# Modeling Biological Networks as Logic Programs

Carito Guziolowski

DOCTORAT BRETAGNE LOIRE / MATHSTIC

Nantes Université

# HABILITATION À DIRIGER DES RECHERCHES

NANTES UNIVERSITÉ

ÉCOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies de l'Information et de la Communication*
Spécialité : *INFO*

Par
**Carito GUZIOLOWSKI**

## Modeling Biological Networks as Logic Programs

**Habilitation présentée et soutenue à Nantes, le 25/01/2024**
**Unité de recherche : Laboratoire des Sciences du Numérique de Nantes (LS2N)**

### Rapporteurs avant soutenance :

| | |
|---|---|
| Elisabeth REMY | DR CNRS, I2M, Université d'Aix-Marseille |
| Pedro MONTEIRO | Associate Professor, University of Lisboa, Portugal |
| Mohamed ELATI | Professeur des Universités, Université de Lille |

### Composition du Jury :

| | | |
|---|---|---|
| Président : | Prénom NOM | Fonction et établissement d'exercice *(à préciser après la soutenance)* |
| Examinateurs : | Jérémie BOURDON | Professeur des Universités, Nantes Université |
| | Damien EVEILLARD | Professeur des Universités, Nantes Université |
| | Marie-France SAGOT | DR INRIA, Université Claude Bernard, Lyon 1 |
| | Anne SIEGEL | DR CNRS, IRISA, Université de Rennes 1 |

# Acknowledgement

I would like to thank the people in my teaching department, in my research team, and in the lab who encourage me to finish this work. Also all the PhD students, Postdocs, and Engineers, who I had the chance to work with. Also my colleagues in science with whom we had discussions to move forward these research subjects. Finally, the members of this jury, for the time and energies put on reading this manuscript; as well as to all colleagues who offered me advice to improve the presentation of this manuscript.

I acknowledge also the time and load of work this HDR writing implied. The overall process that was very fluid and without stress, contrary to my expectations. What makes this possible I guess, is those many persons around me professionaly that encourage and respect balance in our work, which is vital for me. In particular I visualize here the faces of those colleagues and friends for whom family, children, and human relationships in general, are as mysterious, amazing and passionate as for me.

In case reading this document comes to their curiosity one day, I would like to acknowledge life for giving me the gift to share precious moments with my small and sometimes isolated (from remote family members) family. To this beautiful children of life, who taught and continuously teach me balance since they came to my life. The fact of me writing this manuscript was not in their priorities, all the contrary; but recently it comes to my mind that I continue on this challenging research path for something, and even if I'm not completely certain, they appear as one serene reason. Lastly, this document would just not have happened, without the calm company of my partner of life; whose quest of meaning, questioning, and evolution, are a gift and privilege to observe.

# Table of Contents

**Résumé en français** **1**

**1 Introduction** **7**

**2 Coloring genes and proteins regulation** **9**
2.1 Introduction . . . 9
2.2 Biological problem . . . 10
2.2.1 Genes and proteins regulation . . . 10
2.2.2 Experimental technologies measuring gene expression . . . 11
2.2.3 Regulatory networks . . . 11
2.3 Formalization of gene regulation: the sign consistency modeling . . . 14
2.3.1 Motivation: one modeling framework upon many others . . . 16
2.3.2 The core of sign consistency . . . 17
2.3.3 Sign consistency as a (brief) logic program . . . 24
2.4 Applications . . . 29
2.5 Limitations and further modeling . . . 29

**3 Applying the sign consistency model to Multiple Myeloma** **32**
3.1 Introduction . . . 32
3.2 Biological context of this study . . . 33
3.3 Integrative Approaches . . . 33
3.4 Biological knowledge . . . 35
3.4.1 Graph generation . . . 35
3.4.2 Experimental data and discretization . . . 36
3.5 Results . . . 37
3.5.1 Sign consistency modeling: interaction graph and partial labelings . 37
3.5.2 Key nodes identification in Multiple Myeloma patients . . . 37
3.5.3 Nodes perturbation . . . 40
3.5.4 JUN/FOS activity as specific marker . . . 42
3.5.5 FOXM1 activity as survival marker . . . 43

3.5.6 Improvement of the current prognostic model in MM using Iggy's predictions . . . 44
3.6 Discussion . . . 45
3.6.1 Multiple Myloma sign consistency modeling . . . 45
3.7 Conclusions and perspectives . . . 47

**4 Logic-programs' application in personalised medicine 50**
4.1 Introduction . . . 50
4.2 Regulatory and Signaling networks as Logical programs . . . 51
4.3 Multiple Myeloma . . . 51
4.3.1 Motivation and background . . . 51
4.3.2 Methods . . . 53
4.3.3 Case studies . . . 56
4.4 Acute Myeloide Leukemia . . . 59
4.4.1 Motivation and background . . . 59
4.4.2 Methods . . . 59
4.4.3 Case study - Acute Myeloid Leukemia . . . 61
4.5 Conclusions . . . 63

**5 Learning Boolean Networks 64**
5.1 Introduction . . . 64
5.2 Motivation and background . . . 65
5.3 Methods . . . 66
5.3.1 Learning Boolean logic models . . . 66
5.3.2 Learning Boolean logic models with ASP . . . 68
5.3.3 Software: caspo . . . 69
5.4 Results . . . 70
5.4.1 Family of Optimal Models . . . 70
5.4.2 Sub-optimal Models: Enumeration and Structure . . . 71
5.4.3 Input-output behavior . . . 73
5.5 Conclusion . . . 74

**6 Learning Dynamical Boolean Networks 78**
6.1 Introduction . . . 78
6.2 Motivation and background . . . 78

6.3 Materials and Methods . . . 80
6.3.1 Data acquisition . . . 80
6.3.2 *Caspo-ts* modeling framework . . . 81
6.3.3 Graph similarity measure . . . 86
6.4 Results . . . 87
6.4.1 Prior knowledge network . . . 87
6.4.2 Cell line specific Boolean networks . . . 87
6.4.3 Evaluation . . . 89
6.5 Conclusion . . . 95
6.6 Perspective . . . 96

**7 Conclusion** **98**
7.1 Short summary of contributions . . . 98
7.1.1 Biological insights . . . 98
7.1.2 Bioinformatics - automatic recovery of interaction networks . . . 99
7.1.3 Modeling Biology . . . 99
7.2 Perspectives . . . 100
7.2.1 Modeling Human embryonic development . . . 101
7.2.2 Modeling biology with SAT and ASP hybrid solvers . . . 102
7.2.3 Modeling gene and metabolic networks' integration . . . 102
7.3 Research Project . . . 103
7.3.1 Context, positioning and objectives . . . 103
7.3.2 Partnership . . . 107

**8 CV** **117**

**A Appendix** **125**
A.1 Sign consistency . . . 126
A.2 Multiple Myeloma sign-consistency modelling . . . 140
A.3 Learning Boolean Networks . . . 153
A.4 Learning Dynamical Boolean Networks . . . 161

**Bibliography** **185**

# Résumé en français

Ce mémoire a l'intention de résumer et d'assembler plusieurs des découvertes de la recherche et des incompréhensions dans lesquelles j'ai travaillé avec bonheur pendant les 15 dernières années de ma carrière scientifique. Cet ouvrage contient une partie de l'histoire de mes recherches, réalisées en collaboration avec de nombreux collègues, dans la conception d'un monde virtuel de population de cellules, nommé dans notre communauté *modèle informatique*. Son objectif ambitieux est de prédire l'état futur des cellules et de comprendre le comportement cellulaire.

Un modèle informatique ne reflète pas nécessairement la réalité (biologique). Sa conception repose sur des hypothèses parfois fortes, et les résultats qu'elle produit doivent être considérés selon les hypothèses de départ. Il peut cependant nous permettre d'avoir un aperçu de la complexité des organismes vivants. Je recommande vivement de s'en éloigner, de ne pas s'identifier à tout cadre de modélisation en particulier, et le questionner autant que l'énergie et le temps peut le permettre. Il est sage et respectueux d'admettre que la nature est merveilleuse et que les ordinateurs seront toujours limités par rapport à elle.

Cela dit, l'exercice de modélisation d'un système biologique est passionnant, et ses résultats peuvent être utiles pour intégrer des informations et des mesures massives (produites grâce à des nouvelles technologies), pour guider humblement la compréhension biologique et peut-être proposer une vue complémentaire à la décision médicale. Le processus de modélisation d'un système biologique est précieux quand aucun intérêt financier n'est derrière. Ce fut un privilège de travailler en science avec un tel environnement libre. Concevoir, intégrer des données, exécuter et vérifier les calculs informatiques concernant les prédictions d'un modèle, prend du temps et le temps nécessaire pour y parvenir dépendra de la personne qui modélise, du cadre choisi de modélisation, de la réalité biologique, et de l'affinité existante entre eux trois.

J'ai sélectionné cinq principaux groupes de recherches que j'ai menées entre 2013 et 2019 en tant que postdoctorante et maîtresse de conférences. J'en ai ignoré d'autres, qui ont aussi été le terrain de collaborations étonnantes en raison des contraintes de temps. Je n'ai pas tenu compte de traiter en détail les sujets de recherche que j'encadre actuellement. Cependant, j'aurai pour ces travaux des mots particuliers dans le dernier chapitre. Le

Chapitre 2 commence par la vision du sujet de la *consistance des signes*, qui m'accompagne depuis ma thèse, dans 2010, et qui se poursuit en 2022. La méthode est décrite au Chapitre 2 et son application dans le complexe domaine du Myélome Multiple est présentée au Chapitre 3. Dans le Chapitre 4 nous abordons le sujet de la médecine personnalisée, et comment nos méthodes proposent des solutions dans ce contexte. Enfin, les Chapitres 5 et 6 donnent un aperçu du processus d'apprentissage des réseaux Booléens à partir d'un type particulier de données, la phospho-protéomique, qui permet de perturber et mesurer en cas de perturbation les espèces du système biologique de manière à ce que les modèles booléens peuvent être appris efficacement.

## Chapitre 2 : coloration de la régulation des gènes et de protéines

Le processus de régulation des gènes et des protéines au sein d'une cellule peut être abordé de différentes manières. Dans notre approche, nous faisons l'abstraction que nous pouvons utiliser un objet mathématique, un graphe, pour représenter les interactions entre ces composantes au sein d'une cellule. Les interactions des gènes et protéines, au sein d'une cellule, peuvent avoir un temps spécifique, une force spécifique, peuvent survenir ou non selon des scénarios spécifiques. Ils peuvent être coordonnés, coopératives ou mutuellement exclusives. Ils peuvent être connus ou pas encore découverts. Cette complexité sera représentée à l'aide du cadre de modélisation de la consistance des signes, qui a été l'un de mes premiers efforts pour modéliser un système vivant. L'apparente simplicité de cette approche peut cacher le problème combinatoire complexe sous-jacente. En utilisant ce cadre de modélisation, nous avons modélisé certains mécanismes de régulation dans les maladies humaines, tels que le Myelome Multiple.

## Chapitre 3: application de la consistance des signes à la modélisation du Myelome Multiple

Nous présentons dans ce chapitre une partie de l'étude de recherche que nous avons publiée en 2017 [47]. Les lecteurs qui souhaitent approfondir la compréhension des méthodes et des résultats de ce travail sont référés à l'annexe A.2. Ce travail est le fruit d'une collaboration avec Stéphane Minvielle et Florence Magrangeas, du CRCINA (Centre de

Recherche en Cancérologie et Inmunologie Nantes Angers). Cette collaboration a été financée par GRIOTE (Groupement de Recherche en Intégration de données Omics à Très grande Echelle) un projet rassemblant la recherche bioinformatique dans la région Pays de la Loire. Le premier auteur du travail était Bertrand Miannay, qui a fait une thèse de doctorat sous ma supervision. Nos collègues biologistes nous ont fourni un ensemble de données non publiés. Il s'agissait des données de puces à ARNm de patients qui ont développé un cancer du sang nommé Myelome Multiple. Ils étaient intéressés à découvrir les réseaux de signalisation ou de régulation des gènes sous-jacents à ces données expérimentales, avec l'optique de personnaliser le traitement de ce cancer. Nous commençons cette partie en indiquant le contexte du myélome multiple (Section 3.2). Après, nous présentons une revue des approches dites intégratives (Section 3.3), qui sont des méthodes utilisées classiquement pour relier un profil d'expression génique à un réseau ou une voie biologique. Notre objectif ici est de clarifier comment l'approche de modélisation de la consistance des signes peut être relié, et à quel niveau, à ce processus d'intégration. Plus tard, dans la section 3.4, nous expliquerons les données biologiques que nous avons utilisées dans cette recherche. Dans la section 3.5, nous montrerons les résultats que nous avons obtenus.

## Chapitre 4: les programmes logiques appliqués dans la médecine personnalisée

Ce chapitre rappelle les travaux de recherche, basés sur des programmes logiques inspirés par de problèmes sur des systèmes biologiques, qui nous ont permis de distinguer différents profils de patients. A différence des méthodes classiques de classification des données omiques des patients, nos méthodes incluent des réseaux biologiques. La collaboration avec plusieurs personnes a contribué aux méthodes et aux résultats présentés ici. Je peux citer : Misbah Razzaq, Lokmane Chebouba, Pierre Le Jeune, Bertrand Miannay, Dalila Boughaci et Jérémie Bourdon. Je suis très reconnaissante pour l'énergie et l'intérêt mis pour explorer ce domaine de recherche, qui dans la plupart de nos études n'a pas été l'objectif principal du projet de recherche, mais des pistes que nous avons fini par explorer guidées par l'intérêt aux données fournis par la communauté *DREAM challenges*. Ces défis de l'ingénierie inverse consistent à proposer des données *omiques* à la communauté méthodologique. Ces données ont une haute qualité et sont générés pour aider dans la recherche sur le cancer. Nous avons soigneusement étudié trois ensembles

de données de ces défis relatifs au: (1) myélome multiple, (2) leucémie myéloïde aiguë et (3) cancer du sein. Les deux premiers jeux de données seront présentés dans ce chapitre respectivement dans les sections 4.3 et 4.4. Le troisième sera présenté au chapitre 6. Pour chacun d'eux nous avons proposé des méthodes basées sur des programmes logiques. Un article de recherche présentant ces trois travaux ensembles a été publié dans [101]. De nombreux paragraphes écrits ici sont tirés de cette publication.

## Chapitre 5: apprendre des réseaux booléens

Ce chapitre résume une méthode que nous avons proposé pour apprendre des réseaux booléens. La méthode fonctionne à partir d'expériences de phosphoprotéomique, où l'expression d'un ensemble des protéines est observée lors de multiples perturbations du système. Ce travail [125] nous a conduit à proposer l'outil *caspo*. Le système *caspo*, conçu en Answer Set Programming, a été inspiré par CellNOpt [126], où la recherche du modèle optimal a été conçue à l'aide d'algorithmes génétiques. *caspo* a inspiré d'autres travaux que nous avons menés plus tard, comme [33], [127], ou [121] (voir chapitre 4). Nous en avons présenté une version plus récente dans [128] intégrant de multiples fonctionnalités. *caspo* continue d'inspirer des travaux similaires tels que [38, 17, 129]. Les applications de ce système se présentent de manière naturelle lors de l'élucidation des mécanismes biologiques cachés dans une masse de données à l'échelle du génome. C'est le cas avec le sujet de thèse de Mathieu Bolteau, doctorant que je co-encadre actuellement. Son sujet de recherche est dans le domaine du développement de l'embryon humain. Ce chapitre est organisé en suivant principalement les résultats que nous avons présentés dans [46] (voir Annexe A.3). La recherche conduite avec *caspo* a été possible grâce aux efforts de tous les co-auteurs de nos publications. Je citerai, spécialement, en raison de l'impact de leur contribution : Santiago Videla, Anne Siegel, Julio Saez-Rodriguez et Irina Konokotina.

## Chapitre 6: apprendre des réseaux booléens dynamiques

Dans ce dernier chapitre, il est présenté le problème d'inférer des réseaux booléens à partir de données d'expression de séries temporelles sous perturbation. Ce travail a nécessité l'interaction de plusieurs anciens et nouveaux collaborateurs dans le cadre d'une thèse de doctorat qui a donné de bons résultats, malgré le fait qui était un défi au début. Dans ce projet de thèse j'ai co-encadré les travaux de Misbah Razzaq. L'étude présentée

dans ce chapitre a été publiée dans [150] (voir Annexe A.4), les résultats présentés ici sont extraits de cette publication. L'histoire complète de cette recherche n'a pas commencé ou terminé avec cet article, des travaux de recherche précédents (publiés dans [151, 152]), et futurs (publiées dans [150]) ont été faits et doivent être considérées pour comprendre l'ensemble de cette contribution. L'apport de toutes les personnes co-autrices de ces publications a été important pour que cette recherche apparaisse. Je mentionne spécialement deux collègues, Max Ostrowski et Loïc Paulevé, qui ont apporté un appui essentiel à ce travail.

