic Acids Research*, 40(Database Issue):48–53, January 2012.

[BKW+77] F. C. Bernstein, T. F. Koetzle, G. J. Williams, E. F. Meyer, M. D. Brice, J. R. Rodgers, O. Kennard, T. Shimanouchi, and M. Tasumi. The Protein Data Bank: a computer-based archival file for macromolecular structures. *Journal of Molecular Biology*, 112(3):535–542, May 1977.

[BOGK98] H. Bono, H. Ogata, S. Goto, and M. Kanehisa. Reconstruction of Amino Acid Biosynthesis Pathways from the Complete Genome Sequence. *Genome research*, 8:203–210, 1998.

[Bon72] P. Bonacich. Factoring and weighting approaches to status scores and clique identification. *The Journal of Mathematical Sociology*, 2(1):113–120, January 1972.

[BPSM+08] T. Bray, J. Paoli, C. Sperberg-McQueen, E. Maler, and F. Yergeau. Extensible markup language (XML) 1.0. *www.w3.org - W3C recommendation*, pages 1–10, 2008.

[Bri11] C. Brinkrolf. Realisierung einer neuen Petri-Netz-Simulations- und Analyse-Umgebung in der Systembiologie. *Master thesis at the Bielefeld University, Germany*, pages 41–46, November 2011.

[BSR+08] B. J. Breitkreutz, C. Stark, T. Reguly, L. Boucher, A. Breitkreutz, M. Livstone, R. Oughtred, D. H. Lackner, J. Bähler, V. Wood, K. Dolinski, and M. Tyers. The BioGRID Interaction Database: 2008 update. *Nucleic Acids Research*, 36(Database Issue):637–640, January 2008.

[BWF+00] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, January 2000.

[BYYO11] M. D. Brazas, D. S. Yim, J. T. Yamada, and B. F. Ouellette. The 2011 Bioinformatics Links Directory update: more resources, tools and databases and features to empower the bioinformatics community. *Nucleic Acids Research*, 39(Web Server issue):3–7, July 2011.

[CA08] A. Camargo and F. Azuaje. Identification of dilated cardiomyopathy signature genes through gene expression and network data integration. *Genomics*, 92(6):404–413, December 2008.

[CAB+09] G. Cochrane, R. Akhtar, J. Bonfield, L. Bower, F. Demiralp, N. Faruque, R. Gibson, G. Hoad, T. Hubbard, C. Hunter, M. Jang, S. Juhos, R. Leinonen, S. Leonard, Q. Lin, R. Lopez, D. Lorenc, H. McWilliam, G. Mukherjee, S. Plaister, R. Radhakrishnan, S. Robinson, S. Sobhany, P. T. Hoopen, R. Vaughan, V. Zalunin, and E. Birney. Petabyte-scale innovations at the European Nucleotide Archive. *Nucleic Acids Research*, 37(Database Issue):19–25, January 2009.

[CAD+12] R. Caspi, T. Altman, K. Dreher, C. A. Fulcher, P. Subhraveti, I. M. Keseler, A. Kothari, M. Krummenacker, M. Latendresse, L. A. Mueller, Q. Ong, S. Paley, A. Pujar, A. G. Shearer, M. Travers, D. Weerasinghe, P. Zhang, and P. D. Karp. The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic Acids Research*, 40(Database Issue):742–753, January 2012.

[CBC+12] W. B. Copeland, B. A. Bartley, D. Chandran, M. Galdzicki, K. H. Kim, S. C. Sleight, C. D. Maranas, and H. M. Sauro. Computational tools for metabolic engineering. *Metabolic Engineering*, 14(3):270–280, May 2012.

[CCB+07] Y. B. Chen, A. Chattopadhyay, P. Bergen, C. Gadd, and N. Tannery. The Online Bioinformatics Resources Collection at the University of Pittsburgh Health

Sciences Library System - a one-stop gateway to online bioinformatics databases and software tools. *Nucleic Acids Research*, 35(Database Issue):780–785, January 2007.

[CH02] M. Chen and R. Hofestädt. Quantitative Petri net model of gene regulated metabolic networks in the cell. *In Silico Biology*, 3(3):347–365, December 2002.

[CHA+12] J. M. Cherry, E. L. Hong, C. Amundsen, R. Balakrishnan, G. Binkley, E. T. Chan, K. R. Christie, M. C. Costanzo, S. S. Dwight, S. R. Engel, D. G. Fisk, J. E. Hirschman, B. C. Hitz, K. Karra, C. J. Krieger, S. R. Miyasato, R. S. Nash, J. Park, M. S. Skrzypek, M. Simison, S. Weng, and E. D. Wong. Saccharomyces Genome Database: the genomics resource of budding yeast. *Nucleic Acids Research*, 40(Database Issue):700–705, January 2012.

[CHL08] R. Cheong, A. Hoffmann, and A. Levchenko. Understanding NF-kappaB signaling via mathematical modeling. *Molecular Systems Biology*, 4:192–196, May 2008.

[CJK+11] M. Courtot, N. Juty, C. Knüpfer, D. Waltemath, A. Zhukova, A. Dräger, M. Dumontier, A. Finney, M. Golebiewski, J. Hastings, S. Hoops, S. Keating, D. B. Kell, S. Kerrien, J. Lawson, A. Lister, J. Lu, R. Machne, P. Mendes, M. Pocock, N. Rodriguez, A. Villeger, D. J. Wilkinson, S. Wimalaratne, C. Laibe, M. Hucka, and N. Le Novère. Controlled vocabularies and semantics in systems biology. *Molecular Systems Biology*, 7:1–12, October 2011.

[CLN+03] A. A. Cuellar, C. M. Lloyd, P. F. Nielsen, D. P. Bullivant, D. P. Nickerson, and P. J. Hunter. An Overview of CellML 1.1, a biological model description language. *Simulation*, 79(12):740–747, December 2003.

[Cri58] F. H. C. Crick. Ideas on Protein Synthesis. *Symposia of the Society for Experimental Biology XII*, pages 1–2, July 1958.

[Cri70] F. H. C. Crick. Central dogma of molecular biology. *Nature*, 227(5258):561–563, August 1970.

[CSLR01] T. H. Cormen, C. Stein, C. E. Leiserson, and R. L. Rivest. *Representations of graphs*, volume 1, pages 527–531. MIT Press and McGraw-Hill, 2nd edition, 2001.

[DCP+10] E. Demir, M. P. Cary, S. Paley, K. Fukuda, C. Lemer, I. Vastrik, G. Wu, P. D'Eustachio, C. Schaefer, J. Luciano, F. Schacherer, I. Martinez-Flores, Z. Hu, V. Jimenez-Jacinto, G. Joshi-Tope, K. Kandasamy, A. C. Lopez-Fuentes, H. Mi, E. Pichler, I. Rodchenkov, A. Splendiani, S. Tkachev, J. Zucker, G. Gopinath, H. Rajasimha, R. Ramakrishnan, I. Shah, M. Syed, N. Anwar, Ö. Babur, M. Blinov, E. Brauner, D. Corwin, S. Donaldson, F. Gibbons, R. Goldberg, P. Hornbeck, A. Luna, P. Murray-Rust, E. Neumann, O. Ruebenacker, O. Reubenacker, M. Samwald, M. van Iersel, S. Wimalaratne, K. Allen, B. Braun, M. Whirl-Carrillo, K. H. Cheung, K. Dahlquist, A. Finney, M. Gillespie, E. Glass, L. Gong, R. Haw, M. Honig, O. Hubaut, D. Kane, S. Krupa, M. Kutmon, J. Leonard, D. Marks, D. Merberg, V. Petri, A. Pico, D. Ravenscroft, L. Ren, N. Shah, M. Sunshine, R. Tang, R. Whaley, S. Letovksy, K. H. Buetow, A. Rzhetsky,

V. Schachter, B. S. Sobral, U. Dogrusoz, S. McWeeney, M. Aladjem, E. Birney, J. Collado-Vides, S. Goto, M. Hucka, N. Le Novère, N. Maltsev, A. Pandey, P. Thomas, E. Wingender, P. D. Karp, C. Sander, and G. D. Bader. The BioPAX community standard for pathway data sharing. *Nature Biotechnology*, 28(9):935–942, September 2010.

[DFM+04] A. Doi, S. Fujita, H. Matsuno, M. Nagasaki, and S. Miyano. Constructing biological pathway models with hybrid functional Petri nets. *In Silico Biology*, 4(3):271–291, 2004.

[DG08] O. Demin and I. Goryanin. *Kinetic Modelling in Systems Biology*. Chapman & Hall/CRC, 2008.

