CAUSALITY ANALYSIS IN BIOLOGICAL NETWORKS

A DISSERTATION SUBMITTED TO
THE DEPARTMENT OF COMPUTER ENGINEERING
AND THE INSTITUTE OF ENGINEERING AND SCIENCE
OF BİLKENT UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

By
Özgün Babur
January, 2010

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Assoc. Prof. Dr. Uğur Doğrusöz (Advisor)

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Asst. Prof. Dr. Ali Aydın Selçuk

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Asst. Prof. Dr. Özlen Konu

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Assoc. Prof. Dr. Uğur Güdükbay

I certify that I have read this thesis and that in my opinion it is fully adequate, in scope and in quality, as a dissertation for the degree of doctor of philosophy.

Asst. Prof. Dr. Tolga Can

Approved for the Institute of Engineering and Science:

Prof. Dr. Mehmet B. Baray
Director of the Institute

ABSTRACT

CAUSALITY ANALYSIS IN BIOLOGICAL NETWORKS

Özgün Babur
Ph.D. in Computer Engineering
Supervisor: Assoc. Prof. Dr. Uğur Doğrusöz
January, 2010

Systems biology is a rapidly emerging field, shaped in the last two decades or so, which promises understanding and curing several complex diseases such as cancer. In order to get an insight about the *system* – specifically the molecular network in the cell – we need to work on following four fundamental aspects: experimental and computational methods to gather knowledge about the system, mathematical models for representing the knowledge, analysis methods for answering questions on the model, and software tools for working on these. In this thesis, we propose new approaches related to all these aspects.

In this thesis, we define new terms and concepts that helps us to analyze cellular processes, such as positive and negative paths, upstream and downstream relations, and distance in process graphs. We propose algorithms that will search for functional relations between molecules and will answer several biologically interesting questions related to the network, such as neighborhoods, paths of interest, and common targets or regulators of molecules.

In addition, we introduce ChiBE, a pathway editor for visualizing and analyzing BioPAX networks. The tool converts BioPAX graphs to drawable process diagrams and provides the mentioned novel analysis algorithms. Users can query pathways in Pathway Commons database and create sub-networks that focus on specific relations of interest.

We also describe a microarray data analysis component, PATIKA*mad*, built into ChiBE and PATIKA*web*, which integrates expression experiment data with networks. PATIKA*mad* helps those tools to represent experiment values on network elements and to search for causal relations in the network that potentially explain dependent expressions. Causative path search depends on the presence of transcriptional relations in the model, which however is underrepresented in most of the databases. This is mainly due to insufficient knowledge in the literature.

We finally propose a method for identifying and classifying modulators of transcription factors, to help complete the missing transcriptional relations in the pathway databases. The method works with large amount of expression data, and looks for evidence of modulation for triplets of genes, i.e. modulator - factor - target. Modulator candidates are chosen among the interacting proteins of transcription factors. We expect to observe that expression of the target gene depends on the interaction between factor and modulator. According to the observed dependency type, we further classify the modulation. When tested, our method finds modulators of Androgen Receptor; our top-scoring result modulators are supported by other evidence in the literature. We also observe that the modulation event and modulation type highly depend on the specific target gene. This finding contradicts with expectations of molecular biology community who often assume a modulator has one type of effect regardless of the target gene.

Keywords: Computational biology, bioinformatics, systems biology, pathway informatics, causality analysis.

ÖZET

BİYOLOJİK AĞLARDA NEDENSELLİK ANALİZİ

Özgün Babur
Bilgisayar Mühendisliği, Doktora
Tez Yöneticisi: Doçent Dr. Uğur Doğrusöz
Ocak, 2010

Sistem biyolojisi son birkaç on yılda şekillenmiş, ve kanser gibi karmaşık hastalıklara çözüm vadeden bir alandır. Sistem hakkında (daha spesifik olarak hücresel ağlar hakkında) bir kavrayış geliştirebilmek için şu dört temel alanda calışmalar yapmak gerekir: sistem hakkında bilgi toplamak için deneysel ve hesapsal metotlar, bilgiyi göstermek için matematiksel modeller, model hakkındaki sorulara cevap bulan analiz yöntemleri, ve bütün bunlar üzerinde çalışmamıza yardımcı olacak yazılım araçları. Bu tezde, bahsedilen bütün alanlarla ilgili yeni yaklaşımlar sunuyoruz.

Bu tezde, pozitif ve negatif etkili yolaklar, akışyukarı ve akışaşağı ilişkiler, ve süreç çizgelerinde uzaklık gibi terimler ve kavramlar tanımlıyoruz. Komşuluk, ilgilenilen ağlar, ve ortak hedef ve ortak düzenleyiciler gibi biyolojik açıdan ilginç sorulara cevap üretecek, çizge üzerinde fonksiyonel ilişkiler arayan algoritmalar öneriyoruz.

Buna ek olarak, BioPAX çizgelerini görselleyen ve analiz eden ChiBE isimli yazılımı sunuyoruz. Bu yazılım, BioPAX çizgelerini çizilebilir süreç çizgelerine çeviriyor ve yukarıda bahsi geçen algoritmaları sağlıyor. ChiBE kullanıcıları Pathway Commons veritabanını sorgulayabiliyor ve kendi ilgilendikleri süreçlere odaklı çizge parçaları üretebiliyorlar.

Ayrıca PATIKA*mad* isimli bir mikrodizi veri analiz yazılımı geliştirdik, ve mikrodizilerle biyolojik ağları entegre edecek şekilde ChiBE ve PATIKA*web* yazılım araçlarında kullandık. PATIKA*mad* sayesinde bu araçlar mikrodizi değerlerini molekül düğümleri üzerinde gösterebiliyor ve ağ üstündeki ifadeler arasındaki bağlantıları açıklama potansiyeline sahip sonuçsal yolakları ortaya çıkarabiliyor. Sonuçsal yolakların analizi, modellenen biyolojik ağ üzerinde yazılım ilişkilerinin de modellenmiş olmasına dayanır. Fakat yazılım ilişkileri biyolojik ağ veritabanlarında olması gerekenin çok altında bulunmaktadır. Bunun temel nedeni literatürde bu konuda görece daha az bilgi bulunmasıdır.

Son olarak, veritabanlarındaki eksik yazılım bilgisini tamamlama potansiyeline sahip, yazılım faktörlerinin modülatörlerini tahmin eden ve sınıflayan bir yöntem öneriyoruz. Bu yöntem çok sayıda mikrodizi verisini kullanarak modulatör - faktör - hedef gen üçlüleri arasında modülasyon ilişkisi arıyor. Modulatör adaylarını yazılım faktörünün etkileştiği bilinen proteinler arasından seçiyoruz ve hedef genin ifadesinin faktör ve modülatör arasındaki etkileşimden etkilenmesini bekliyoruz. Gözlenen etki şekline göre modülatörleri ayrıca sınıflandırıyoruz. Metodumuzu Androjen Reseptörü üzerinde denediğimiz zaman görüyoruz ki yüksek puanlı modülatörler literatürdeki başka kanıtlarla da destekleniyor. Bu araştırmada gözlediğimiz diğer bir olgu ise modülatörlerin etkisinin ve sınıfının çoğunlukla hedef gene göre farklılık göstermesidir. Halbuki literatürdeki çalışmalar modülatörleri genellikle hedef genden bağımsız tek tip etkiye göre (aktifleştirici ve engelleyici) sınıflandırmaya çalışıyor.

Anahtar sözcükler: Biyoenformatik, Nedensellik analizi.

Acknowledgement

I would like to thank to my advisor, Uğur Doğrusöz for his great support during my PhD studies. He never gave up showing me the right way. I would happily continue working with him if there was no such thing called graduation.

This thesis would be in a totally different shape without Emek Demir. We worked together at all parts of this work. He contributed with his mind, heart, and friendship.

While working on GEM, we needed a help in statistics; that's how I met with Mithat Gönen. Then we started on very hot discussions, which was a true joy to me.

I would like to thank Chris Sander for letting me to work in MSKCC for nine months during my PhD period. It was very motivating for me to work in his group, and I really enjoyed discussing research with him.

Merve Çakır and Shatlyk Ashyralyev worked on local queries in ChiBE during their summer project. Recep Çolak worked on PATIKA*mad* during his senior project. We shared an office with Alptuğ Dilek and Esat Belviranlı in the last three years, and we have been in many encouraging discussions. Cihan Küçükkeçeci developed Chisio during his MS study; it would be very hard to build ChiBE without it.

I would like to thank Çiğdem, Zeynep, Kamer, Burcu, Önder, Burak, Cihan, Duygu, Sengör, Engin, and Özlem for their friendship, and Funda for her love, and lastly, my family for being there when I needed.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Biological Pathways

Events happening in the cell has always been interesting to the scientific community since most of the diseases are related to malfunctioning of a component or interruption of normal function by external factors, like chemicals or viruses. Modeling of these events creates *pathways*, which can be diverse in structure and the encoded information. Structure of pathways is generally affected from the viewpoint of researchers, and the experimental methods that supply information about the event.

There are some customary types of pathways such as metabolic, protein interaction, signaling, gene regulatory, and genetic interaction networks. Today there are various efforts to integrate these different representations and create a standard representation.

Metabolic networks focus on enzymatic reactions, specifying substrates and products. The identity of the enzyme itself can be unknown and reactants can be generic, i.e. representing a set of molecules that have a common chemical property (Figure 1.1). Reactions are generally discovered with biochemical assays, performed in a test tube (*in vitro*). These assays can also identify several rate

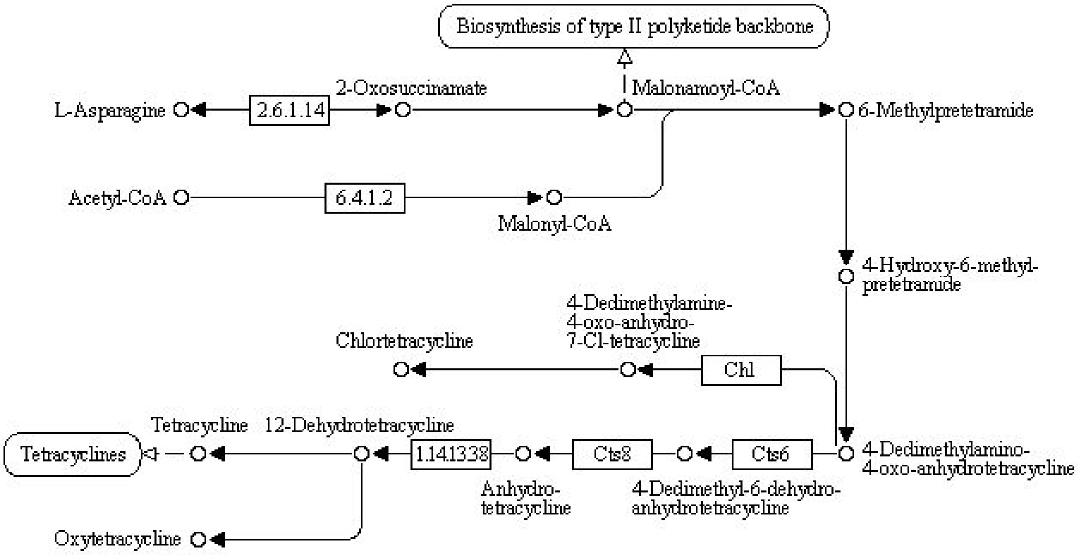


Figure 1.1: Metabolic network example from KEGG database [46].

constants of the reaction. When rate constants and molecule identities are known, it is possible to simulate metabolic pathways on computer (*in silico*) [65, 12].

Signaling networks are the bridges between external signals and metabolic events. Complexity of a signaling network increases with the complexity of the associated organism. This kind of networks capture the signal flow between the signaling molecules (Figure 1.2). Phosphorylation and de-phosphorylation of proteins constitute a great deal of signaling events, performed by kinases and phosphotases, detected with kinase assays.

Protein interaction networks are the simplest and the most popular type. An edge between two protein nodes indicates an interaction (Figure 1.3). High-throughput experiments, like yeast two-hybrid assays, provide massive amounts of protein interaction data.

Gene regulatory networks capture the relation between genes in terms of the regulation of expression. Edges in these directed networks indicate the activity of the source gene affecting the expression of the target gene. These networks are generally inferred using gene expression datasets [58].

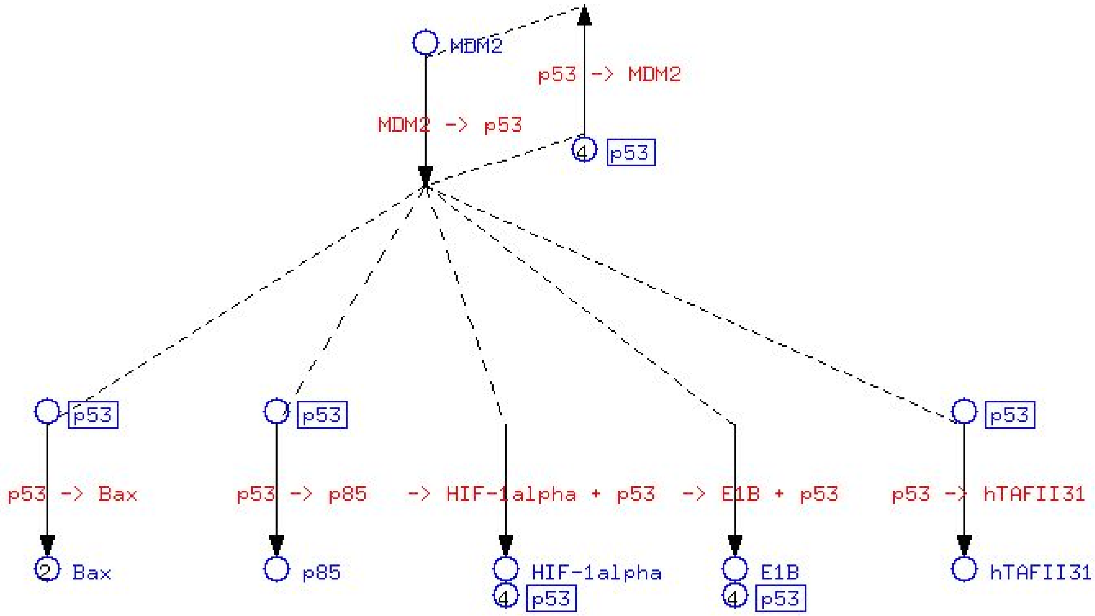


Figure 1.2: Signaling network example from CSNDB database [74].

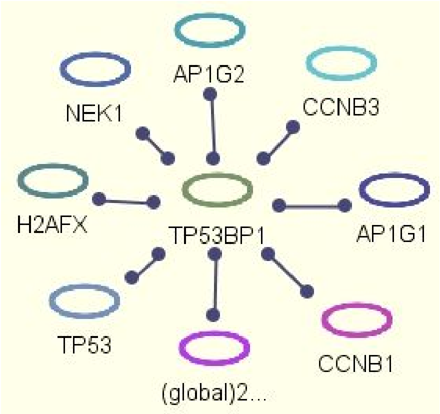


Figure 1.3: Protein-protein interaction network from PATIKA database [28].

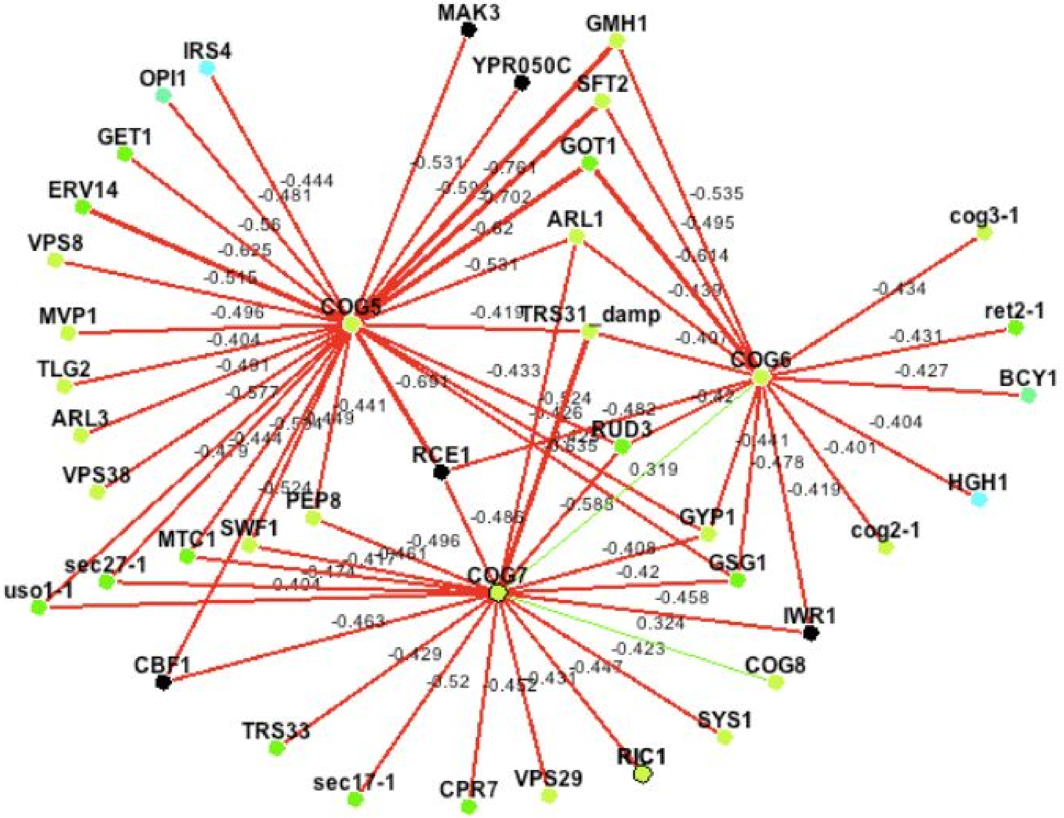


Figure 1.4: Genetic interaction network from DRYGIN database [52].

Genetic interaction networks relate genes whose interaction (probably indirect) is associated with a certain phenotype (Figure 1.4). For instance, if a yeast strain dies when genes A and B are knocked out together, but is not affected when only one of them is knocked out, then we say there is a genetic inteaction between genes A and B. A recent high-throughput technique called Synthetic Genetic Array (SGA) analysis is developed for quantitatively identifying genetic interactions based on synthetic lethality [75].

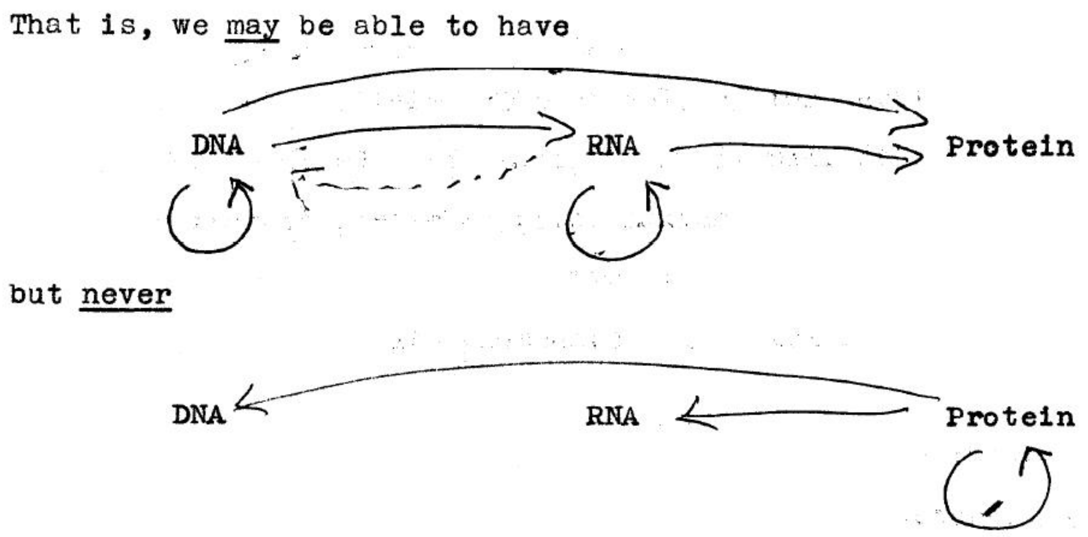


Figure 1.5: Idea of Central Dogma of Molecular Biology, drawn by Francis Crick [19].

1.2 Gene Expression

In 1958, Francis Crick reported in a symposium: “Once information has got into a protein it can’t get out again” [19]. On his report, Crick draws a model of information flow of the genetic code in the cell (Figure 1.5). This model is recognized as the Central Dogma of molecular biology. Today, we name each part of the flow as:

- DNA → DNA: DNA replication by DNA polymerases
- DNA → RNA: Transcription by RNA polymerases
- RNA → Protein: Translation by ribosomes
- RNA → RNA: RNA replication by RNA dependent RNA polymerases of some viral genomes like poliovirus
- RNA → DNA: Reverse transcription by reverse transcriptase enzyme of some viral genomes like HIV

Second and third relations (DNA → RNA → Protein) are collectively called as "gene expression". The cell changes its expressed set of genes according to changing conditions and external signals. This results in a different set of functional proteins in the cell, which also is key to differentiation in multicellular organisms.