# Conclusion

Ce manuscrit a présenté une partie de mes recherches qui consistaient à concevoir des modèles informatiques à partir de ou pour des systèmes biologiques. Les méthodes que nous avons proposées sont principalement basées sur la programmation logique à l'aide du paradigme de *Answer Set Programming* (programmation par ensemble de réponses). Ils interagissent pourtant avec des méthodes d'apprentissage automatique, des analyses statistiques et une expertise biologique, afin de offrir des prédictions computationnelles significatives répondant à des problèmes biologiques réels. Ce chapitre résume mes résultats de recherche et propose des perspectives simples.

**Des contributions à la biologie** Les questions biologiques que nous avons abordées et auxquelles nous avons répondu avec les méthodes examinées dans ce manuscrit sont :

- La compréhension des mécanismes de régulation, en termes de gènes ou de protéines, concernant des interactions présentes dans le cancer. Principalement, du Myélome Multiple [47] (voir chapitre 3) et Carcinome hépatocellulaire [48]. Ces recherches ont été menées avec des biologistes, experts dans le domaine spécifique du cancer étudié. Les résultats de notre méthode étaient conformes avec la littérature biologique ou complémentaires aux techniques classiques pour trouver des marqueurs.
- L'apprentissage de modèles informatiques à partir d'ensembles de données en protéomique, qui peuvent prédire efficacement les ensembles de données de test. Ces données ont été fournies par les défis DREAM (Dialogue on Revearse Engineering Assessment and Methods). Nous avons étudié, en particulier, la spécificité des patients ou des lignées cellulaires appartenant à différentes classes. Ce type de

recherche a été menée dans l'étude de la Leucémie Aigüe Myéloïde [121] (voir chapitre 4) et du cancer du sein [150] (voir chapitre 6). Nos résultats ont été comparés en détail avec d'autres méthodes analysant les mêmes données. Ils ont produit, sur le contexte des publications citées, des résultats concrets sur des mécanismes non découverts auparavant et de meilleurs taux de prédiction.

**Bioinformatique - récupération automatique des réseaux d'interaction** Concernant le domaine de la bioinformatique (plus spécifiquement les méthodes qui génèrent des réseaux d'interactions à partir de bases de données), même s'il n'est pas abordé dans ce manuscrit, il convient de mentionner le travail que nous avons entrepris dans [11], qui exploite les connaissances à l'intérieur des bases de données d'interactions de gènes et de protéines afin de construire des graphes d'interactions causales automatiquement. Ce travail était basé sur le Web sémantique, et a produit des résultats très intéressants sur les graphes générés, par rapport aux méthodes de l'état de l'art. Il est important à mentionner que le réseau d'interactions, constitue ce que nous appelons *Prior Knowledge Networks*, et qui est un élément de départ essentiel à toutes les méthodes proposées dans ce manuscrit.

**Modelisation de la biologie** Enfin, ce qui a surtout occupé l'espace dans ma carrière de chercheuse jusqu'à présent a été la modélisation de la biologie. Écrire un programme (logique) qui calcule des hypothèses biologiques à capturé mon attention depuis déjà 15 ans. Les modèles que nous avons proposés sont en général très intuitifs, puisqu'ils testent des règles définissant le lien entre une espèce biologique avec ses voisins. Cependant, comme les graphes reliant les espèces biologiques sont à grande échelle (composés dans certains cas de milliers d'espèces) et incomplètes, leur analyse doive être soutenu par une programmation efficace. Aussi, compte tenu de la grande masse de données expérimentales, les approches automatiques offrent ici une excellente aide vers l'intégration et compréhension des données expérimentales. La plupart de mes recherches reposent sur deux structures de modélisation : la consistance de signes et l'apprentissage des réseaux booléens.

Chapter 1

# Introduction

*"Faire des plans d'avenir :*
*c'est aller à la pêche où il n'y a pas d'eau.*
*Rien ne se passe jamais comme tu l'as voulu ou craint.*
*Laisse donc tout cela derrière toi."*
— Christiane Singer

This memoir intends to summarize and mix together many of the research discoveries and misunderstandings in which I have happily worked for the last 15 years of my scientific path. This work contains a part of the story of my research, done in collaboration with many colleagues, in the conception of a virtual world of cells population, named in our community *computational model*. Its ambitious purpose is to predict the future cell state and to understand cellular behavior.

A computational model is not necessarily reflecting (biological) reality. Its conception is based on sometimes strong hypotheses, and the results it produces need to be considered according to the original hypotheses. It can allow us, however, to have a glance of the complexity of living organisms. I highly recommend to take a distance from it, not to identify to any modeling framework in particular, and question it as much as energy and time can allow. It is wise and respectful to admit that Nature is marvellous and computers will always be limited compared to it.

Having said that, the exercise of modeling a biological system is exiting, and its results can be useful to integrate massive information and measurements (produced with recent technologies), to humbly guide biological understanding and maybe propose a complementary view to medical decision. The process to model a biological system is precious when no financial interest is behind. It has been a privilege to work in science with such a free environment. Conceiving, integrating data, running, and verifying the computational predictions of a model, is time consuming and the necessary time to fulfill this will depend on the modeller, on the modeling-framework, on the biological reality, and on the affinity

between the three of them.

I have selected five main groups of research studies I conducted between 2013 and 2019 as a post-doc and as a *maître de conférences*. I have ignored others, which also have been the ground of amazing collaborations because of time restrictions. Specially I am ignoring to treat in detail the research subjects which I am currently supervising, but for whom I will have special words on the last chapter. Chapter 2 starts with the vision of the subject of *sign consistency*, which has accompanied me since my PhD thesis, in 2010, and continues in 2022. The method is described in Chapter 2 and a challenging application of it in the domain of Multiple Myeloma is presented in Chapter 3. Chapter 4 handles the subject of personalized medicine, and how our methods proposed solutions on this context. Finally, Chapters 5 and 6 give insights on the process of learning Boolean networks from a particular type of data, *phospho-proteomics*, which allows to perturb and measure upon perturbation the biological system species in a way so that Boolean models can be effectively learned.

Chapter 2

# Coloring genes and proteins regulation

*"There is a state, 'where' problems are no longer settled in any particular way. In the course of your life you have after careful consideration come to a decision on many questions, have you not? But now you will have to realize that no solution is ever conclusive; in other words, you will have to go beyond the level where there is certainty and uncertainty."*

— Sri Ma Anandamayi

## 2.1 Introduction

The process of gene and protein regulation within a cell can be approached in different ways. In our approach we make the abstraction that we can use a mathematical object, a graph, to represent the interactions between these compounds within a cell. Genes and proteins interactions within a cell may have an specific time, an specific strength, may occur or not according to specific *scenari*. They can be coordinated, cooperative, or mutually exclusive. They may be known, or not yet discovered (Section 2.2). This complexity will be constrained using the sign-consistency modeling framework (Section 2.3), which was one of my first efforts to model a living system. The appearing simplicity of this approach may hide the complex combinatorial problem behind it. Using this framework, we have approached some regulatory mechanisms in human diseases, such as Multiple Myeloma (Chapter 3).

## 2.2 Biological problem

### 2.2.1 Genes and proteins regulation

Each cell contains the knowledge of how much and which type of proteins they require to function. Gene expression is the process in which the double-stranded DNA molecule is converted into a string of aminoacids called proteins. Gene expression process has multiple steps as shown in Fig. 2.1, among which transcription, when the double-stranded DNA is converted in a single stranded RNA molecule; and translation, when the RNA is converted into a protein. In some cases RNA itself may accomplish already a function. Transcription is carried out mainly by the molecule RNA-polymerase, while translation is carried out by the Ribosome protein.

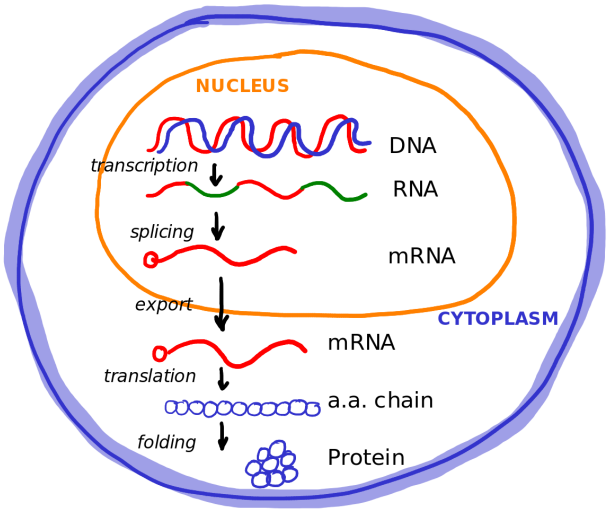



Figure 2.1: **Gene expression.** The sequence of steps needed to transform a double-stranded DNA molecule into a functional protein.

Gene products (proteins) have different functional roles and are expressed under different stresses. The decision concerning which genes should be turned *on* or switched *off* is executed by transcription factors (TFs). TFs use metabolites/signals as an input information from the environmental state and give a transcriptional response as an output [1]. Regulation of gene expression gives a cell the control over its structure and function. It is the basis for cellular differentiation, as well as for versatility and adaptability of any organism. Gene expression can be regulated at several levels, as shown in Fig. 2.1: initiation of transcription (*e.g.* by repressor or activator proteins), premature termination of

transcription, initiation of translation, and by post-translational effects. The challenge in modeling gene expression regulation is to approach how all the components of the regulatory process interact in order to perform complex biological functions; this may result on novel hypotheses that can help the understanding of a living system and support medical treatment and decision.

### 2.2.2 Experimental technologies measuring gene expression

The analysis of gene expression is an area that has evolved tremendously over the last decades. The development in 1977 of the Northern blot, to characterize the relative abundance of an RNA sequence, led to the first analyzes of gene expression by measuring the amount of messenger RNA (mRNA). This method, simple to set up and inexpensive is still widely used today [2], and has seen many improvements concomitant with the evolution of biological knowledge and the automation of analysis [3].

Three decades ago DNA chips (or microarrays) appeared [4], been able to measure the mRNA concentration of thousands of genes [5] at the same time and in a specific cellular state or condition. DNA chips were very useful in the comparison of different gene expression profiles [6, 7, 8]. This technological advance has motivated the apparition of the systems biology research field, having as a main challenge to propose methods and concepts to exploit these observations [9].

A new revolution in the analysis of gene expression took place around 2008 [10] with the first high-speed sequencers, or NGS (next-generation sequencing). NGS allowed sequencing all of the DNA fragments present in a sample, this made it possible to sequence entire genomes and transcriptomes.

In the Multiple Myeloma case study presented in Chapter 3, gene expression was measured using RNA microarrays obtained via Affymetrix technology. Later, in Chapter 4 we showed how gene expression measured by NGS was used to model and classify Multiple Myeloma patients having different disease prognostics.

### 2.2.3 Regulatory networks

**From biology to networks** One of the types of gene regulatory mechanism we address in this manuscript is the regulation of the transcription initiation. During this process a molecule RNA-polymerase attaches to the promoter region of the DNA sequence (or gene) and begins transcription along all the DNA strand. The RNA-polymerase transcription

can be regulated by TFs that are proteins or proteins-complexes. They can be activators, which enhance the interaction between RNA polymerase and a particular promoter encouraging the expression of the gene, or repressors, which bind to non-coding sequences on the DNA strand impeding the progress of RNA polymerase along the strand, thus, impeding the expression of the gene. This type of regulatory process can therefore be represented by a transcriptional regulatory network (TRN), as shown in Fig. 2.2.

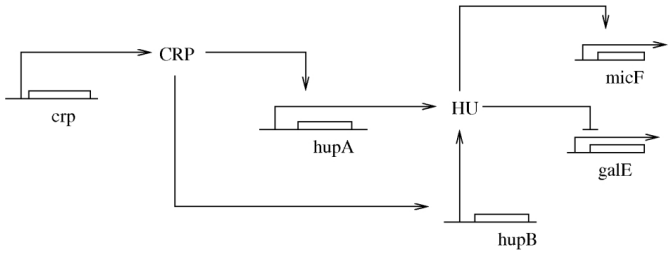



Figure 2.2: **Small transcriptional regulatory network.** Extract of the transcriptional network of genes and proteins in *Escherichia coli*. The names in capital letters correspond to TFs (proteins): HU and CRP, that can activate or repress other genes transcription. Arrows ending with "->" or "-|" imply that the initial product activates or, respectively, represses production of the product of arrival.

When including post-transcriptional regulation, biological networks, can become more complex. Processes like translocation, modification, state transition, can appear and need to be included in the modeling framework. For example, in Fig. 2.3 another type of biological network is built, which includes transcriptional and post-transcriptional events. It is nowadays possible to access the detailed information of regulatory events affecting genes or proteins within a cell by using public (or private) Pathway Databases. In a paper published in 2021 [11] we proposed a method to automatically interpret the information available in such pathways databases to construct biological causal and signed graphs.

**From networks to mathematical objects: graphs** The mathematical formalisation of gene regulation is applied on a biological regulatory network represented in the form of an *influence graph* or *interaction graph*. An influence graph is a common representation for biochemical systems where arrows show activations or inhibitions. Basically, an arrow between $A$ and $B$ means that an increase of $A$ tends to increase or decrease the production rate of $B$ depending on the shape of the arrow head. For example, in the influence graph representing the network depicted in Fig. 2.2, common sense can be used to state that an increase of HU should result in a decrease of the production rate of galE. However, if

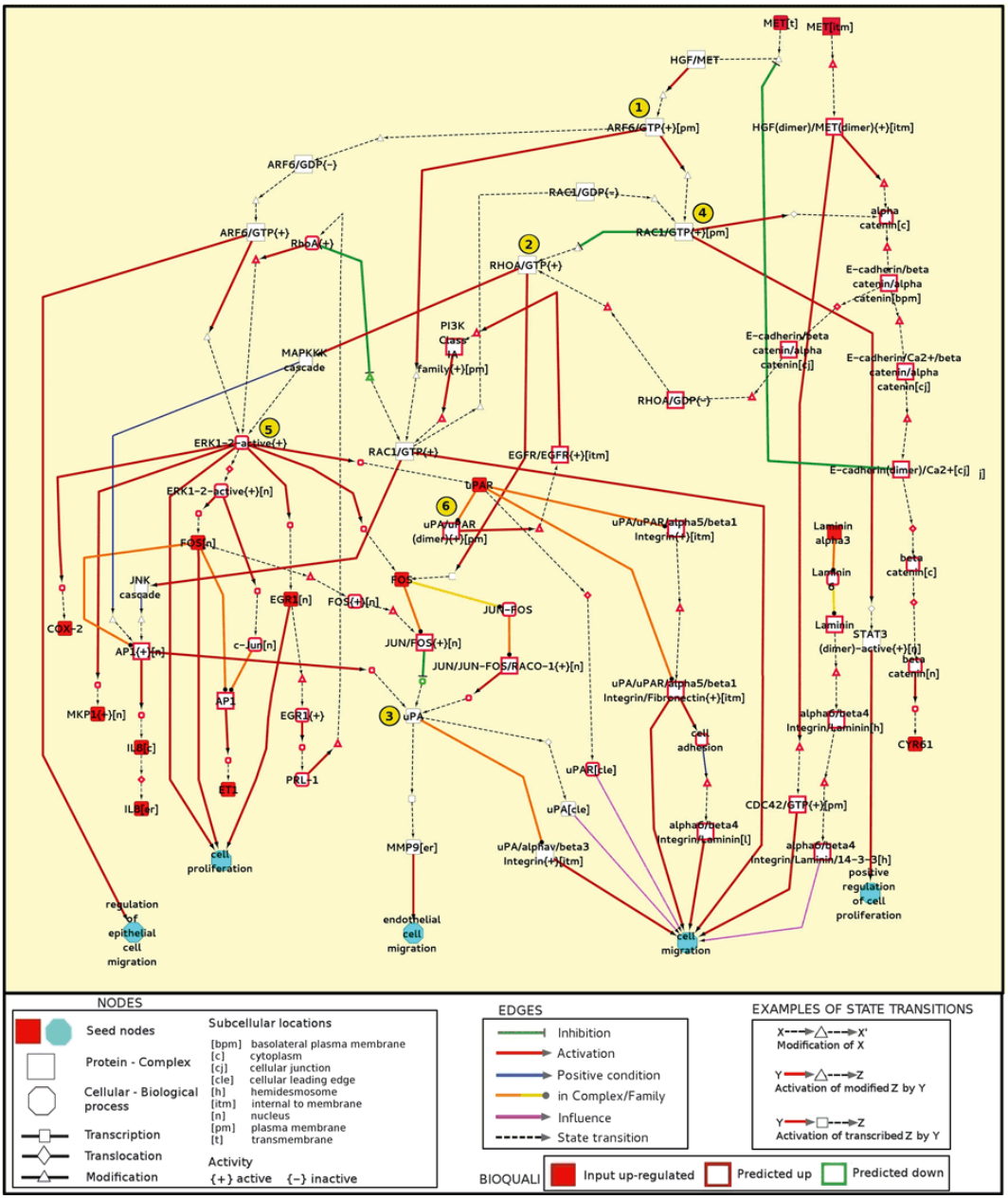



Figure 2.3: **RSTC (receptor - signaling - transcription - cellular process) network** This network was generated from the Pathway Interaction Database explaining the up-regulated genes induced after HGF (Hepatocyte Growth Factor) stimulation in Human cells. The graph legend is provided in the lower box. The seed nodes are composed of: HGF receptor nodes, the protein nodes where two-fold up-regulated genes could be overlayed, and the cell migration and proliferation nodes. These nodes were used to generate the RSTC network.

hupA increases and hupB decreases, then nothing can be said about the variation of HU.