[DHAF+12] E. C. Dimmer, R. P. Huntley, Y. Alam-Faruque, T. Sawford, C. O'Donovan, M. J. Martin, B. Bely, P. Browne, W. Mun Chan, R. Eberhardt, M. Gardner, K. Laiho, D. Legge, M. Magrane, K. Pichler, D. Poggioli, H. Sehra, A. Auchincloss, K. Axelsen, M. C. Blatter, E. Boutet, S. Braconi-Quintaje, L. Breuza, A. Bridge, E. Coudert, A. Estreicher, L. Famiglietti, S. Ferro-Rojas, M. Feuermann, A. Gos, N. Gruaz-Gumowski, U. Hinz, C. Hulo, J. James, S. Jimenez, F. Jungo, G. Keller, P. Lemercier, D. Lieberherr, P. Masson, M. Moinat, I. Pedruzzi, S. Poux, C. Rivoire, B. Roechert, M. Schneider, A. Stutz, S. Sundaram, M. Tognolli, L. Bougueleret, G. Argoud-Puy, I. Cusin, P. Duek-Roggli, I. Xenarios, and R. Apweiler. The UniProt-GO Annotation database in 2011. *Nucleic Acids Research*, 40(Database Issue):565–570, January 2012.

[Die00] R. Diestel. *Graphentheorie*, volume 2, pages 1–25. Springer, Berlin, 2000.

[DIKI10] P. S. Demenkov, T. V. Ivanisenko, N. A. Kolchanov, and V. A. Ivanisenko. ANDVisio: A new tool for graphic visualization and analysis of literature mined associative gene networks in the ANDSystem. *In Silico Biology*, 11(3):149–161, December 2010.

[DL04] V. Danos and C. Laneve. Formal molecular biology. *Theoretical Computer Science*, 325(1):69–110, September 2004.

[DLRF10] J. De Las Rivas and C. Fontanillo. Protein-protein interactions essentials: key concepts to building and analyzing interactome networks. *PLoS Computational Biology*, 6(6):1–8, June 2010.

[DPG93] F. Dorkeld, G. Perrière, and C. Gautier. Object-oriented modelling in molecular biology. *Proceedings of the Artificial Intelligence and Genome Workshop, JCAI*, pages 99–106, 1993.

[DRWM05] P. C. De Ruiter, V. Wolters, and J. C. Moore. *Dynamic Food Webs*, pages 3–10. Multispecies Assemblages, Ecosystem Development, and Environmental Change. Academic Press, December 2005.

[DS99] K. R. Dronamraju and E. Schrödinger. Erwin Schrödinger and the origins of molecular biology. *Genetics*, 153(3):1071–1076, November 1999.

[EE93] G. B. Ermentrout and L. Edelsteinkeshet. Cellular Automata Approaches to Biological Modeling. *Journal of Theoretical Biology*, 160(1):97–133, 1993.

[EFZ08] F. Erhard, C. C. Friedel, and R. Zimmer. FERN - a Java framework for stochastic simulation and evaluation of reaction networks. *BMC Bioinformatics*, 9:356, 2008.

[EHC03] S. Efroni, D. Harel, and I. R. Cohen. Toward rigorous comprehension of biological complexity: modeling, execution, and visualization of thymic T-cell maturation. *Genome Research*, 13(11):2485–2497, October 2003.

[EHG89] O. Enger, B. Husevåg, and J. Goksøyr. Presence of the fish pathogen Vibrio salmonicida in fish farm sediments. *Applied and Environmental Microbiology*, 55(11):2815–2818, October 1989.

[Eig71] M. Eigen. Selforganization of matter and the evolution of biological macromolecules. *Naturwissenschaften*, 58(10):465–523, October 1971.

[EN10] R. Elmasri and S. Navathe. *Fundamentals of Database Systems*, pages 3–27. Addison Wesley, 6th edition, April 2010.

[EPN00] J. Elek, K. H. Park, and R. Narayanan. Microarray-based expression profiling in prostate tumors. *In Vivo*, 14(1):173–182, 2000.

[FAL+05] P. Fritzson, P. Aronsson, H. Lundvall, K. Nyström, A. Pop, L. Saldamli, and D. Broman. The openmodelica modeling, simulation, and software development environment. *Simulation News Europe*, 44(45):1588–1595, December 2005.

[FBM+05] J. A. Fox, S. L. Butland, S. McMillan, G. Campbell, and B. F. Ouellette. The Bioinformatics Links Directory: a compilation of molecular biology web servers. *Nucleic Acids Research*, 33(Web Server issue):3–24, July 2005.

[FDM09] C. Francesca, M. Daniele, and G. Marco. Modeling Biological Pathways: An Object-Oriented like Methodology Based on Mean Field Analysis. *Transactions of the IRE*, 1:117–122, 2009.

[FGB12] R. D. Finn, P. P. Gardner, and A. Bateman. Making your database available through Wikipedia: the pros and cons. *Nucleic Acids Research*, 40(Database Issue):9–12, January 2012.

[FH03] A. Finney and M. Hucka. Systems biology markup language: Level 2 and beyond. *Biochemical Society Transactions*, 31:1472–1473, December 2003.

[Fin93] A. Finkel. A Minimal coverability graph for petri nets. *Papers from the 12th International Conference on Applications and Theory of Petri Nets: Advances in Petri Nets*, 1:210–243, 1993.

[FLM94] A. Frick, A. Ludwig, and H. Mehldau. A fast, adaptive layout algorithm for undirected graphs. *Springer Verlag*, 894 of Lecture Notes in Computer Science:1–16, 1994.

[FMKT03] A. Funahashi, M. Morohashi, H. Kitano, and N. Tanimura. CellDesigner: a process diagram editor for gene-regulatory and biochemical networks. *Biosilico*, 1(5):159–162, 2003.

[Fre77] L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, 40(1):35–41, 1977.

[Fry08] B. Fry. *Visualizing Data*. O'Reilly Media, 1st edition, 2008.

[Gar70] M. Gardner. Mathematical games: The fantastic combinations of John Conway's new solitaire game "life". *Scientific American*, 1:120–123, 1970.

[GBR+99] R. Gupta, H. Birch, K. Rapacki, S. Brunak, and J. E. Hansen. O-GLYCBASE version 4.0: a revised database of O-glycosylated proteins. *Nucleic Acids Research*, 27(1):370–372, January 1999.

[GBSH+08] E. Grafahrend-Belau, F. Schreiber, M. Heiner, A. Sackmann, B. H. Junker, S. Grunwald, A. Speer, K. Winder, and I. Koch. Modularization of biochemical networks based on classification of Petri net t-invariants. *BMC Bioinformatics*, 9(1):90–117, 2008.

[Ger04] C. Gershenson. Introduction to random boolean networks. *eprint arXiv*, nlin/0408006:1–14, August 2004.

[GFS12] M. Y. Galperin and X. M. Fernandez-Suarez. The 2012 Nucleic Acids Research Database Issue and the online Molecular Biology Database Collection. *Nucleic Acids Research*, 40(Database Issue):1–8, January 2012.

[GGM+10] M. Gizzatkulov, I. Goryanin, E. A. Metelkin, E. A. Mogilevskaya, K. V. Peskov, and O. V. Demin. DBSolve Optimum: a software package for kinetic modeling which allows dynamic visualization of simulation results. *BMC Systems Biology*, 4(1):109, 2010.

[GMTH06] M. R. Grant, K. E. Mostov, T. D. Tlsty, and C. A. Hunt. Simulating properties of in vitro epithelial cell morphogenesis. *PLoS Computational Biology*, 2(10):129–134, October 2006.

[GP98] P. J. E. Goss and J. Peccoud. Quantitative modeling of stochastic systems in molecular biology by using stochastic Petri nets. *Proceedings of the National Academy of Sciences of the United States of America*, 95(12):6750–6755, 1998.

[Gra08] S. H. Gray. *Food Webs: Interconnecting Food Chains*, pages 1–48. Interconnecting Food Chains. Exploring Science, January 2008.

[GTL06] A. B. Goryachev, D. J. Toh, and T. Lee. Systems analysis of a quorum sensing network: Design constraints imposed by the functional requirements, network topology and kinetic constants. *Biosystems*, 83(2-3):178–187, February 2006.

[HBCW03] H. Huang, W. C. Barker, Y. Chen, and C. H. Wu. iProClass: an integrated database of protein family, function and structure information. *Nucleic Acids Research*, 31(1):390–392, January 2003.

[HC86] G. H. Hamm and G. N. Cameron. The EMBL data library. *Nucleic Acids Research*, 14(1):5–9, January 1986.

[HdHTDG07] M. Huisman, E. de Heer, P. Ten Dijke, and J. Grote. Transforming growth factor beta and wound healing in human cholesteatoma. *Laryngoscope*, 118(1):94–98, December 2007.