1.2.1 Transcription

Gene expression starts with transcription, which means production of mRNA using a DNA template. This mRNA is later used as a template for protein synthesis. Transcription is performed by the enzyme RNA Polymerase, a multicomponent protein, found in all living cells. RNA Polymerase binds to promoter region of genes in the presence of specific transcription factors (TFs), and initiate transcription at the start site (Figure 1.6).

TFs are DNA binding proteins that recognize specific binding regions in the promoter or enhancer region of genes. In eucaryotes, TFs can be generic like TATA Binding Factor (TBF), or can be specific, like STATs, targeting a restricted set of genes. Each gene posesses binding sites of a specific set of TFs in their promoter, thus needs presence of a specific set of factors for their expression.

At any given time, depending on the context and cellular stimuli, a transcription factor will affect only of a subset of its all possible target genes. This specificity is often provided by *modulators*, proteins that control transcription factor activity through several different mechanisms, including: post translational modifications, protein degradation, and non-covalent interactions. Modulators help a cell to combine different external signals and make complex downstream decisions. Elucidating their function is necessary for understanding and controlling cell's response to external stimuli at gene expression level.

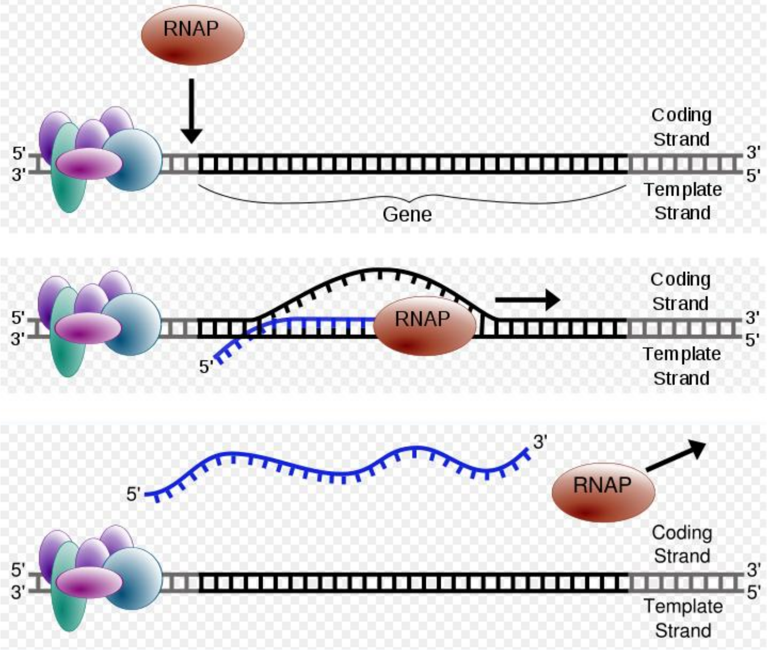


Figure 1.6: Sketches showing transcription of a gene. **Top:** RNA polymerase recognizes the transcription initiation complex, formed by several transcription factors and their binding proteins. **Middle:** RNA polymerase starts transcription, reading the coding strand of the DNA and synthesizing mRNA. **Bottom:** mRNA is synthesized and RNA polymerase dissociated from DNA [80]

1.2.2 Expression Microarrays

The most popular way of detecting gene expression is to measure the mRNA level of the gene. Since mRNAs in a cell are constantly produced and degraded, mRNA concentration cannot last without an active transcription; thus it is an indication of transcription. Translation phase is generally assumed by the presence of mRNA; however, there are studies showing that another kind of RNA, microRNA, can interfere with and inhibit the translation process [54, 67].

Microarray technology is an advancement over the Southern Blotting technique for detecting DNA, and first described by [71]. Southern Blotting is very similar to catching a fish, where the complementary DNA (cDNA) of the queried DNA is bait, and fragmented DNA of the cell separated with gel electrophoresis are the fish [73]. Bait is spread over DNA fragments, and catches (hybridizes with) the DNA whose sequence is complementary. Excess bait is washed out, remaining bait indicates the location of the DNA in query.

Microarrays detect expression of thousands of genes, sometimes the complete genome of an organism, in a single experiment. The method uses DNA fragments attached to a surface (array). Each spot on the array contains a specific DNA sequence. The mRNA, extracted from the cell, are used for production of labelled with fluorescence cDNAs, which are later hybridized with the DNA attached on the microarray. Attached cDNAs are detected with laser scanning, measuring signals coming from spots, each one indicating the expression level of a gene (Figure 1.7).

There are two main types of microarrays, cDNA arrays and oligonucleotide-arrays. cDNA arrays are historically first emerged type and use the whole expressed sequence as the probe. Each spot cDNA arrays is specific to a gene. Oligonucleotide-arrays, on the other hand, use short oligomers – 25 to 60 bases – in each spot. These oligomers are matched with fragments of genes, one gene represented by several spots on the array. These arrays are mainly produced by corporations Agilent and Affymetrix, they are relatively cheap, and most popular.

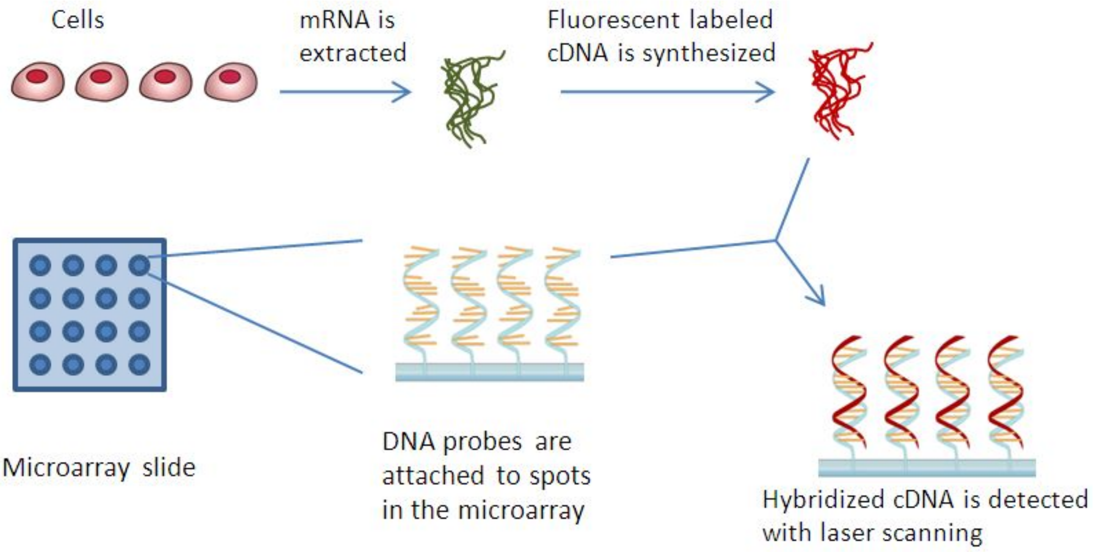

Cells
mRNA is extracted
Fluorescent labeled cDNA is synthesized
Microarray slide
DNA probes are attached to spots in the microarray
Hybridized cDNA is detected with laser scanning

Figure 1.7: Steps of a microarray experiment

Measured expression values are rarely used as is, since there is no standard for normal expression of a gene. The most common method used for interpreting gene expression is to perform the microarray experiment for two different conditions, and compare expressions of the same gene in these conditions. If there is a significant difference in the expressions, then the gene is said to be *differentially expressed*. However, comparing just two arrays for assessing differentially expressed genes would not be wise because expression values have high variation due to the expreimental technique. It is said that, differential expressions can be false positive up to 75% [2]. A way to overcome this problem is to perform many microarray expreiments on the same condition and decide the expression by evaluating collectively.

1.3 Contribution

In this thesis, we formulate several analysis methods for process description graphs. Chapter 3 discusses the characteristics of process description graphs and describes some graph traversal algorithms adapted to these graphs. We use these algorithms for answering several biologically relevant questions like paths between molecules or common targets. We discuss the semantics of integrating microarray data to pathways, and define the concept of *causative paths* that can be used for elucidating dependencies between gene expressions through paths on the network. In Chapter 4, we describe a probabilistic method, GEM [4], for inferring and characterizing modulators of transcription factors based on expression profiles. We treat interacting molecules of transcription factors as potential modulators, and use known targets of factors for measuring the dependency of factor-target correlation on the modulator expression. In Chapter 5, we present two tools, PATIKA*mad* [3] and ChiBE [5], that facilitate pathway visualization and analysis. PATIKA*mad* is the microarray data integration component of PATIKA*web* [28]. ChiBE is a BioPAX pathway editor that represents BioPAX graphs using the process description notation. It also supports local and distant querying of BioPAX models in process description graphs; and adapts PATIKA*mad* for expression data integration and analysis.

Chapter 2

Related Work

2.1 Biological Pathway Ontologies

Biological pathways are represented in many different forms, mainly determined by the pathway provider itself. Databases that focus on a specific type of network use a model which is simplest to fit their data. For instance Database of Interacting Proteins (DIP) [69] represent their data in PSI-MI [41], which covers molecule and interaction details (like reference or evidence), but cannot model reactions, regulations or abstractions.

There are several efforts for integrating pathway data from different sources. Such an effort have to use a model that can accommodate different types of information. PATIKA project [24] defines such an ontology, and integrates pathway data from BIND [6], HPRD [50], and Reactome [59] databases. BioPAX [25] is another ontology, being developed with community effort, for modeling many kinds of networks, and offered as a pathway exchange language. SBML [44] is a similar project with a focus on simulation. SBGN [53] is another community effort for determining standards of pathway graphical notation.

An extension of graph-based representation, namely hierarchically structured or compound graphs, in which a member of a biological network may recursively contain a sub-network of other pathway elements, can be used for representing sub-pathways, molecular complexes and subcellular location. Compound graphs also help managing complexity by interactively decomposing a pathway into distinct components or modules [35, 23]. The recently introduced visualization standard SBGN [53] also uses compound graphs extensively.

2.1.1 PATIKA Ontology

In 2000, PATIKA project was launched to create a pathway visualization and analysis platform with a comprehensive database. Towards this goal, the PATIKA ontology [23], that models pathways in two levels of detail, was defined. Less detailed first level (bioentity level) includes bioentities and their binary interactions, while the second level (mechanistic level) models mechanistic details of events.

A molecule node in the mechanistic level is called *state*, and each state is associated with a *bioentity*, which keeps references to the sequence databases such as Entrez Gene [56] or UniProt [1], and to small chemical databases such as ChEBI [22] or PubChem [15]. States represent molecular state of an entity in a cellular location with some modification, like phosphorylation.

Events in the mechanistic level are modeled with *transitions*, which have substrate, product, activator, and inhibitor edges that link transitions to states (Figure 2.1). Transitions have several types such as *chemical modification*, *complex formation*, and *transport*.

Homologous states and transitions are modeled with homology abstractions (Figure 2.2), and *regular abstraction* structure is used for defining any kind of groupings, like pathways.

PATIKA ontology can also handle incomplete information using the model elements *incomplete state* and *incomplete transition* (Figure 2.3). When any of the *incomplete* element is associated with an edge, this means that this edge is actually associated with one of the members of the abstraction, but we do not know which one.

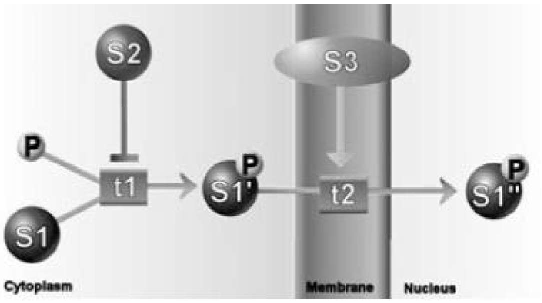


Figure 2.1: Basics of PATIKA ontology [23].

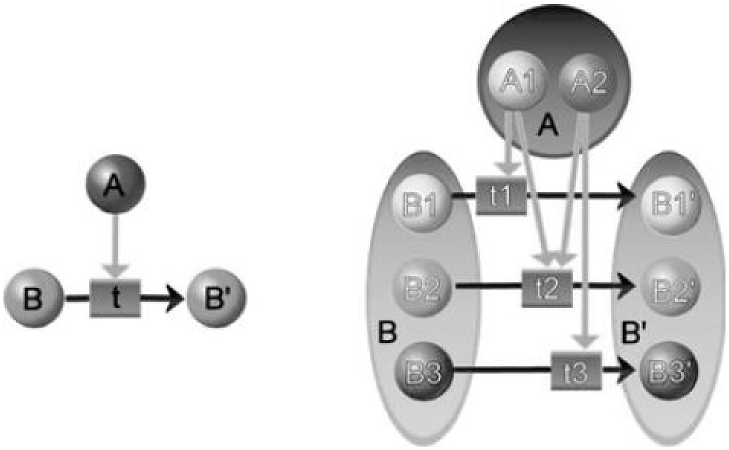


Figure 2.2: Demonstration of the use of homology abstraction in PATIKA ontology [23].

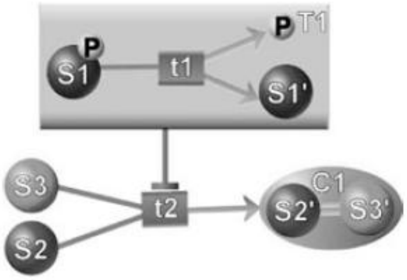


Figure 2.3: When the associated state is not certain, but a candidate set exists, *incomplete states* are used in PATIKA ontology for representing this information. T1 in the drawing is an incomplete state, indicating either S1 or S1′ inhibits the reaction [23].

2.1.2 BioPAX

BioPAX (Biological Pathway Exchange Language) project was initiated for creating a common format that will facilitate the data transfer between biological pathway databases. Each version of BioPAX language is called a *level*. The first BioPAX level (level 1) is released in 2004, modeling biochemical reactions (Figure 2.4). Their model associates `PhysicalEntity` objects (molecules) to *Conversions* through utility objects called `PhysicalEntityParticipant`, which also keeps stoichiometry, cellular location, and chemical modifications of the associated `PhysicalEntity`. Level 2 was released with an extension to include physical interactions between molecules.

BioPAX level 3 improves over the previous levels by explicitly putting molecular states in the model. This level completely abandons `PhysicalEntityParticipant`, and changes semantics of PhysicalEntity to represent molecular states instead of entities, and semantics of previous `PhysicalEntity` class migrates to `EntityReference` (Figure 2.5). Level 3 also supports gene regulatory interactions and genetic interactions. Figure 2.6 shows the progress of BioPAX language over time.

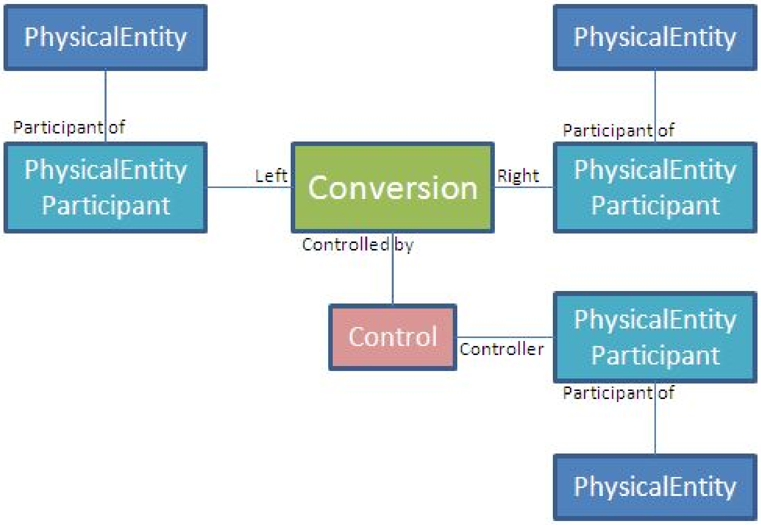


Figure 2.4: Part of the data model of BioPAX Level 1.

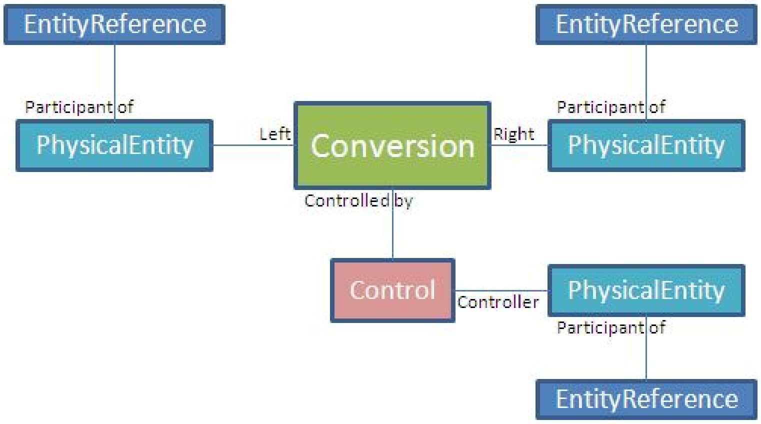
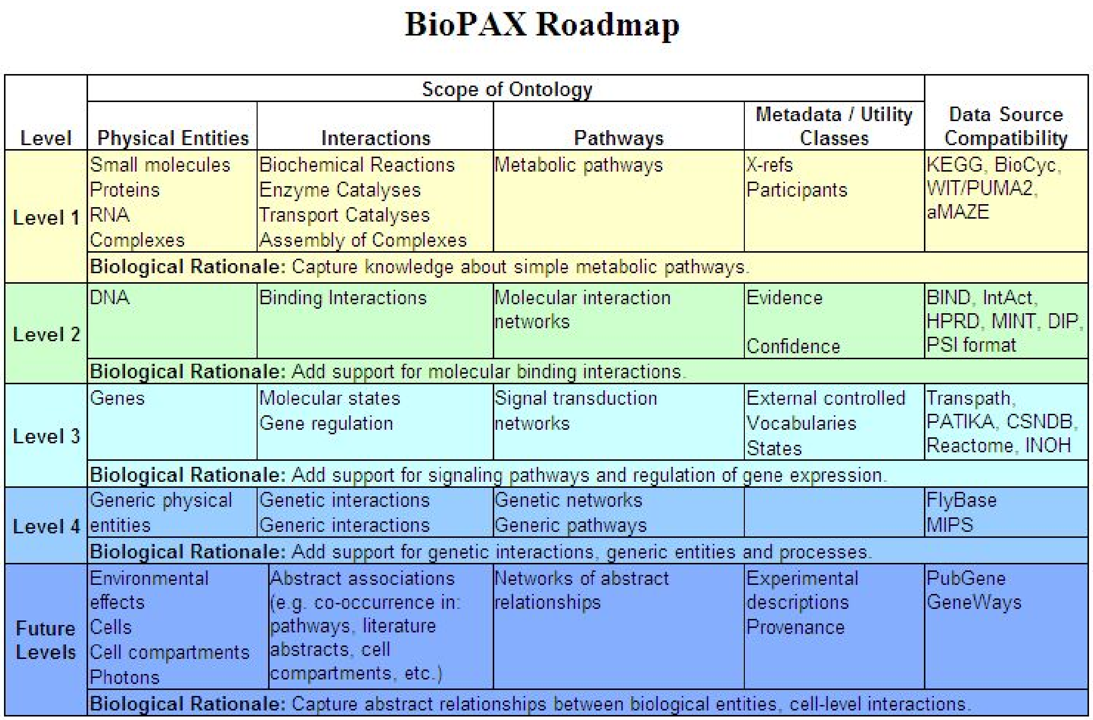


Figure 2.5: Part of the data model of BioPAX Level 3.



BioPAX Roadmap

Level	Scope of Ontology				Data Source Compatibility
	Physical Entities	Interactions	Pathways	Metadata / Utility Classes	
Level 1	Small molecules Proteins RNA Complexes	Biochemical Reactions Enzyme Catalyses Transport Catalyses Assembly of Complexes	Metabolic pathways	X-refs Participants	KEGG, BioCyc, WIT/PUMA2, aMAZE
	Biological Rationale: Capture knowledge about simple metabolic pathways.				
Level 2	DNA	Binding Interactions	Molecular interaction networks	Evidence Confidence	BIND, IntAct, HPRD, MINT, DIP, PSI format
	Biological Rationale: Add support for molecular binding interactions.				
Level 3	Genes	Molecular states Gene regulation	Signal transduction networks	External controlled Vocabularies States	Transpath, PATIKA, CSNDB, Reactome, INOH
	Biological Rationale: Add support for signaling pathways and regulation of gene expression.				
Level 4	Generic physical entities	Genetic interactions Generic interactions	Genetic networks Generic pathways		FlyBase MIPS
	Biological Rationale: Add support for genetic interactions, generic entities and processes.				
Future Levels	Environmental effects Cells Cell compartments Photons	Abstract associations (e.g. co-occurrence in: pathways, literature abstracts, cell compartments, etc.)	Networks of abstract relationships	Experimental descriptions Provenance	PubGene GeneWays
	Biological Rationale: Capture abstract relationships between biological entities, cell-level interactions.				

Figure 2.6: Progress of BioPAX language as new levels are released [10].