Influence graphs can be built using a natural passage from transcriptional regulatory networks. This is because TRNs hold interactions of the form TF-gene, in which the rate of production of the gene is affected by the concentration change of the TF that transcribes it. Even so, any kind of biological network may be studied as long as their interactions can be mapped as influence, *i.e. A influences B if increasing or decreasing concentration of A affects the production rate of concentration of B.*

The edges of an influence graph must be labeled with discrete signs as '+', '-', and '?', where '+' represents a positive influence (*e.g.* activation of gene transcription, recognition of a gene promoter region, or formation of proteins complexes), '-' a negative influence (*e.g.* inhibition of gene transcription, inactivation of a protein), and '?' a dual or complex regulation. The transformation from biological networks pictures or representations towards influence graphs could be thought as trivial or direct (see Fig. 2.4). As shown in [11], this is currently not the case.

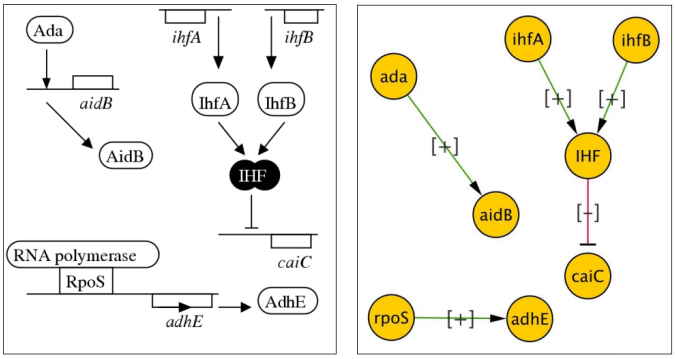



Figure 2.4: **Biological network represented as an influence graph** A regulatory network (A) mapped into an influence graph (B). Interactions among molecules create an influence graph. The arrows in the influence graph represent a positive (+) or negative (-) influence.

## 2.3 Formalization of gene regulation: the sign consistency modeling

The objective of formalizing gene regulation is to integrate in a formal mathematical framewok, and later on a computational system, the main concepts explained in Section

2.2: gene expression regulation and gene expression measurement, so that they can *speak by themselves*. These words may be understood in logic as *verification of satisfiability*. In more intuitive terms, inspired by an Artificial Intelligence paradigm, the idea will be to generate new knowledge from a given (known) information and given a set of (logical) rules of deduction, limiting what is accepted or not. We will refer in Section 2.3.2 to this constrained deduction as *consistency*.

Gene expression regulation will be formalized using a mathematical object, known as an interaction graph; while gene expression measurements, as discrete colors in the previous graph. Back ago during my PhD thesis in 2006 at the *Université de Rennes 1* in the research team *Symbiose* of the INRIA research laboratory, I joined a group of persons that set up the basis of this mathematical formalization [12]. If my memory is correct I can mention here my PhD supervisor, Anne Siegel, and precious colleagues and scientific mentors Ovidiu Radulescu, Michel Le Borgne, and Philippe Veber. The first computational modeling framework developed was based on Ternary Decision Diagrams, the tool name was Sigali; it switched then to Bioquali [13], also based on decision diagrams, but providing an interactive user friendly visualisation tool via Cytoscape; and after a long collaboration with the University of Potsdam colleagues (Torsten Schaub, Sven Thiele, Martin Gebser) the computational framework evolved into answer set programs [14, 15]; finally with a collaboration with the Max Plank Institute for Dynamics of Complex Technical Systems (Steffen Klamt), our computational tool became stable with the name of Iggy in 2015. This is nowadays a stable framework to model gene expression regulation, well documented and maintened at `https://bioasp.github.io/iggy/`. It has been the basis to many applications I have conducted; and it continues to be in movement (see for example an application to experimental design in [16]), and a source of inspiration to different type of consistency modelings frameworks of gene regulation such as [17].

Telling this long research story is not an easy exercice, the choice here is to refer to the reader that searches for details to the Appendix A.1, to one of, up to now, the most complete paper we published about the recent version of Iggy. In this section, I will repeat some of the key notions and concepts that were formulated on that work. These notions will be important to understand how Iggy has been applied recently to model Multiple Myeloma and Hepatocellular carcinoma. These application part will be presented in Section 2.4.

### 2.3.1 Motivation: one modeling framework upon many others

As mentionned in Section 2.2, the advancements of measurement technologies and high-throughput methods in molecular biology have led to a tremendous increase in the availability of factual biological knowledge as well as of data capturing the response of biological systems to experimental conditions. Knowledge about metabolic, signaling, and gene regulatory interactions and networks has been available in databases such as KEGG, RegulonDB, PID, or Reactome which can be used as a starting point to build causal models of bio-molecular networks [18]. We deepen this research area, from databases to causal models, in a work published in 2021 [11]. Specifically, signaling and gene regulatory networks carrying signal and information flows can be represented as interaction or influence graphs [13, 15, 19, 20, 21], Bayesian networks [22], some form of logic (including Boolean or constrained fuzzy logic) modeling [19, 23, 24], or ordinary differential equations [25, 26, 27]. However, there is an increasing need to relate large-scale network models to high-throughput data in order to (in)validate network topologies or to decide which regulatory or signaling interactions are present in a particular biological system, cell type, or environmental condition.

Significant work has been published on this subject, attempting to detect inconsistencies among measured high-throughput data and signaling and regulatory networks and to subsequently identify missing or inactive interactions such that the optimized network structure maximizes consistency with experimental data [13, 19, 28, 29, 30, 31, 32, 33]. Some of these approaches use signed directed graphs, also called interaction or influence graphs (IG), as underlying model where edges indicate either positive or negative effect of one node upon another. Although these models are qualitative and simple, they have frequently been used to study signal flows in a wide range of biological systems. Moreover, the fact that every Boolean and every ODE model has an underlying interaction graph renders their analysis directly relevant for other modeling formalisms and it has been shown that some important global properties of Boolean or ODE models are determined by the structure of their associated IG [21, 34, 35]. IGs have also been used for qualitative reasoning, to describe physical systems where a detailed quantitative description is unavailable [36]. In fact, this has been one motivation for using IG in the context of biological systems [35] where knowledge and data are usually uncertain.

One important class of methods relating IG with experimental data is based on the notion of sign consistency. The key idea here is to represent the potential network behaviors resulting from steady-state shift experiments (such as upregulation or downregulation of

node activation levels after network perturbations) by certain kinds of discrete constraints. A first computational approach based on sign consistency was introduced in [13]. There, experimentally measured changes in node activities were represented by two labels (increase, decrease) on the IG nodes. Constraints relating nodes labels and IG are introduced to model the propagation of regulatory effects. Later, in [14, 15], Answer Set Programming (ASP) [37] was used to find admissible node labelings adhering to the posed constraints, and optimal repairs to restore sign-consistency were proposed. A related formalism was presented in [32], which more recently has been adapted by using sign-consistency to propose experimental designs [38].

The objective of this section was to introduce the history of sign-consistency approaches and to briefly situate them related to other biological network modeling approaches. We continue, in next section, with the core of the mathematical formalisation of this framework.

### 2.3.2 The core of sign consistency

#### Definitions

An **influence or interaction graph (IG)** is a signed directed graph $(V, E, \sigma)$, where $V$ is a set of nodes, $E$ a set of edges, and $\sigma : E \to \{+, -\}$ a labeling of the edges. Every node in $V$ represents a species in the modeled system and an edge $j \to i$ means that the change of $j$ in time influences the level of $i$. Every edge $j \to i$ of an IG can be labeled with a sign, either '+' or '-', denoted by $\sigma(j, i)$, where '+' ('-') indicates that $j$ tends to increase (decrease) $i$. Examples of IG are given in Figs. 2.5 and 2.2.

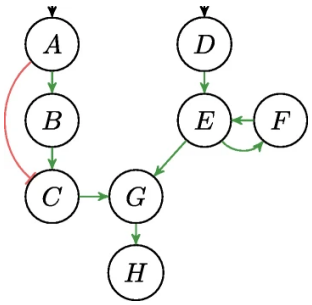



Figure 2.5: **Influence or interaction graph** IG with a positive feedback loop between E and F.

In this modeling framework, we confront the IG with **experimental profiles**. Experimental profiles are supposed to come from steady-state shift experiments where, initially,

the system is at steady-state, then externally perturbed in certain nodes, settles eventually into another steady-state. For some species $S \subseteq V$ (genes, proteins, or metabolites) concentrations are measured in the initial and final state. The raw data is given by a real value *obs(s)* for every measured species $s \in S$ specifying the difference of the node states at the beginning and in the new steady state. As defined below, we determine for these nodes whether the concentration has increased, decreased or not significantly changed. These discrete values can be represented as signs or colors.

### Data discretization

Our approach will use discretized experimental measurements. For this, we thought on using four (condition-dependent) thresholds $t_1 \leq t_2 < 0 < t_3 \leq t_4$, allowing one to consider uncertainties in the discretization process. As illustrated in Fig. 2.6, these thresholds define a mapping $\mu : S \to \{-, \triangledown, 0, \triangle, +\}$ as follows:

$$\mu(S) = \begin{cases} - & \text{if} \quad obs(s) \leq t_1 \\ \triangledown & \text{if} \quad t_1 < obs(s) \leq t_2, \\ 0 & \text{if} \quad t_2 < obs(s) \leq t_3, \\ \triangle & \text{if} \quad t_3 \leq obs(s) \leq t_4, \\ + & \text{if} \quad t_4 < obs(s) \end{cases}$$

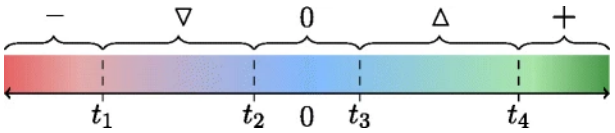



Figure 2.6: **Iggy : discretization of experimental observations** We assume the real values of observed experimental measurements on molecular species given in a continuous scale. When imposed a set of thresholds $t_i$, the real values are discretized into 5 discrete values.

Experimental measurements which are smaller than $t_1$, bigger than $t_4$, and between $t_2$ and $t_3$ are considered as certain (decrease '-', increase '+', no-change '0'), while measurements that are between $t_1$ and $t_2$ (resp. $t_3$ and $t_4$) are uncertain (uncertain-decrease $\triangledown$, uncertain-increase $\triangle$) and not exactly classifiable. Having said that, an experimental profile $(S, I, \mu)$ is defined by the set of measured species $S$, the set of input nodes $I \subseteq S$ (the experimentally perturbed species) whose changes are trivially explained, and the mapping $\mu$ as defined above.

### Local consistency rules

Given an IG $(V, E, \sigma)$ and an experimental profile $(S, I, \mu)$ one can describe the rules that relate both. For this purpose we look for total labelings (or coloring models) $\mu^t : V \to \{-, 0, +\}$ that satisfy the local constraints defined below. It is important to notice that $\mu^t$ will define a total labeling using the three labels $\{-, 0, +\}$ whereas $\mu$ defines a partial labeling (only measured nodes are labeled) based on the five labels $\{-, \triangledown, 0, \triangle, +\}$ representing the discretized measurements.

With the first constraint, we look for total labelings $\mu^t$ that satisfy the observed measurements captured in the partial node labeling given by $\mu$:

***Constraint 1 (satisfy observations)*** Let $(V, E, \sigma)$ be an IG, $(S, I, \mu)$ an experimental profile, $\mu^t : V \to \{-, 0, +\}$ be a total labeling, and let $i \in V$ be a node with $\mu^t(i) \in \{+, 0, -\}$. Then $\mu^t$ satisfies *Constraint 1* for node $i$ iff:

- $i \notin S$, or
- $\mu(i) \in \{+, \triangle\}$ and $\mu^t(i) = +$, or
- $\mu(i) \in \{\triangle, 0, \triangledown\}$ and $\mu^t(i) = 0$, or
- $\mu(i) \in \{\triangledown, -\}$ and $\mu^t(i) = -$

Uncertain measurements restrict the labeling of a node to two out of the three values $\{+, -, 0\}$, while measurements with high certainty fix a node's label to exactly one value.

In the following constraint we demand for every non-input node $i$, that its change $\mu^t(i)$ must be explained by the total influence of its predecessors in the IG. The influence of $j$ on $i$ is given by the product $\mu^t(j)\sigma(j, i) \in \{+, -, 0\}$.

***Constraint 2 (change must be justified by a change in a predecessor)*** Let $(V, E, \sigma)$ be an IG, $(S, I, \mu)$ an experimental profile, $\mu^t : V \to \{-, 0, +\}$ be a total labeling, and let $i \in V \setminus I$ be a non-input node with $\mu^t(i) \in \{+, -\}$. Then, $\mu^t$ satisfies *Constraint 2* for node $i$ if there is some edge $j \to i$ in $E$ such that $\mu^t(i) = \mu^t(j)\sigma(j, i)$. *Constraint 2* demands that increases and decreases of a node in the network must be explained by at least one of its direct predecessors.

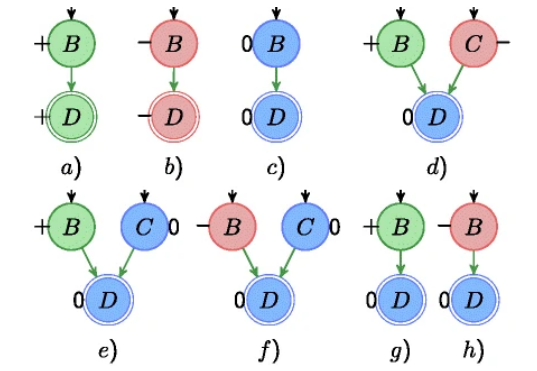
This is a very generic or permissive constraint, that allows that many coloring models, or total labelings $\mu^t$, exist. It is, however, subject to further exploration in case a clear description of the regulatory mechanism is given. For example, a cooperation, a complex-formation, or a competition. In [39] we explored a principle of maximising the direct

predecessors sign agreement with the target sign. Currently, in the context of the PhD thesis of Sophie Le Bars, we have proposed a *majority rule* in which the sign of a node reflects (in a more quantitative way) the part of the parents that is more sign-dominant (article accepted for publication in *BMC Bioinformatics*).

***Constraint 3 (0-change must be justified)*** Let $(V, E, \sigma)$ be an IG, $(S, I, \mu)$ an experimental profile, $\mu^t : V \to \{-, 0, +\}$ be a total labeling, and let $i \in V \setminus I$ be a non-input node with $\mu^t(i) = 0$. Then $\mu^t$ satisfies *Constraint 3* for node $i$ if there is either no edge $j \to i$ in $E$ such that $\mu^t(j)\sigma(j, i) \in \{+, -\}$ or there exists at least two edges $j_1 \to i$ and $j_2 \to i$ in $E$ such that $\mu^t(j_1)\sigma(j_1, i) + \mu^t(j_2)\sigma(j_2, i) = 0$.

*Constraint 3* restricts the occurrence of 0-changes. A node is only allowed to show 0-change if it receives either no influence or contradictory influences. This constraint thus assumes that each influence has indeed an effect and only contradictory influences can cancel each other out.

In Fig. 2.7, we illustrate the local constraints in IGs with different labelings where green stands for increase, red for decrease and blue for 0-change. Notice, that Constraint 2 intentionally allows situations like in labeling g and h, where $D$ is labeled as 0-change even if the predecessor $B$ is showing an increase resp. decrease. On the other hand, Constraint 2 forbids $D$ to increase or decrease, if all predecessors are labeled as 0-change.




| | *a* | *b* | *c* | *d* | *e* | *f* | *g* | *h* |
|---|---|---|---|---|---|---|---|---|
| Constraint 2 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Constraint 3 | ✓ | ✓ | ✓ | ✓ | × | × | × | × |

Figure 2.7: **Iggy : summary of local reasoning constraints.** IGs with different labelings where green stands for increase, red for decrease, and blue for 0-change. All labelings satisfy the basis of Constraint 2 for node D, but only the labelings a-d satisfy also Constraint 3.

### From local to global reasoning

There might exist several total labelings that satisfy the local constraints for some nodes. We are interested in checking global consistency, where a total labeling exists such that the local constraints are satisfied for all nodes. In Fig. 2.8, we illustrate an IG together with a partial labeling which is locally consistent but globally inconsistent. In other words, there exists two total labelings such that the local consistency rules (Constraints 1, 2 and 3) are satisfied, for either A or B, but there exists no single total labeling that satisfies these constraints for all nodes.

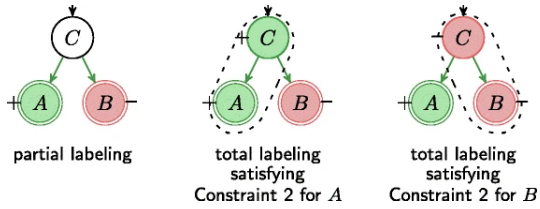



Figure 2.8: **Iggy : global sign consistency.** Example for an IG with partial labeling, which is locally consistent for A and B, but globally inconsistent because there exists no single total labeling satisfying Constraint 2 for A and B.