[HFS+03] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish-Bowden, A. A. Cuellar, S. Dronov, E. D. Gilles, M. Ginkel, V. Gor, I. I. Goryanin, W. J. Hedley, T. C. Hodgman, J. H. Hofmeyr, P. J. Hunter, N. S. Juty, J. L. Kasberger, A. Kremling, U. Kummer, N. Le Novere, L. M. Loew, D. Lucio, P. Mendes, E. Minch, E. D. Mjolsness, Y. Nakayama, M. R. Nelson, P. F. Nielsen, T. Sakurada, J. C. Schaff, B. E. Shapiro, T. S. Shimizu, H. D. Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner, and J. Wang. The systems biology markup language (SBML): a medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531, March 2003.

[HH95] P. Hage and F. Harary. Eccentricity and centrality in networks. *Social networks*, 17(1):57–63, 1995.

[HK05] M. W. Hahn and A. D. Kern. Comparative genomics of centrality and essentiality in three eukaryotic protein-interaction networks. *Molecular Biology and Evolution*, 22(4):803–806, April 2005.

[HKA+05] R. Hoffmann, M. Krallinger, E. Andres, J. Tamames, C. Blaschke, and A. Valencia. Text mining for metabolic pathways, signaling cascades, and protein networks. *Signal Transduction Knowledge Environment (STKE)*, 1(283):1–21, 2005.

[HKJ+11] K. Hippe, B. Kormeier, S. Janowski, T. Töpel, and R. Hofestädt. DAWIS-M.D. 2.0 - A Data Warehouse Information System for Metabolic Data. *Proceedings of the 7th International Symposium on Integrative Bioinformatics*, 1:720–725, 2011.

[HLH+08] E. Hjerde, M. Lorentzen, M. T. G. Holden, K. Seeger, S. Paulsen, N. Bason, C. Churcher, D. Harris, H. Norbertczak, M. A. Quail, S. Sanders, S. Thurston, J. Parkhill, N. Willassen, and N. R. Thomson. The genome sequence of the fish pathogen Aliivibrio salmonicida strain LFI1238 shows extensive evidence of gene decay. *BMC Genomics*, 9(1):616–630, 2008.

[HMPB+04] H. Hermjakob, L. Montecchi-Palazzi, G. Bader, J. Wojcik, L. Salwinski, A. Ceol, S. Moore, S. Orchard, U. Sarkans, C. von Mering, B. Roechert, S. Poux, E. Jung, H. Mersch, P. Kersey, M. Lappe, Y. Li, R. Zeng, D. Rana, M. Nikolski, H. Husi, C. Brun, K. Shanker, S. G. Grant, C. Sander, P. Bork, W. Zhu, A. Pandey, A. Brazma, B. Jacq, M. Vidal, D. Sherman, P. Legrain, G. Cesareni, I. Xenarios, D. Eisenberg, B. Steipe, C. Hogue, and R. Apweiler. The HUPO PSI's molecular interaction format–a community standard for the representation of protein interaction data. *Nature Biotechnology*, 22(2):177–183, February 2004.

[Hof94] R. Hofestädt. A Petri net application to model metabolic processes. *Systems Analysis Modelling Simulation*, 16(2):113–122, 1994.

[Hol98] J. R. Holum. *Fundamentals of general, organic, and biological chemistry*. Wiley, 6th edition, 1998.

[Hop82] F. C. Hoppensteadt. *Mathematical Methods of Population Biology*, pages 1–28. Cambridge University Press, February 1982.

[HRSR08] M. Heiner, R. Richter, M. Schwarick, and C. Rohr. Snoopy-A Tool to Design and Execute Graph-Based Formalisms. *Petri Net Newsletter*, 74:8–22, 2008.

[HSG+06] S. Hoops, S. Sahle, R. Gauges, C. Lee, J. Pahle, N. Simus, M. Singhal, L. Xu, P. Mendes, and U. Kummer. COPASI - a COmplex PAthway SImulator. *Bioinformatics*, 22(24):3067–3074, December 2006.

[HT98] R. Hofestädt and S. Thelen. Quantitative modeling of biochemical networks. *In Silico Biology*, 1(1):39–53, 1998.

[HV04] R. Hoffmann and A. Valencia. A gene network for navigating the literature. *Nature genetics*, 36(7):664–664, June 2004.

[HZ09] M. V. Han and C. M. Zmasek. phyloXML: XML for evolutionary biology and comparative genomics. *BMC Bioinformatics*, 10:356, 2009.

[IK01] L. Itti and C. Koch. Computational modelling of visual attention. *Nature Review Neuroscience*, 2(3):194–203, March 2001.

[JA11] D. Jayasinghe and A. Azeez. *Apache Axis2 Web Services*. Packt Publishing, 2nd edition, February 2011.

[Jan08] S. J. Janowski. An integrative bioinformatics solution to visualize and examine biological networks. *Master thesis at the Bielefeld University, Germany*, October 2008.

[Jan09] S. Janowski. Erstellung und Realisierung eines Corporate Designs für die Software-Applikation Netzwerkeditor. *Bachelor thesis at the Bielefeld University, Germany*, June 2009.

[JGW04] C. G. Johnson, J. P. Goldman, and J. G. William. Simulating complex intracellular processes using object-oriented computational modelling. *Progress in Biophysics and Molecular Biology*, 86(3):379–406, 2004.

[JKH+11] S. J. Janowski, B. Kormeier, K. Hippe, Q. Nguyen, S. Hong, R. Hofestädt, J. Stoye, B. Kaltschmidt, and C. Kaltschmidt. Reconstruction and analysis of biological networks based on large scale data from the NF-κB pathway. *Proceedings of IB 2011 (International Symposium on Integrative Bioinformatics 2011)*, 1:1–3, 2011.

[JKT+10] S. Janowski, B. Kormeier, T. Töpel, K. Hippe, R. Hofestädt, N. Willassen, R. Friesen, S. Rubert, D. Borck, P. Haugen, and M. Chen. Modeling of Cell-to-Cell Communication Processes with Petri Nets Using the Example of Quorum Sensing. *In Silico Biology*, 10(1):27–48, 2010.

[JMBO01] H. Jeong, S. P. Mason, A. Barabási, and Z. N. Oltvai. Lethality and centrality in protein networks. *Nature*, 411(6833):41–42, May 2001.

[JS11] B. H. Junker and F. Schreiber. *Analysis of Biological Networks*. Wiley-Interscience, 1st edition, September 2011.

[KAAEV05] M. Krallinger, M. Alonso-Allende Erhardt, and A. Valencia. Text-mining approaches in molecular biology and biomedicine. *Drug Discovery Today*, 10(6):439–445, 2005.

[KAB+12] S. Kerrien, B. Aranda, L. Breuza, A. Bridge, F. Broackes-Carter, C. Chen, M. Duesbury, M. Dumousseau, M. Feuermann, U. Hinz, C. Jandrasits, R. C. Jimenez, J. Khadake, U. Mahadevan, P. Masson, I. Pedruzzi, E. Pfeiffenberger, P. Porras, A. Raghunath, B. Roechert, S. Orchard, and H. Hermjakob. The IntAct molecular interaction database in 2012. *Nucleic Acids Research*, 40(Database Issue):841–846, January 2012.

[Kar00] P. D. Karp. An ontology for biological function based on molecular interactions. *Bioinformatics*, 16(3):269–285, March 2000.

[Kau69] S. A. Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *Journal of Theoretical Biology*, 22(3):437–467, February 1969.

[KBMCV+09] I. M. Keseler, C. Bonavides-Martinez, J. Collado-Vides, S. Gama-Castro, R. P. Gunsalus, D. A. Johnson, M. Krummenacker, L. M. Nolan, S. Paley, I. T. Paulsen, M. Peralta-Gil, A. Santos-Zavaleta, A. G. Shearer, and P. D. Karp. EcoCyc: A comprehensive view of Escherichia coli biology. *Nucleic Acids Research*, 37(Database Issue):464–470, January 2009.

[KBT+06] J. Köhler, J. Baumbach, J. Taubert, M. Specht, A. Skusa, A. Rüegg, C. Rawlings, P. Verrier, and S. Philippi. Graph-based analysis and visualization of experimental results with ONDEX. *Bioinformatics*, 22(11):1383–1390, June 2006.

[KCH01] N. Kam, I. R. Cohen, and D. Harel. Proceedings IEEE Symposia on Human-Centric Computing Languages and Environments (Cat. No.01TH8587). In *HCC 2001. IEEE Symposium on Human-Centric Computing Languages and Environments*, volume 1, pages 15–22. IEEE, 2001.

[Kea06] J. D. Kearns. IκB provides negative feedback to control NF-κB oscillations, signaling dynamics, and inflammatory gene expression. *The Journal of Cell Biology*, 173(5):659–664, June 2006.

[Kep07] F. Kepes. *Biological networks*. World Scientific Pub Co Inc, 1st edition, 2007.

[KGDO05] V. Kunin, L. Goldovsky, N. Darzentas, and C. A. Ouzounis. The net of life: Reconstructing the microbial phylogenetic network. *Genome Research*, 15(7):954–959, June 2005.