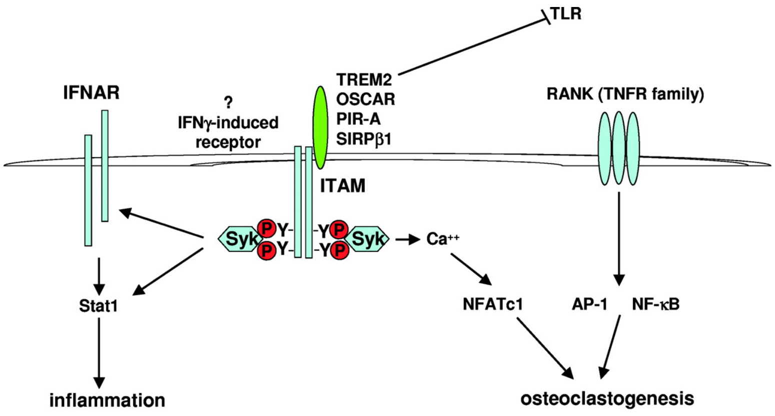


Figure 2.7: A manually drawn pathway appeared in [42].

2.1.3 SBML

SBML [44] was developed as a modeling standard for pathways at the level of biochemical reactions. Focus of SBML is simulation of the modeled events; thus, their model is very low level. Each pool of chemically identical molecules in a specific cellular compartment is represented with a *Species*, which are also inputs and outputs of the *Reaction*s. SBML has structures for reaction rules and parameters, while it completely ignores any generalizations or incomplete information.

2.1.4 SBGN

Biological pathways are generally visualized using graphical models. Most graphically pleasing pictures are still manually drawn ones that generally appear in published materials. However, these nice pictures generally lack a consistent notation, and it is impossible to understand them without an explanatory text. For instance, the graph in Figure 2.7 describes regulation of Stat signaling by ITAM-dependent pathways [42]. Two arrows that go to Stat1 have completely different meanings. One edge is for the activation of Stat1, while the other is helping this event, which we only understand after reading the related paper.

SBGN [53] is a standardization effort for pathway drawings. It aims the drawings to be self-explanatory and to cover most of the biological phenomena. SBGN defines three kinds of drawings: process diagrams, entity relationship graphs, and activity flow graphs. For each graph type, several *glyphs* are defined as units of the notation. Process diagrams (Figure 2.8) explicitly draw processes with participant molecular states. This graph type is most similar to the notation used by popular pathway databases like Reactome, KEGG, and PATIKA. Entity relationship diagrams (Figure 2.9) draw each entity once, and show processes by edges between entities, their features, and other edges.

2.2 Pathway Editors

A limited number of software tools for biological pathway visualization and analysis was developed such as Cytoscape [51], CellDesigner [36], PATIKA*web* [28], Pathway Tools [49], and VisANT [43]. These tools differ in their focus and capabilities. Several of these tools are compared in Table 2.1 with their support to BioPAX and SBGN standards, layout capability, compound graph support, availability and type of software. More details about Cytoscape, CellDesigner, and PATIKA*web* are given below.

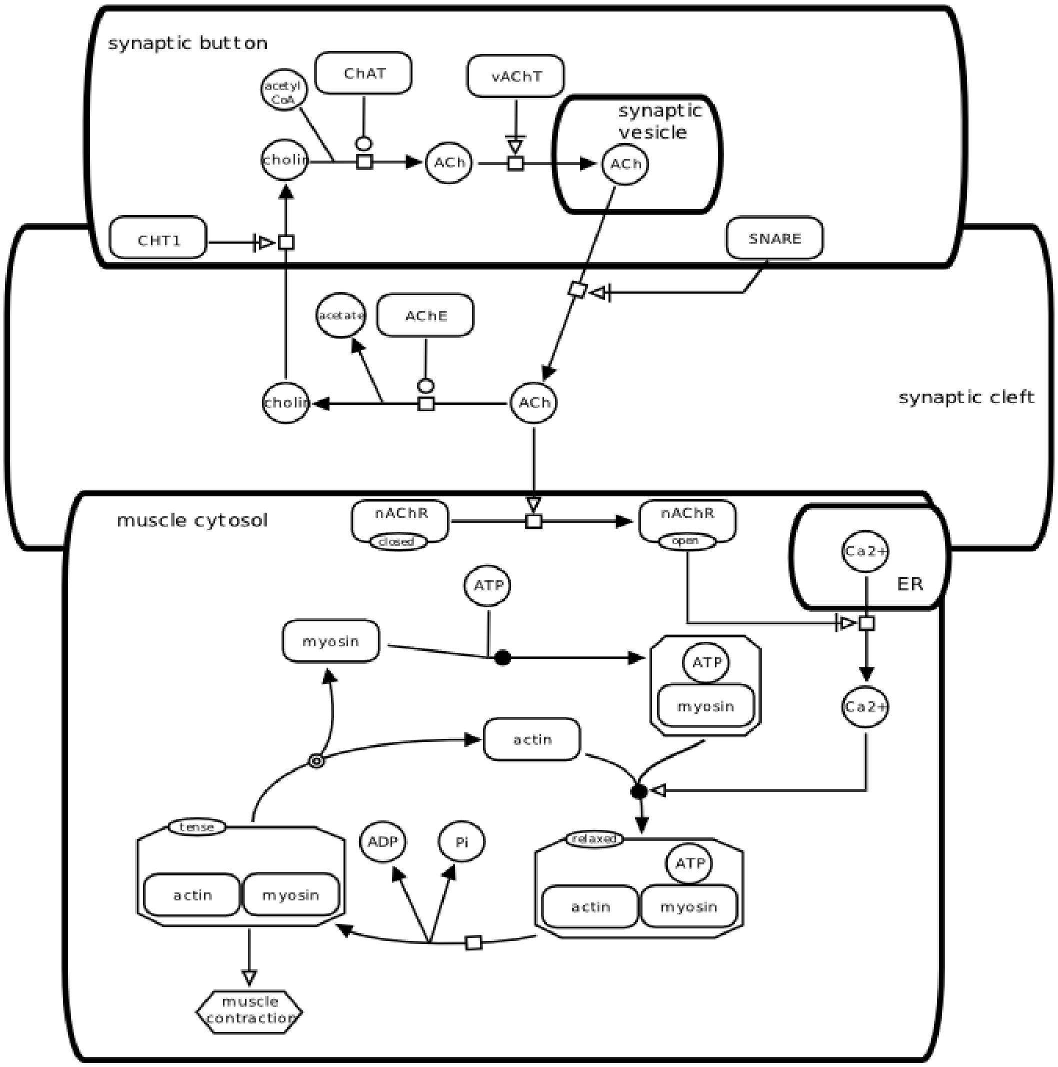


Figure 2.8: Stimulation events in the neuro-muscular junction, drawn as a SBGN Process Diagram [70].

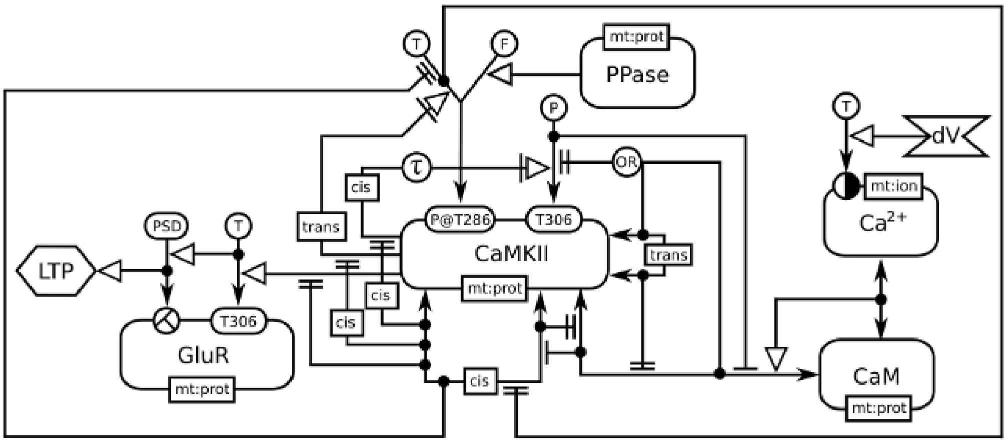


Figure 2.9: SBGN entity relationship diagram describing the effect of a depolarization (dV) on the intracellular calcium, that binds to calmodulin, that itself binds to the calcium/calmoduline kinase II (CaMKII) [70].

	BioPAX Support	Layout	Compound Support	SBGN	Availability	Tool Type
Cytoscape [51]	Yes	Automated	No	No	Open Source	Application
BiNoM [82]	Yes	Automated	No	Yes	Open Source	Application[1]
Reactome [59]	No	Automated	No	Planned	Open Source	Web
KEGG Tools [46]	No	Manual/Static	Limited[2]	No	Free	Web
BioCyc [47]	Export Only	No	No	No	Free	Application
VisANT [43]	Yes	Yes[3]	Yes	No	Open Source	Application, Applet
CellDesigner [36]	No	Yes	Yes	Yes	Free	Application
PathCase [30]	Yes[4]	Yes	No	No	TSS License[5]	Application
VISIBIO*web*	Yes	Yes	Yes	Yes	Open Source	Web
PATIKA*web* [28]	Yes	Yes	Yes	No	TSS Lisence	Web

[1] BiNoM is a plug-in for Cytoscape

[2] Only cellular locations are represented as compounds; complexes are shown with simple nodes

[3] Compound support in the layout seems unreliable

[4] BioPAX support seems unreliable

[5] Tom Sawyer Software License is required

Table 2.1: A comparison of several popular pathway visualization tools [26].

2.2.1 Cytoscape

Cytoscape is an open source pathway editor, based on yFiles graph editing framework. The project aims the tool to be easily extendable by plugins, so that researchers can write their own analyzers for pathways. Today, there are many Cytoscape plugins, written by different groups, implementing a diversity of pathway analysis algorithms [1].

One pitfall of Cytoscape is that they do not make use of compound graphs, so they have an unusual way of representing complex molecules (Figure 2.10). Cellular locations are shown as text next to the name of the related molecule.

2.2.2 CellDesigner

CellDesigner [36] is a diagram editor for drawing gene-regulatory and biochemical networks. They use SBGN Entity Relationship and Process Diagram representations in drawings (Figure 2.11), and they can save the created models in SBML language. CellDesigner lets user to adjust kinetic parameters of the reactions and concentrations of the molecules, and performs simulations on the model.

2.2.3 PATIKA*web*

PATIKA*web* [28] is the front-end of the PATIKA database [24]. It is a web based pathway editor, which was built on JSP (JavaServer Pages technology) edition of the Tom Sawyer Visualization technology. Pathway representations are similar to SBGN Process Diagrams. PATIKA*web* draws pathways on a cell model; i.e., drawing area is divided into compartments representing cellular locations (Figure 2.12). The tool uses compound nodes for displaying molecular complexes, homologies, and abstractions. Graphs are laid out using the CoSE algorithm [29], specially designed for graphs with compound structures.

[1] For a complete list of plugins, refer to [21]

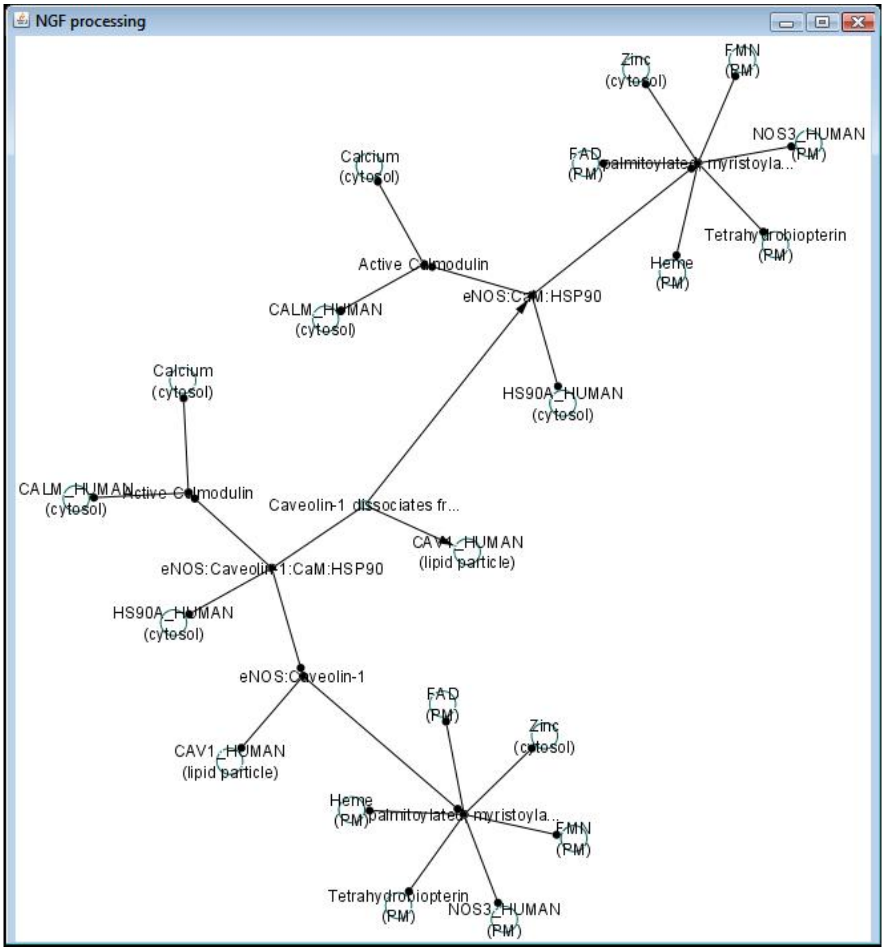


Figure 2.10: Cytoscape pathway showing dissociation of CAV1 from a big complex.

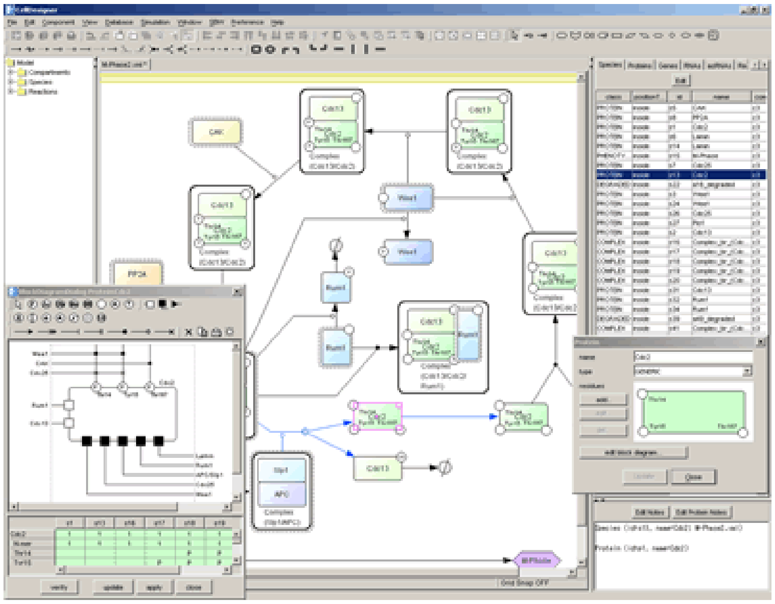


Figure 2.11: A screenshot of CellDesigner [14].

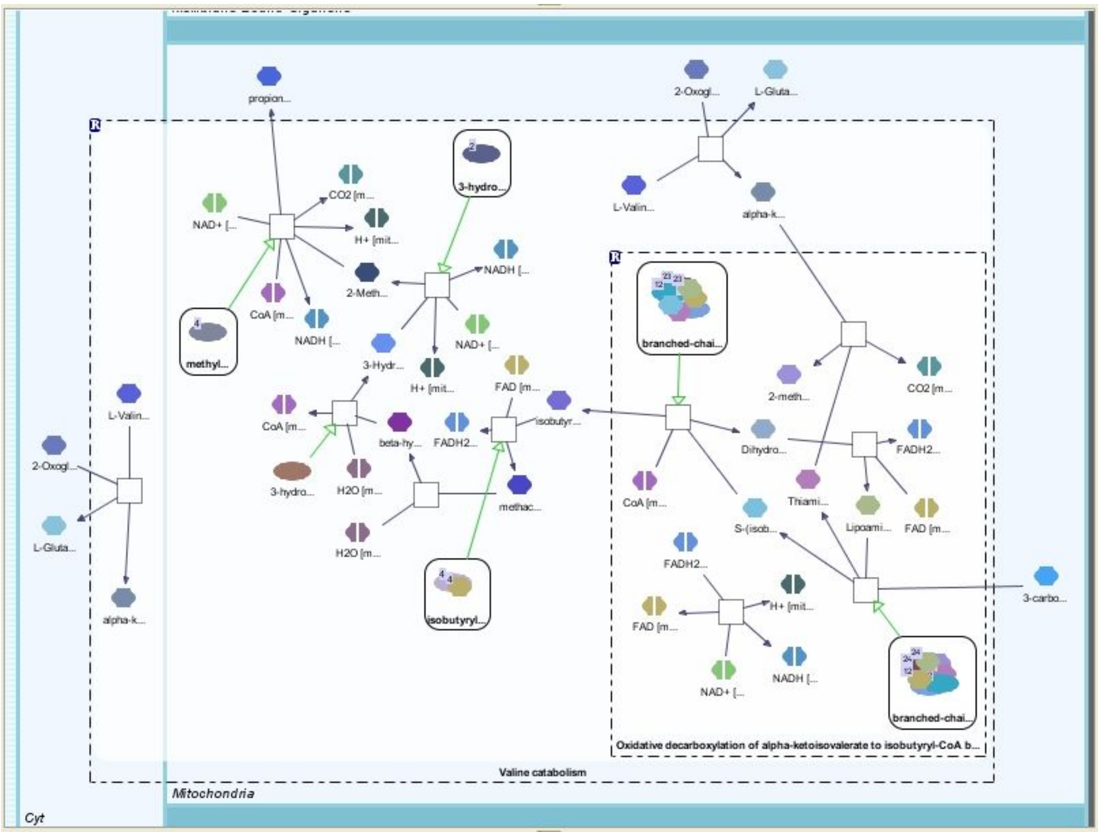


Figure 2.12: PATIKA*web* view of the pathway “Valine Catabolism”.

2.3 Reverse Engineering of Gene Regulatory Networks

Identification of a gene's regulatory process with controlled experiments is a costly procedure, which requires experiment setups that should involve the gene's promoter with a reporter, and all related transcription factors and their potential modulators. Since it is not practical to test all potential regulators in combinatorially many settings, studies generally focus on few regulators in limited conditions.

Gene expression microarrays can measure expression levels of all genes in a specific condition, thus have potential to provide insights on dependencies of genes to each other for expression. There are many studies that try to re-construct the gene regulatory network using large numbers of expression data. A review by Margolin and Califano [57] classifies reverse engineering methods as linear, probabilistic, and information theoretic, basis of which are summarized below.

All of these methods assume that expression of a gene is a function of other genes, and expression of genes are indicator of their protein activity.

2.3.1 Linear Models

Gene expression at time *t+1* can be formulated as the linear combination of other genes' expressions at time t plus some constant (Eq. 2.1).

$$x_i^{t+1} = \sum_j a_j x_j^t + c_i \tag{2.1}$$

$$X^{t+1} = A \times X^t + C \tag{2.2}$$

The relation is more formally represented using matrices, like in Eq. 2.2, where, X is the gene expression vector of size n, A is a $n \times n$ matrix, and C is the constants vector of size n. This formulation comes natural when worked on time-series expression data, where samples in the expression dataset belong to equally distributed time intervals, starting after a perturbation is applied to the system.

2.3.2 Probabilistic Models

Probabilistic frameworks try to estimate a *probability of expression* for the gene under condition of the expression status of other genes. The most popular probabilistic framework is the Bayesian network, which is a directed acyclic graph, where nodes represent gene expression statuses and edges represent dependencies between expressions. A sample formulation is given in Eq. 2.3, where x_i is the expression status of the i^{th} gene, $\pi_{i,j}$ represents the j^{th} parent of i^{th} gene, $a_{i,j}$ is the weight of the effect of j^{th} parent on i^{th} gene.

$$P(x_i) = \sum_j a_{i,j}\pi_{i,j} + c_i \tag{2.3}$$

Probabilistic approaches are generally applicable when gene expressions can be discretized, like high and low. While working on steady-state expression datasets of differing conditions, probabilistic framework is easier to use than a linear system because of absence of *time* in the formulation.

2.3.3 Information Theoretic Models

The information theoretic measure, mutual information (MI), can capture the dependency between gene expressions. MI is calculated using independent and joint entropies of gene expressions as in Eq. 2.4, where, S is the information theoretic entropy, and X_i is the expression vector of the i^{th} gene.

$$MI_{i,j} = S(X_i) + S(X_j) - S(X_i, X_j) \tag{2.4}$$

MI is similar to Pearson correlation in the sense that it measures a kind of dependence between variables; however, Pearson correlation assumes a linear relationship between variables while MI measures any kind of dependence. MI is guaranteed to be non-zero unless variables are statistically independent.