We use the previously defined constraints to define the following global consistency notions.

***Consistency Notion 1 (weak propagation, WP)*** We call an IG and an experimental profile $(S, I, \mu)$ consistent under weak propagation (WP), iff there exists a total labeling $\mu^t$ such that Constraints 1 and 2 are satisfied for all nodes.

***Consistency Notion 2 (strong propagation, SP)*** We call an IG and an experimental profile $(S, I, \mu)$ consistent under strong propagation (SP), iff there exists a total labeling $\mu^t$ such that Constraints 1, 2 and 3 are satisfied for all nodes.

In the paper shown at the Appendix A.1 we mentionned two others consistency constraints, the reader interested to understand all available Iggy functionalities is invited to read them in detail.

### Consistency checking

We can now apply the previously defined consistency notions to enumerate consistent total labelings and to verify the consistency of network and observation data for a given

experimental profile. We consider an IG consistent with an experimental profile $(S, I, \mu)$ if there exists at least one consistent total labeling (consistent with respect to the chosen Notion WP, SP). Consider Fig. 2.9 which shows the total labelings of the IG in Fig. 2.5 consistent with an example experimental profile (A and D were increased resulting in a measured 0-change in H) under the different consistency notions. Note that the notions become more strict, accepting less labelings as consistent and therefore excluding certain system behaviors. The set of admissible labelings under SP is a subset of the admissible labelings under WP.

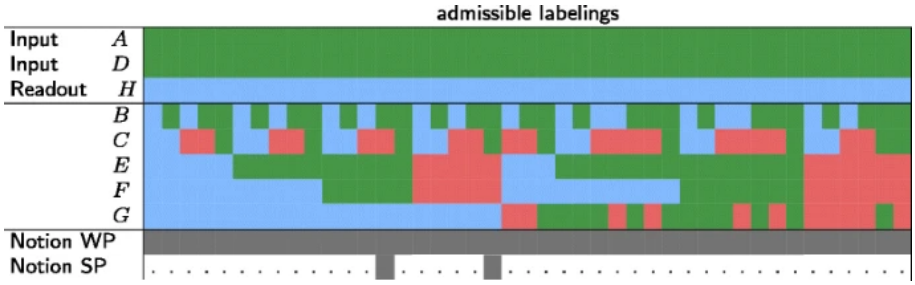



Figure 2.9: **Iggy: checking for consistency.** Consistent total labelings of the example in Fig. 2.5 under different consistency notions. In this example, there is a partial labeling given over 3 nodes in the graph, and 43 possible coloring models or labelings consistent at least under one consistency notion (WP or SP) are fully displayed. Note that all possible labelings are $3^5$. A grey cell indicates that the labeling above is consistent and a white cell with "." means that it is not a consistent labeling.

### Predictions under consistency

The consistency check of network and experimental data is the first analysis that is performed with the sign consistency approach. If network and data are consistent the sign consistency approach can be used to predict the behavior of unmeasured entities in the network. This can also be used to predict the outcome of a planed experiment and reversely to plan an experiment that should result in a specific desired behavior.

In the sign consistency approach, we call a statement (or sign) that holds in all consistent labelings, under the given consistency notion, a *prediction*. We assume that if some species of the system have the same sign in all consistent labelings, their sign can be predicted. We predict that a species increases '+' (resp. decreases '-', does not change '0') if it increases (resp. decreases, does not change) in all consistent labelings. We call these strong predictions, because the possible behaviors (or signs) of a species are reduced

to exactly one. Further, we can predict that a species does not increase (resp. does not decrease, does change) if it does not increase (resp. not decrease, does change) in all consistent labelings. Therefore, we can also predict weak increase $\oplus$, when a species does not decrease, but increases in at least one consistent labeling, and does not change in another consistent labeling. Likewise, we predict weak decrease $\ominus$ when a species does not increase, but decreases in at least one consistent labeling, and does not change in another. Finally, we predict change $\pm$ when a species does always change, it increases in at least one consistent labeling and decreases in another. We call $\oplus$, $\ominus$, and $\pm$ weak predictions because one possible sign is excluded while one degree of freedom is left.

For example, in Fig. 2.9, for the consistency notion SP, we observe that there are 2 consistent labelings. In which node $B$ is predicted as increase ('+'), and $G$ as '0', strong predictions. While nodes $C$, $E$, $F$ are predicted as $\pm$, corresponding to a weak prediction. For the consistency notion WP, only $B$ can be weakly predicted as $\oplus$; the rest of the nodes will not be predicted because their values can take signs in $\{+, -, 0\}$ in all the consistent labelings.

Formally, for a set $V$ of nodes in our network and the set $M$ of labelings consistent with our experimental profile, we define the prediction function $pred : V \to \{+, -, 0, \oplus, \ominus, \pm\}$ as follows:

$$pred(x) = \begin{cases} + & \text{if} \quad \forall \mu \in M : \mu(x) = +, \\ - & \text{if} \quad \forall \mu \in M : \mu(x) = -, \\ 0 & \text{if} \quad \forall \mu \in M : \mu(x) = 0, \\ \oplus & \text{if} \quad \forall \mu \in M : \mu(x) \neq -, \exists \mu \in M : \mu(x) = +, \exists \mu \in M : \mu(x) = 0, \\ \ominus & \text{if} \quad \forall \mu \in M : \mu(x) \neq +, \exists \mu \in M : \mu(x) = -, \exists \mu \in M : \mu(x) = 0, \\ \pm & \text{if} \quad \forall \mu \in M : \mu(x) \neq 0, \exists \mu \in M : \mu(x) = -, \exists \mu \in M : \mu(x) = +, \\ none & else. \end{cases}$$

### Repairing inconsistent networks and data

If network and data are inconsistent the natural question arising is how to repair networks and/or data, that is, how to modify network and/or data in order to re-establish their mutual consistency. A major challenge lies in the range of possible repair operations, since an inconsistency can be explained by missing interactions or inaccurate information in a network as well as by measurement errors. The sign consistency approach can be used

to determine a set of repair operations that are suitable to restore consistency. Typically, plenty of suitable repair operations are possible, in particular, if multiple repair operations are admitted. However, one usually is only interested in repairs that make few changes on the model and/or data. These minimal repair sets cannot only be used for hypotheses generation (*e.g.*, which data might be questionable or which edges might be missing or inactive) but as a quantitative measure for the fitness of model and data. Also note that once consistency is re-established, network and data can again be used for predicting behaviors of unmeasured entities. We will focus here on explaining the Minimal Correction Sets (MCOS). We refer to the reader to two studies detailing different repair operations [15, 32] to see all possible repairs implemented currently in Iggy.

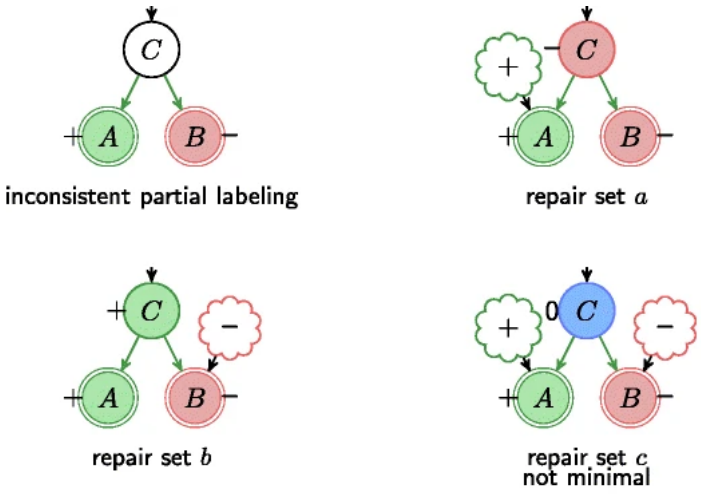
***Minimal Correction Sets (MCOS)*** To resolve inconsistencies we chose to add new influences to the model. Adding an influence can be used to indicate missing (unknown) regulations or oscillations of regulators that would explain the (topology-inconsistent) measurements. We use minimal correction sets (MCOS) as minimal sets of new signed (positive or negative) input influences that restore consistency of model and data. MCOS are defined as signed influences and are specific for a single experiment; they might be incompatible with other experiments. Note that every inconsistency can be repaired by adding a new influence. Therefore, adding influences is always suited to restore consistency. Also the MCOS can be interpreted as a measure of consistency of model and data. Fig. 2.10 illustrates how repair through addition of influences works.

#### Prediction under minimal repair

The sign consistency approach enables prediction even if model and data are mutually inconsistent. Predictions under minimal repair are obtained from the identification of consequences (or predictions) shared by all consistent labelings under all possible minimal repairs. Note that this approach although it confines to minimal repairs following the law of parsimony, does not favour any of the possible minimal repairs but only considers a statement a prediction if it holds under every minimal repair.

### 2.3.3 Sign consistency as a (brief) logic program

It is not our objective here to provide a full explanation of logic programming. Readers interested on the theory and practice of Answer Set Programming (ASP) are referred to




| Repair set | $a$ | $b$ | $c$ |
|---|---|---|---|
| Number of repairs | 1 | 1 | 2 |
| WP | ✓ | ✓ | ✓ |
| SP | ✓ | ✓ | ✓ |
| Minimality | ✓ | ✓ | × |

| Node | $A$ | $B$ | $C$ |
|---|---|---|---|
| repair set $a$ | + | − | − |
| repair set $b$ | + | − | + |
| prediction under minimal repair | + | − | ± |

Figure 2.10: **Iggy : minimal correction sets (MCOS) repair** Given the inconsistent labeling presented in the top-left IG, there are three alternative repair sets: repair set (a) adds a positive influence to $A$, repair set (b) includes a negative influence on $B$, and repair set (c) includes a positive influence on $A$ and a negative influence on $B$. Repair sets (a) and (b) are minimal (1 repair), while repair set (c) is not minimal (2 repairs). From the set of consistent labelings under minimal repairs, (a) and (b), the strong predictions are $pred(A) = +, pred(B) = -$, and the weak predictions is $pred(C) = \pm$.

[37, 40, 41]. In this section, we aim to explain how sign-consistency constraints can be encoded in a logic program. For a more detailed exploration of the logic programs behind Iggy we refer the reader to two papers [14, 15] where the ASP encodings are carefully explained.

***Brief ASP explanation*** In ASP a logic program is composed of a set of logical rules, which are themselves written using first-order predicates that relate objects (constants or terms). For example stating that $a$ is a node in a graph can be represented by the predicate `node(a)`. Logical rules can express different forms of *ideas*, with respect to predicates: predicates can be facts, always true; they can be inferred according to an specific logic (read from right to left); and they have to respect certain constraints. The solution of a logic program is called the *answer sets*. This solution is not written by the

human modeling the problem, but deduced automatically by the solver. For providing this solution, the solver checks if there is a stable Herbrand model [42] composed by a set of predicates justifying each rule of the logic program.

In modeling ASP logic programs one has to have in mind this particular set of answer-sets (predicates), composed of predicates derived from a combinatorial space of possible candidates, that satisfy all the given rules, in which can exist bonus elements (not disrespecting any rule), and that can be inferred by the program logic. A classical form of a logical rule in ASP is given by:

```
1    a_0 :- a_1, ... , a_m, not a_{m+1}, ..., not a_n.
```

This rule can be read as: if $a_1, \ldots, a_m$ are true and non $a_{m+1}, \ldots, a_n$ can be proved to be true, then $a_0$ must be true. $a_i$ are predicates, also named *atoms*. The *head* of the rule is the left part of it ($a_0$). The body, the right part. A rule without a body is called a fact: what is in the head is always true in the answer-sets of the logic program. A rule without a head is called *integrity constraint*: what is in the body is forbidden to be included in the answer-sets of the logic program. The answers of such logic programs refer to stable Herbrand models. They can be understood as a set of predicates that hold (true) for each rule of the program. A logic program can also give an *usatisfiable* answer, meaning that there does not exist a set of predicates satisfying all the rules given in the logic program.

***Sign consistency logic program*** ASP programs are used in search combinatorial problems, as coloring graphs; this is why we adopted this strategy in modeling the sign-consistency approach. ASP proposes an interesting paradigm to encode or model problems aimed at exploring discrete domains with specific simetries, as the one formalised in Section 2.3, when IGs have thousands of species. In the following, we formulate a simplified version of the sign-consistency modeling in a logic program. We aim with this example to give an idea of the logical rules implemented in Iggy and to show a reader not familiar with the ASP paradigm the semantics and syntax of a logic program.

To explain the logic program that decides on sign-consistency we use the IG and expression profile depicted in Fig. 2.11. The logic program shown in Listing 2.1 aims to express the logic of the *Constraint 1: weak propagation (WP)* (Section 2.3). Lines $1-5$ express what is called the *problem instance*, referring to the set of facts (heads of the rules) that are always true. In particular these lines will set the constants (words starting by a lowercase letter) of our problem. Line 1 expresses a predicate `signs` which sets up the

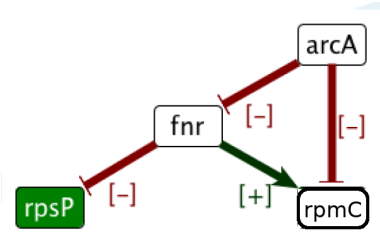



Figure 2.11: **IG and expression profile.** Signed and oriented IG: red arrows refer to inhibitions, while red to activations. The partial labeling is given for two nodes: *rpsP* and *rpmC*, both up-regulated (observed as '+').

domain of signs we will explore in this problem $\{+, -\}$. Notice that the ';' facilitates the writing of the rule: it is identical to write two rules: `signs(down). signs(up).` Notice that all logical rules end with a '.'. Line 2, expresses the nodes of the graph by using the predicate `vertex`. Line 3, expresses the partial labeling $\mu$ of the graph by using the binary predicate `observedV`. That is $\mu(rpsP) = +$. Lines 4 and 5 express the edges of the IG, their orientation and sign by using the predicate `observedE`.

```
1 sign(down;up).
2 vertex(rpsP;fnr;arcA;rpmC).
3 observedV(rpsP,up).
4 observedE(fnr,rpsP,down). observedE(fnr,rpmC,up).
5 observedE(arcA,fnr,down). observedE(arcA,rpmC,down).
6
7 {labelV(I,S):sign(S)} = 1 :- vertex(I).
8 labelV(I,S) :- observedV(I,S).
9 {receive(I,S):sign(S)} = 1 :- vertex(I).
10 receive(I,up) :- observedE(J,I,S), labelV(J,S).
11 receive(I,down) :- observedE(J,I,S), labelV(J,T), S!=T.
12 :- labelV(I,S), not receive(I,S).
```

Listing 2.1: Logic program sc.lp

The combinatorial choice of possibilities is given in Line 7. This type of rules are named as *choice rules*. Notice that variables are given in uppercase letters. The rule reads as: for each vertex I, generate *one* possible labeling `labelV` relating $I$ with a variable $S$ given by the predicate `sign(S)`. Executing this rule (together with the problem instance) will give answer sets composed by a fully affectation of signs in $\{+, -\}$ to each node of the graph. All $2^4$ answer sets will correspond to different possible affectations.

Then, lines $8 - 12$ aim to constraint these possible affectations so that they respect

the *WP constraint*: *"The sign of a node having predecessors in the graph, should be equal to the sign of the influence of at least one of its direct predecessors"*. The influence of a node $j$ over its successor $i$ in the graph is equal to the product of its sign *sign(j)* and the sign of the edge $(j, i)$, *i.e.* $sign(j) \times sign(edge(j, i))$. To encode this logic, we start by first fixing the possible `labelV` of a node that appears in the expression profile $\mu$ (Line 8): given an observed sign $S$ on a node $I$, then the `labelV` of node $I$ should be $S$. Because of the cardinality choice constraint (Line 7), which limits the number of possible signs affectations of a node to 1, Line 8 fixes the sign of each observed node. Line 9 expresses a second choice rule by using the predicate `receive` to refer to the influences that a node can receive. Line 9 expresses: each node can only have one predicate of type `receive` with a sign $S$ assigned to it. In lines 10 and 11, the choice of sign $S$ associated to a node with predicate `receive` is narrowed as follows: it will be `up` for node $I$, if its predecessor $J$ has a sign that agrees with the sign of the edge $(J, I)$ (Line 10). A similar logic is written for the opposite signs in Line 11. Line 9 allows assigning to nodes without a predecessor one possible sign, unconstrained. Finally Line 12 expresses the full constraint: *there cannot be a labeling over a node I, having a sign S, in which S does not agree with the sign of a received influence*. Line 12 expresses a bound between the choice of the predicate `receive` (Line 9) and the choice of the labeling `labelV` (Line 7). Both choices need to be selected in such a way so that rule at Line 12 holds.