[KGJ10] A. Kozomara and S. Griffiths-Jones. miRBase: integrating microRNA annotation and deep-sequencing data. *Nucleic Acids Research*, 39(Database Issue):152–157, December 2010.

[KGM+08] R. Kottmann, T. Gray, S. Murphy, L. Kagan, S. Kravitz, T. Lombardot, D. Field, and F. O. Glöckner. A standard MIGS/MIMS compliant XML Schema: toward the development of the Genomic Contextual Data Markup Language (GCDML). *OMICS*, 12(2):115–121, June 2008.

[KGS+12] M. Kanehisa, S. Goto, Y. Sato, M. Furumichi, and M. Tanabe. KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Research*, 40(Database Issue):109–114, January 2012.

[KHA+10] B. Kormeier, K. Hippe, P. Arrigo, T. Töpel, S. Janowski, and R. Hofestädt. Reconstruction of biological networks based on life science data integration. *Journal of integrative bioinformatics*, 7(2):146–159, 2010.

[KHH11] B. Kormeier, K. Hippe, and R. Hofestädt. Data Warehouses in Bioinformatics: Integration of Molecular Biological Data. *it-Information Technology*, 5:241–248, 2011.

[KJB+12] C. Klenke, S. Janowski, D. Borck, D. Widera, L. A. Ebmeyer, J. Kalinowski, A. Leichtle, R. Hofestädt, T. Upile, C. Kaltschmidt, B. Kaltschmidt, and H. Sudhoff. Identification of Novel Cholesteatoma-related Gene Expression Signatures Using Full-genome Microarrays. *PloS One*, 7(12):1–14, 2012.

[KK09] B. Kaltschmidt and C. Kaltschmidt. NF-κB in the nervous system. *Cold Spring Harbor Perspectives in Biology*, 1(3):1–13, August 2009.

[KKKJ06] K. H. Kwon, S. J. Kim, H. J. Kim, and H. Jung. Analysis of gene expression profiles in cholesteatoma using oligonucleotide microarray. *Acta Oto-Laryngologica*, 126(7):691–697, June 2006.

[KLH+07] E. Klipp, W. Liebermeister, A. Helbig, A. Kowald, and J. Schaber. Systems biology standards–the community speaks. *Nature Biotechnology*, 25(4):390–391, April 2007.

[KLP+05] D. Koschützki, K. A. Lehmann, L. Peeters, S. Richter, D. Tenfelde-Podehl, and O. Zlotowski. *Network Analysis: Centrality Indices - Lecture Notes in Computer Science*, volume 3418. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[KM69] R. M. Karp and R. E. Miller. Parallel program schemata. *Journal of Computer and system Sciences*, 3(2):147–195, 1969.

[KOMK+05] P. D. Karp, C. A. Ouzounis, C. Moore-Kochlacs, L. Goldovsky, P. Kaipa, D. Ahrén, S. Tsoka, N. Darzentas, V. Kunin, and N. López-Bigas. Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. *Nucleic Acids Research*, 33(19):6083–6089, October 2005.

[Kos11] D. Koschützki. Phd thesis: Zentralitätsanalyse molekularbiologischer Netzwerke. *Institut für Informatik der Naturwissenschaftlichen Fakultät III der Martin-Luther-Universität Halle-Wittenberg*, 2011.

[KPE03] K. J. Kauffman, P. Prakash, and J. S. Edwards. Advances in flux balance analysis. *Current Opinion in Biotechnology*, 14(5):491–496, September 2003.

[KPGK+09] T. S. Keshava Prasad, R. Goel, K. Kandasamy, S. Keerthikumar, S. Kumar, S. Mathivanan, D. Telikicherla, R. Raju, B. Shafreen, A. Venugopal, L. Balakrishnan, A. Marimuthu, S. Banerjee, D. S. Somanathan, A. Sebastian, S. Rani, S. Ray, C. J. Harrys Kishore, S. Kanth, M. Ahmed, M. K. Kashyap, R. Mohmood, Y. L. Ramachandra, V. Krishna, B. A. Rahiman, S. Mohan,

P. Ranganathan, S. Ramabadran, R. Chaerkady, and A. Pandey. Human Protein Reference Database - 2009 update. *Nucleic Acids Research*, 37(Database Issue):767–772, January 2009.

[KPJ+10] M. Kalas, P. Puntervoll, A. Joseph, E. Bartaseviciute, A. Topfer, P. Venkataraman, S. Pettifer, J. C. Bryne, J. Ison, C. Blanchet, K. Rapacki, and I. Jonassen. BioXSD: the common data-exchange format for everyday bioinformatics web services. *Bioinformatics*, 26(18):540–546, September 2010.

[KPV+06] M. Krull, S. Pistor, N. Voss, A. Kel, I. Reuter, D. Kronenberg, H. Michael, K. Schwarzer, A. Potapov, C. Choi, O. Kel-Margoulis, and E. Wingender. TRANSPATH: an information resource for storing and visualizing signaling pathways and their pathological aberrations. *Nucleic Acids Research*, 34(Database Issue):546–551, January 2006.

[KSA08] E. Klussmann, J. Scott, and E. M. Aandahl. *Protein-protein interactions as new drug targets*. Springer Verlag, 1st edition, 2008.

[Kur07] W. Kurth. Specification of morphological models with L-systems and relational growth grammars. *Journal of Interdisciplinary Image Science*, 5:1–25, 2007.

[LBP+12] L. Licata, L. Briganti, D. Peluso, L. Perfetto, M. Iannuccelli, E. Galeota, F. Sacco, A. Palma, A. P. Nardozza, E. Santonico, L. Castagnoli, and G. Cesareni. MINT, the molecular interaction database: 2012 update. *Nucleic Acids Research*, 40(Database Issue):857–861, January 2012.

[LC03] Z. Lacroix and T. Critchlow. *Bioinformatics: Managing Scientific Data*. Morgan Kaufmann, 1st edition, August 2003.

[LDD+09] S. Lee, A. M. Dudley, D. Drubin, P. A. Silver, N. J. Krogan, D. Pe'er, and D. Koller. Learning a Prior on Regulatory Potential from eQTL Data. *PLoS Genetics*, 5(1):1–24, January 2009.

[Lew12] M. Lewinski. Mathematischer Netzwerkvergleich anhand multipler Netzwerk Charakteristika. *Master thesis at the Bielefeld University, Germany*, pages 58–70, January 2012.

[LJH+08] C. Liang, P. Jaiswal, C. Hebbard, S. Avraham, E. S. Buckler, T. Casstevens, B. Hurwitz, S. McCouch, J. Ni, A. Pujar, D. Ravenscroft, L. Ren, W. Spooner, I. Tecle, J. Thomason, C. W. Tung, X. Wei, I. Yap, K. Youens-Clark, D. Ware, and L. Stein. Gramene: a growing plant comparative genomics resource. *Nucleic Acids Research*, 36(Database Issue):947–953, January 2008.

[LNHM+09] N. Le Novère, M. Hucka, H. Mi, S. Moodie, F. Schreiber, A. Sorokin, E. Demir, K. Wegner, M. I. Aladjem, S. M. Wimalaratne, F. T. Bergman, R. Gauges, P. Ghazal, H. Kawaji, L. Li, Y. Matsuoka, A. Villeger, S. E. Boyd, L. Calzone, M. Courtot, U. Dogrusoz, T. C. Freeman, A. Funahashi, S. Ghosh, A. Jouraku, S. Kim, F. Kolpakov, A. Luna, S. Sahle, E. Schmidt, S. Watterson, G. Wu, I. Goryanin, D. B. Kell, C. Sander, H. Sauro, J. L. Snoep, K. Kohn, and H. Kitano. The Systems Biology Graphical Notation. *Nature Biotechnology*, 27(8):735–741, August 2009.

[LPW+06] T. J. Lee, Y. Pouliot, V. Wagner, P. Gupta, D. W. Stringer-Calvert, J. D. Tenenbaum, and P. D. Karp. BioWarehouse: a bioinformatics database warehouse toolkit. *BMC Bioinformatics*, 7:170, 2006.

[LS72] D. J. Lim and W. H. Saunders. Acquired Cholesteatoma - Light and Electron-Microscopic Observations. *Ann Otol Rhinol Laryngol. 1*, 81(1):1–11, February 1972.

[LS94] M. Lee and L. L. J. Starr. Object-Oriented Analysis in the Real-World. *Embedded Systems Programming*, 7(6):24–37, 1994.

[LSG+06] C. Li, S. Suzuki, Q. W. Ge, M. Nakata, H. Matsuno, and S. Miyano. Structural modeling and analysis of signaling pathways based on Petri nets. *Journal of Bioinformatics and Computational Biology*, 4(5):1119–1140, September 2006.