2.4 MINDY - Identifying Transcription factor modulators

Wang *et al.* propose an information theoretic approach for inferring modulators of transcription factors (TFs) from microarray data. They measure the mutual information between the TF and the target gene (t), conditional to the modulator candidate (M); i.e., $CMI(TF, t|M)$. The mutual information between TF and the target gene indicates dependency of expression of the target gene on the expression of the TF. In the presence of a modulation, they expect this value to be different in low and high values of the modulator (Eq. 2.5), and infer modulators with a high ΔCMI.

$$\Delta CMI = CMI(TF, t|M+) - CMI(TF, t|M-) \tag{2.5}$$

They test their method on B cells, 254 expression profiles, and identify modulators of Myc oncogene. They test all genes as potential targets, and all signaling proteins and other TFs as potential modulators. Low and high values of a modulator are determined by rank-ordering the expression data values, selecting first quartile as low and third quartile as high, and not using the second quartile. Among 542 signaling proteins and 598 transcription factors, MINDY identifies 91 signaling proteins and 99 TFs as modulators of Myc.

Chapter 3

Analysis of Process Description Graphs

This chapter provides the theoretical basis for pathway analysis as implemented in the software tools PATIKA*mad* and ChiBE.

3.1 Basics

Let $G = (V, E)$ be a graph with a non-empty node set V and an edge set E. An edge, $e = x, y$ or simply xy, joining nodes x and y is said to be incident with both x and y. Node x is called a *neighbor* of y and vice versa. A pathway graph $G = (V, E)$ is a graph, where some of the edges in E are marked as inhibition edges (e.g., an interaction that disables or impedes the target reaction node via the source state node).

A path between two nodes n_0 and n_k is a non-empty graph $P = (V', E')$ with $V' = n_0, n_1, ..., n_k$ and $E' = n_0n_1, n_1n_2, ..., n_{k-1}n_k$, where n_i are all distinct. n_0 and n_k are called the end points of path $P = n_0n_1...n_k$, whose *length*, denoted by $|P|$ is the number of edges on it. A path is said to be *directed* if all its ordered edges are directed in the same direction. A directed path P is called an *incoming* (*outgoing*) path of node n if P ends at *target* (starts at *source*) node n. A directed path is called *positive* (*negative*) if it contains an *even* (*odd*) number of inhibitors (i.e., inhibition edges).

Given node sets A and B, an $A-B$ path is a path with its ends in A and B, respectively, and no node of P other than its ends is from either set A or B. An $A-path$ is a path where one of its end nodes is in A, and no other nodes and interactions are from A.

The graph-theoretic distance $d_G(x, y)$, between two nodes x and y in graph G, is the length of a shortest $x-y$ path in G. If $G' = (V', E')$ is a subgraph of $G = (V, E)$, and G' contains all the edges $xy \in E$ with $x, y \in V'$, then G' is an vertex-induced or simply *induced subgraph* of G; we say that V' induces G' in G and write $G' = G[V']$. If node x is the starting node of a directed path that ends up at node y, then node y is said to be in the *downstream* of node x; similarly, node x is said to be in the *upstream* of node y. A node y in the downstream of a node x is a potential target of x; similarly, x is a potential regulator of y.

The graph type assumed in the rest of this Chapter (except section 3.5) is biological graphs that are similar to PATIKA mechanistic graphs or SBGN process diagram, which we call *process description graph*. The characteristic property of such graphs is that they follow the biochemical reaction paradigm, events are represented with a special node type (*transitions* in PATIKA), molecule nodes, or *states*, are related to the events through input, output, and effector relations (Figure 3.1).

Edges in a process description graph always have a direction. When two states are connected through a directed path, this implies that the state at the start of the path can have influence on the existence (or concentration) of the state at the end of the path. For instance, in Figure 3.1 the path from S1 to S4 implies that concentration change of S1 can affect the concentration of S4. Because of the presence of such a path, we say that S1 is at the *upstream* of S4, and S4 is at the *downstream* of S1.

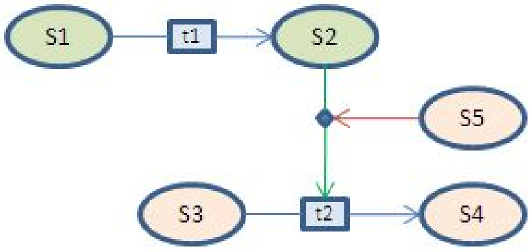


Figure 3.1: Sample process description graph with three different node types and 5 different edge types. Direction of the edges without an arrow is from the state to the non-state node. Green edge is for activation, while red edge represents inhibition.

Only the effector edges in a process description graph can be negative. In Figure 3.1 S1 is at the *positive upstream* of S4, and S4 is at the *negative downstream* of S5.

3.2 Visualization of BioPAX Using Process Description Graph

BioPAX language have some structural differences from process description graphs, which needs a conversion before visualization and analysis. BioPAX level 2 uses `PhysicalEntityParticipant` (`PEP`) objects as a link from `PhysicalEntity` (`PE`) to `Interaction` (`Conversion` and `Control`) objects. `PEP`s in BioPAX are not reusable objects, they are created per interaction, because `PEP`s also store the stoichiometry information which is specific to the `Interaction`. During conversion, each PEP that has the same modification features and the same cellular compartment corresponds to a unique *state* in process description graph (Figure 3.2).

`Conversion` in BioPAX can be bidirectional (reversible), however a transition in process description graph is strictly unidirectional. Any reversible `Conversion` in BioPAX is represented with two transitions in process description graph (Figure 3.3).

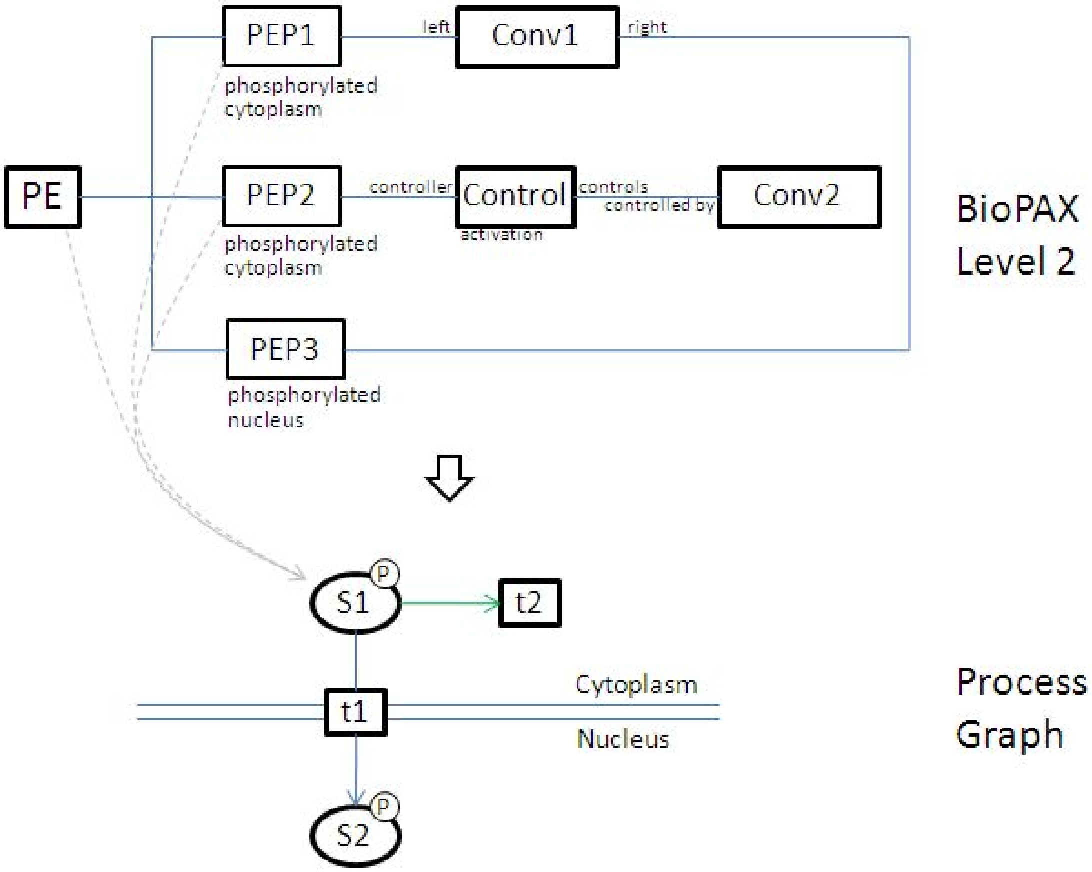


Figure 3.2: PEPs in BioPAX level 2 graph are grouped according to their modifications and cellular locations during conversion. Each group is represented with a unique *state* in the process description graph.

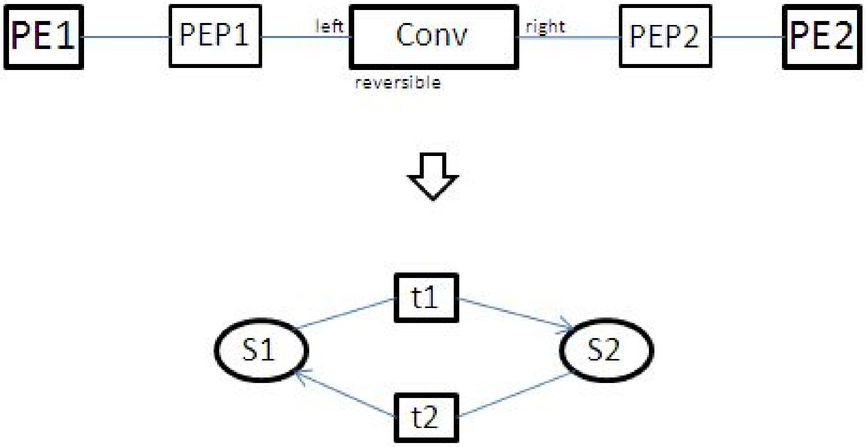


Figure 3.3: Reversible Conversion in BioPAX is represented with two *transitions* in process description graph.

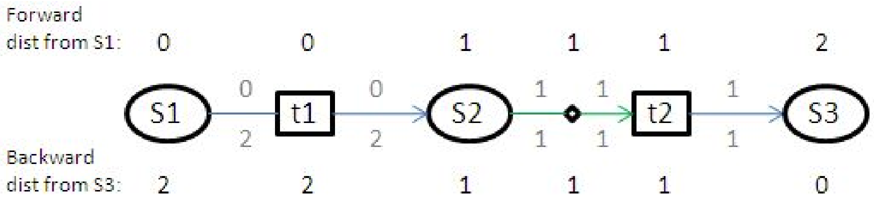


Figure 3.4: Desired distance labeling in a process description graph when states are in focus of the traversal. Nodes and edges on the S1-S3 path is labeled according to the distance from S1 (upper labels, forward distance), or distance from S3 (lower labels, backward distance). Distances are defined between *states*, however, there are advantages of defining distance labels for all nodes and edges. For instance, the sum of forward distance and backward distance of a node or edge on the S1-S3 path is equal to the state-based length of the path, which is 2.

3.3 Paths and Distances

Distances between nodes are often used in graph traversal algorithms. For instance the breadth-first search (BFS) guarantees that no node in distance $i + 1$ will be traversed before all nodes in distance i is traversed. When a graph has a single type of node and a single type of edge, distance between two nodes is simply calculated by the number of the edges on the path that connects them. However, when node and edge types are multiple, distances between a node type of focus can be different from the graph theoretical distance. In that case, we need to modify graph traversal algorithms to run using the *specific node-based* distance.

In process description graphs, molecular states are connected through other *non-state* nodes, like *transition* and *control* nodes. When the focus of a traversal algorithm is the molecular states on the network, it needs to define the *distance* as the distance between *states*. For instance in Figure 3.1, S5 – the upstream inhibitor of S4 – has a graph theoretic distance of 3 from S4. However, S5 is an immediate inhibitor of S4, and the state-based distance is 1.

Figure 3.4 shows an example *state-based* distance labeling on a path whose graph theoretical distance is 5. The state-based distance from S1 to S3 is 2 because they are connected by 2 events. Here, the non-state nodes and edges between states get the label of the state at their upstream. Upper labels in the figure show a forward labeling, i.e, distance from S1; while the lower labels show a backward labeling, i.e., distance from S3. The sum of forward label and backward label of an object on the path is equal to the state-based length of the path, which is 2 in this example.

3.4 Querying Paths on a Network

We previously designed a graph-theoretic querying framework, answering some important biological questions for PATIKA (mechanistic and bioentity) graphs [27], such as neighborhood, shortest path, graph of interest, paths of interest, and common stream. Algorithms implemented in this framework did not have to consider the heterogeneity of node types because PATIKA mechanistic graphs are bipartite; thus state-based distance of a path is always half of its graph theoretic distance.

Here we generalize these algorithms to process description graphs, which are not necessarily bipartite; however, node types can be labeled as *state* and *non-state*. All these algorithms are based on breadth-first search (BFS), so we first modify the well-known BFS to use the state-based distances in process description graphs, and then build other algorithms on this modified BFS.

3.4.1 BFS for Process Description Graphs

Algorithm 1 is a modified BFS, where search starts from the source set S, ends in target set T, runs in the direction specified by *dir* parameter, and continues until the limit distance k is reached. If there is no target set, T can be left empty; and if there is no search distance limit, k can be defined as infinite. The complexity of this modified BFS is the same with the ordinary BFS, which is $O(|V|+|E|)$ time complexity, where $|V|$ and $|E|$ are the number of nodes and edges, respectively.

The difference of this BFS from the regular BFS is that *breadth* is defined with the closest level of *states*. This is realized by using a priority queue instead of a regular queue. States are added at the end of the queue to be processed at as the next breadth (line 25), while non-state nodes are added at the head of the queue to be processed immediately (line 27). This BFS also uses a state-based labeling, like the labeling in Figure 3.4. It increments labels only when reaching a state from an edge during forward traversal (line 18), and when reaching an edge from a state during backward traversal (line 12). The algorithm uses color labels for the processing statuses of nodes: white means *not processed*, gray means *in queue*, and black means *processed*. The algorithm assumes initial node colors are white.

3.4.2 Neighborhood

Neighborhood of a set S of source nodes is defined as:

$$\begin{aligned} NB(S,k) = S \;\cup\; &\{x \mid x \text{ is a node on a S-path } P, \text{ and } |P| \leq k\} \\ \cup\; &\{e \mid e \text{ is an edge on a S-path } P, \text{ and } |P| \leq k\} \end{aligned}$$

Upstream or downstream neighborhoods of states in a process description graph can be queried with simple BFS calls (Algorithm 2).

3.4.3 Paths of Interest

Simplest strategy for searching relations between states is to search paths in between. We define the paths-of-interest (PoI) algorithm for searching paths between two given sets of states – source set S, and target set T – within a search distance limit k. This algorithm does not enumerate paths, but returns a merge graph of the related paths. Paths-of-interest is formally defined as:

$$PoI(S,T,k) = G\left[B\right], \text{ where } B = \{x \mid x \text{ is on a S-T path } P, \text{ and } |P| \leq k\}$$

Algorithm 1 BFS(S, T, dir, k)

```
Require: $dir$ is FWD or BKWD
Require: $S$ and $T$ contain only STATE-NODE
for all vertex $n \in S$ do
    $n.color \leftarrow$ GRAY
    $n.label(dir) \leftarrow 0$
$Q \leftarrow \emptyset$
$R \leftarrow S$
$Q.enqueue(S)$
while $Q \neq \emptyset$ do
    $u \leftarrow Q.dequeue()$
    for all incident edge $e$ of $u$ going in $dir$ do
        $R \leftarrow R \cup \{e\}$
        if $dir =$ BKWD and $u$ is STATE-NODE then
            $e.label(dir) \leftarrow u.label(dir) + 1$
        else
            $e.label(dir) \leftarrow u.label(dir)$
        $n \leftarrow e.otherEnd(dir)$
        if $n.color =$ WHITE then
            if $dir =$ FWD and $n$ is STATE-NODE then
                $n.label(dir) \leftarrow e.label(dir) + 1$
            else
                $n.label(dir) \leftarrow e.label(dir)$
            $R \leftarrow R \cup \{n\}$
            if $n \notin T$ and ($n.label(dir) < k$ or $n$ is not STATE-NODE) then
                $n.color \leftarrow$ GRAY
                if $n$ is STATE-NODE then
                    $Q.enqueue(n)$
                else
                    $Q.addFirst(n)$
            else
                $n.color \leftarrow$ BLACK
    $u.color \leftarrow$ BLACK
return $R$
```

Algorithm 2 Neighborhood(S, dir, k)

```
Require: $dir$ is FWD, BKWD, or BOTH
$R \leftarrow \emptyset$
if $dir$ = FWD or $dir$ = BOTH then
    $R \leftarrow R \cup \text{BFS}(S, \emptyset, \text{FWD}, k)$
if $dir$ = BKWD or $dir$ = BOTH then
    $R \leftarrow R \cup \text{BFS}(S, \emptyset, \text{BKWD}, k)$
return $R$
```

Algorithm 3 PoI(S, T, k)

```
$C \leftarrow \text{BFS}(S, \emptyset, \text{FWD}, k)$
ResetColors($C$)
$C \leftarrow C \cup \text{BFS}(T, \emptyset, \text{BKWD}, k)$
$R \leftarrow \emptyset$
for all vertex $u \in C$ do
    if $u.label(\text{FWD}) + u.label(\text{BKWD}) \leq k$ then
        $R \leftarrow R \cup \{u\}$
return $R$
```

An alternative version of the PoI algorithm uses the shortest path distance between source and target sets as the search distance limit. This is useful especially when we do not have any idea on the distances from S to T, so we can not provide a realistic k. So, we define the algorithm PoI-Shortest, searching paths from S to T using a length limit $shortest + k$ (Algorithm 4). Both PoI and PoI-Shortest algorithms have $O(|V| + |E|)$ time complexity.

Algorithm 4 PoI-Shortest(S, T, k)

```
$C \leftarrow \text{BFS}(S, \emptyset, \text{FWD}, \infty)$
ResetColors($C$)
$C \leftarrow C \cup \text{BFS}(T, \emptyset, \text{BKWD}, \infty)$
$R \leftarrow \emptyset$
$sd \leftarrow min(u.label(\text{FWD}) + u.label(\text{BKWD}))$ where $u \in C$
for all vertex $u \in C$ do
    if $u.label(\text{FWD}) + u.label(\text{BKWD}) \leq sd + k$ then
        $R \leftarrow R \cup \{u\}$
return $R$
```

3.4.4 Graph of Interest

Often times researchers do not have source and target sets to search a relation in between, but they have just a set of states, and want to learn if they are related. We define the graph-of-interest (GoI) algorithm for searching any path between a given set of states, within a search distance k, which is in fact a PoI call using the state set as both source and target (Algorithm 5). GoI is defined as:

$$GoI(S,k) = G\left[B\right], \text{ where } B = \{x \mid x \text{ is on a S-S path } P, \text{ and } |P| \leq k\}$$

GoI can alternatively use PoI-Shortest for searching paths within a distance of $shortest + k$.

Algorithm 5 GoI(S, k)

```
1: return PoI($S$, $S$, $k$)
```

3.4.5 Common Stream

There are already a number of algorithms for inferring highly connected or co-regulated subnetworks of cellular interactions and processes often called modules or pathways [13, 81, 9]. When analyzing these modules, we often want to know if there is a process or gene that is upstream of the genes in the module, which can provide a causal explanation for the co-regulation, and ultimately a way to control the module. Similarly, two pathways affecting the same mechanism in the cell is interesting since it suggests that a specific phenotype can have more than one molecular cause. For instance, Engelman et al. [31] discuss that drug resistance in lung cancer is related to an alternative pathway that leads to PI3K activation. Searching for common targets of signaling proteins can help to develop alternative treatment strategies.

Common downstream (upstream) of a source entity set S is the set of potential common target (regulator) entities that are in the downstream (upstream) of all entities in S. We describe the common-stream algorithm for identifying common upstream and downstream (determined by *dir* parameter) within a search distance limit k (Algorithm 6). Common downstream is defined as:

$$CD(S,k) = \{x \mid \forall a \in S \ (\exists P \mid P \text{ is a path from } a \text{ to } x, \text{ and } |P| \leq k)\}$$

Common upstream is defined similarly. The algorithm simply executes a BFS search from each source node and increment *reached* count of the nodes in the resulting BFS tree. When a node is reached from all of the source nodes, it is collected in the resulting common stream. This algorithm has $O((|S| \times |V|) + |E|)$ running time complexity.

Algorithm 6 CommonStream(S, *dir*, k)

```
Require: $dir$ can be FWD or BKWD
$C \leftarrow R \leftarrow \emptyset$
for all vertex $u \in S$ do
    $C \leftarrow \text{BFS}(\{u\}, \emptyset, dir, k)$
    for all vertex $n \in C$ do
        $n.reached \leftarrow n.reached + 1$
    ResetLabel($C, dir$)
    ResetColor($C$)
for all vertex $v \in C$ do
    if $v.reached = |S|$ then
        $R \leftarrow R \cup \{v\}$
return $R$
```

3.4.6 Inter-Compartment Paths

A signal in or outside of the cell is transmitted through cellular locations towards its destination. This is mainly controlled through receptors on the boundary surfaces of compartments, and carrier molecules that assist other molecules in their transmission. Paths between different cellular locations often capture these signaling events. We define Inter-compartment-paths query (Algorithm 7) as a special application of paths-of-interest query. This query executes a paths-of-interest query from states in a compartment to the states in the other compartment.