Running the clingo [43] solver on this program we obtained that: from the initially possible $2^4$ answer sets (rules in lines $1-5$), when the partial labeling is fixed (rules $1-7$) we go to 8 answer sets, and finally when the sign-consistency constraint is defined (rules $1-12$) we end with one possible consistent model, as shown in Fig. 2.12. We can observe that the answer set found assigns, using predicate `labelV`, consistent signs (or colors) to the nodes in the IG (see Fig. 2.11), according to the rules given in Listing 2.1.

```
cguziolo@cguziolo-Latitude-7480:~/HDR/TheseUBL-v1-13/vL/encodings$ ~/src/clingo-4.5.4-linux-x86_64/clingo sc.lp
clingo version 4.5.4
Reading from sc.lp
Solving...
Answer: 1
labelV(rpsP,up) labelV(fnr,down) labelV(arcA,up) labelV(rpmC,down)
SATISFIABLE

Models       : 1
Calls        : 1
Time         : 0.003s (Solving: 0.00s 1st Model: 0.00s Unsat: 0.00s)
CPU Time     : 0.000s
```

Figure 2.12: **Answers sets.** Result from the execution with clingo v4 of the encoding in Listing 2.1. There is one answer set showing the different `labelV` chosen.

**Iggy software** The different consistency notions as well as the methods for consistency checking and quantification, prediction, and all data and network repair operations were implemented in an open source application Iggy. Iggy uses ASP as logical modeling and constraint solving paradigm, it is part of the BioASP software collection. ASP is used to model problems from NP and provides state-of-the-art solvers. In particular, Iggy uses the solver clasp [43] via the Rust wrapper, available with the package *clingo-rs*[1]. For further information visit `http://bioasp.github.io/iggy`.

## 2.4 Applications

We have applied the sign-consistency modeling to different biological systems, containing tens, hundreds, and thousands of species. These works were all published and appear rapidly listed bellow.

- *Escherichia coli* gene regulatory network (GRN) [13, 44]
- Human biological systems:
  - GRN on Ewing's tumor development [45].
  - Signaling and gene regulation on hepatocyte growth factor-stimulated cell migration and proliferation [46].
  - Signaling and gene regulation of Multiple Myeloma patients [47].
  - Signaling and gene regulation of Hepatocellular carcinoma [48].
  - HIF signaling pathway, impacting neurodegenerative diseases [49].

The studies [13, 44] aimed to validate the sign-consistency frameworks; more recently in [49] we investigated the comparison between Iggy's discrete computational prediction and a probabilistic quantitative computational prediction system. In [46, 47, 48] the purpose was, together with biological experts, to provide a novel understanding of the systemic biological behavior in a specific context. We refer to these last category of works as *applications*. In the following Chapter 3, we show the results and the hypotheses behind the sign-consistency modeling when applied to model Multiple Myeloma.

## 2.5 Limitations and further modeling

The sign-consistency approach proposed in Iggy remains of discrete nature. In order to use Iggy, the biological data needs to be discretized. Also, the nature of Iggy's predictions

1. https://github.com/potassco/clingo-rs

is discrete, reflecting up- or down-regulations in network components without giving a precise information of the given intensity of the changes. This discrete nature is a recurrent question that arises when discussing with bioinformaticians or biologists. To answer this, we have explored two separate paths. The first one avoids discretizing the gene expression profiles; the second one, proposes an idea of weights in the model predictions. The following paragraphs explain the main ideas of these paths.

In [39], the objective was to confront a dataset of expression profiles with a network topology. The confrontation, however, was done as a post-processing (similarity measures) of the answer of the logic program applied only on the network. We started by modeling the biological network representing its underlying structure as a logic-program. This model pointed to reachable network discrete states that maximize a notion of harmony between the molecular species active or inactive possible states and the directionality of the pathways reactions according to their activator or inhibitor control role. Afterwards, we confronted these network states with the gene expression profiles (GEPs). From this confrontation independent graph components are derived, each of them related to a fixed and optimal assignment of active or inactive states. These components allowed us to decompose a large-scale network into subgraphs and their molecular species state assignments had different degrees of similarity when compared to the same GEP. In Chapter 4, we explain this system with more details applied to the classification of Multiple Myeloma patients.

In [50] we proposed a comparison between Iggy and Probregnet [51], which is a method that outputs quantitative predictions on the network components by using probabilistic models. This comparison was studied in the context of Sophie Le Bars's PhD project, that aimed to evaluate the impact of Iggy's predictions as input for metabolic networks modeling based on linear programming. Our results modeled the HIF-1 (Hypoxia-inducible factor) signaling network using gene expression measurements from patients with Alzheimer Disease. They showed that Iggy's predictions are comparable and closer to experimental data with respect to Probregnet's predictions. As a continuation of this work, we explored the impact of a sign-consistency weighted approach over Iggy's predictions in order to better approximate the measure of optimal ATP, which is the output of linear programming modeling of the metabolic network of Human brain. Our results (article accepted for publication in *BMC Bioinformatics*) suggest that this new tool *MajS* provides a better way, when compared to Iggy, to bridge GRN modeling with metabolic network analysis.

Iggy's *simple* idea to confront the logic between experimental data and network topol-

ogy, to automatically find errors, and propose minimal repairs, has inspired a research work [17], where the authors proposed an approach to check Boolean Network consistency and propose automatic repairs. Iggy's main advantage is its scalability, since it can analyse networks composed of 3383 nodes and 13771 edges, as it was the case in a study of Hepatocellular carcinoma [48] using regulatory and signaling information extracted from the KEGG datbase, in a minute (standard laptop computer). A path we have not yet explored is the space of minimal repairs, their properties, and the connexion with biological phenomena, such as mutations for example. Another line of research we have not explored, and that we start noticing worth to invest energies in, is the connection between sign-consistency and genome assembly through graph representations of genomes.

Chapter 3

# Applying the sign consistency model to Multiple Myeloma

*"Reality is much kinder than thoughts about reality"*
— Byron Katie

## 3.1 Introduction

We present in this chapter part of the research study we published in 2017 [47]. Readers who wish to deepen the methods and results understanding of this work are referred to Appendix A.2. This work was a result of a collaboration with Stéphane Minvielle and Florence Magrangeas, from the CRCINA (*Centre de Recherche en Cancérologie et Inmunologie Nantes Angers*). This collaboration was funded by GRIOTE (*Groupement de Recherche en Intégration de données Omics à Très grande Echelle*) a project assembling bioinformatics research in the french region *Pays de la Loire*. The first author of the work was Bertrand Miannay, who did a doctoral thesis under my co-supervision. Our biological partners provided us with a set of unpublished mRNA chips data of patients that developed a blood cancer named Myelome Multiple. They were interested to discover the signaling or gene regulatory networks underlying this experimental data, with the optic to personalize the treatment of this cancer.

We start this section stating the context of Multiple Myeloma (Section 3.2). After, we present a review of integrative approaches (Section 3.3), which are methods used classically to relate a gene expression profile (GEP) with a biological network or pathway. Our objective here is to clarify how the sign-consistency modeling approach can be related, and at which part, to this integrative process. Later, in Section 3.4 we will explain the biological data we used in this research. In Section 3.5 we will show the results we obtained.

## 3.2 Biological context of this study

Multiple myeloma (MM) is a neoplasm of plasma cells with an incidence rate of approximately 5/100,000 in Europe. The median survival of MM patients has improved substantially over the past decade. Owing to the establishment of high-dose therapy followed by autologous stem cell transplantation as a routine procedure, significant improvements in supportive care strategies, and the introduction and widespread use of the immunomodulatory drugs thalidomide and lenalidomide, and the proteasome inhibitor bortezomib. Nevertheless, almost all MM patients ultimately relapse, and new drugs and new combinations for the treatment of MM are warranted. MM is a heterogeneous disease at both the clinical and molecular levels. Recent large scale genomics analysis based on the landscape of copy-number alterations and on whole exome sequencing have revealed the hallmarks of genetic changes in MM such as hyperdiploidy, translocations involving the IgH locus, and mutations in the RAS/MAP and NF-kB pathways and in TP53 [52]. These genetic changes as well as gene-expression profiling (GEP) have been widely used in the molecular classification of newly diagnosed patients to define diagnostic entities and identify promising new therapeutic targets [53, 54, 55, 56, 57, 58]. However, at present a standard of classification based on subgroups that could be targeted therapeutically is still being debated. Clearly, there is a need for innovative tools to improve the identification of the prognostically relevant entities, clinically and biologically, in newly diagnosed MM patients. It is tempting to use the mutational spectrum based on whole-exome sequencing as a gold standard; however Stephane Minvielle, Florence Mangreas and colleagues have previously shown that a large number of exome mutant alleles are not expressed clinically or biologically [59]. In addition, exome sequencing may miss potential driver mutations in the non coding regulatory elements known to affect enhancer activity, which thereby affect the transcriptional program [60]; therefore GEP remains a tool of choice. However, GEP alone is limited and must be integrated with innovative approaches that use biological regulatory networks to extract biological information relative to gene expression datasets to provide significant clues about the etiology of myeloma.

## 3.3 Integrative Approaches

During the past decade, many methods of so-called pathway analysis or active pathways detection have been developed. These methods use as a knowledge base a biological

pathway or regulatory network, that compiles a series of molecular phenomena that lead to activation (or inhibition) of gene expression, a cell product such as a hormone, or a physical modification of the cell. Regulatory network information is currently available through databases such as Gene Ontology (GO) [61], the Kyoto Encyclopedia of Genes and Genomes (KEGG) [62], the Pathway Interaction Database (PID) [63], Wikipathway [64], Transfac [65], and Causal Biological Networks (CBN) [66]. The main objective of pathway analysis methods is to confront or integrate GEP data with regulatory networks or pathways to distinguish two or more classes of cells (*e.g.* healthy vs ill) from GEP data by inferring a specific signature for each class. We can identify three principal categories of approaches that have been used to associate GEP with specific pathways [67].

The **Over-Representation Analysis (ORA)** group of approaches [68, 18] includes approaches that are based on differentially expressed (DE) genes. These approaches score single pathways based on the proportion of DE genes (identified with statistical tests or with a threshold) contained in each pathway. In most cases, these methods use a hypergeometric test to score each pathway. Moreover, the majority of ORA approaches that use functional annotation (GO) or pathway maps (KEGG) consider the consequences of the DE genes (leading to the differentially expression of proteins) in the associations between gene and pathway. Martin *et al.* [69] called this type of reasoning forward assumption compared to the backward assumption [18], which considers the causes of those DE genes in the gene-pathway association.

The **Functional Class Scoring (FCS)** group of approaches uses the full datasets without any pre-selection, allowing integration of the effects of low gene expression variations in the identification of the pathways involved. FCS approaches can use forward [70, 71] or backward [72, 73] reasoning. Although these methods improve the problem of genes selection, the pathways in which individual genes are involved are still studied independently. Moreover, the position of the genes in the topology is not used in the analysis.

The **Pathway Topology (PT)** approaches are very similar to the FCS approaches, but in addition, they score genes according to the pathways to which they belong. Whereas some of these approaches only include interactions between genes [74, 75, 76, 77], others consider different types of relationships between genes [69, 78] generally activation and inhibition. The majority of methods study each pathway independently. Within this group, we can also identify methods that use both forward [74, 75, 76, 77] and backward [69, 78] reasoning.

In this case-study, we integrated the GEPs obtained from myeloma cells (MC) of 602

MM patients and from normal plasma cells (NPC) of 9 healthy donors with the whole compendium of the PID-NCI public pathway repository so as to better understand the mechanisms of plasma cell carcinogenesis. To integrate this data, we first automatically build a directed (and labeled) graph using the whole compendium of the PID-NCI public pathway repository. This graph connects signaling pathways to the transcription of the genes in the GEP dataset. We then integrate the graph with the expression data by reasoning on its logic using Iggy (see Section 2.3.3). Our combined approach could be considered to fall within the PT category since it takes into account the causality and activation/inhibition logic of graph edges. However, unlike previous cited methods, it uses a global logic to analyze experimental and pathway data. In this formalism, both forward and backward modes are included as reasoning modes (causes-consequences). We proposed an integrative method that does not correlate protein activation with gene expression; the two entities are identified separately in the graph. The non-measured protein activations necessary to satisfy the GEP according to the entire pathway database topology (*i.e.* Iggy's predictions, Section 2.3.2) are used later to propose a signature for each dataset profile. This global signature can be used to characterize the dataset classes. Moreover, our model also allows us to *in silico* quantify the effect of perturbations on this global pathway for each single patient. We show how this type of method, which combines large-scale information in terms of number of patients, the complete GEP, and the entire compendium database, can be applied to identify new specificities of MM disease compared to normal cells. As a result, we inferred information on the states of specific proteins in the cell that may cause these disorders, and we identified specific markers of MC compared to NPC that can be used to identify survival markers. Furthermore, these markers can be studied as therapeutic targets because of their over-representation and their impact on the involved pathways.

## 3.4 Biological knowledge

### 3.4.1 Graph generation

We used the 2012 version of the entire pathways database PID-NCI (Pathway Interaction Database) [63] and downloaded it in PID-XML format. This database is specialised to include regulatory pathways involved in cancer. The complete graph contains 17932 nodes (proteins, complexes, genes, transcription or protein modification events) and 27976

edges (activation or inhibition). To orient our analysis to the expression profiles and to the biological problem at hand, we built a subgraph with signed edges by extracting the downstream events from three signaling pathways (IL6/IL6-R, IGF1/IGF1-R and CD40), all of which are known to include cellular receptors involved in MM [79], to the over- and underexpressed variant genes from all datasets in the gene expression profiles by the shortest paths. This cycled, directed subgraph was then filtered by deleting all nodes that are not observed and with one predecessor or one successor [29]. This filtering step involves no loss of information with respect to the graph coloring model and allowed us to reduce the complexity of the analysis while maintaining the dependencies between the nodes.

### 3.4.2 Experimental data and discretization

**Data: source and measurement** The MM data was obtained from plasma cells, isolated from the bone marrow of 602 newly diagnosed cases of MM. The samples were obtained during standard diagnostic procedures conducted at the *Intergroupe Francophone du Myélome* (IFM) centers. The subjects included patients that were enrolled in the IFM trials of 2005 and 2007, and 9 normal healthy plasma cells donors. The Affymetrix Human Exon1.0 chip technology was used to measure the RNA quantity for each sample extracted from each subject. In total there were 611 $(602 + 9)$ GEPs to be analyzed.

**Gene differential expression and data discretization** Let $V^s \in \mathbb{R}^n$ be a $n$-dimensional vector representing the $n$ measured values of species (gene) $s$ in each of the $n = 611$ subjects. Since there are two types of subjects, MM patients and NPC (normal plasma cells) donors, we can say $V^s = M^s \cup N^s$; where $M^s$ refers to the vector of measured values for $s$ in the MM patients and $N^s$, in NPC donors. Recall that $|M^s| = 602$ and $|N^s| = 9$. Each element of $V^s$, denoted as ${v^s}_i$, represents the measured expression of gene $s$ in MM subject $i$, taken from the Microarray chip analyses. Let $P^s \in \mathbb{R}^n$ be a $n$-dimensional vector representing the $n$ differentially expressed values of $s$ in each of the $n = 611$ subjects. Each element of $P^s$, denoted as ${p^s}_j$, represents the differential expression of gene $s$ in subject $j$ with respect to the average expression of gene $s$ in NPC donors: ${p^s}_j = \frac{{v^s}_i}{N^s}$.

Once a differential expression is calculated, then we can transform its real value in a discrete domain of $\{+, -, 0\}$. Recall that Iggy (Section 2.3.3) requires a partial labeling of the nodes in such discrete domain. Similarly to what presented in Section 2.3.2, we will use two thresholds $0 < k_1 < k_2$. Recall that $S$ referred to the set of species (RNA or genes)

measured experimentally. These thresholds will define a mapping $\mu^j : S \to \{-, 0, +\}$, for each MM patient $j$, as follows:

$$\mu^j(S) = \begin{cases} - & \text{if} \quad {p^s}_j \leq -k_2 \\ 0 & \text{if} \quad -k_1 < {p^s}_j \leq k_1, \\ + & \text{if} \quad k_2 < {p^s}_j \end{cases}$$

The thresholds $k_1$ and $k_2$ were chosen to limit a proportion of signed genes in each subject $j$ to 50% and to provide a better precision of the Iggy's predictions (see Appendix A.1 Materials and Methods, for details).

## 3.5 Results

### 3.5.1 Sign consistency modeling: interaction graph and partial labelings

The NCI-PID integration allowed us to find 634 genes (a protein preceded by a transcription event). Independently, the discretization of the experimental data proposed signed observations in $\{+, -, 0\}$ on microarray probes corresponding to 15418 proteins identified in Uniprot. Merging both lists allowed us to identify 557 genes present in the NCI-PID and observed as over- and under-expressed or invariant in the GEPs. By extracting the downstream events from three signaling pathways (IL6/IL6-R, IGF1/IGF1-R and CD40) to the variant genes, we generated an induced subgraph from NCI-PID containing 2269 nodes, 2683 edges and connecting 529 variant genes. This graph was then compacted to a new graph with 596 nodes and 960 edges (Fig. 3.1) and composed of 529 observed nodes (genes) and 67 unobserved nodes, including 23 proteins, 33 complexes, 2 biological processes, 9 proteins reactions (translocation, phosphorylation, etc.).