[LVC+08] N. Y. K. Li, K. Verdolini, G. Clermont, Q. Mi, E. N. Rubinstein, P. A. Hebda, and Y. Vodovotz. A Patient-Specific in silico model of Inflammation and Healing Tested in Acute Vocal Fold Injury. *PLoS ONE*, 3(7):1–11, July 2008.

[MCR+11] D. Machado, R. S. Costa, M. Rocha, E. C. Ferreira, B. Tidor, and I. Rocha. Modeling formalisms in Systems Biology. *AMB Express*, 1(1):45–59, December 2011.

[MDDM00] H. Matsuno, A. Doi, R. Drath, and S. Miyano. Genomic object net: object oriented representation of biological systems. *Genome Informatics Series*, 11:229–230, 2000.

[MDNM00] H. Matsuno, A. Doi, M. Nagasaki, and S. Miyano. Hybrid Petri net representation of gene regulatory network. *Pacific Symposium on Biocomputing*, 1:341–352, 2000.

[Men93] P. Mendes. GEPASI: a software package for modelling the dynamics, steady states and control of biochemical and other systems. *Computer Applications in the Biosciences*, 9(5):563–571, September 1993.

[Men97] P. Mendes. Biochemistry by numbers: simulation of biochemical pathways with Gepasi 3. *Trends in Biochemical Sciences*, 22(9):361–363, August 1997.

[Mey97] E. F. Meyer. The first years of the Protein Data Bank. *Protein Science*, 6(7):1591–1597, July 1997.

[MFM+06] R. Machné, A. Finney, S. Müller, J. Lu, S. Widder, and C. Flamm. The SBML ODE Solver Library: a native API for symbolic and fast numerical analysis of reaction networks. *Bioinformatics*, 22(11):1406–1407, May 2006.

[MGG+09] L. Matthews, G. Gopinath, M. Gillespie, M. Caudy, D. Croft, B. de Bono, P. Garapati, J. Hemish, H. Hermjakob, B. Jassal, A. Kanapin, S. Lewis, S. Mahajan, B. May, E. Schmidt, I. Vastrik, G. Wu, E. Birney, L. Stein, and P. D'Eustachio. Reactome knowledgebase of human biological pathways and processes. *Nucleic Acids Research*, 37(Database Issue):619–622, January 2009.

[Mil98] G. Miller. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*. A Bradford Book, 1st edition, May 1998.

[MK96] A. R. Mushegian and E. V. Koonin. A minimal gene set for cellular life derived by comparison of complete bacterial genomes. *Proceedings of the National Academy of Sciences*, 93(19):10268–10273, September 1996.

[MKA+00] R. McEntire, P. Karp, N. Abernethy, D. Benton, G. Helt, M. DeJongh, R. Kent, A. Kosky, S. Lewis, D. Hodnett, E. Neumann, F. Olken, D. Pathak, P. Tarczy-Hornoch, L. Toldo, and T. Topaloglou. An evaluation of ontology exchange languages for bioinformatics. *International Conference on Intelligent Systems for Molecular Biology*, 8:239–250, December 2000.

[MKMF+06] V. Matys, O. V. Kel-Margoulis, E. Fricke, I. Liebich, S. Land, A. Barre-Dirrie, I. Reuter, D. Chekmenev, M. Krull, K. Hornischer, N. Voss, P. Stegmaier, B. Lewicki-Potapov, H. Saxel, A. E. Kel, and E. Wingender. TRANSFAC and its module TRANSCompel: transcriptional gene regulation in eukaryotes. *Nucleic Acids Research*, 34(Database Issue):108–110, January 2006.

[MMR+10] A. K. Miller, J. Marsh, A. Reeve, A. Garny, R. Britten, M. Halstead, J. Cooper, D. P. Nickerson, and P. F. Nielsen. An overview of the CellML API and its implementation. *BMC Bioinformatics*, 11:178–180, 2010.

[MRC+09] L. Milanesi, P. Romano, G. Castellani, D. Remondini, and P. Lio. Trends in modeling Biomedical Complex Systems. *BMC Bioinformatics*, 10(12):1–13, 2009.

[MTA+02] H. Matsuno, Y. Tanaka, H. Aoshima, A. Doi, M. Matsui, and S. Miyano. Biopathways representation and simulation on hybrid functional Petri net. *In Silico Biology*, 3(3):389–404, December 2002.

[MZR03] L. A. Mueller, P. Zhang, and S. Y. Rhee. AraCyc: a biochemical pathway database for Arabidopsis. *Plant Physiology*, 132(2):453–460, June 2003.

[NdBC+10] L. M. A. Nunes, A. L. M. de Barros, R. V. R. Cal, C. T. A. Nunes, and F. D. Lima. Giant Cholesteatoma: Case and Literature Review Report. *International Archives of Otorhinolaryngology*, 14(1):113, 2010.

[NHE12] Q. Nguyen, S. Hong, and P. Eades. TGI-EB: A New Framework for Edge Bundling integrating Topology, Geometry and Importance. *Proceedings of Graph Drawing 2011*, 1:123–135, 2012.

[NSJ+10] M. Nagasaki, A. Saito, E. Jeong, C. Li, K. Kojima, E. Ikeda, and S. Miyano. Cell Illustrator 4.0: a computational platform for systems biology. *In Silico Biology*, 10(1):5–26, 2010.

[OCN+11] J. D. Orth, T. M. Conrad, J. Na, J. A. Lerman, H. Nam, A. M. Feist, and B. O. Palsson. A comprehensive genome-scale reconstruction of Escherichia coli metabolism. *Molecular Systems Biology*, 7:1–9, October 2011.

[PB11] S. Proß and B. Bachmann. An Advanced Environment for Hybrid Modeling of Biological Systems Based on Modelica. *Journal of integrative bioinformatics*, 8(1):152, 2011.

[PCE+12] M. Punta, P. C. Coggill, R. Y. Eberhardt, J. Mistry, J. Tate, C. Boursnell, N. Pang, K. Forslund, G. Ceric, J. Clements, A. Heger, L. Holm, E. L. Sonnhammer, S. R. Eddy, A. Bateman, and R. D. Finn. The Pfam protein families database. *Nucleic Acids Research*, 40(Database Issue):290–301, January 2012.

[PCTK+10] E. Portales-Casamar, S. Thongjuea, A. T. Kwon, D. Arenillas, X. Zhao, E. Valen, D. Yusuf, B. Lenhard, W. W. Wasserman, and A. Sandelin. JASPAR 2010: the greatly expanded open-access database of transcription factor binding profiles. *Nucleic Acids Research*, 38(Database Issue):105–110, January 2010.

[Pe'05] D. Pe'er. Bayesian network analysis of signaling networks: a primer. *Audio, Transactions of the IRE Professional Group on*, 2005(281):14–24, April 2005.

[Pet62] C. A. Petri. Dissertation: Kommunikation mit Automaten. *Schriften des Rheinisch-Westfälischen Institutes für Instrumentelle Mathematik an der Universität Bonn*, 1962.

[Pim02] S. L. Pimm. *Food Webs*, pages 1–34. University Of Chicago Press, May 2002.

[PJB+12] S. Proß, S. J. Janowski, B. Bachmann, C. Kaltschmidt, and B. Kaltschmidt. PNlib- A Modelica Library for Simulation of Biological Systems Based on Extended Hybrid Petri Nets. *Proceedings of the 3rd International Workshop on Biological Processes and Petri Nets*, 852:1–16, 2012.

[PJHB12] S. Proß, S. J. Janowski, R. Hofestädt, and Bachman B. A New Object-Oriented Petri Net Simulation Environment Based On Modelica. *Online proceedings of the 2012 Winter simulation Conference, IEEE*, pages 1–12, 2012.

[PJS+06] S. K. Palaniswamy, S. James, H. Sun, R. S. Lamb, R. V. Davuluri, and E. Grotewold. AGRIS and AtRegNet - a platform to link cis-regulatory elements and transcription factors into regulatory networks. *Plant Physiology*, 140(3):818–829, March 2006.

[PL90] P. Prusinkiewicz and A. Lindenmayer. *The algorithmic beauty of plants*, pages 1–46. Springer Verlag, 1990.

[PP08] A. Panchenko and T. Przytycka. *Protein-Protein Interactions and Networks*, pages 1–53. Identification, Computer Analysis, and Prediction. Springer-Verlag New York Incorporated, 1st edition, August 2008.

[PR08] C. Petri and W. Reisig. Petri net. *Scholarpedia*, 3(4):6477, 2008.

[PSM+11] G. A. Pavlopoulos, M. Secrier, C. N. Moschopoulos, T. G. Soldatos, S. Kossida, J. Aerts, R. Schneider, and P. G. Bagos. Using graph theory to analyze biological networks. *BioData Mining*, 4(1):10, April 2011.