Algorithm 7 InterCompartmentPaths(*compartment_1*, *compartment_2*, k)

```
$S \leftarrow$ states in $compartment\_1$
$T \leftarrow$ states in $compartment\_2$
return PoI($S$, $T$, $k$)
```

3.5 Expression Data on Pathways

Microarray experiments take a snapshot of the cell, showing expression of almost the entire genome. Since gene expressions can be affected from their upstream in a cellular network, and since they can affect expression of their downstream, analyzing expression data on pathways can be informative. One of the simple way of integrating expression data with pathways is visualization of data values on the molecule nodes of the network.

Visualization of expressions on pathways needs a mapping from expression values to molecules in the pathway. This mapping can be obtained by matching external references on the expression data and pathway. However, often times more than one row of expression data match with a molecule on the pathway, and these rows can have dramatically different values. Presence of multiple rows per gene is generally due to presence of several isoforms of that gene, which are measured separately on the expression profile. The most accurate way of visualization in that case is to represent all the related values on the molecule nodes; however, this is generally not a practical solution because of space limitation and increasing visual complexity of the pathway drawing. An approximation is to display only the highest value that is matched with the molecule; so that we define expression of a gene as the expression of at least one of its isoforms. This approximation fails when isoforms have different functions, but this time the cause of the failure is not incorrect mapping but absence of sufficient details on the network.

Expression data is a measure for the concentration of RNA molecules in the cell. Thus, the correct way of mapping is to map expressions to RNA states on the network. Unfortunately, RNA states are highly under-represented in popular pathway databases. Representing expressions on the protein states is the next option, and applied by several pathway editors [51, 5]. When represented on proteins, expression values tend to be interpreted as indicators of protein levels, or activities in the cell. This is a kind of an approximation ignoring translational and post-translational control of proteins, which is probably wrong in many cases, requiring caution when used.

Visualizing a single profile is generally not very informative since expression values do not tell much about activity of gene products. However, comparisons between two profiles tell many things; a change in expression values is often interpreted as a change of activity of gene products in the same direction.

3.6 Causative Paths

Pathway databases contain information about possible interactions and reactions between molecules in a cell. Usually, this data is created by manually curating biological literature and can span multiple experiments from different tissues, organisms and contexts. When taken as an interconnected network, these interactions and reactions offer a causal model of a cell's response to stimuli. For instance, in a typical microarray experiment, relatively small portions of this network are differentially active between the control and the sample, and determining these parts can be extremely useful for finding causal explanations for the correlations observed in the data.

Change of an expression value can be related to change of other gene expressions through a path in the cellular network. If a path in the network potentially explains expression change of the end-state with the expression change of the start-state, then we call it a *causative path*.

The last *transition* in a causative path should be a *transcription*, or should at least be related to gene expression. A positive causative path will have similar expression changes at its start-state and end-state. Similarly, a negative path can be causative only when it has different expression changes at its start-state and end-state (Figure 3.5).

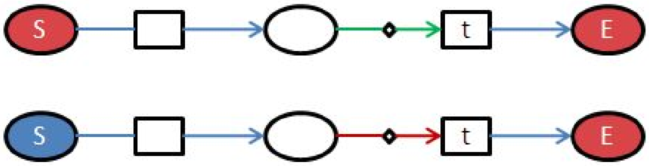


Figure 3.5: Examples of two causative paths. Red state: upregulated, blue state: downregulated. Transitions with label t are transcription events. Red edge: inhibition, green edge: activation.

Chapter 4

Discovering Modulators of Gene Expression

Our current knowledge of the modulation of transcription factors comes mainly from experimental studies that measure the expression levels of a few target genes (such as [62] and [66]) or the expression level of an artificial reporter gene with a "canonical promoter" (such as [77]). While these experiments provide invaluable insight, they do not tell the whole story. In order to detect context-dependent, target-specific effects of modulators, system-scale methods are required. Gene expression profiles are now extensively used for inferring causal relationships between transcription factors and target genes. The models produced from gene expression profiles, often referred as "gene regulatory networks", or simply "gene networks", differ significantly in their semantics and level of detail. Margolin and Califano [57] provide a comprehensive review of these methods and classify them under three groups: linear, graph-theoretic, and information-theoretic models (Section 2.3). The majority of these methods focus on modeling either causal relationships between gene expression levels as binary interactions, or linear integration of expression values.

Expression level of genes can also be affected by non-modulator proteins such as alternative transcription factors, generic inhibitors of transcriptional machinery or regulators of mRNA degradation. A modulator is defined by its dependency on the transcription factor in order to exert its effect on the target. When the transcription factor is not present, at least a part of the modulator activity should be rendered ineffective. This implies a ternary, non-linear relationship, analogous to the electrical transistor, between the activity levels of the two "inputs", the transcription factor and the modulator, and the "output", the target gene expression. Using a sufficiently large set of expression profiles, these relationships can be detected by looking at the correlations between expression levels of candidate modulators with the expression level of a transcription factor and its target genes. Assuming that the expression level is an indicator of modulator and transcription factor activity, the dependency between modulator and target expression must increase as the concentration of the transcription factor increases. Therefore, we expect to observe a transcription factor-dependent correlation between modulator and target.

Wang *et al.* [76] propose MINDy, an information theoretic algorithm for detecting modulators. They test the conditional mutual information (CMI) between the transcription factor and the target gene, and its dependency on the modulator candidate (Section 2.4). This is, in essence, the aforementioned non-linearity principle. Building upon the same principle, we present GEM (Gene Expression Modulation) [4], a probabilistic method for detecting modulators of transcription factors using *a priori* knowledge and gene expression profiles. For a modulator / transcription factor / target triplet, GEM predicts how a modulator-factor interaction will affect the expression of the target gene. GEM improves over MINDy by detecting two new classes of interaction that would result in strong correlation but low ΔCMI, can filter out *logical-or* cases and offers a more precise classification scheme. A detailed comparison of GEM and MINDy is provided in Section 4.2.2.

In the following sections, we explain our method and assumptions and apply GEM to predict modulators of Androgen Receptor (AR). We compare our results with a recent literature review on modulators of AR and show that GEM correctly predicts a significant number of its modulators and can provide additional insight into the mechanism of modulation and affected targets. We observe that these

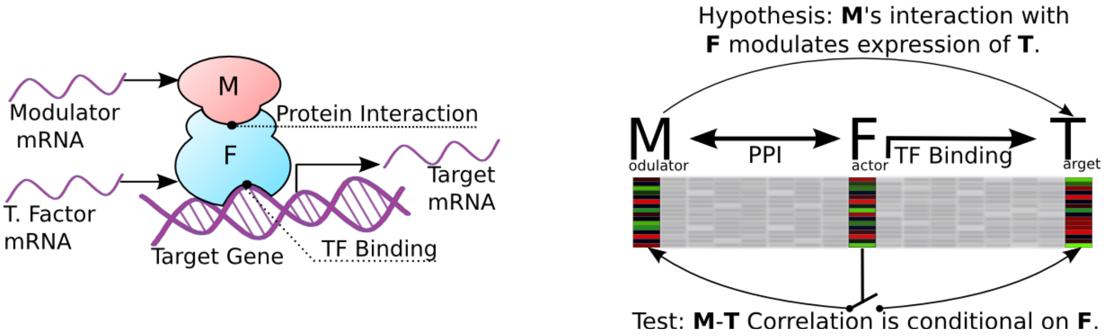

Modulator mRNA
M
Protein Interaction
F
Target mRNA
T. Factor mRNA
Target Gene
TF Binding
Hypothesis: **M**'s interaction with **F** modulates expression of **T**.
Modulator
PPI
Factor
TF Binding
Target
Test: **M**-**T** Correlation is conditional on **F**.

Figure 4.1: Left: GEM is based on a simple model of gene regulation. A modulator interacts with a transcription factor to affect the expression of a target. Right: Initial hypotheses are generated by combining known protein-protein and protein-DNA interactions which are then tested against a set of gene expression profiles.

modulators cannot be easily classified into co-activator/co-repressor categories. Most modulators will selectively increase the expression level of some AR targets while decreasing the others, a property we call *bimodality*.

4.1 GEM Method

GEM uses three types of input, protein-protein interactions, transcription factor-target relations, and gene expression profiles. Proteins that are known to interact with the transcription factor are considered as potential modulators and transcription factor-target binding data are used to obtain a list of target genes for each transcription factor. These two types of interactions are combined to build a large number of small causal hypotheses of the form: "Modulator protein M, via transcription factor F affects the expression of the target gene T". The modulator hypothesis predicts that correlation between the expression levels of the modulator and the target must change as the level of transcription factor changes. We use this dependency as a metric of the interaction between the modulator candidates and the transcription factor to select most likely modulators (Figure 4.1).

We can estimate this relation with the following model:

$$E(t) = h_c + h_m(m) + h_f(f) + g(m, f) \tag{4.1}$$

Where, m, f and t are expression levels of the **m**odulator, transcription **f**actor, and **t**arget, respectively. $E(\mathrm{t})$ is the expected value of t. h_m and h_f represent the effect of m and f, respectively, on t by themselves alone (main effects), while g represents the effect of their interaction. If interaction of m and f has an effect on t, we expect g to be *non-zero.*

There is reason to believe that h_m and h_f can be approximated with linear functions [17]. On the other hand, the nature of g can vary significantly from triplet to triplet, and cannot be covered by a single class of continuous functions. If g is monotonic, however, we can use a discrete model such as the one described by Wang *et al.* [76]. This allows us to look for non-zero g components without worrying about the actual mechanism. When we transform the expression values of genes to activity levels 0 and 1, our model becomes:

$$P(\mathrm{t}' = 1) = \alpha_c + \alpha_m \mathrm{m}' + \alpha_f \mathrm{f}' + \gamma \mathrm{m}'\mathrm{f}' \tag{4.2}$$

Given a set of expression profiles, we estimate alpha coefficients by calculating the observed proportions of $\mathrm{t}' = 1$, conditional on m' and f'. We then select triplets with a high γ coefficient that satisfy a false discovery rate threshold after multiple hypothesis testing correction.

A high γ alone, however, is not sufficient to infer modulation. Some non-linear relationships, such as *logical-or* of M and F cannot be explained by modulation. To remove these false positives, and to infer the mode of action of the modulator, we classify the non-linear triplets based on their proportion patterns and select those that can be explained by a simple, direct modulation. We report these modulators along with their respective targets and their mode of action.

4.1.1 Construction of Triplets

To construct our initial set of hypotheses, in the form of a modulator-factor-target triplet, we combine existing protein-protein and transcription factor-target interactions. Proteins known to interact with a transcription factor, but not targets of the factor themselves, are considered as potential modulators for all targets of the transcription factor. Large integrated protein-protein interaction datasets are already available [50], and known targets of transcription factors can be obtained from literature curation [45] [60], sequence based prediction [45], and ChIP-Chip experiments [11].

4.1.2 Selection of Expression Data

Using gene expression profiles we can directly measure the level of expression for target genes and estimate activities of M and F from their expression levels. For this estimation to be accurate, expression profiles must satisfy the following two conditions:

- *There is a steady state expression level for genes.* A change in the expression levels of M and F will be reflected in their protein abundance and expression after a delay. Without steady state property, we cannot correlate m, f, and t in the same expression profile.
- *Expression levels of* M *and* F *are correlated with their protein abundance.* Studies demonstrated that there is a lower correlation between expression levels and protein abundance than expected [39]. This correlation, however, increases significantly if the variance of expression values are high.

In addition to these conditions, f and m should have sufficient variance in the expression dataset. If one or both genes have relatively constant expression, then this may cause three problems:

- A low correlation between mRNA and protein abundance is expected.

- There will not be enough "perturbation" in the data set to infer M and F's effect on T.

- There is a possibility of detecting fine-tuning feedback loops as modulations.

Ideally, m and f should have high variation and low correlation in the samples.

Gene expression profiles of 2158 human tumor samples published by expO (Expression Project for Oncology) is currently the best publicly available dataset for our purposes [32]. The variety of tumor samples used in this study increases variation and thus helps reduce correlations between m and f due to the context. There are, however, some genes in the expO set that have inadequate variation in their expression levels (variation of log values less than 1) and these are left out of our analysis.

4.1.3 Discretization and Conditional Proportions

We divide rank-ordered expression values of a gene by tertiles and further discretize the triplets using:

$$x' = \begin{cases} 1, & \text{if } x \text{ is in upper tertile} \\ \mathit{null}, & \text{if } x \text{ is in middle tertile} \\ 0, & \text{if } x \text{ is in lower tertile} \end{cases} \tag{4.3}$$

This simple strategy has been shown to maximize entropy among groups [18] and is similar to the one used by Wang *et al.* We also explored more sophisticated (and computationally expensive) strategies including dynamically determining optimal threshold for each triplet that maximizes entropy; however, these did not yield substantial changes in our results.

After discretization, each experiment falls into one of the 27 possible bins based on the ternary state of m$'$, f$'$, and t$'$ (Figure 4.2, Left). While calculating

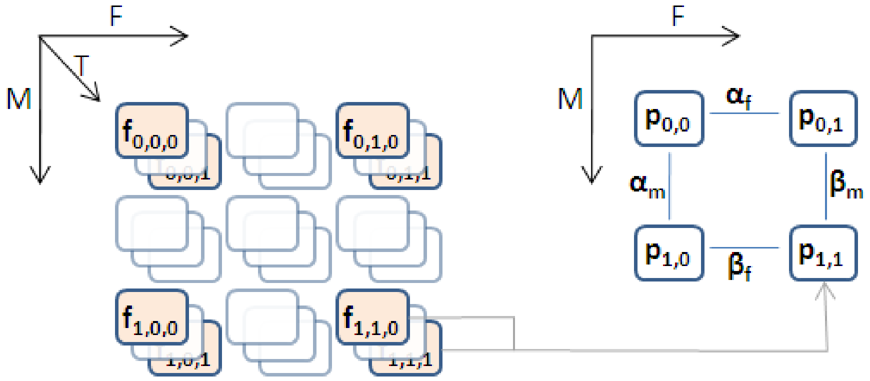


Figure 4.2: Left: Samples are ranked and divided into 27 possible bins. Samples with middle values are discarded and frequencies from 8 "corner" bins are used for the rest of the analysis. Right: For each combination of m,f states, proportions of t being high are derived from frequencies. Pairwise differences of proportions provide estimates for α and β values.

the interactions, we only consider the 8 bins, where none of the genes has *null* value. Observed frequencies of these states are denoted by $\hat{f}_{\mathrm{m}',\mathrm{f}',\mathrm{t}'}$.

We then calculate the proportions of $\mathrm{t}' = 1$ for each combination of states of f' and m':

$$\hat{p}_{\mathrm{m}',\mathrm{f}'} = \frac{\hat{f}_{\mathrm{m}',\mathrm{f}',1}}{\hat{f}_{\mathrm{m}',\mathrm{f}',0} + \hat{f}_{\mathrm{m}',\mathrm{f}',1}} \tag{4.4}$$

4.1.4 Selection of Significant Triplets

Observed proportions are conceptually similar to biological experiments. $\hat{p}_{1,1}$ is our test case, where both f and m are high; thus, an interaction is expected. $\hat{p}_{0,0}$, $\hat{p}_{1,0}$ and $\hat{p}_{0,1}$ are the controls; here, we do not expect an interaction to occur as at least one of the interacting partners is missing.

By using the differences of observed proportions we can estimate the α coefficients in Eq 4.2 (Figure 4.2, Right):

$$\hat{\alpha}_c = \hat{p}_{0,0} \tag{4.5}$$

$$\hat{\alpha}_f = \hat{p}_{0,1} - \hat{p}_{0,0} \tag{4.6}$$

$$\hat{\alpha}_m = \hat{p}_{1,0} - \hat{p}_{0,0} \tag{4.7}$$

We can also estimate the effect of F and M when their interacting partner is present:

$$\hat{\beta}_f = \hat{p}_{1,1} - \hat{p}_{1,0} \tag{4.8}$$

$$\hat{\beta}_m = \hat{p}_{1,1} - \hat{p}_{0,1} \tag{4.9}$$

Finally, $\hat{\gamma}$ gives us a metric for the effect of interaction:

$$\hat{\gamma} = \hat{\beta}_f - \hat{\alpha}_f = \hat{\beta}_m - \hat{\alpha}_m = \hat{p}_{1,1} - \hat{p}_{0,1} - \hat{p}_{1,0} + \hat{p}_{0,0} \tag{4.10}$$

Any significant triplet must have a non-zero $\hat{\gamma}$. This, however, is not sufficient, as a synergistic effect can result from relationships other than direct modulation. For example, consider the case where M and F are two transcription factors competing for the same binding site to activate expression of T. When F is high, there will be low M-T correlation – a non-linear relation which might have significant γ. Such cases occur when effects of M and F are similar but independent, and there is a cap on the T expression levels due to a third factor, such as the DNA binding site. The nature of such a relationship between M and F is a *logical-or* as opposed to *logical-and* in modulation. Although interesting, we can not apply our statistical inference to these relationships due to the hidden third factor.

If M is affecting T directly through F, it must be *active when F is high*. More formally, $\hat{\beta}_m$ must be significantly different than zero, and must either have a larger absolute value or have a different sign than $\hat{\alpha}_m$.

As a result, all of the following null hypotheses must be rejected for a triplet to be inferred as a direct modulation:

$$H_1 : \gamma = 0 \qquad H_2 : \beta_m = 0 \qquad H_3 : \frac{\alpha_m}{\beta_m} \geq 1 \tag{4.11}$$

4.1.5 Significance of the Difference of Proportion Pairs

α and β values are estimated using independent proportions $p_{0,0}$, $p_{0,1}$, $p_{1,0}$ and $p_{1,1}$ (Eq. 4.6 - 4.9). When M and F has no effect on T expression, these proportions will be approximately normally distributed with mean zero. Similarly, the difference between two proportions are approximately normally distributed with mean zero when the change in the conditions does not have an effect on T.

Each $p_{i,j}$ was calculated using frequencies as in Eq. 4.12.

$$p_{i,j} = \frac{f_{i,j,1}}{n_{i,j}} \tag{4.12}$$

$$n_{i,j} = f_{i,j,0} + f_{i,j,1} \tag{4.13}$$

The variance of proportion difference $p_{i,j} - p_{k,l}$ is estimated in Eq. 4.14 [34].

$$Var(p_{i,j} - p_{k,l}) = p_{ijkl}\, q_{ijkl}\, (\frac{1}{n_{i,j}} + \frac{1}{n_{k,l}}) \tag{4.14}$$

$$p_{ijkl} = \frac{f_{i,j,1} + f_{k,l,1}}{n_{i,j} + n_{k,l}} \tag{4.15}$$

$$q_{ijkl} = 1 - p_{ijkl} \tag{4.16}$$

Using the variance we can asses the probability of the measured difference to belong to this distribution:

$$P(x \in G(0, Var(x))) = 1 - erf(\frac{x}{\sqrt{2Var(x)}}) \tag{4.17}$$

$$erf(y) = \frac{2}{\sqrt{\pi}} \int_0^y e^{-t^2} dt \tag{4.18}$$

4.1.6 Significance of γ

γ is estimated using proportions as in Eq. 4.10. When the interaction between M and F does not affect T, γ will be approximately normally distributed with mean zero. Variance of this distribution is estimated in Eq. 4.19.

$$Var(\gamma) = p'q'(\frac{1}{n_{0,0}} + \frac{1}{n_{0,1}} + \frac{1}{n_{1,0}} + \frac{1}{n_{1,1}}) \tag{4.19}$$

$$p' = \frac{f_{0,0,1} + f_{0,1,1} + f_{1,0,1} + f_{1,1,1}}{n_{0,0} + n_{0,1} + n_{1,0} + n_{1,1}} \tag{4.20}$$

$$q' = 1 - p' \tag{4.21}$$

We use Eq. 4.17 for assessing the probability of a measured γ to belong to this normal distribution around zero.

4.1.7 Category of Action

Using $\hat{\alpha}_f$, GEM classifies unmodulated F activity into three classes: activator, inhibitor, and inactive. Similarly, by comparing $\hat{\alpha}$ and $\hat{\beta}$ coefficients, modulators are classified into three classes – they can enhance, attenuate, or invert the activity of the transcription factor. There are 6 possible categories that would have a high γ. These cases and their interpretations are listed in Table 4.1, Table 4.2, and in Figure 4.3.