### 3.5.2 Key nodes identification in Multiple Myeloma patients

**Methodology** We proposed here a method to analyze Iggy's sign predictions (see Section 2.3.2), under MCOS minimal repair by using statistical and machine-learning approaches. Our objective was to identify, using Iggy's predictions (function *pred*), specific markers of MM subjects in comparison to NPC donors. For this, we generated for each node $x$ in the graph, and each possible sign $u \in \{+, -\}$, a set of triplets of the form $(x, u, b)$ (where $b$ is a Boolean value) in the following way:

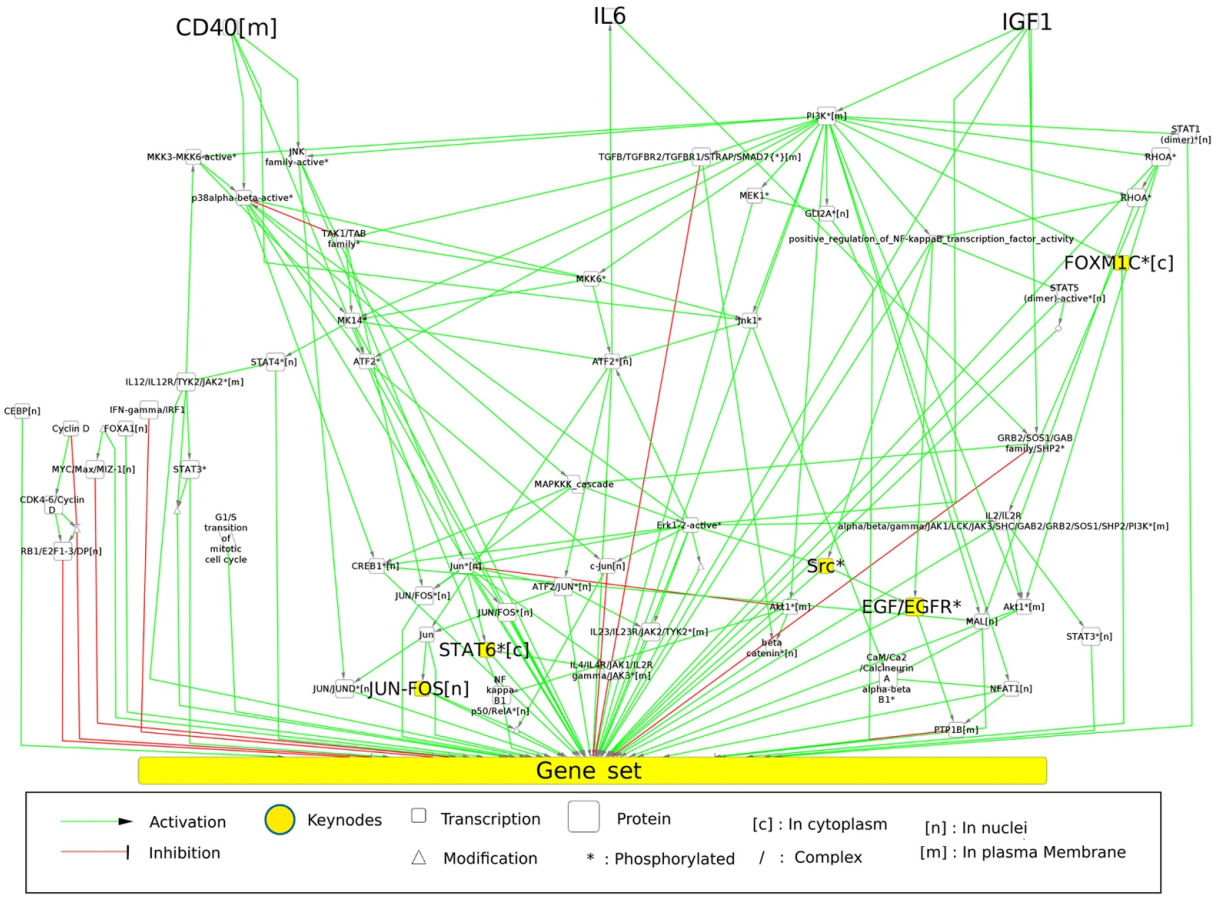



Figure 3.1: **Interaction graph explaining the differentially expressed genes in Multiple Myeloma.** Representation of the subgraph obtained from the PID-NCI database. CD40, IL6 and IGF1 (the nodes in the top portion of the graph) are the 3 queried pathways. The 529 genes that are differentially expressed across all profiles are merged for this representation in the node "Gene set". We used the same syntax for all nodes in this study. The edges from the "Gene set" node to proteins have been deleted for the sake of clarity.

$$(x,u,b) = \begin{cases} (x,u,T) & \text{if} \quad pred(x) = u \\ (x,u,F) & \text{if} \quad \nexists\, pred(x) \end{cases}$$

This set of triplets was represented by a Boolean matrix $M \in \mathbb{B}^{m \times 2n}$, where $m$ represents the number of nodes predicted by Iggy in at least one GEP (or subject) and $n$ represents the total number of GEPs ($n = 611$). Let us consider that all even columns of $M$ refer to a '+' prediction, while odd columns, to a '-' prediction. Then, the matrix element $m_{ij} = T$ expresses that $pred(i) = +$ in profile $j$, when $j$ is even; and $pred(i) = -$

in profile $j$, when $j$ is odd. Likewise, $m_{ij} = F$ expresses that $\nexists\ pred(i) = +$ in profile $j$, when $j$ is even; and $\nexists\ pred(i) = -$ in profile $j$, when $j$ is odd.

To identify specific markers of MM, we analyzed matrix $M$ and looked for overrepresented values when comparing $M$ columns belonging to MM subjects with respect to those belonging to NPC. For this, we used two approaches: a machine-learning approach based on supervised learning and a statistical approach based on frequency classification. For the supervised learning, we used decision trees [80] and random forest classification [81]; due to the underrepresentation of the NPC class, we multiplied the NPC weight by 67 so as to balance both classes (9 NPC and 602 MC). For the frequency approach, we calculated the frequency score ($F$) for each class $C$ (MM or NPC) and for each assignment $(i, u)$, where $i$ is the predicted node and $u$ the sign in $\{+, -\}$, as follows:

$$F^{C}{}_{i,+} = \frac{1}{c} \sum_{1 \leq j \leq c,\ j\ even} m_{ij} \qquad or \qquad F^{C}{}_{i,-} = \frac{1}{c} \sum_{1 \leq j \leq c,\ j\ odd} m_{ij}$$

Where $c$ represents the number of elements in the $C$ class. We then sorted our results based on a Fisher test between the proportions for NPC and MM to determine the most specific node assignments for the MM datasets.

**Results** Fig. 3.2 shows the decision tree obtained by comparing Iggy's predictions of MM subjects with respect to normal subjects. It shows that the combination of the assignments (JUN/FOS[n], -) and (FOXM1*[c], -) is associated with the majority of MM subjects (73%) and that the method can distinguish MM from NPC GEPs. JUN/FOS[n] represents the protein complex composed of JUN and FOS, which is located in the nucleus, whereas FOXM1*[c] represents the FOXM1 protein, which is phosphorylated and located in the cytoplasm. We can identify another important group of MM subjects (13%) that is characterised by the presence of (JUN/FOS[n], -) and the absence of (FOXM1*[c],-) and (SRC*, -). Similar results were obtained using a random forest classification.

To characterize the shared specificity for all MM GEPs, we computed the frequency scores ($F$) for our predictions. In Table 3.1, we show the 5 best p-values associated with a Fisher test with $F^{MM} > F^{NPC}$. For these selected nodes, we checked the number of input/output observed as variant $\{+, -\}$ genes connected to each node in the graph (Table 2, column connectivity). We observe that inhibition of the complex JUN/FOS[n] is predicted for 95.6% of the MM GEPs. The activity levels of FOXM1*[c] and STAT6*[c]

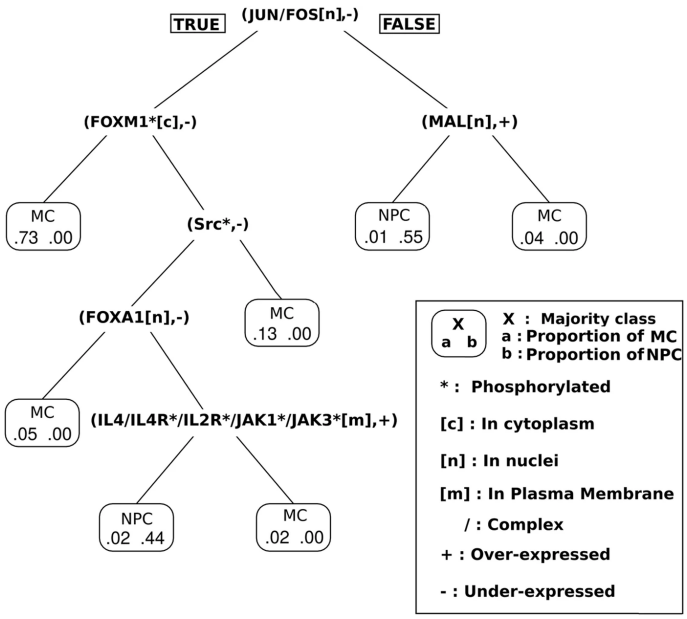



Figure 3.2: **Decision tree** based on Iggy's predictions of MM subjects RNA expression data with respect to normal ones.

were predicted to decrease. This decrease, in terms of protein activity, is correlated with the level of gene expression in 76% and 93%, respectively, of the MM datasets (column OVE). The frequency score classification identified the presence of (Src*, +) as an interesting marker for MM datasets. Interestingly, the decision tree approach identified the absence of (Src*, -) to distinguish MM datasets that were previously characterized by (JUN/FOS[n], -) and (FOXM1*[c], -). Both the machine-learning and statistical methods identified (JUN/FOS[n], -) and (FOXM1*[c], -) as important markers of MC datasets. In Fig. 3.1, we show (marked as yellow nodes) how these 5 main proteins or protein complexes appear connected following the PID-NCI representation.

### 3.5.3 Nodes perturbation

**Methodology** In this analysis, we quantified the effectiveness of a node perturbation to simulate *in silico* the activation or inhibition of a protein. The quantification of these *in silico* perturbations was performed in two steps.

1. For each node $i$, predicted in at least one GEP, and for each GEP $j$, we generated a new dataset of observations $\mu_{ij}$ identical to the original dataset of profile $j$ ($\mu^j$) except that we added an observation on node $i$ fixed to '+' or '-' depending whether

| Predicted node | Sign | $F^{NPC}$ | $F^{MM}$ | p.val (Fisher) | References | Connectivity | OVE | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | + | - |
| JUN/FOS[n] | - | 0.444 | 0.956 | $2.65e-005$ | [82, 83, 84, 85, 86] | 8/529 | 373 | 137 |
| FOXM1*[c] | - | 0.222 | 0.774 | $7.97e-004$ | [87, 88] | 529/529 | 85 | 265 |
| STAT6*[c] | - | 0.222 | 0.764 | $1.05e-003$ | | 8/529 | 30 | 429 |
| EGF/EGFR*[m] | + | 0.556 | 0.935 | $2.08e-003$ | [89, 90, 91] | 529/529 | 79 | 4 |
| Src* | + | 0.556 | 0.935 | $2.08e-003$ | [92, 93, 94] | 529/529 | 110 | 48 |

Table 3.1: **Frequency score of Iggy's predictions for the NPC and MM subjects**. The references column lists the publications that agreed with our sign prediction. Connectivity refers to the ratio of genes connected to each predicted node. The OVE (observed variant expression) shows the number of variant gene expressions, using the best precision threshold, across all the GEPs.

$j$ was even or odd. We then computed the SCENFIT score $s_{ij}$ between the graph $G$ and $\mu_{ij}$. In a similar way to the MCOS score, SCENFIT computes the number of minimal repairs that can be performed on the dataset to restore consistency after flipping the sign in the dataset of observations. We used a matrix $S \in \mathbb{Z}^{m \times 2n}$ to store all the SCENFIT scores, where $m$ stands for the number of nodes predicted by Iggy in at least one GEP (or subject) and $n$ represents the total number of GEPs ($n = 611$).

2. We computed the Top Perturbation Score (TPS) for each assignment of node $i$ to a sign in $\{+, -\}$ as follows:

$$TPS^{C}{}_{i,+} = \frac{1}{c} \sum_{1 \leq j \leq c,\ j\ even} f(i,j) \qquad or \qquad TPS^{C}{}_{i,-} = \frac{1}{c} \sum_{1 \leq j \leq c,\ j\ odd} f(i,j)$$

where

$$f(i,j) = \begin{cases} 1 & \text{if} \quad s_{ij} \geq S_{*,j_{(-X)}} \\ 0 & \text{otherwise} \end{cases}$$

and

$$S_{*,j_{(-X)}} = max(s\ such\ that\ \#\{e \in S_{*,j} \mid e \geq s\} = X)$$

$C$ represents the class MM or NPC, $c$ the number of elements in class $C$. $S_{*,j}$ refers to j-th column of matrix $S$. Note that $S_{*,j_{(-X)}}$ selects the threshold that separates the top $X$-scores of column (or GEP) $j$. For our analyses we use $X$ such

that it separates the 10% of the SCENFIT scores across the GEP. Thus, $TPS$ measures the percentage of cases, across all GEPs of a specific class, in which the perturbation of node $i$ was relevant.

**Results** From the computation of all *in silico* node perturbations, we evaluated the impact of perturbing the key nodes found with the frequency score method (see Section 3.5.2, Table 3.1). Our results are shown in Table 3.2. A unilateral Fisher test allowed us to evaluate the significance of each perturbation compared to the NPC datasets. We can see that the activation of JUN/FOS generates a top-ranked (10% top) score of conflicts and therefore proposes repairs in 74.6% of the MM datasets, whereas it proposes repairs only on 22.2% of the NPC datasets. Interestingly, *in vitro* JUN overexpression in MM cell lines results in cell death and growth inhibition [85]. A similar tendency (more conflicts in MM than in NPC) is observed when FOXM1 is activated, but the difference cannot be considered significant. Nonetheless, we note that of the 36.4% of profiles in which the activation of FOXM1 is top-ranked, 96.8% correspond to patient profiles with the prediction (FOXM1*[c], -) (see Supplementary Material, Table S3 of the paper in Appendix A.2). For the other proteins and complexes, we can see that the difference between MM and NPC is not significant. It is worth noting that the p-value of a perturbation that goes in the opposite direction of the prediction shown in Table 3.1 is in all cases lower than the one of a perturbation which goes in the same direction of the prediction.

### 3.5.4 JUN/FOS activity as specific marker

The FOS and JUN proteins form a heterodimer complex that is responsible for AP-1 activity. This activity is known to play a role in tumorigenesis because it has been implicated in the induction of apoptosis, in the promotion of cell survival and in proliferation. The classification methods showed that (JUN/FOS[n], -) is the best assignment to distinguish MM from NPC subjects and revealed that AP-1 activity is lower in almost all MM patients than in normal controls. Inspection of individual patients' subgraphs showed predominantly underexpression (65% of the observed expression in MM) of the proapoptotic protein BIM. These results are in agreement with the results of in vitro studies demonstrating that in myeloma cell lines IL6 protects against apoptosis via AP-1 inactivation [83].

| Node | Dir | $TPS^{NPC}$ | $TPS^{MM}$ | p.val |
|---|---|---|---|---|
| JUN/FOS[n] | + | 22.2% | 74.6% | 0.001 |
| | - | 44.4% | 0.5% | 1 |
| FOXM1*[c] | + | 11.1% | 36.4% | 0.107 |
| | - | 55.6% | 19.1% | 0.997 |
| STAT6*[c] | + | 33.3% | 55.0% | 0.169 |
| | - | 44.4% | 21.9% | 0.970 |
| EGF/EGFR*[m] | + | 0.0% | 0.3% | 0.971 |
| | - | 0.0% | 3.5% | 0.728 |
| Src* | + | 0.0% | 1.3% | 0.887 |
| | - | 11.1% | 33.4% | 0.150 |

Table 3.2: **Node's perturbation results**. Each node was perturbed in two directions (column Dir): +, activation and -, inhibition. TPS represents the frequency with which perturbing a node in a specific direction was significant (*i.e.* it generated a high, 10% top, SCENFIT score) across the MM profiles ($TPS^{MM}$) or NPC profiles ($TPS^{NPC}$). The highlighted rows contain percentages which refer to perturbations that have a direction opposite to that of the predicted signs obtained with the frequency score (Table 3.1). P.val was obtained using a unilateral Fisher test.