[PUD+01] J. D. Peterson, L. A. Umayam, T. Dickinson, E. K. Hickey, and O. White. The Comprehensive Microbial Resource. *Nucleic Acids Research*, 29(1):123–125, January 2001.

[PW08] L. Priese and H. Wimmel. *Petri-Netze*, pages 49–90. Springer Verlag, 2nd edition, March 2008.

[QLH+00] J. Quackenbush, F. Liang, I. Holt, G. Pertea, and J. Upton. The TIGR gene indices: reconstruction and representation of expressed gene sequences. *Nucleic Acids Research*, 28(1):141–145, January 2000.

[RBB+03] S. Y. Rhee, W. Beavis, T. Z. Berardini, G. Chen, D. Dixon, A. Doyle, M. Garcia-Hernandez, E. Huala, G. Lander, M. Montoya, N. Miller, L. A. Mueller, S. Mundodi, L. Reiser, J. Tacklind, D. C. Weems, Y. Wu, I. Xu, D. Yoo, J. Yoon, and P. Zhang. The Arabidopsis Information Resource (TAIR): a model organism database providing a centralized, curated gateway to Arabidopsis biology, research materials and community. *Nucleic Acids Research*, 31(1):224–228, January 2003.

[RE91] J. Rumbaugh and F. Eddy. *Object-Oriented Modeling and Design*, pages 15–57. Prentice-Hall, 1991.

[Rei92] W. Reisig. *A primer in Petri net design.* Springer Verlag, 1st edition, 1992.

[RMH10] C. Rohr, W. Marwan, and M. Heiner. Snoopy- a unifying Petri net framework to investigate biomolecular networks. *Bioinformatics*, 26(7):974–975, April 2010.

[RML92] V. N. Reddy, M. L. Mavrovouniotis, and M. N. Liebman. Petri net representations in metabolic pathways. *International Conference on Intelligent Systems for Molecular Biology*, 1:328–336, December 1992.

[RS06] N. C. Reading and V. Sperandio. Quorum sensing: the many languages of bacteria. *FEMS Microbiology Letters*, 254(1):1–11, January 2006.

[RSO+07] M. E. Ritchie, J. Silver, A. Oshlack, M. Holmes, D. Diyagama, A. Holloway, and G. K. Smyth. A comparison of background correction methods for two-colour microarrays. *Bioinformatics*, 23(20):2700–2707, 2007.

[Sab66] G. Sabidussi. The centrality index of a graph. *Psychometrika*, 31(4):581–603, December 1966.

[SAK+09] C. F. Schaefer, K. Anthony, S. Krupa, J. Buchoff, M. Day, T. Hannay, and K. H. Buetow. PID: the Pathway Interaction Database. *Nucleic Acids Research*, 37(Database Issue):674–679, January 2009.

[San03] P. Sandhu. *The MathML Handbook*, pages 1–10. Charles River Media, 2003.

[SAO+03] P. Shannon, M. Andrew, O. Ozier, N. S. Baliga, J. T. Wang, D. Ramage, N. Amin, B. Schwikowski, and T. Ideker. Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome Research*, 13(11):2498–2504, November 2003.

[SB08] H. M. Sauro and F. T. Bergmann. Standards and ontologies in computational systems biology. *Essays in Biochemistry*, 45:211–222, 2008.

[Sch55] E. Schrödinger. *What is life? The physical aspect of the living cell*, pages 1–12. The University Press, 1955.

[Sch01] S. Schauder. The languages of bacteria. *Genes & Development*, 15(12):1468–1480, June 2001.

[SCH05] P. Sethupath, B. Corda, and A. G. Hatzigeorgiou. TarBase: A comprehensive database of experimentally supported animal microRNA targets. *RNA*, 12(2):192–197, December 2005.

[SD10] B. J. Strasser and M. O. Dayhoff. Collecting, comparing, and computing sequences: the making of Margaret O. Dayhoff's Atlas of Protein Sequence and Structure, 1954-1965. *Journal of the History of Biology*, 43(4):623–660, 2010.

[SFK+11] D. Szklarczyk, A. Franceschini, M. Kuhn, M. Simonovic, A. Roth, P. Minguez, T. Doerks, M. Stark, J. Muller, P. Bork, L. J. Jensen, and C. von Mering. The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Research*, 39(Database Issue):561–568, January 2011.

[SGC+05] G. K. Smyth, R. Gentleman, V. J. Carey, W. Huber, R. A. Irizarry, and S. Dudoit. *Statistics for Biology and Health.* Springer-Verlag, New York, 2005.

[SGC+11] M. Scheer, A. Grote, A. Chang, I. Schomburg, C. Munaretto, M. Rother, C. Söhngen, M. Stelzer, J. Thiele, and D. Schomburg. BRENDA, the enzyme information system in 2011. *Nucleic Acids Research*, 39(Database Issue):670–676, January 2011.

[SGTS11] N. B. Shunyu, S. D. Gupta, A. Thakar, and S. C. Sharma. Histological and immunohistochemical study of pars tensa retraction pocket. *Transactions of the IRE Professional Group*, 145(4):628–634, September 2011.

[SHF+02] H. M. Sauro, M. Hucka, A. Finney, C. Wellock, H. Bolouri, J. Doyle, and H. Kitano. Next generation simulation tools: the Systems Biology Workbench and BioSPICE integration. *OMICS: A Journal of Integrative Biology*, 7(4):355–372, December 2002.

[SHK06] A. Sackmann, M. Heiner, and I. Koch. Application of Petri net based analysis techniques to signal transduction pathways. *BMC Bioinformatics*, 7(1):482–499, 2006.

[SHL07] L. Strömbäck, D. Hall, and P. Lambrix. A review of standards for data exchange within systems biology. *Proteomics*, 7(6):857–867, March 2007.

[SKS+10] B. Sommer, J. Künsemöller, N. Sand, A. Husemann, M. Rumming, and B. Kormeier. CELLmicrocosmos 4.1: An Interactive Approach to Integrating Spatially Localized Metabolic Networks into a Virtual 3D Cell Environment. *Proceedings of Bioinformatics*, 1:1–3, 2010.

[SKSB00] C. Schönbach, P. Kowalski-Saunders, and V. Brusic. Data warehousing in molecular biology. *Briefings in Bioinformatics*, 1(2):190–198, May 2000.

[SMS+02] P. T. Spellman, M. Miller, J. Stewart, C. Troup, U. Sarkans, S. Chervitz, D. Bernhart, G. Sherlock, C. Ball, M. Lepage, M. Swiatek, W. L. Marks, J. Goncalves, S. Markel, D. Iordan, M. Shojatalab, A. Pizarro, J. White, R. Hubley, E. Deutsch, M. Senger, B. J. Aronow, A. Robinson, D. Bassett, C. J. Stoeckert, and A. Brazma. Design and implementation of microarray gene expression markup language (MAGE-ML). *Genome Biology*, 3(9):1–9, August 2002.

[SOR+11] M. E. Smoot, K. Ono, J. Ruscheinski, P. L. Wang, and T. Ideker. Cytoscape 2.8: new features for data integration and network visualization. *Bioinformatics*, 27(3):431–432, February 2011.

[SPPB06] C. D. Schmid, R. Perier, V. Praz, and P. Bucher. EPD in its twentieth year: towards complete promoter coverage of selected model organisms. *Nucleic Acids Research*, 34(Database Issue):82–85, January 2006.

[SS67] M. L. Stein and P. R. Stein. Enumeration of Linear Graphs and Connected Linear Graphs up to p = 18 Points. *Report LA-3775, Los Alamos Scientific Laboratory of the University of California*, 1967.

[SS03] G. K. Smyth and T. Speed. Normalization of cDNA microarray data. *Methods*, 31(4):265–273, December 2003.

[SSR+03] E. Segal, M. Shapira, A. Regev, D. Pe'er, D. Botstein, D. Koller, and N. Friedman. Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature genetics*, 34(2):166–176, 2003.

[ST07] H. Sudhoff and M. Tos. Pathogenesis of sinus cholesteatoma. *European Archives of Otorhinolaryngology*, 264(10):1137–1143, 2007.

[STK+10] B. Sommer, E. S. Tiys, B. Kormeier, K. Hippe, S. J. Janowski, T. V. Ivanisenko, A. O. Bragin, P. Arrigo, P. S. Demenkov, A. V. Kochetov, V. A. Ivanisenko, N. A. Kolchanov, and R. Hofestädt. Visualization and analysis of a cardio vascular disease- and mupp1-related biological network combining text mining and data warehouse approaches. *Journal of Integrative Bioinformatics*, 7(1):148, January 2010.

[Sut08] S. Suthram. *Dissertation: Understanding cellular function through the analysis of protein interaction networks.* ProQuest, 2008.