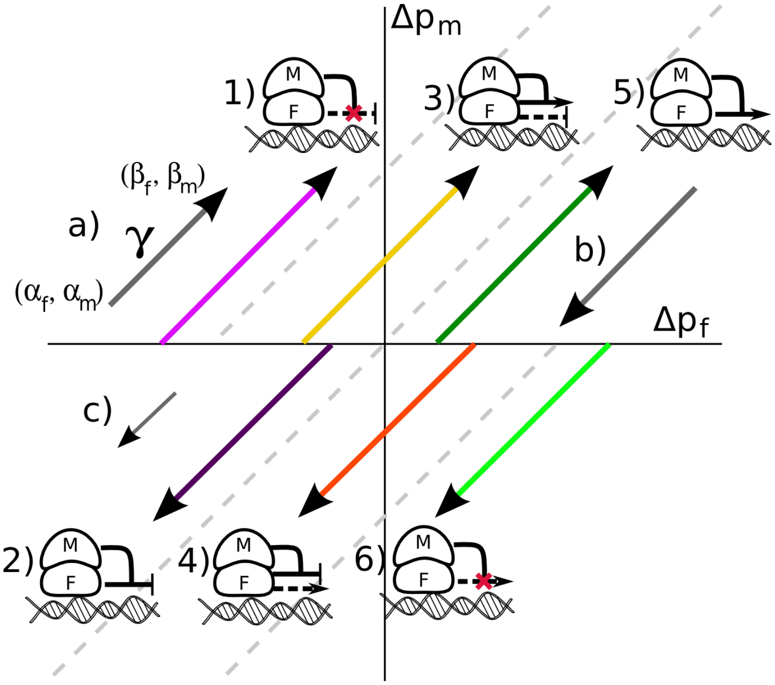


Figure 4.3: Classifying modulators using proportion differences: a) A triplet can be represented as a vector $\langle(\alpha_f, \alpha_m), (\beta_f, \beta_m)\rangle$. The size of the vector is proportional to γ. b) An example of *logical-or* case. c) An example of too small γ. Most of the triplets fall into one of these categories and are filtered out by GEM. 1-6) Representative vectors for each category of action in Tables 4.1 and 4.2, drawn assuming $\alpha_m = 0$.

Modulation Category	Explanation
Attenuates Inhibition	F, alone, inhibits T – M attenuates F activity.
Enhances Inhibition	Modulated F inhibits T.
Inverts Inhibition	F, alone, inhibits T – M inverts F activity.
Inverts Activation	F, alone, activates T – M inverts F activity.
Enhances Activation	Modulated F activates T.
Attenuates Activation	F, alone, activates T – M attenuates F activity.

Table 4.1: Interpretation of the categories of modulation.

Modulation Category	γ	α_f	β_f	β_m	$\alpha_f + \beta_m$
Attenuates Inhibition	+	-			
Enhances Inhibition	-		-	-	-
Inverts Inhibition	+	-	+	+	+
Inverts Activation	-	+	-	-	-
Enhances Activation	+		+	+	+
Attenuates Activation	-	+			

Table 4.2: Inequality constraints that the category of modulation should satisfy. “+” and “−” signs in the columns indicate significantly positive and negative values, respectively. Note that this categorization is formulated for triplets for which the null hypotheses in Eq. 4.11 were also rejected.

4.2 Results and Discussion

4.2.1 Inferring Modulators of the Androgen Receptor

Androgen Receptor (AR) is critical to the development and maintainance of male sexual phenotype and is also implicated as a central component in development of prostate cancer. Heemers *et al.* provide an extensive list of AR modulators and targets [40]. In the AR literature, modulators are often classified as co-activators or co-repressors. However, the semantics of this binary classification can be ambiguous; for example, “Is a modulator that attenuates the inhibitory action of a transcription factor a co-activator or co-repressor?” Another implicit assumption is that most modulators are unimodal; that is, they have a single type of effect which is either a co-activator or a co-inhibitor for all targets. Heemers *et al.* list only 12 out of 192 modulators as bimodal. Since for most modulators only a few targets are examined in the literature, we expect to have an observation bias towards unimodality. The extent of this bias, however, is not obvious. To answer these questions, and gain insight to the AR biology, we have applied GEM to infer modulators of AR.

For this experiment, we used the expression dataset provided by Expression Project for Oncology (expO), which contains 2158 profiles from various cancer tissue samples. Target genes were compiled by combining 40 known AR targets in Heemers *et al.* and 30 AR targets listed in TRED [45]. 134 proteins were listed

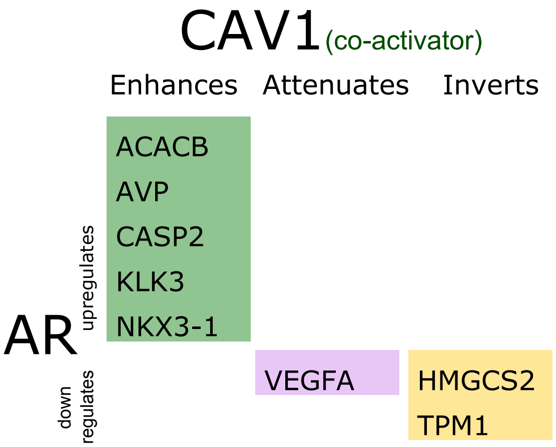


Figure 4.4: Target genes of the Androgen Receptor detected to be modulated by CAV1. KLK3, also known as PSA, is upregulated as well as 4 other important tumor growth related genes.

in HPRD [50] as interactors of AR forming the modulator candidate set. We removed genes that are both modulators and targets of AR as well as those that lacked adequate variability in the expression profiles. We used GEM to detect which of these 134 proteins modulate AR and compared our results with the list provided in Heemers *et al.*

For each modulator, GEM predicts its targets and its category of action. For example, Figure 4.4 lists the inferred target genes of CAV1 modulation. CAV1 was previously shown to positively regulate AR activity [55] and was associated with prostate cancer and aggressive PSA (KLK3) recurrence. We observe that expression levels of all 8 predicted targets were increased in response to CAV1, including PSA. Four of the eight genes have various growth promoting functions including fatty acid metabolism (ACACB), ketogenesis (HMGCS2), and angiogenesis (AVP and VEGFA). CASP2 and NKX3-1 have, however, tumor suppressor functions and are also upregulated by CAV1. These results show a complicated picture of modulation by CAV1 but are in agreement with previous studies that show both anti-tumor and metastatic functions for CAV1 [33].

CAV1 fits in nicely with the co-activator classification in the review by Heemers *et al.* Most targets of CAV1 fall into "Enhances Upregulation" class and inverting or even attenuating downregulation can be classified as co-activating.

Following from this observation, we looked at whether the results inferred by GEM agree with the review for the other modulators.

Using a 1% false discovery rate, we identified 47 modulators, covering 33 of the 192 modulators listed in Heemers *et al.* The 25 modulators with the most targets detected by GEM are listed in Figure 4.5 along with their classification in Heemers *et al.* Since we are limiting ourselves to direct modulators, and have a very conservative false discovery rate, this is a quite good recall. On the other hand we have predicted 14 modulators that were not listed in the review, including two master regulators of AR – EGFR and RUNX1. When we searched the literature for unlisted modulators with the most targets (EGFR, RUNX1, CDC2, CASP1, and MED1), we were able to find supporting evidence for modulation. Recchia *et al.* demonstrated the cross-talk between EGFR and AR pathways by investigating their effect on CD1 expression [66]. They claim that CD1 expression requires both EGFR and AR activity. Ning *et al.* identified modulation of mouse Slp by RUNX1 via AR [62]. Moilanen *et al.* show that CDC2 phosphorylates N-terminal domain of AR, which contains the major transactivation function [61]. Wellington *et al.* report cleavage of AR by CASP1 [79]. Wang *et al.* detect that MED1 plays an important co-regulatory role in AR-mediated gene expression [77]. These results show that GEM can complement literature reviews and can identify likely modulators from protein interactors of transcription factors. More importantly, GEM can infer target-specific mechanisms for each modulator.

Unlike CAV1, we observe that most modulators are bimodal. Of the top 25, only JUN and PIAS2 are listed as bimodal in Heemers *et al.* This difference in the frequency of bimodal modulators predicted by our method and those found in the literature supports our supposition that many modulators are classified as co-activators or co-repressors only because they were tested on a restricted set of target genes. We also observe that the number of targets for each modulator varies from 1 to 27. Although the target sets are far from being complete, they are sufficiently large so we expect the distribution of targets to be representative. Our results show that there is a spectrum of very specific modulators with a few targets to few master regulators that affect a majority of AR targets.

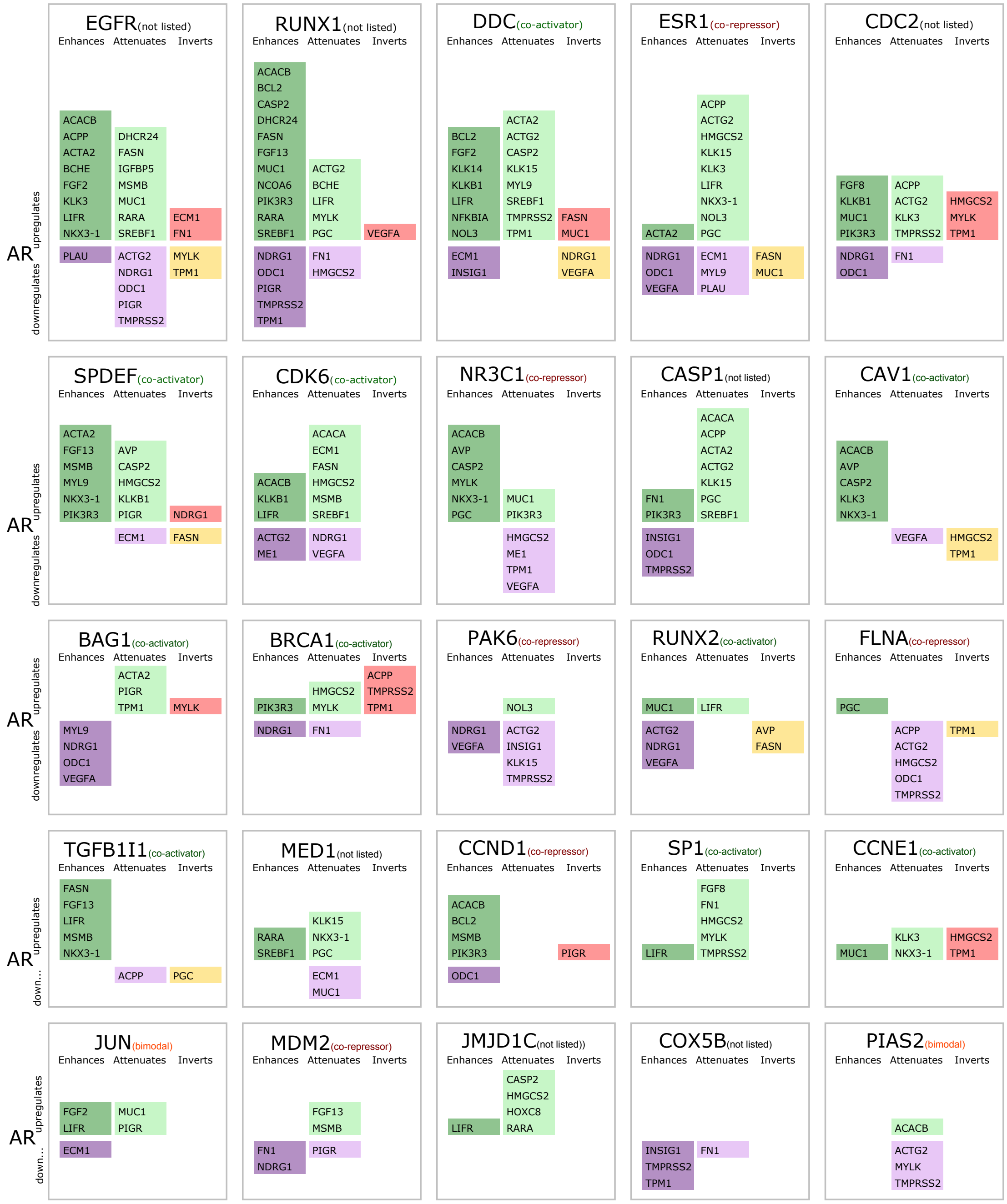


Figure 4.5: Top modulators of Androgen Receptor: each box contains targets affected by the modulator organized by categories of action and color coded. If the modulator is listed in the review by Heemers *et al.*, it is noted next to the name of the modulator. Most modulators have different effects for different targets and do not necessarily follow the classification in the review.

As previously mentioned, GEM requires high variance in expression values. When we do not filter out low variance genes, GEM detects NCOA3 as negative modulator of AR for most of the target genes. NCOA3 is a generic nuclear receptor co-activator whose expression does not change much in the cell. Heemers *et al.* show that NCOA3 expression is negatively regulated up to 0.5 fold by AR activation. When the expression of a candidate has low variation, such feedback loops can lead to false inference. In the same study, the effect of AR activation on other known modulators including some of the modulators in Figure 4.5 (DDC, BRCA1, BAG1, CAV1, FLNA, TGFB1I1, and PAK6), were also reported. Since these genes have very high variance in the dataset, however, these feedback effects can only account for a small fraction of the observed expression level changes.

We performed a second analysis using GEM on all cancer related transcription factors and their targets in TRED. Using interactors in HPRD as modulator candidates we identify 435 M-F pairs in the result. These include 57 TFs and 295 modulators, in which we also observe that the type of modulation depends on the target gene.

4.2.2 Comparison with MINDy

Both MINDy and GEM infer modulation of transcription factors based on factor-dependent correlations between modulators and targets. MINDy measures the differential conditional mutual information (ΔCMI) between transcription factor and target in low and high conditions of modulator (*M-* and *M+*). Since mutual information is a non-negative measure, however, ΔCMI does not differentiate between the negative and positive modes of modulation. This can be a problem when the factor has opposite effects under *M-* and *M+*, which results in high mutual information in both cases, and in turn low ΔCMI. An example of such a relation is the effect of EGFR on the relation of AR with its target MYLK. GEM detects that AR inhibits MYLK in EGFR- and activates MYLK in EGFR+. 10% of GEM result triplets with AR have non-significant ΔCMI and would not been able to be detected by MINDy.

MINDy treats all signaling proteins as modulator candidates, whereas we propose a much more conservative approach – we use only known interacting proteins. Using known protein interactors has the advantage of producing hypotheses about direct interactions that are immediately testable. There are combinatorially many indirect modulators and to test them, one has to supply the intermediary molecules to the system. This makes indirect modulators harder to test, especially *in vitro*. Also, dependency between M and F activity on T can be a result of non-causal relations – if any of the M, F, and T genes were replaced with a highly correlated substitute, there would still be a non-linear dependency. When we use *a priori* interactions to construct our triplets, a substantial amount of indirect and non-causative cases are filtered out. As a tradeoff our method loses some coverage due to missing or incorrect information in the source databases.

Similar to γ, ΔCMI would also detect a *logical-or* relation between M and F. In the case of AR, one third of our result triplets were classified as *logical-or* and filtered out. Unlike our approach, MINDy would not differentiate *logical-or* from modulation. These relationships can be meaningful in other contexts, such as genetic interactions. They, however, do not fit into the biological description of modulation, where the modulator affects the target through the factor. We believe that there is a value in basing the method on a biological model and fine tuning assumptions and restrictions based on it, so that the biological interpretation of the results are not ambiguous and they are more testable. To support other biological models (e.g. genetic interactions) we are developing a customizable GEM service where the user can select different *a priori* data and filtering options.

Chapter 5

Tools

Most methods developed as part of this thesis have been put into practice within software tools. In the remainder of this chapter, we discuss two such tools.

5.1 PATIKA*mad*

There are many microarray specific statistical tools that normalize and cluster the data, and provide a variety of visualization options using tables and plots. Similarly, many pathway databases and tools for creating, storing, querying and analyzing biological networks exist [7]. But, there are only a few tools that bring both worlds together. One such tool is GenMAPP [68], which provides static pathway diagrams and the ability to map color coded expression values on top of entities in the diagram. MAPPFinder is a tool for finding overrepresented Gene Ontology (GO) terms in a microarray experiment, and for searching GenMAPP pathways for the ones that have genes related with these overrepresented GO terms. However, GenMAPP lacks an integrated database, thus it is incapable of producing dynamic pathways related with experiments. Cytoscape [51] has a plugin that loads tab-delimited array data, and performs several statistical analyses. These values can be visualized on Cytoscape pathways via color coding. Reactome [59] database shows an overview map of the reactions in the database, which is laid out according to the module that the reaction belongs to. They support loading of microarray values and show them on an overview graph by color coding, so that users have an idea about the affected module. None of these tools are, however, capable of connecting microarray data with graph-theoretic queries or any other advanced graph analysis operations.

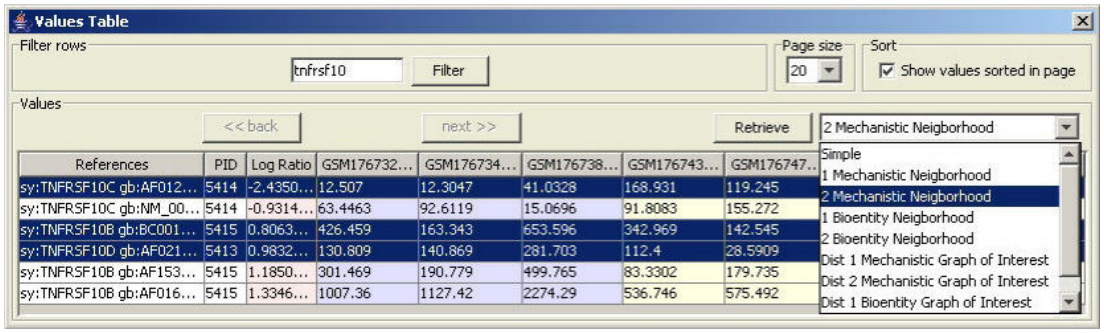
We have built a microarray data integration component, called PATIKA*mad* [3], within PATIKA*web* [28], which is a Web interface to the PATIKA database for querying, visualizing, and analyzing biological networks. Its ontology supports pathway graphs at two levels: bioentity level and mechanistic level. Bioentity-level graphs contain less detailed information, such as protein-protein interactions or transcriptional regulations between biological entities. Mechanistic-level graphs have state information (e.g. different phosphorylated states) and compartment of molecules. This level models reactions with its inputs, outputs, and effectors.

About graphs at the bioentity level or other levels of similar detail, there is a small body of literature regarding microarray data integration and co-analysis [20]. The common goal in almost all these works is to detect regions or pathways where significant microarray data is somehow "dense". This approach makes sense when the mechanism of interactions is not clear in the graph. However, in the case of mechanistic graphs, interesting paths do not necessarily have to be rich in microarray annotation. Many reactions are post-translational events and can be part of a differentially active network without any change of expression in their actors. Expression changes may be linked through paths, whose activity change is independent from expressions. In PATIKA*mad*, we supply a facility to query for paths between significant nodes (according to users' significance criteria) in an integrated pathway knowledgebase, in order to compile a "graph of interest".

PATIKA*mad* accepts tab-delimited microarray data files containing data values, and external database references. Such files are available from well known public microarray databases such as Gene Expression Omnibus, Stanford Microarray Database, and ArrayExpress. Supported external references are GenBank, Unigene, Entrez Gene, HUGO Gene Symbol, SWISSPROT, OMIM, Entrez RefSeq Protein ID, and Entrez RefSeq Transcript ID. During the processing of tab-delimited files, rows of the array are matched to the objects in the PATIKA database, and a ".pmad" (PATIKA microarray data format) file is created for later use in PATIKA*mad*. Alternatively, one may load their local model, for instance in BioPAX [25] format, containing external references. Then, microarray data with compatible external references may be loaded and mapped to this model, facilitating one to work on their proprietary data independent of PATIKA database.

After loading a set of experiments specified in a ".pmad" file, the user may set an experiment of interest, or choose to average a group of experiments, or compare log-2 ratios of two groups. These settings are managed using the Data Management dialog. This selection determines the value to be used for each row, directly affecting visualization and querying events. Expression values, calculated from current experiments of interest, are visualized on the graph through node coloring and labeling. Visualization options can be modified using the Visual Settings dialog. Besides the default red/green coloring, the user may customize coloring by assigning colors to values. Values in between are shown with colors in between. Rows of the loaded experiment may be visualized in the Values Table, which also provides an interface for querying the PATIKA database associated with the selected rows (Figure 5.1). The rows displayed may be filtered by keywords, which partially exist in external references. Selected rows may be used for retrieving related PATIKA objects from the database, or for running neighborhood or graph-of-interest queries using related nodes as seed in the database. These queries may run on either bioentity or mechanistic levels.

An experiment-scale graph-of-interest query using the Graph-of-Interest dialog is also supported. This dialog displays the user's significance criteria for the rows, length of search path, and type of graph, on which to execute the query. This query maps significant rows to significant nodes and searches paths between significant nodes. All paths not longer than the search length are included in the



tnfrsf10

20

☑ Show values sorted in page

2 Mechanistic Neigborhood

- Simple
- 1 Mechanistic Neigborhood
- 2 Mechanistic Neigborhood
- 1 Bioentity Neigborhood
- 2 Bioentity Neigborhood
- Dist 1 Mechanistic Graph of Interest
- Dist 2 Mechanistic Graph of Interest
- Dist 1 Bioentity Graph of Interest

References	PID	Log Ratio	GSM176732...	GSM176734...	GSM176738...	GSM176743...	GSM176747..
sy:TNFRSF10C gb:AF012...	5414	-2.4350...	12.507	12.3047	41.0328	168.931	119.245
sy:TNFRSF10C gb:NM_00...	5414	-0.9314...	63.4463	92.6119	15.0696	91.8083	155.272
sy:TNFRSF10B gb:BC001...	5415	0.8063...	426.459	163.343	653.596	342.969	142.545
sy:TNFRSF10D gb:AF021...	5413	0.9832...	130.809	140.869	281.703	112.4	28.5909
sy:TNFRSF10B gb:AF153...	5415	1.1850...	301.469	190.779	499.765	83.3302	179.735
sy:TNFRSF10B gb:AF016...	5415	1.3346...	1007.36	1127.42	2274.29	536.746	575.492

Figure 5.1: Part of the Values Table, where experiment rows are filtered with string "tnfrsf10" in ascending order, according to the log-ratio values. Any number of rows may be selected and used for executing neighborhood or graph-of-interest queries.

resulting graph of interest.