### 3.5.5 FOXM1 activity as survival marker

FOXM1, a transcriptional factor known to be associated with MM, has been studied as a therapeutic target [88]. Based on the graph reduction and on our reasoning model, FOXM1*[c] is equivalent to FOXM1*[n] and is representative of the FOXM1 transcriptional activity. Firstly, we analysed FOXM1 gene expression in the MM groups in which FOXM1 activity was predicted. We found that decreased FOXM1 activity is associated with reduced expression of the FOXM1 gene (Fig. 3.3, left). Since our model identified a subgroup of patients with decreased activity of FOXM1 and since decreased expression of FOXM1 is associated with superior survival, we wanted to know whether FOXM1 activity could impact survival. We compared overall survival (OS) in both predicted groups in the larger cohort of patients that received comparable treatment (Velcade-dexamethasone induction followed by high-dose melphalan and autologous stem cell transplantation; $n = 450$) (Fig. 3.3, right). A log-rank test between these groups yielded a p-value of $< 0.1$, allowing us to conclude that low FOXM1 activity is associated with a trend towards better survival.

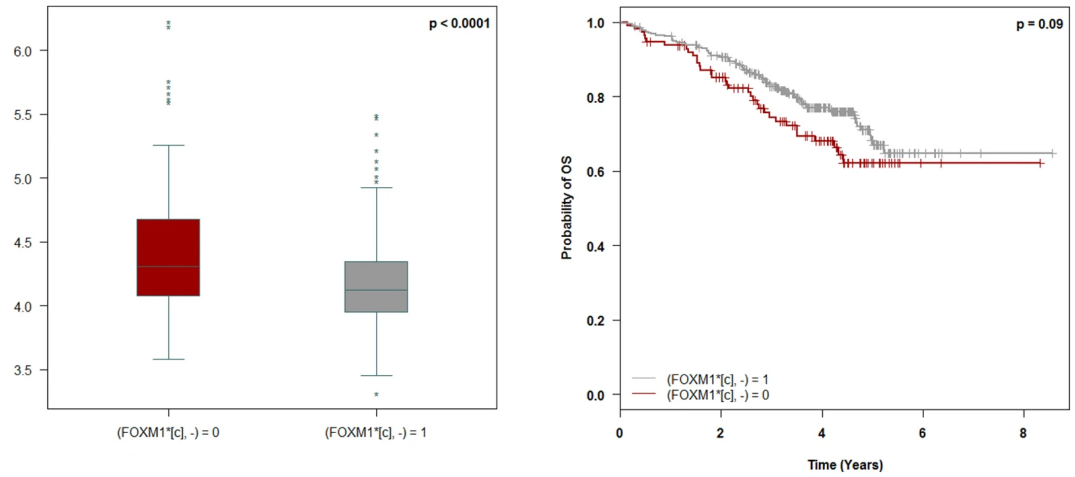



Figure 3.3: **Survival curve from Iggy's predictions applied to MM.** (Left) Gene expression of FOXM1 among MM datasets with or without the prediction (FOXM1*[c], -). (Right) Overall survival (OS) of patients with or without prediction (FOXM1*[c], -).

### 3.5.6 Improvement of the current prognostic model in MM using Iggy's predictions

**Methodology** Cox proportional hazard analyses [95] are used to estimate the effect of a treatment or feature measured on a group of patients that received it compared to a group of patients that did not receive it. This statistical analysis is used in survival analyses and composed of different terms. The $HR$ term, stands for the hazard ratio; which is the ratio of *events* reported in the group with a particular feature, with respect to the number of events repported in the group without the feature. The *events* considered in survival analyses are the number of deaths in a time unit. An $HR > 1$ means that the feature has an impact on increasing death; $HR = 1$, no impact over death or survival; and $HR < 1$, impact on survival. The *P.value* of this analysis measures the significance of the prediction model, being considered as significant a $P.value < 0.05$. Univariate analyses measure the impact of a single feature on prognosis; while multivariate analyses measure the impact of a set of variables.

**Results** Univariate and multivariate Cox proportional hazards analyses were performed on the cohort of 450 MM patients who received comparable treatment to determine the relative prognostic values of the 201 couples combining unobserved nodes and all signs $\{+, -, 0\}$ and the three strongest known prognostic variables in MM (Table 3.3); these were the translocation of chromosomes 4 and 14 (t(4;14)), the deletion in the short arm

of chromosome 17 (del(17p)) and serum 2-microglobulin $\geq 5.5mg/L$ ($\beta$ 2-microglobulin) for OS determination [96].

In the multivariate analysis, considering the five features, the estimation of $HR$ for death indicates that both (G1/S transition of mitotic cell cycle, -) and (RB1/E2F1-3/DP[n], +) were independent powerful prognostic factors (see Fig. 3.4).

| Parameters | Univariate analysis | | | Multivariate analysis | | |
|---|---|---|---|---|---|---|
| | HR | 95%CI | P.value | HR | 95%CI | P.value |
| $\beta$ 2-microglobulin, $mg/L \geq 5.5$ *vs.* $< 5.5$ | 2.03 | $1.35 - 3.05$ | 0.001 | 1.53 | $0.99 - 2.35$ | 0.056 |
| t(4,14), yes *vs.* no | 3.19 | $2.08 - 4.89$ | $< 0.01$ | 2.41 | $1.49 - 3.90$ | $< 0.01$ |
| del(17p) $> 60$ *vs.* $\leq 60$ | 4.16 | $2.53 - 6.83$ | $< 0.01$ | 3.16 | $1.80 - 5.56$ | $< 0.01$ |
| (G1/S transition of mitotic cell cycle, -), yes *vs.* no | 0.33 | $0.22 - 0.47$ | $< 0.01$ | 0.47 | $0.30 - 0.72$ | $< 0.01$ |
| (RB1/E2F1-3/DP[n], +), yes *vs.* no | 0.49 | $0.33 - 0.75$ | 0.001 | 0.58 | $0.36 - 0.93$ | 0.025 |

Table 3.3: **Parameters associated with MM overall survival**. HR stands for hazard ratio and CI, for confidence interval.

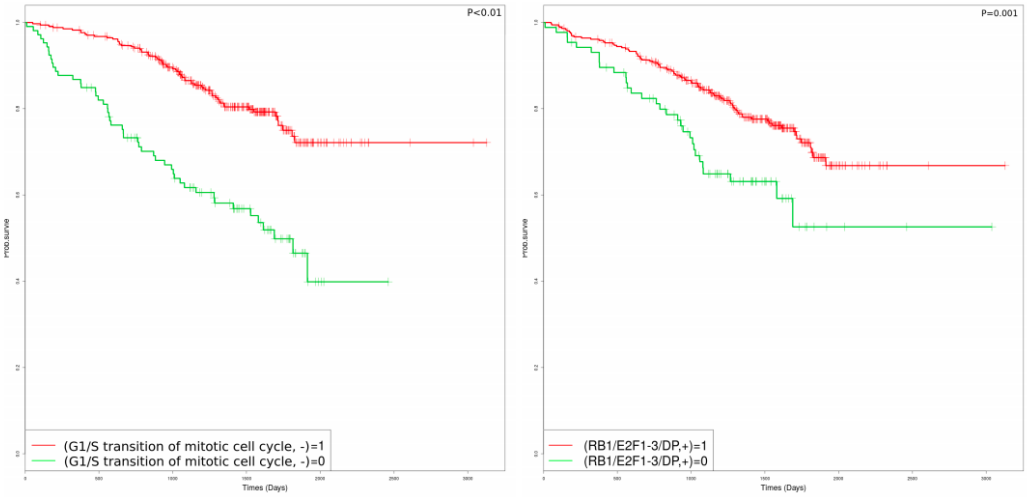



Figure 3.4: **Survival curves.** Overall survival (OS) in patients depending of inhibition of the *G1/S transition of mitotic cell cycle* node (left) and inhibition of *RB1/E2F1-3/DP[n]* (right).

## 3.6 Discussion

### 3.6.1 Multiple Myeloma sign consistency modeling

**Data discretization and graph generation** Our method incorporates both variant $\{+, -\}$ and invariant (0) expressions in its reasoning. Several pathway analysis methods,

as those reviewed in Section 3.3, use only genes that are significantly up- or down-regulated between the two classes of subjects to extract what they called signatures, *i.e.*, subsets of genes from the gene expression dataset main responsible of the differences. We believe, however, that adding information on similar expression enables us to better capture cellular behavior. Indeed, the precision of recovering 50% of the data using a two-signs $\{+, -\}$ model that receives and predicts only over- and under-expressed values, was of 48%. Thus, the two-signs model has a precision closer to a random precision distribution (50%), whereas the precision obtained using a three-signs model is farther from the random precision (43%).

Our method differs from classic pathway analysis methods in that it incorporates the notion of automatic reasoning. Within the context of MM, we are able to automatically detect repairs. These repairs are specific for each GEP and could represent cancer mutations, regulatory network incompleteness or experimental errors. The graph used in this study contains 529 genes; we can therefore observe the strong connectivity that exists among PID pathways since the total number of genes in the PID is 634. This strong connectivity is important for methods such as ours that are able to reason on the information content of the whole database. We observe, however, that the number of genes connected to cancer pathways in PID is far below the total number of human genes. This underrepresentation of regulatory knowledge is an important limitation of PID. Apart from this fact, PID-NCI includes important modeling information that identifies transcription events. This information allows us to separate gene expression from protein activity. These two parameters are not necessarily correlated, especially in cases involving phosphorylated proteins or complexes such as JUN/FOS.

**Key nodes identification** Our analysis of the predictions made by the method allowed us to identify nodes associated with a sign specific to MM compared to NPC datasets. Among these assignments, we found the inhibition of JUN/FOS[n] and FOXM1*[c]. These proteins are known to be involved in cancer in general [97, 98] and in hematological malignancies in particular [87, 88]. In the case of FOXM1, we showed that this transcription factor can represent a survival marker when its activity decreases. We can draw a parallel with the bibliography, which identifies the activation of the FOXM1 pathway as a risk factor. For JUN/FOS, our analysis identified this pathway as a potential therapeutic target but not as a survival marker. We observed that inhibition of the associated pathways has been already identified in MM patients [83, 85] and in patients with other cancers.

Moreover, this pathway is targeted in some therapeutic approaches [82, 84]. We identified two couples that improve classical prognostic models. In the case of the first couple, (G1/S transition of the mitotic cell cycle, -), we can associate this node with the proliferation pathway. The computed prognostic model showed that the prediction of inhibited proliferation can be a protective factor for MM patients. The second node, (RB1/E2F1-3/DP[n], +), was also identified as a protective factor by the 5-parameter model. This complex is known to be involved in the RB pathway, which influences cell growth pathways by regulating the initiation of DNA replication. This pathway is usually altered in cancer, leading to a loss of function[99], and current therapeutic approaches aim to activate this pathway[100].

Using the *in silico* node perturbation method, we were able to estimate the effect of perturbing a node within a particular dataset (*i.e.* single patient cancer cell). This method represents a powerful tool for analyzing the consequences of perturbations of oncogenic pathways in a given patient, especially as *in vitro* experiments are limited due to the small amount of viable myeloma cells that are obtained after bone marrow aspiration. The results of this *in silico* analysis show that activation of JUN/FOS[n] had a significant impact on 75% of MM profiles; all of these JUN/FOS[n]='+' sensible MM profiles had the prediction (JUN/FOS[n], -). In addition, activating FOXM1*[c] had a significant impact on 36.4% of the profiles; 96.8% of the FOXM1*[c]='+' sensible MM profiles had the prediction (FOXM1*[c], -). The difference in the percentages of JUN/FOS[n] and FOXM1*[c] can be explained by the graph topology and the connectivity of the individual nodes. JUN/FOS[n] is connected to eight genes through a distance of 1 molecular species; therefore, perturbing JUN/FOS[n] will impact these genes directly since they are strongly constrained by the sign of JUN/FOS[n]. On the other hand, FOXM1 is connected to 529 genes through longer paths through distances of from 4 to 77 molecular species. These genes may have other predecessors that are independent of FOXM1; this could explain why activation of FOXM1 has a strong effect on only 37% of the MM profiles. Overall, we think that this *in silico* method could be used to reinforce the choice of a therapeutic target for a specific patient profile.

## 3.7 Conclusions and perspectives

In this study, we used a specific approach to study and understand the heterogeneous gene expression profiles of approximately 600 multiple myeloma (MM) patients. Our pri-

mary goal was to provide mechanistic scenarios by identifying protein activity states of molecules that may be central to the diversity of gene expression. Our approach relies heavily on reasoning based on graphs and on changes in gene expression in the form of logical programs that combine these two types of information. The method proposed here can be summarized in the following steps. First, we obtained a directed graph, allowing us to connect significantly up-/down-regulated genes to upstream MM-related cellular receptors. Second, we confronted this graph to transcriptomic data with Iggy, which is a tool that reasons on the logic of the graph and on shifts of expression in the data so as to predict (node, sign) assignments representing the specific states of biological entities. Using two approaches of classification, we were able to identify specific assignments for MM datasets compared to NPC datasets. Finally, taking advantage of our modeling framework, we studied the effect of performing single *in silico* perturbations.

One advantage of this method is that it makes it possible to infer information about protein states from transcriptomic data by using the causal nature of the interactions as documented in PID. This can be interesting when constructing biological models and, more specifically, when developing cancer models for which proteomic data are not always available and extractable, whereas transcriptomic data are easier to obtain. Moreover, compared to the previously presented classical pathway analysis methods, we identify not only the specific biological processes that are implicated in cancer profiles but also the mechanisms associated with those phenomena. After statistically testing the quality of the method's predictions, we proposed a set of five top-scoring proteins based on their respective changes in activity in MM compared with NPC. We found the AP-1 complex and the FOXM1 transcription factor to be concomitantly inactivated in a strong majority of patients regardless of treatment or age. Interestingly, this method identified a subgroup of MM patients with increased FOXM1 activity associated with poor survival. These findings allow us to validate the predictions of our approach and show that it is feasible to individualize or restrict the analysis of multiple expression profiles to identify markers within subgroups of profiles and to identify parameters associated with survival in these subgroups. The 5-parameter model including the two predicted nodes improves the standard prognostic model in MM. In addition to its strong prognostic value, our model revealed two nodes, (G1/S transition of mitotic cell cycle, -) and (RB1/E2F1-3/DP[n], +), that are of potential biological interest in the understanding of the molecular mechanisms underlying resistance to treatment. Note that these nodes can only be predicted with the graph and coloring model, since they are a (logical) consequence of the GEP.

Our results on *in silico* perturbations of a system are also encouraging because they show that changes in the activity of the predicted proteins can serve as input information for conducting efficient perturbations.

In this work, we focused only on single perturbations, since they are more experimentally realistic. One possible perspective was to deepen the graph *vs.* gene-expression confrontation analysis so as to understand the differences between MM subgroups based on age, prognosis and other criteria. In this context, one line of research would be to study minimal subsets of perturbations. Another possible line of research would be the classification of gene expression profiles based on plausible graph-coloring models, and this perspective was explored in [39].

Chapter 4

# Logic-programs' application in personalised medicine

*"Having given up attachment to the results of action,*
*he who is ever-contented, dependent on nothing,*
*he really does not do anything, even though engaged in action."*
— Chapter 4, Verse 20, Bhagavad Gita.

## 4.1 Introduction

The following chapter recalls the research works, based on logic programs inspired by biological systems, that allowed us to distinguish different profiles of patients. Different from classical ways of classifying patients' omic data, our methods use biological networks. The collaboration with several people has contributed to the methods and results presented here. I may mention: Misbah Razzaq, Lokmane Chebouba, Pierre Le Jeune, Bertrand Miannay, Dalila Boughaci, and Jérémie Bourdon. I am greatly thankful for the energy and interest put to explore this research area, which was in most of our studies not the main goal of the research project, but paths we ended by exploring guided by interesting datasets provided by the DREAM challenges community. These Reverse Engineering challenges proposed high quality Human omic datasets in concrete problematics targeting cancer research to the methodological community. We have carefully studied three datasets from these challenges: (1) Multiple Myeloma, (2) Acute Myeloid Leukemia, and (3) Breast Cancer. The two first datasets will be presented in this chapter in respectively Sections 4.3 and 4.4. The third one will be presented in Chapter 6. For all of them we have proposed methods based on logic programs. A research article presenting these three works together was published in [101]. Many of the paragraphs written here are taken from that publication.

## 4.2 Regulatory and Signaling networks as Logical programs

The methods described in the following sections are mainly implemented in Answer Set Programming (ASP) [37]. This declarative programming approach allows us to express a problem in the form of a logic program (LP). The syntax of ASP is close to Prolog's syntax because the grammatical structure of both LPs rules expresses a logical implication from the right terms of the rule towards the left terms of the rule. However, ASP semantics, allows a different type of solving mechanism. While in Prolog there is an inference process to search for an answer to a query, ASP programs allow to find all (Herbrand stable) models satisfying all the LP rules. ASP semantics allows declaring variables and domains, as well as imposing constraints and solving global optimizations. It is close to SAT (propositional satisfiability) and is typically used in the study of the solution space of combinatorial search problems. A small example of a LP in ASP is presented in Chapter 2, Section 2.3.3.