[SWL+05] U. Stelzl, U. Worm, M. Lalowski, C. Haenig, F. H. Brembeck, H. Goehler, M. Stroedicke, M. Zenkner, A. Schoenherr, S. Koeppen, J. Timm, S. Mintzlaff, C. Abraham, N. Bock, S. Kietzmann, A. Goedde, E. Toksöz, A. Droege, S. Krobitsch, B. Korn, W. Birchmeier, H. Lehrach, and E. E. Wanker. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–968, September 2005.

[SZT12] B. Schwartz, P. Zaitsev, and V. Tkachenko. *High Performance MySQL: Optimization, Backups, and Replication.* O'Reilly Media, 3rd edition, April 2012.

[TB11] T. Triplet and G. Butler. Systems biology warehousing: Challenges and strategies toward effective data integration. *DBKDA 2011, The Third International Conference on Advances in Databases, Knowledge, and Data Applications*, 1:34–40, 2011.

[TCN03] J. Tyson, K. C. Chen, and B. Novak. Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. *Current Opinion in Cell Biology*, 15(2):221–231, March 2003.

[TD90] R. Thomas and R. T. R. D'Ari. *Biological Feedback*, pages 1–316. CRC Press, 1990.

[Tho73] R. Thomas. Boolean formalization of genetic control circuits. *Journal of Theoretical Biology*, 42(3):563–585, 1973.

[THT+99] M. Tomita, K. Hashimoto, K. Takahashi, T. S. Shimizu, Y. Matsuzaki, F. Miyoshi, K. Saito, S. Tanida, K. Yugi, and J. C. Venter. E-CELL: software environment for whole-cell simulation. *Bioinformatics*, 15(1):72–84, 1999.

[TIS04] F. L. Thompson, T. Iida, and J. Swings. Biodiversity of Vibrios. *Microbiological reviews*, 68(3):403–431, September 2004.

[TKHT04] K. Takahashi, K. Kaizu, B. Hu, and M. Tomita. A multi-algorithm, multi-timescale method for cell simulation. *Bioinformatics*, 20(4):538–546, February 2004.

[TKKH08] T. Töpel, B. Kormeier, A. Klassen, and R. Hofestädt. BioDWH: a data warehouse kit for life science data integration. *Journal of integrative Bioinformatics*, 5(2):1–9, 2008.

[TM04] M. Tory and T. Möller. Human factors in visualization research. *IEEE Transaction on Visualization and Computer Graphics*, 10(1):72–84, January 2004.

[TN88] G. K. Totland and A. Nylund. An ultrastructural study of morphological changes in Atlantic salmon, Salmo salar L., during the development of cold water vibriosis. *Journal of Fish Diseases - Wiley Online Library*, 11(1):1–13, 1988.

[USFL98] C. Ullmer, K. Schmuck, A. Figge, and H. Lübbert. Cloning and characterization of MUPP1, a novel PDZ domain protein. *FEBS Letter*, 424(1):63–68, March 1998.

[Val78] R. Valk. Self-modifying nets, a natural extension of Petri nets. *Automata, Languages and Programming*, 62:464–476, 1978.

[vdB11] H. van den Berg. *Mathematical Models of Biological Systems (Oxford Biology)*. Oxford University Press, USA, 1st edition, January 2011.

[VJ85] R. Valk and M. Jantzen. The Residue of Vector Sets with Applications to Decidability Problems in Petri Nets. *Acta Informatica*, 21(6):643–674, 1985.

[VSC+92] R. A. VanBogelen, P. Sankar, R. L. Clark, J. A. Bogan, and F. C. Neidhardt. The gene-protein database of Escherichia coli: edition 5. *Electrophoresis*, 13(12):1014–1054, December 1992.

[WAB+06] C. H. Wu, R. Apweiler, A. Bairoch, D. A. Natale, W. C. Barker, B. Boeckmann, S. Ferro, E. Gasteiger, H. Huang, R. Lopez, M. Magrane, M. J. Martin, R. Mazumder, C. O'Donovan, N. Redaschi, and B. Suzek. The Universal Protein Resource (UniProt): an expanding universe of protein information. *Nucleic Acids Research*, 34(Database Issue):187–191, January 2006.

[Wak05] G. Waksman. *Proteomics and protein-protein interactions*, volume 3 of *biology, chemistry, bioinformatics, and drug design*, pages 50–89. Springer Verlag, 2005.

[WBH05] S. L. Werner, D. Barken, and A. Hoffmann. Stimulus specificity of gene expression programs determined by temporal control of IKK activity. *Science*, 309(5742):1857–1861, September 2005.

[WCE+04] D. L. Wheeler, D. M. Church, R. Edgar, S. Federhen, W. Helmberg, T. L. Madden, J. U. Pontius, G. D. Schuler, L. M. Schriml, E. Sequeira, T. O. Suzek, T. A. Tatusova, and L. Wagner. Database resources of the National Center for Biotechnology Information: update. *Nucleic Acids Research*, 32(Database Issue):35–40, January 2004.

[WD10] I. E. Wertz and V. M. Dixit. Signaling to NF-κB: Regulation by Ubiquitination. *Cold Spring Harbor perspectives in Biology*, 2(3):1–19, March 2010.

[Wie01] W. Wiechert. 13C Metabolic Flux Analysis. *Metabolic Engineering*, 3(3):195–206, July 2001.

[WIN+05] J. Westbrook, N. Ito, H. Nakamura, K. Henrick, and H. M. Berman. PDBML: the representation of archival macromolecular structure data in XML. *Bioinformatics*, 21(7):988–992, April 2005.

[WMP99] J. U. Wurthner, A. K. Mukhopadhyay, and C. J. Peimann. A cellular automaton model of cellular signal transduction. *Computers in Biology and Medicine*, 30(1):1–21, December 1999.

[WXS+12] Y. Wang, J. Xiao, T. O. Suzek, J. Zhang, J. Wang, Z. Zhou, L. Han, K. Karapetyan, S. Dracheva, B. A. Shoemaker, E. Bolton, A. Gindulyte, and S. H. Bryant. PubChem's BioAssay Database. *Nucleic Acids Research*, 40(Database Issue):400–412, January 2012.

[WYA+05] D. S. Wishart, R. Yang, D. Arndt, P. Tang, and J. Cruz. Dynamic cellular automata: an alternative approach to cellular simulation. *In Silico Biology*, 5(2):139–161, 2005.

[YBL06] J. Yoon, A. Blumer, and K. Lee. An algorithm for modularity analysis of directed and weighted biological networks based on edge-betweenness centrality. *Bioinformatics*, 22(24):3106–3108, December 2006.

[YKW+06] M. Yoshikawa, H. Kojima, K. Wada, T. Tsukidate, N. Okada, H. Saito, and H. Moriyama. Identification of specific gene expression profiles in fibroblasts derived from middle ear cholesteatoma. *Archives of Otolaryngology - Head & Neck Surgery*, 132(7):734–742, June 2006.

[YNF+09] A. Yilmaz, M. Y. Nishiyama, B. G. Fuentes, G. M. Souza, D. Janies, J. Gray, and E. Grotewold. GRASSIUS: a platform for comparative regulatory genomics across the grasses. *Plant Physiology*, 149(1):171–180, January 2009.

[ZAD07] L. Zhang, C. A. Athale, and T. S. Deisboeck. Development of a three-dimensional multiscale agent-based tumor model: Simulating gene-protein interaction profiles, cell phenotypes and multicellular patterns in brain cancer. *Journal of Theoretical Biology*, 244(1):96–107, January 2007.

[Zha09] A. Zhang. *Protein Interaction Networks: Computational Analysis*, pages 1–62. Cambridge University Press, 1st edition, April 2009.

[ZOS02] I. Zevedei-Oancea and S. Schuster. Topological analysis of metabolic networks based on Petri net theory. *In Silico Biology*, 3(3):323–345, December 2002.

[ZZP+09] J. Zheng, D. Zhang, P. F. Przytycki, R. Zielinski, J. Capala, and T. M. Przytycka. SimBoolNet - A Cytoscape plugin for dynamic simulation of signaling networks. *Bioinformatics*, 26(1):141–142, December 2009.

About the author

Meaning of the PhD

Research means discovering new techniques that aid in building or using new mechanisms. And the best framework for me to learn to do this and to take it beyond the academic life was working on my PhD. I have always been fascinated by science and wished to follow this academic path. Biological network studies are an exciting area and due to my position at Bielefeld University, I had the possibility of developing my capacities and deepening knowledge in this field.

While pursuing my PhD degree, I learned how to plan a project, calculate how to execute it, and, more generally, how to develop approaches fit to tackle any new techniques or research questions. In addition to the volume of scientific knowledge, I learned to overcome problems both in research and in my personal life. Indeed, a PhD is four years of intense and determined work, but also very rewarding and quite enjoyable.