5.1.1 Clustering

Clustering is one of the most popular microarray data analysis methods. The aim here is to group similarly behaving genes, thus to have an idea about modules and genes whose function is not clear. PATIKA*mad* supports k-means and hierarchical clustering of the loaded experiments. Users have the option for scale normalization, standard normalization, and filtering out a certain percentage of genes that show low variance. Clustering results can be saved in a ".pcaf" (PATIKA cluster analysis file) file for later use. Clusters in loaded clustering results are visualized on pathways using compound graphs or by highlighting nodes (Figure 5.2).

5.2 ChiBE

Computational biologists have advanced pathway knowledge representation, created standards and formats [25, 44, 47, 46, 59, 23], and built more than 300 pathway and interaction databases [7] in recent years. However, current bioinformatics infrastructure is still lacking in software tools for visualizing and analyzing

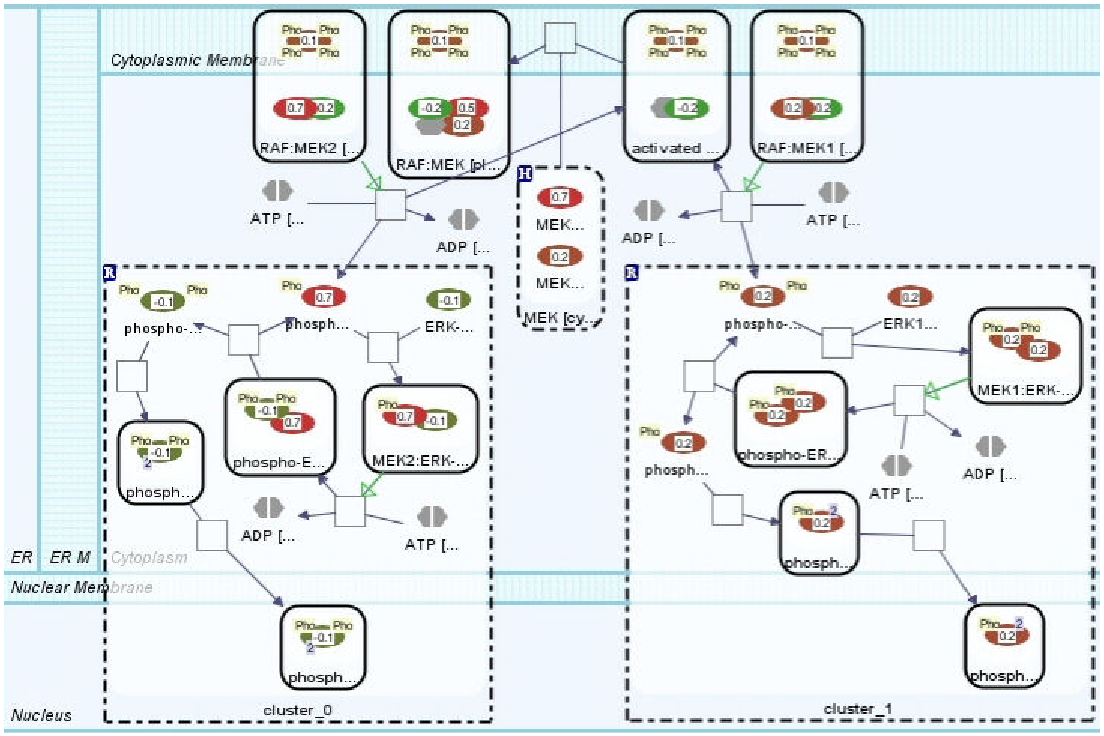


Figure 5.2: Part of a MAP Kinase pathway where two clusters are shown using compound nodes. Loaded microarray values are shown with labels and colors on nodes.

pathways. A main obstacle in this direction has been the fragmented, incomplete, and incompatible nature of pathway knowledge, making representation and integration of pathways extremely difficult.

A number of interesting pathway visualization tools [51, 36, 43, 30, 82] have been developed over the past decade, with diverse analysis focus, from analyzing gene expression profiles to effective database querying, to discovering graph-theoretic properties in biological networks. Such tools can benefit substantially from standardization of knowledge representation, pathway-specific layout algorithms, and representation of compound graphs.

5.2.1 Knowledge Representation

BioPAX has made great progress in developing a standard exchange format for biological pathway data, as a result of several years of community effort. Pathway Commons (PC) [63], based on BioPAX, was developed as an integrated single point of access to publicly available pathway information. PC covers major pathway databases and already provides integration at the level of molecular identifiers. Therefore, the community now has an emerging platform for building software tools and services without worrying about compatibility and fragmentation issues.

5.2.2 Pathway Layout

General graph layout algorithms do not address the specific needs and established conventions of pathway graphs. So far, work on pathway layout algorithms [48, 8, 72, 78, 38] has primarily focused on biochemical pathways. Thus, the need for layout of complex pathways, such as signal transduction, remains to be addressed.

Tool	Layout	Compound Support	Compartment Visualization	Experiment Data Visualization	Open Source	Plugin Support
Cytoscape	yFiles, Cytoscape and JGraph layouts	No	No	Yes	Yes	Yes
BiNoM	Uses Cytoscape layouts	No	No	N/A	Yes	N/A
VisAnt	Circular and spring embedder	Yes	No	Yes	Yes	No
PathCase	Hierarchical	No	No	No	No	No
ChiBE	CoSE (compound spring embedder) layout	Yes	Yes	Yes	Yes	No

Table 5.1: Comparison of ChiBE and 4 other tools that support BioPAX visualization. Tools are compared in the aspects of automated layout support, compound graph support, compartment visualization and experiment data visualization.

5.2.3 Compound Graphs

ChiBE [5] is an open source visualization tool, which for the first time, brings together compound graph based BioPAX visualization, seamless Pathway Commons access, pathway specific layout, and strong visualization and data analysis capabilities. Table 5.1 compares ChiBE with similar visualization tools.

Chapter 6

Conclusion

ChiBE ([5]) is a standalone pathway editor that we developed for working with BioPAX pathways. It uses Paxtools for reading, manipulating and saving BioPAX files, and for querying Pathways Commons database. ChiBE draws easy-to-understand views of BioPAX when the graph is not very large. These views are similar to SBGN process description language and are generated per pathway. Most interesting part of ChiBE is its ability to generate small size pathways out of cluttered files according to user's point of interest. We realize this by providing a local querying mechanism, enabling to search for neighborhoods, paths between molecules, or common upstream or downstream of molecules. We aim to enrich editing and querying support in ChiBE, provide support for the recently released BioPAX Level 3 ontology and format, and for the graphical notation standard SBGN.

Pathway databases collect interactions and reactions in the cell, which were discovered in different laboratories with different experimental settings. However, one often wants to restrict the network to a specific cellular context, such as a tissue with a disease. Expression profiles provide a clue about the active part of the network by showing expression levels of genes. PATIKA*mad* ([3]) integrates expression data to networks. It is not an independent software but a concept that we implemented as a component in both PATIKA*web* [28] and ChiBE. Its function includes reading the expression data and showing expression values on related proteins in the network. Alternatively, one can compare two microarray data and visualize the fold change of expression values. We defined the term "causative path", which infers the causes of dependency between expression value changes through the network. When causative paths are searched and integrated, PATIKA*mad* can produce a candidate "network of change" based on the compared profiles.

One drawback of causative path analysis is that the last step of the path must be a gene regulation, and gene regulations are poorly covered in pathway databases. However, many transcription factors are known or predicted along with their target genes and binding sites in the promoter. We developed a new method, GEM ([4]), for identification and characterization of modulators of transcription factors (TF). GEM tests if the known binding proteins of TFs has modulator activity using large number of expression data. It is based on the assumption that gene expressions are correlated with their protein activity, and it tests if the correlation between modulator and target gene expression depends on the TF expression. We select modulator candidates among interacting proteins of TFs to eliminate indirect relations and non-causative correlations. GEM also identifies the specific mode of action of the modulator on a target.

We have observed that most modulators affect multiple targets and are bimodal – they do not have a single mode of action but can act as an enhancer or attenuator based on the target. The co-activator and co-inhibitor classifications in the literature reflect a very simplified version of gene regulation as they generalize the effect of a modulator for a single gene or binding site to all targets. GEM provides a much larger scope for picking up likely targets and inferring modulator-target relationships.

Ideally, the regulation at each promoter should be modeled including all major actors at the site, considering a modulation affects the collective activity of actors instead of just a single TF. GEM can be extended to model the control at each promoter. This model would be a basis for predicting effects of upstream events to gene expressions.

Transcription factors and their modulators are potential drug targets since their activity affect expression of target genes whose activity is related to diseases such as cancer. Here, the idea is to detect the malfunctioning part of the network and repair or disable it by modifying the control at the upstream. This would be a straightforward operation if upstream events were composed of linear paths. However, signaling paths are interlinked, and TFs and modulators affect multiple target genes that function in diverse mechanisms. For instance, GEM infers that CAV1 modulates AR on at least 8 target genes, some of which have metastatic activity while some of other have tumor suppressor activity.

In drug discovery research, one big problem is to find manipulation points that will affect the targeted downstream mechanism while causing minimal side effects in healthy cells. This can be achieved by exploiting the robustness of cellular networks. For instance, one can predict that removal of a modulator will not make drastic effect while other similarly functioning modulators are abundantly present. In this direction, we can search for manipulation points whose undesired effects are mostly through robust control points. Note that robustness here depends on the specific set of genes expressed, varying in each individual and tissue type.

Bibliography

[1] R. Apweiler, M. J. Martin, C. O'Donovan, M. Magrane, Y. Alam-Faruque, R. Antunes, D. Barrell, B. Bely, M. Bingley, D. Binns, et al. The Universal Protein Resource (UniProt) in 2010. *Nucleic Acids Res.*, 38:D142–148, Jan 2010.

[2] H. Auer, D. L. Newsom, and K. Kornacker. Expression Profiling Using Affymetrix GeneChip Microarrays. *Methods Mol. Biol.*, 509:35–46, 2009.

[3] O. Babur, R. Colak, E. Demir, and U. Dogrusoz. PATIKAmad: putting microarray data into pathway context. *Proteomics*, 8:2196–2198, Jun 2008.

[4] O. Babur, E. Demir, M. Gonen, C. Sander, and U. Dogrusoz. GEM: Discovering Modulators of Gene Expression. *Submitted to Nucleic Acids Res.*

[5] O. Babur, U. Dogrusoz, E. Demir, and C. Sander. ChiBE: interactive visualization and manipulation of BioPAX pathway models. *Bioinformatics*, Dec 2009.

[6] G. D. Bader, D. Betel, and C. W. Hogue. BIND: the Biomolecular Interaction Network Database. *Nucleic Acids Res.*, 31:248–250, Jan 2003.

[7] G. D. Bader, M. P. Cary, and C. Sander. Pathguide: a pathway resource list. *Nucleic Acids Res.*, 34:D504–506, Jan 2006.

[8] M. Y. Becker and I. Rojas. A graph layout algorithm for drawing metabolic pathways. *Bioinformatics*, 17:461–467, 2001.

[9] A. H. Bild, G. Yao, J. T. Chang, Q. Wang, A. Potti, D. Chasse, M. B. Joshi, D. Harpole, J. M. Lancaster, A. Berchuck, J. A. Olson, J. R. Marks, H. K. Dressman, M. West, and J. R. Nevins. Oncogenic pathway signatures in human cancers as a guide to targeted therapies. *Nature*, 439:353–357, Jan 2006.

[10] http://www.biopax.org.

[11] E. Birney, J. A. Stamatoyannopoulos, A. Dutta, R. Guig, T. R. Gingeras, E. H. Margulies, Z. Weng, M. Snyder, E. T. Dermitzakis, R. E. Thurman, M. S. Kuehn, and *et al.* Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project. *Nature*, 447:799–816, Jun 2007.

[12] C. Bodei, A. Bracciali, and D. Chiarugi. On deducing causality in metabolic networks. *BMC Bioinformatics*, 9 Suppl 4:S8, 2008.

[13] S. Brohee and J. van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, 7:488, 2006.

[14] http://www.celldesigner.org.

[15] B. Chen, D. Wild, and R. Guha. PubChem as a source of polypharmacology. *J Chem Inf Model*, 49:2044–2055, Sep 2009.

[16] http://www.cs.bilkent.edu.tr/∼ivis/chisio.html.

[17] E. Chudin, R. Walker, A. Kosaka, S. X. Wu, D. Rabert, T. K. Chang, and D. E. Kreder. Assessment of the relationship between signal intensities and transcript concentration for Affymetrix GeneChip arrays. *Genome Biol.*, 3:RESEARCH0005, 2002.

[18] T. M. Cover and J. A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

[19] F. H. Crick. On protein synthesis. *Symp. Soc. Exp. Biol.*, 12:138–163, 1958.

[20] R. K. Curtis, M. Oresic, and A. Vidal-Puig. Pathways to the analysis of microarray data. *Trends Biotechnol.*, 23:429–435, Aug 2005.

[21] http://www.cytoscape.org/plugins2.php.

[22] P. de Matos, R. Alcntara, A. Dekker, M. Ennis, J. Hastings, K. Haug, I. Spiteri, S. Turner, and C. Steinbeck. Chemical Entities of Biological Interest: an update. *Nucleic Acids Res.*, 38:D249–254, Jan 2010.

[23] E. Demir, O. Babur, U. Dogrusoz, A. Gursoy, A. Ayaz, G. Gulesir, G. Nisanci, and R. Cetin-Atalay. An ontology for collaborative construction and analysis of cellular pathways. *Bioinformatics*, 20:349–356, Feb 2004.

[24] E. Demir, O. Babur, U. Dogrusoz, A. Gursoy, G. Nisanci, R. Cetin-Atalay, and M. Ozturk. PATIKA: an integrated visual environment for collaborative construction and analysis of cellular pathways. *Bioinformatics*, 18:996–1003, Jul 2002.

[25] E. Demir et al. BioPAX - A Community Standard for Pathway Data Sharing. *in preparation.*

[26] A. Dilek. VISIBIO*web*: A web-based visualization and layout service for biological pathways. Master's thesis, Bilkent University, August 2009.

[27] U. Dogrusoz, A. Cetintas, E. Demir, and O. Babur. Algorithms for effective querying of compound graph-based pathway databases. *BMC Bioinformatics*, 10:376, 2009.

[28] U. Dogrusoz, E. Z. Erson, E. Giral, E. Demir, O. Babur, A. Cetintas, and R. Colak. PATIKAweb: a Web interface for analyzing biological pathways through advanced querying and visualization. *Bioinformatics*, 22:374–375, Feb 2006.

[29] U. Dogrusoz, E. Giral, A. Cetintas, A. Civril, and E. Demir. A layout algorithm for undirected compound graphs. *Information Sciences*, 179(7):980 – 994, 2009.

[30] B. Elliott, M. Kirac, A. Cakmak, G. Yavas, S. Mayes, E. Cheng, Y. Wang, C. Gupta, G. Ozsoyoglu, and Z. M. Ozsoyoglu. PathCase: Pathways Database System. *Bioinformatics*, 24(21):2526–33, August 2008.

[31] J. A. Engelman, K. Zejnullahu, T. Mitsudomi, Y. Song, C. Hyland, J. O. Park, N. Lindeman, C. M. Gale, X. Zhao, J. Christensen, T. Kosaka, A. J. Holmes, A. M. Rogers, F. Cappuzzo, T. Mok, C. Lee, B. E. Johnson, L. C. Cantley, and P. A. Jnne. MET amplification leads to gefitinib resistance in lung cancer by activating ERBB3 signaling. *Science*, 316:1039–1043, May 2007.

[32] http://www.intgen.org/expo.cfm.

[33] F. Felicetti, I. Parolini, L. Bottero, K. Fecchi, M. C. Errico, C. Raggi, M. Biffoni, F. Spadaro, M. P. Lisanti, M. Sargiacomo, and A. Car. Caveolin-1 tumor-promoting role in human melanoma. *Int. J. Cancer*, 125:1514–1522, Oct 2009.

[34] J. L. Fleiss. *Statistical Methods for Rates and Proportions, 2nd Edition.* Wiley-Interscience, 1981.

[35] K. Fukuda and T. Takagi. Knowledge representation of signal transduction pathways. *Bioinformatics*, 17(9):829–837, 2001.

[36] A. Funahashi, N. Tanimura, M. Morohashi, and H. Kitano. Celldesigner: a process diagram editor for gene-regulatory and biochemical networks. *BIOSILICO*, 1:159–162, 2003.

[37] http://www.eclipse.org/gef.

[38] B. Genc and U. Dogrusoz. A layout algorithm for signaling pathways. *Information Sciences*, 176:135–149, 2006.

[39] D. Greenbaum, C. Colangelo, K. Williams, and M. Gerstein. Comparing protein abundance and mRNA expression levels on a genomic scale. *Genome Biol.*, 4:117, 2003.

[40] H. V. Heemers and D. J. Tindall. Androgen receptor (AR) coregulators: a diversity of functions converging on and regulating the AR transcriptional complex. *Endocr. Rev.*, 28:778–808, Dec 2007.

[41] H. Hermjakob, L. Montecchi-Palazzi, C. Lewington, S. Mudali, S. Kerrien, S. Orchard, M. Vingron, B. Roechert, P. Roepstorff, A. Valencia, H. Margalit, J. Armstrong, A. Bairoch, G. Cesareni, D. Sherman, and R. Apweiler. IntAct: an open source molecular interaction database. *Nucleic Acids Res.*, 32:D452–455, Jan 2004.

[42] X. Hu, J. Chen, L. Wang, and L. B. Ivashkiv. Crosstalk among Jak-STAT, Toll-like receptor, and ITAM-dependent pathways in macrophage activation. *J. Leukoc. Biol.*, 82:237–243, Aug 2007.

[43] Z. Hu, J. H. Hung, Y. Wang, Y. C. Chang, C. L. Huang, M. Huyck, and C. DeLisi. VisANT 3.5: multi-scale network visualization, analysis and inference based on the gene ontology. *Nucleic Acids Res.*, 37:W115–121, Jul 2009.

[44] 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 Novre, 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*, 19:524–531, Mar 2003.

[45] C. Jiang, Z. Xuan, F. Zhao, and M. Q. Zhang. TRED: a transcriptional regulatory element database, new entries and other development. *Nucleic Acids Res.*, 35:D137–140, Jan 2007.

[46] M. Kanehisa, S. Goto, M. Furumichi, M. Tanabe, and M. Hirakawa. KEGG for representation and analysis of molecular networks involving diseases and drugs. *Nucleic Acids Res.*, Oct 2009.

[47] P. D. Karp, C. A. Ouzounis, C. Moore-Kochlacs, L. Goldovsky, P. Kaipa, D. Ahren, S. Tsoka, N. Darzentas, V. Kunin, and N. Lopez-Bigas. Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. *Nucl. Acids Res.*, 33(19):6083–6089, 2005.

[48] P. D. Karp and S. Paley. Automated drawing of metabolic pathways. In *Third International Conference on Bioinformatics and Genome Research*, pages 225–238, Tallahassee, Florida, June 1994.

[49] P. D. Karp, S. M. Paley, M. Krummenacker, M. Latendresse, J. M. Dale, T. J. Lee, P. Kaipa, F. Gilham, A. Spaulding, L. Popescu, T. Altman, I. Paulsen, I. M. Keseler, and R. Caspi. Pathway Tools version 13.0: integrated software for pathway/genome informatics and systems biology. *Brief. Bioinformatics*, Dec 2009.

[50] 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 Res.*, 37:D767–772, Jan 2009.

[51] S. Killcoyne, G. W. Carter, J. Smith, and J. Boyle. Cytoscape: a community-based framework for network modeling. *Methods Mol. Biol.*, 563:219–239, 2009.

[52] J. L. Koh, H. Ding, M. Costanzo, A. Baryshnikova, K. Toufighi, G. D. Bader, C. L. Myers, B. J. Andrews, and C. Boone. DRYGIN: a database of quantitative genetic interaction networks in yeast. *Nucleic Acids Res.*, Oct 2009.

[53] N. Le Novre, 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. Villger, 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. *Nat. Biotechnol.*, 27:735–741, Aug 2009.

[54] R. C. Lee, R. L. Feinbaum, and V. Ambros. The C. elegans heterochronic gene lin-4 encodes small RNAs with antisense complementarity to lin-14. *Cell*, 75:843–854, Dec 1993.

[55] M. L. Lu, M. C. Schneider, Y. Zheng, X. Zhang, and J. P. Richie. Caveolin-1 interacts with androgen receptor. A positive modulator of androgen receptor mediated transactivation. *J. Biol. Chem.*, 276:13442–13451, Apr 2001.

[56] D. Maglott, J. Ostell, K. D. Pruitt, and T. Tatusova. Entrez Gene: gene-centered information at NCBI. *Nucleic Acids Res.*, 35:26–31, Jan 2007.

[57] A. A. Margolin and A. Califano. Theory and limitations of genetic network inference from microarray data. *Ann. N. Y. Acad. Sci.*, 1115:51–72, Dec 2007.

[58] A. A. Margolin, I. Nemenman, K. Basso, C. Wiggins, G. Stolovitzky, R. Dalla Favera, and A. Califano. ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, 7 Suppl 1:S7, 2006.

[59] 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 Res.*, 37:D619–622, Jan 2009.

[60] V. Matys, E. Fricke, R. Geffers, E. Gssling, M. Haubrock, R. Hehl, K. Hornischer, D. Karas, A. E. Kel, O. V. Kel-Margoulis, D. U. Kloos, S. Land, B. Lewicki-Potapov, H. Michael, R. Mnch, I. Reuter, S. Rotert, H. Saxel,

M. Scheer, S. Thiele, and E. Wingender. TRANSFAC: transcriptional regulation, from patterns to profiles. *Nucleic Acids Res.*, 31:374–378, Jan 2003.

[61] A. M. Moilanen, U. Karvonen, H. Poukka, O. A. Jnne, and J. J. Palvimo. Activation of androgen receptor function by a novel nuclear protein kinase. *Mol. Biol. Cell*, 9:2527–2543, Sep 1998.

[62] Y. M. Ning and D. M. Robins. AML3/CBFalpha1 is required for androgen-specific activation of the enhancer of the mouse sex-limited protein (Slp) gene. *J. Biol. Chem.*, 274:30624–30630, Oct 1999.

[63] http://www.pathwaycommons.org.

[64] http://www.biopax.org/paxtools.

[65] K. Raman and N. Chandra. Flux balance analysis of biological systems: applications and challenges. *Brief. Bioinformatics*, 10:435–449, Jul 2009.

[66] A. G. Recchia, A. M. Musti, M. Lanzino, M. L. Panno, E. Turano, R. Zumpano, A. Belfiore, S. And, and M. Maggiolini. A cross-talk between the androgen receptor and the epidermal growth factor receptor leads to p38MAPK-dependent activation of mTOR and cyclinD1 expression in prostate and lung cancer cells. *Int. J. Biochem. Cell Biol.*, 41:603–614, Mar 2009.

[67] G. Ruvkun. Molecular biology. Glimpses of a tiny RNA world. *Science*, 294:797–799, Oct 2001.

[68] N. Salomonis, K. Hanspers, A. C. Zambon, K. Vranizan, S. C. Lawlor, K. D. Dahlquist, S. W. Doniger, J. Stuart, B. R. Conklin, and A. R. Pico. GenMAPP 2: new features and resources for pathway analysis. *BMC Bioinformatics*, 8:217, 2007.

[69] L. Salwinski, C. S. Miller, A. J. Smith, F. K. Pettit, J. U. Bowie, and D. Eisenberg. The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res.*, 32:D449–451, Jan 2004.

[70] http://www.sbgn.org.

[71] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown. Quantitative monitoring of gene expression patterns with a complementary DNA microarray. *Science*, 270:467–470, Oct 1995.

[72] F. Schreiber. High quality visualization of biochemical pathways in BioPath. *In Silico Biology*, 2(2):59–73, 2002.

[73] E. M. Southern. Detection of specific sequences among DNA fragments separated by gel electrophoresis. *J. Mol. Biol.*, 98:503–517, Nov 1975.

[74] T. Takai-Igarashi, Y. Nadaoka, and T. Kaminuma. A database for cell signaling networks. *J. Comput. Biol.*, 5:747–754, 1998.

[75] A. H. Tong and C. Boone. Synthetic genetic array analysis in Saccharomyces cerevisiae. *Methods Mol. Biol.*, 313:171–192, 2006.

[76] K. Wang, M. J. Alvarez, B. C. Bisikirska, R. Linding, K. Basso, R. Dalla Favera, and A. Califano. Dissecting the interface between signaling and transcriptional regulation in human B cells. *Pac Symp Biocomput*, pages 264–275, 2009.

[77] Q. Wang, D. Sharma, Y. Ren, and J. D. Fondell. A coregulatory role for the TRAP-mediator complex in androgen receptor-mediated gene expression. *J. Biol. Chem.*, 277:42852–42858, Nov 2002.

[78] K. Wegner and U. Kummer. A new dynamical layout algorithm for complex biochemical reaction networks. *BMC Bioinformatics*, 6:212, 2005.

[79] C. L. Wellington, L. M. Ellerby, A. S. Hackam, R. L. Margolis, M. A. Trifiro, R. Singaraja, K. McCutcheon, G. S. Salvesen, S. S. Propp, M. Bromm, K. J. Rowland, T. Zhang, D. Rasper, S. Roy, N. Thornberry, L. Pinsky, A. Kakizuka, C. A. Ross, D. W. Nicholson, D. E. Bredesen, and M. R. Hayden. Caspase cleavage of gene products associated with triplet expansion disorders generates truncated fragments containing the polyglutamine tract. *J. Biol. Chem.*, 273:9158–9167, Apr 1998.

[80] http://en.wikipedia.org/wiki/transcription_(genetics).

[81] D. J. Wong, D. S. Nuyten, A. Regev, M. Lin, A. S. Adler, E. Segal, M. J. van de Vijver, and H. Y. Chang. Revealing targeted therapy for human cancer by gene module maps. *Cancer Res.*, 68:369–378, Jan 2008.

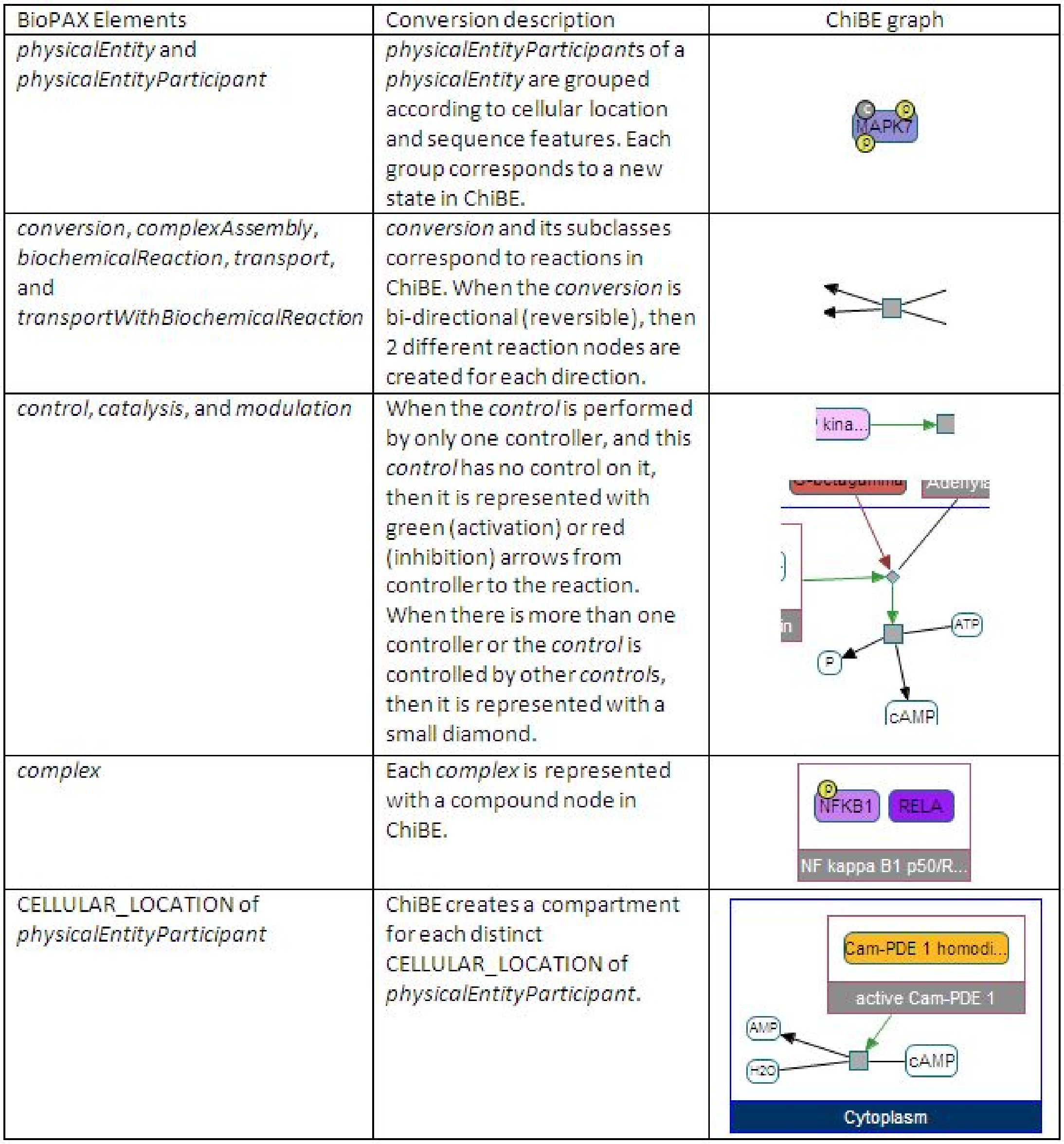
[82] A. Zinovyev, E. Viara, L. Calzone, and E. Barillot. BiNoM: a Cytoscape plugin for manipulating and analyzing biological networks. *Bioinformatics*, 24:876–877, Mar 2008.

Appendix A

Features of ChiBE

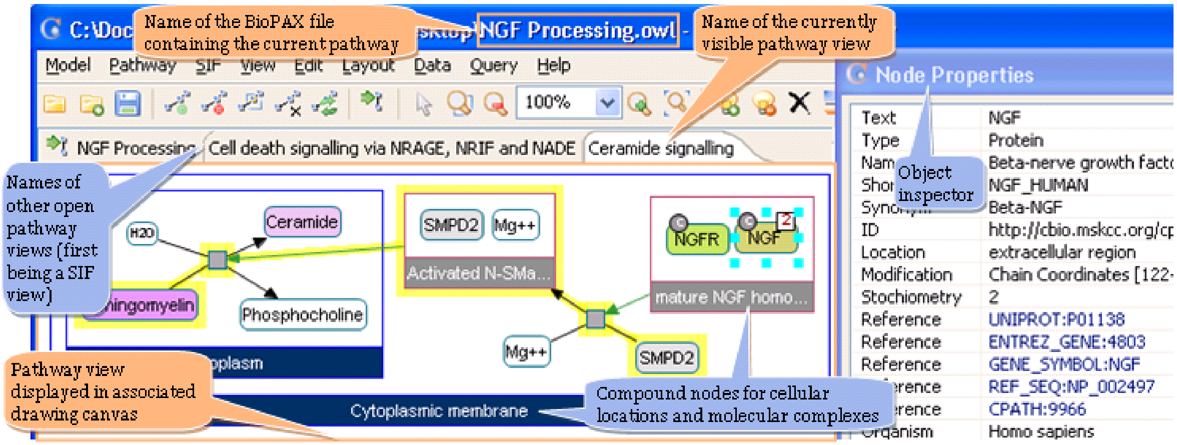
ChiBE accepts data in BioPAX Level 2 format. Section 3.2 and Table A.1 summarizes how BioPAX models are interpreted by ChiBE. We call the entire set of biological information loaded from a BioPAX file a *pathway model*. As defined by BioPAX, a *pathway* is a set or series of interactions, often forming a network, which biologists have assembled for organizational, historic, biophysical or other reasons. We use pathways to determine the boundaries of a coherent view. Each loaded pathway is displayed in a separate canvas, organized with tabs (Figure A.1). A pathway model may be expanded by merging it with another BioPAX file or PC query.

A pathway view is composed of pathway objects and their interactions. Compound nodes are exclusively used to represent *molecular complexes* and *cellular compartments* (Figure A.1). Our notation is similar to that of PATIKA [23]. ChiBE has context-sensitive pop-up menus associated with pathway objects, providing fast access to popular operations for the associated pathway object. All kinds of nodes and edges in a pathway view have distinct properties and UIs. These properties can be changed by using *inspectors* for each pathway object.



BioPAX Elements	Conversion description	ChiBE graph
physicalEntity and *physicalEntityParticipant*	*physicalEntityParticipants* of a *physicalEntity* are grouped according to cellular location and sequence features. Each group corresponds to a new state in ChiBE.	
conversion, *complexAssembly*, *biochemicalReaction*, *transport*, and *transportWithBiochemicalReaction*	*conversion* and its subclasses correspond to reactions in ChiBE. When the *conversion* is bi-directional (reversible), then 2 different reaction nodes are created for each direction.	
control, *catalysis*, and *modulation*	When the *control* is performed by only one controller, and this *control* has no control on it, then it is represented with green (activation) or red (inhibition) arrows from controller to the reaction. When there is more than one controller or the *control* is controlled by other *controls*, then it is represented with a small diamond.	
complex	Each *complex* is represented with a compound node in ChiBE.	
CELLULAR_LOCATION of *physicalEntityParticipant*	ChiBE creates a compartment for each distinct CELLULAR_LOCATION of *physicalEntityParticipant*.	

Table A.1: BioPAX elements and their corresponding visual elements in ChiBE.


C:\Doc
Name of the BioPAX file containing the current pathway
sktop\NGF Processing.owl -
Name of the currently visible pathway view
Model Pathway SIF View Edit Layout Data Query Help
100%
NGF Processing
Cell death signalling via NRAGE, NRIF and NADE
Ceramide signalling
Names of other open pathway views (first being a SIF view)
H2O
Ceramide
hingomyelin
Phosphocholine
oplasm
SMPD2
Mg++
Activated N-SMa...
NGFR
NGF
2
mature NGF homo...
Mg++
SMPD2
Pathway view displayed in associated drawing canvas
Cytoplasmic membrane
Compound nodes for cellular locations and molecular complexes
Node Properties
Object inspector

Text	NGF
Type	Protein
Nam	Beta-nerve growth facto
Shor	NGF_HUMAN
Syno	Beta-NGF
ID	http://cbio.mskcc.org/cp
Location	extracellular region
Modification	Chain Coordinates [122-
Stochiometry	2
Reference	UNIPROT:P01138
Reference	ENTREZ_GENE:4803
Reference	GENE_SYMBOL:NGF
erence	REF_SEQ:NP_002497
erence	CPATH:9966
Organism	Homo sapiens

Figure A.1: ChiBE views are organized in canvasses, each displaying one or more BioPAX pathways.

A.1 Viewing and Editing Pathways

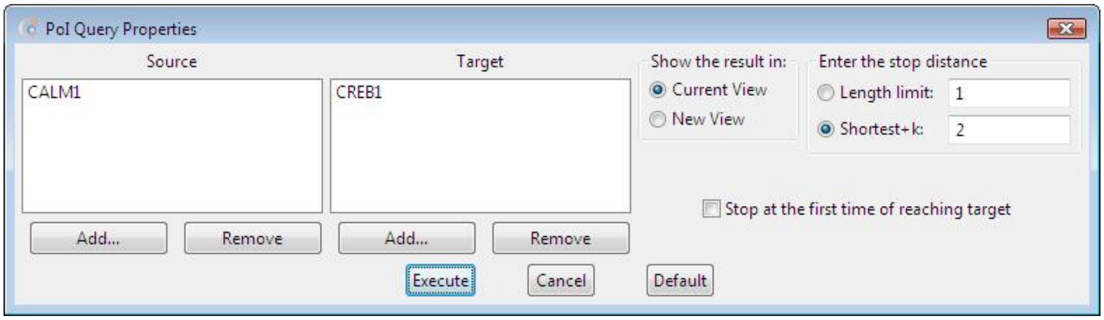
The user has various mechanisms for navigating and editing the topology as well as the geometry of pathway views. These mechanisms range from standard zoom/scroll and highlight operations to modifying the UI associated with each pathway object and to automatic layout of the pathway view.

A.2 Pathway Operations

Any subset of available pathways in a model may be displayed as a separate view, and may be saved as an image or printed. Each subset may then be modified as desired. Also, new pathway views may be created by duplicating, cropping or capturing a neighborhood of a view.

A.3 Querying Pathway Commons

PC is a convenient point of access to biological pathway information collected from public pathway databases, which one can browse or search. ChiBE provides



PoI Query Properties

Source: CALM1

Target: CREB1

Show the result in:
● Current View
○ New View

Enter the stop distance
○ Length limit: 1
● Shortest+k: 2

□ Stop at the first time of reaching target

Figure A.2: Dialog in which a paths-of-interest query is configured. User searchs paths from CALM1 to CREB1 with a length limit of shortest + 2.

a graphical user interface to search this knowledge base to find pathways that contain a specified molecule (using its UniProt or Entrez Gene ID), and present the results in a visual form. The resulting view may contain either the whole pathway or only the immediate neighborhood of the specified molecule in all the pathways in which it appears.

A.4 Querying Local Pathway

ChiBE provides a local querying mechanism which helps the user to work on large models. User can perform neighborhood, paths-of-interest (PoI), graph-of-interest (GoI) and common-stream queries that we defined in Section 3.4. Figure A.2 shows the dialog that user specify parameters of a local PoI query, which looks for paths from CALM1 to CREB1 with a distance limit of shortest path length plus two. When we run this query on the "NGF Processing" pathway from Pathway Commons database, we get the result graph in Figure A.3.

A.5 SIF Operations

SIF (Simple Interaction Format) is a format introduced by Cytoscape [51] for describing interactions in a biological network. ChiBE can reduce BioPAX pathways to SIF using a customizable set of rules to obtain a simpler view. Pathways

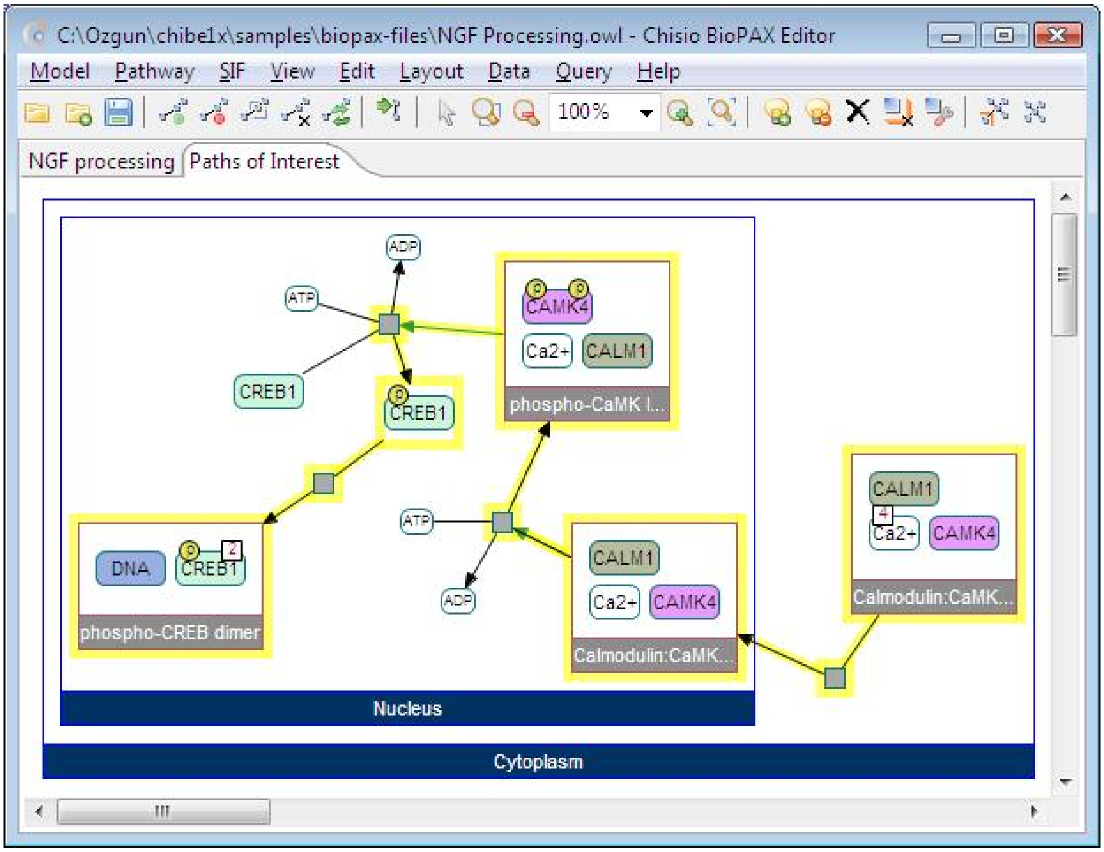


Figure A.3: Result of the paths-of-interest query in Figure A.2, performed on the "NGF Processing" pathway from Pathway Commons database.

can also be saved in SIF.

A.6 Visualizing High-Throughput Data

Multiple types of high-throughput data, such as gene expression or proteomics profiles, copy number variation, and mutation data, can be loaded into ChiBE, and overlaid onto pathway views using color coding or displayed in tables that can be searched and filtered.

A.7 Availability and Components

ChiBE is a free (EPL v1.0) Java application that runs on Windows, Mac OS, and Linux. It was built using Chisio 1.0 [16] and Eclipse GEF 3.1 [37] for graph visualization, Paxtools [64] for accessing and manipulating BioPAX files, and PATIKA*mad* [3] for high-throughput data visualization.