A further thing about doing a PhD for anyone is that it lends credibility to your voice. With this academic path you get a basic common ground, which you share with other scientists, a commonality in formulation of problems and a scientific system of searching for solutions. This is the key, which opens the door to the world of fellow knowledge seekers. Without it you may not really be taken seriously when you voice an opinion but don't seem to have any real knowledge to back it up. The PhD is the credential, which says you can be considered a serious player in the scientific world. In addition, a PhD is the de facto 'calling card' for many interesting and fascinating academic positions and make contact with the brightest minds. Through communication and collaboration with other specialists from around the world I already begun to reach for fantastic ideas beyond our grasp and in so doing, extended my intellectual capabilities.

My actual PhD topic may not necessarily be the only area I will work on. Moreover, it should serve as a solid basis for a scientific direction in life. New questions continuously arise and as time passes and new facts are discovered interests also evolve. Some of the best minds change topics and fields of studies. It keeps them fresh and stimulates thinking. Should I choose to take another path in future, I am pretty sure the PhD experience will enable me to conduct

research beyond academia. From my point of view there is a direct correlation between the skills learned while earning the PhD and the needs of the industry.

Whether working academically or in a commercial capacity, the valuable time as PhD student will support me in any special research areas. Now, I am able to explore, investigate, and contemplate. Furthermore, I am able to adapt to new ideas and willing to search for answers for which no one really knows the questions. My willingness to work to further my knowledge and contribute to the advancement of science through deeper understanding will always support my work. Therefore, I am very grateful to had the chance to work on a PhD thesis and I will always remember the time as one of the most influential periods of my life.

Acknowledgements

Because German is my mother tongue and most of my colleagues and loved ones live in Germany, the following acknowledgement is written in the German language.

In erster Linie möchte ich mich bei meinen Betreuern bedanken. Herr Prof. Dr. Ralf Hofestädt hat mir im Rahmen der Doktorarbeit alle Möglichkeiten gegeben, meine Visionen und Ideen zu verwirklichen. Dabei hat er mir während der ganzen Zeit sein vollstes Vertrauen geschenkt, mich unterstützt und motiviert neue Wege einzuschlagen und Herausforderungen anzunehmen. Mein großer Dank gilt auch Herrn Prof. Christian Kaltschmidt und Frau Prof. Dr. Barbara Kaltschmidt. Ich erinnere mich sehr gerne an viele wertvolle Gespräche zurück, in denen wir gemeinsam die unzähligen Möglichkeiten der Bioinformatik mit großer Freude diskutiert und neue Pläne geschmiedet haben. Gerade diese Zusammenarbeit hat mir einen schärferen Blick für die Biologie gegeben. Weiterer Dank gilt Herrn Prof. Dr. Jens Stoye, der mich immer unterstützend mit fachlichem und persönlichem Rat während dieser Arbeit begleitet hat.

Insbesondere möchte ich Herrn Dr. Benjamin Kormeier, Herrn Dr. Thoralf Töpel und Herrn Klaus Hippe danken, welche maßgeblich an dem Erfolg von VANESA beteiligt waren. Wenn es darum ging biologische Systeme zu analysieren, war Frau Daniela Borck eine sehr wertvolle Kollegin für mich. Dafür danke ich ihr. Generell danke ich der gesamten Arbeitsgruppe von Herrn Prof. Dr. Ralf Hofestädt, mit Herrn Alban Shoshi, Herrn Björn Sommer, Herrn David Braun, Herrn Venus Ogultharhan und Herrn Dr. Hang Mao Lee. Gemeinsam konnten wir viele Projekte angehen und erfolgreich abschließen. Mein herzlichster Dank geht auch an Frau Sabine Klusmann, Herrn Klaus Kulitza und Frau Barbara Davis. Ich kann sagen, dass mit allen Mitarbeitern mich nicht nur eine berufliche Beziehung, sondern auch viel mehr eine Freundschaft verbindet.

Eine wichtige Rolle in dieser Arbeit spielte auch die Graduiertenschule, bei der ich diese Arbeit absolvieren konnte. Hier möchte ich herzlichst Herrn Prof. Karl-Josef Dietz, Herrn Dr. Kolja Henckel und meinen Kommilitonen aus der Graduiertenschule Frau Kalina Mrozek, Herrn

Christoph Schmal und Frau Melanie Gerken danken. Dieses Umfeld war für mich von großem Wert, wo ich neue Ansätze vorstellen und diskutieren konnte. Dabei erfreute ich mich immer den Ansichten und dem Rat der Mitglieder dieser Graduiertenschule.

In Hinblick auf interdisziplinäre Kooperationen möchte ich vor allem Frau Sabrina Proß und Herrn Prof. Dr. Bernhard Bachmann der FH Bielefeld danken, welche die Petri Netz Bibliothek in Modelica realisiert haben und mit uns die gemeinsame Arbeit an der Simulationsumgebung in VANESA weiterführen. Herrn Prof. Dr. Tim Nattkemper danke ich für die gelungene und wertvolle Zusammenarbeit im Bereich der Visualisierung. Weiterhin will ich meine großen Dank an meine Kollegen aus Sydney, Herrn Prof. Dr. Eades, Frau Prof. Dr. Seokhee Hong, Herrn Prof. Dr. Masahiro Takatsuka, Herrn Dr. Olivier Swienty, Quan Nguyen und alle weiteren Mitgliedern der Arbeitsgruppe richten. Sie haben mich herzlichst in Australien aufgenommen und in die Welt der fortgeschrittenen Netzwerkvisualisierung eingeführt. Diese Zeit wird mir immer positiv in Erinnerung bleiben. Daher gilt mein tiefster Dank auch an Herrn Prof. Dr. Falk Schreiber, welcher mir geholfen hat, diesen Forschungsaufenthalt zu realisieren. Des Weiteren spreche ich Frau Dr. Christin Klenke für die Kooperation und die gemeinsame Arbeit in der Systemmedizin meinen Dank aus. Mein herzlichster Dank gilt auch Herrn Prof. Dr. Edgar Wingender. Schon von Anfang an hat er das Potential in VANESA gesehen und mich in meiner Arbeit bekräftigt.

An dieser Stelle bedanke ich mich auch für die Hilfe tatkräftiger Studenten. Ohne sie wäre VANESA, so wie es sich heute präsentiert, nicht realisierbar gewesen. Generell haben alle Studenten sehr gute Arbeit geleistet und das Projekt mit tollen neuen Funktionen bereichert. Herausheben möchte ich doch insbesondere Herrn Martin Lewinski, Herrn Evgeny Anisiforov, Herrn Christoph Brinkrolf und Herrn Arne Sahm. Sie waren immer sehr engagiert und haben eine hervorragende Arbeit geleistet.

Abschließend richte ich meinen Dank an meine Liebsten und Freunde, welche an meiner Seite standen und mich auch hin und wieder erinnert haben, dass das Leben nicht nur aus der Doktorarbeit besteht. Abgesehen davon möchte ich ganz herzlich meiner Mutter danken. Das Schreiben einer Doktorarbeit erfordert viel Kraft, Ausdauer, Disziplin und Hingabe. Nicht alle Tugenden und Werte kann man während der Doktorarbeit erlangen. Viel mehr werden sie einem mit auf den Weg gegeben. Hier war meine Mutter nie müde mich zu fördern. Ich wäre jetzt nicht dort wo ich bin, wenn sie mir nicht so viel ihrer Kraft und Ausdauer geschenkt hätte. Dafür danke ich ihr vom ganzen Herzen. Einen großen Anteil an meinem Werdegang hat auch meine Großmutter. Von Anfang an hat sie meine akademische Laufbahn mit größter Freude begleitet. Begeistert hat sie von den Vorzügen der akademischen Laufbahn gesprochen und mir somit den Geschmack auf weitere Schritte versüßt.

Education

January 1, 2010 - January 1, 2013
Phd thesis in the field of Bioinformatics
Bielefeld University, Germany (full research scholarship)

September 15th, 2010 - 15th January, 2011
Scientific Research Exchange on Network Analysis and Visualization
University of Sydney, Australia

March 1, 2005 - July 15, 2009
Bachelor of Arts in Media Design and Media Informatics
Bielefeld University, Germany and University of Applied Sciences, Bielefeld, Germany
Class best - officially honored, final mark: 1.2

October 1, 2005 - March 26, 2009
Master of Science in Bioinformatics and Genome Research
Bielefeld University, Germany
Class best - officially honored, final mark: 1.3

October 1, 2006 - July 15, 2007
One year Postgraduate Research in Genetics and Health Informatics
Trinity College Dublin, Ireland (full scholarship from the DAAD)

October 1, 2002 - July 15, 2005
Bachelor of Science in Bioinformatics and Genome Research
Bielefeld University, Germany
Graduation with honors - final mark: 1.4

June 28, 2002 - **High School Diploma**
Gymnasium am Waldhof Bielefeld, Germany
Final mark: 3.0