In the following sections we review and discuss two selected methodologies we have proposed to understand medical data using models. The models presented here are of discrete nature, implemented as LPs in ASP. In the first approach (Section 4.3), we built a model integrating gene regulatory networks and experimental observations as facts in a LP interpreted by checking the satisfiability of a constraint named *perfect coloring*. In the second (Section 4.4), we used Boolean networks to model the fact of reproducing the experimental observations with minimal error. While the perfect coloring model is applied for large-scale (thousands of components) networks and gene expression observations, the Boolean models are applied on middle-scale case studies (hundreds of components) using either proteomics data measured across several patients or multiple perturbation time-series phosphoproteomics datasets measured across cell lines.

## 4.3 Multiple Myeloma

### 4.3.1 Motivation and background

Patients suffering from cardiovascular, inflammatory, oncology, infectious, and neuropsychiatric human diseases present a vast heterogeneity in their genome and gene or protein expression profiles. In order to study the underlying mechanisms that explain

these profiles, regulation networks, that summarize the interactions between gene, proteins or metabolites in the cell, are particularly meaningful. Regulatory networks may not necessarily be wired in the same way for two different individuals. Therefore, a concrete treatment may not show the same effect in all patients and in some cases, such as cancer for example, it can encourage disease progress. Classical medical approaches to treat disease provide fixed protocols of treatment independent of the patient's heterogeneity. Systems Medicine is a recent field of research that proposes disease regulatory networks as explanations on how the genes or protein express in an individual. In this way, network analyses shall provide a disease molecular signature that can be connected to clinical observations. While past diagnosis methods focused only on measuring single parameters, the premise of Systems Medicine is to perform multi-parameter analyses that may result on a more plausible explanation of disease. This research field is reinforced by the fact that current technology allows us to measure the state of several species in these regulatory networks in a high-throughput fashion. In this context, we review and discuss here the published research works to compute disease signatures in computational models, built from patients gene or protein expression profiles.

The exponential increase of biological data (genomic, transcriptomic, proteomic) [102] and of biological interaction knowledge in Pathway Databases favors the modeling of cellular regulatory mechanisms. Modeling biological mechanisms can be done by using boolean or ordinary differential equation representations. Those approaches have shown their efficiency in cellular phenomena study [103], disease research [76, 99], and bio-production optimization [104]. However, those modeling approaches cannot take into account the large amount of OMIC data. This limitation requires that the researcher preselects the OMIC data and network, adding bias to the analysis [105]. In this chapter, we review a modeling approach named *perfect coloring* that we published in [39], which is based on exhaustive and global graph coloring approaches such as Iggy [44] (see Chapter 2). The perfect coloring modeling approach is complementary to Iggy in the sens that it looks for *harmonious* or *perfect* coloring models. We will illustrate here how this method was used for Multiple Myeloma (MM) understanding and patients prognosis classification. MM is a hematologic malignancy representing 1% of all cancer [106] with a survival rate of 49.6% after 5 years described in more details in Section 3.2.

### 4.3.2 Methods

#### Perfect coloring model

This methodology was introduced in [39]. The main steps of this method are presented in Fig. 4.1.

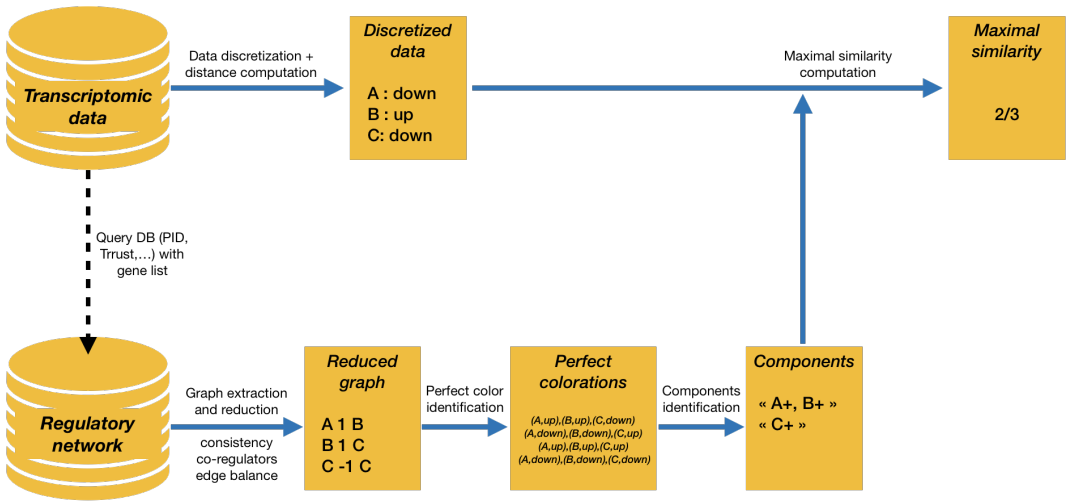



Figure 4.1: **Overview of the perfect coloring modeling framework.** Arrows refer to processing steps and yellow boxes refer to input/output data.

This method works exclusively on the discrete model underlying a regulatory network, and avoids preprocessing the experimental data. The analysis of such a model will predict subgraphs or graph components that are independent according to the up- or down-regulated coloring of their nodes. The input of this method is a graph $G(V, E)$ composed of a set of nodes $V$ and edges $E$; where an edge is a tuple with 2 nodes (source and target), a sign (1 for activation, -1 for inhibition) and a weight. Such graphs can be obtained from publicly available databases, such as the Pathway Interaction Database [107] and Trrust [108] by querying a predefined list of genes. The perfect coloring approach can be summarized in 4 steps:

1. Reduce a large-scale graph by restricting the search space associated to this graph. For this, three graph operations are applied that remove *redundant* nodes or paths. These operations are applied successively over the graph prior to the perfect coloring ASP solving. They identify molecular-species nodes that will be merged in a subcomponent-node. Subcomponents are derived through topological reductions, based on specific graph patterns detections. Molecular-species nodes that belong

to a subcomponent will be correlated to each other, and can also be correlated to molecular-species nodes belonging to other subcomponents. Therefore, a component, such as defined in the Step 3, can be composed by different (topological) subcomponents. The first and second reduction methods identify subcomponents. Aggregating molecular-species nodes within subcomponent nodes reduces the number of nodes in the graph. The third method reduces the number of edges and detects components which are isolated of the rest of the graph.

2. Once the size of the graph was reduced, the next step is to enumerate all the possible ways to color the graph in a perfect (or harmonious) way. In a colored graph all nodes will be associated a sign: "+" standing for up and "-" for down. These signs refer to the qualitative variation that can be experimentally measured in a molecular species of the graph when comparing 2 cellular states, for example after and before a stress condition. In this work we were interested on modeling sets of possible state variations of the graph nodes that satisfy a *perfect coloring* constraint. The intuition behind this constraint is to point to network discrete variation states that maximize the agreement between a target molecular species *up* or *down* variation and the positive or negative influence from its regulators in the graph. The perfect coloring constraint can be expressed in natural language as follows: "for a given node in the graph we impose that its discrete up or down-regulation is explained by a similar (positive or negative) influence from a maximal number of direct predecessors". This statement is inspired from a hypothesis of redundancy in biological networks control, and we use ASP to express this statement and search for coloring models where this property holds for every node in the graph.

   *Perfect coloring models in ASP.* In the following we introduce a fraction of the logic program, written in Clingo 4 syntax, used to identify perfect colorings in a graph. To understand this program, one needs to assume that previous predicates are previously defined to state a specific network (signed and oriented graph), represented with `node/1` [1] and `edge/3` facts. Also a coloring model is previously introduced by associating in an *exhaustive fashion* each node of the graph with one of the two possible colors (up-regulated, "+" and down-regulated, "-"). In ASP this type of constructs are called *choice rules*, and generate all possible solution (coloring models) candidates. For each possible coloring model solution, a different

1. The "/" expresses the predicate arity.

valuation of each node $X$ with a specific sign $S$ will be generated. These solution candidates are represented by the predicate `col(X,S)`. The following logic program computes the coloring models with the minimal imperfections.

```
1 imperfectCol(X) :- col(X,S1),  col(Z,S2), edge(Z,X,1), S1!=S2.
2 imperfectCol(X) :- col(X,S1),  col(Z,S1), edge(Z,X,-1).
3 #minimize{Z : node(Z), imperfectCol(Z)}.
```

In lines 1 and 2 we identify if node $X$ is associated to an imperfect coloring model. The predicate `imperfectCol/1` will *mark* all nodes in the given graph, for a given coloring model solution, associated to an imperfect coloring. An `imperfectCol` predicate is assigned to a node $X$ in 2 possible situations (lines 1 and 2). First, line 1, when the given color of node $X$ is a sign $S1$ (`col(X,S1)`), and one of its direct positive regulators in the graph (node $Z$, `edge(Z,X,1)`), is colored with a color $S2$ different than $S1$. Recall that there are only 2 colors allowed in this model. The second case, line 2, expresses the case where $Z$ is an inhibitor of $X$ (`edge(Z,X,-1)`) and its color (given by the variable $S1$) is the same as X's color. Finally, we express on line 3 that we search to minimize the number of predicates `node(Z)` in which $Z$ is associated to an imperfect coloring model.

3. Among the possible coloring models that satisfy the *perfect coloring* logic program, many of them can be clustered together on account of the symmetry of our approach created by the duality of our knowledge representation: positive-negative influence (edges), up- or down-regulation of molecular species states (nodes). A *component* is defined as a set of molecular-species nodes which are color-dependent or color-correlated. That is, by fixing the color of one molecular-species node in this component, the colors of the other molecular-species nodes can be established so that the perfect coloring constraints hold. Given a graph, it is possible to identify its entire set of components by using ASP constraints or by building a correlation matrix from the perfect coloring models obtained in Step 2 for each couple of nodes.
4. The last step consists on measuring how the up or down-regulation coloring of the nodes in the graph perfect colored components compare to the experimental data. As shown in Fig. 4.1, this comparison can be done without discretization of the experimental data, by measuring a distance between the discrete coloring and the continuous data.

Step 1 is implemented on Python 2.7, step 2 on ASP (clingo 4.5.4), and steps 3-4 on R and Python 2.7. A usage example and the sources of this method are publicly available

at [109].

**Patient classification**

Given a patient gene expression profile (GEP) and given a regulatory network $G$, the perfect coloring approach described above can propose a similarity vector of size $k$, where $k$ is the number of components identified for $G$. This similarity vector is specific to the patient expression profile, and could be understood as a vector of features. Given a cohort of patients, in which each patient is assigned a *good* or *bad* prognostic label, we can use machine learning techniques to learn a classifier from the similarity feature vectors of all patients in a training database. When a new patient arrives, this classifier can predict the patient's good or bad prognostic according to the training patient set. We have implemented a software named IGUANA publicly available for Windows and Mac OS in which such classifier is built using XGBoost (Extreme Gradient Boosting). A complete user guide and use case examples are provided online at [110]. Our objective here was to provide the complete framework via a user-friendly human interface.

### 4.3.3 Case studies

**PID-NCI network** The perfect coloring method was applied to transcriptomic data from myeloma cells (MC) of 602 MM patients and from normal plasma cells (NPC) of 9 healthy donors (see Appendix A.2). We used the PID-NCI database to generate a graph by extracting the downstream events from three signaling pathways (IL6/IL6-R, IGF1/IGF1-R and CD40) to significantly differentially expressed genes of the patients profiles. The obtained subgraph from NCI-PID 2012, contained 2269 nodes, 2683 edges and connected 529 differentially expressed genes. The perfect coloring method identified 16384 coloring models, grouped in 15 components or subgraphs (see Fig 4.2). One of these components (422 nodes and 167 genes) was found statistically specific to MC in comparison to NPC. Using gene ontology enrichment analysis with PANTHER [111] we were able to associate this component to oncogenic phenomena.

**TRRUST network** The perfect coloring approach and the classifier were applied to the data of the Multiple Myeloma DREAM challenge [2]. The objective of this challenge was to classify the MM patients labeled as high risk. They provided to the methodological

2. `https://www.synapse.org/MultipleMyelomaChallenge`

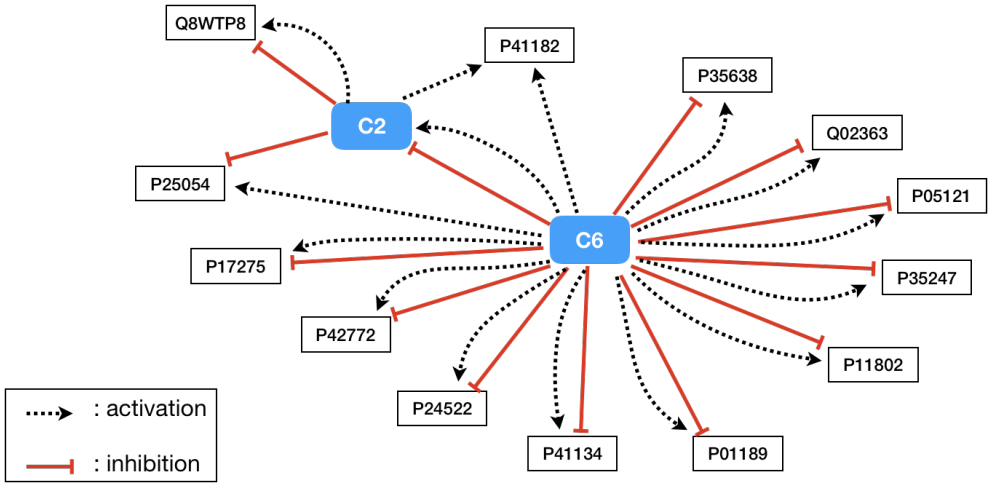



Figure 4.2: **Components identification by perfect coloring approach**. The 15 components identified from all the perfect coloring models generated from the PID-NCI database (2269 nodes, 2683 edges). The components composed only of one gene are labeled with the Uniprot identifier.

community large MM patient cohorts (25000 patients) where patient gene expression profiles and risk information were measured by different US laboratories. We tested our method with 2 sets of gene expression profiles: HOVON (GSE19784, 274 GEPs) and UAMS (GSE24080, 558 GEPs). The graph was a gene regulatory network generated with the Trrust database [112] by querying the significantly expressed genes in the intersection of both datasets. The graph of 447 nodes and 600 edges, was reduced to 30 components with the perfect coloring approach. After this, we applied XGBoost to learn a classifier from the HOVON dataset to predict the high or low risks of patients associated with the UAMS dataset and vice versa, and obtained precision rates of 0.75 and 0.71, respectively. Our precision rate was not satisfactory when comparing it to the one obtained by the other teams participating in the DREAM challenge using gene expression profiles provided by different research institutes, other than HOVON and UAMS. We believe our method is very sensitive to the initial graph; it is important that this graph contains all the significantly expressed genes across all GEPs provided by all the research centers. We were unable to verify this since for this DREAM challenge in particular the testing data was not made available to the community.

Finally, this approach can be used to study divergences among the datasets provided by different experimental platforms or in this case by different research laboratories. Such study is crucial to check if multiple datasets can be merged in order to create a larger

one. A large set would provide more training examples for the perfect coloring model, and this would certainly improve its accuracy. For this, we calculated the expected value as well as the standard deviation for the distributions of similarity scores for each of the 30 components across both sets of profiles (HOVON vs. UAMS). We observe that 7 out of 30 distributions have an expected value of the similarity score at a distance equal or greater than 0.07, such as component 7 for example (see Fig. 4.3). This means that we can identify regulatory mechanisms within the network pointing to regions where the experimental data provided diverges. Note that in this analysis, we supposed that the similarity scores of each component are normally distributed, so that we are able to plot their distributions and compare them. Similarity scores are linear combinations of gene expression levels and they will be normally distributed if and only if all gene expression levels can be modeled as independent random variables normally distributed.

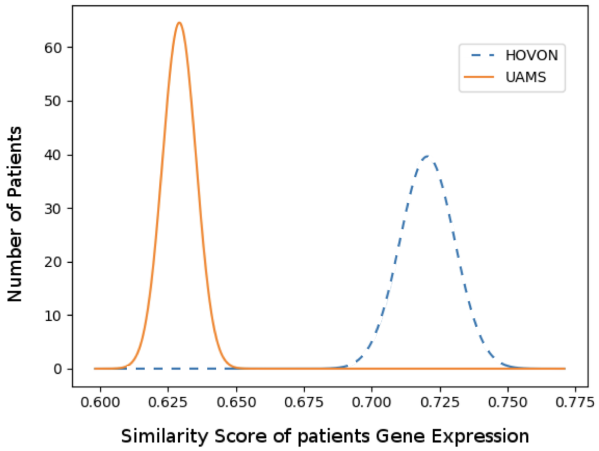



Figure 4.3: **Distribution of similarity score from the perfect coloring approach across two expression profiles of two different patient cohorts (UAMS and HOVON) for the same graph component**. The perfect coloring method detected 30 components in the graph obtained from the Trrust database using the differentially expressed genes of the gene expression profiles (GEP) of 2 research centers (UAMS, HOVON). These GEPs were provided by the Multiple Myeloma DREAM challenge. The similarity scores of each patient with respect to the genes of the component are shown in the x-axis and represent how the perfect coloring values from the component match the continuous data of the GEPs provided by both independent platforms.

## 4.4 Acute Myeloide Leukemia

### 4.4.1 Motivation and background

Patients' response classification is usually approached by methods that find statistically significant markers from the transcriptomic or proteomic data at hand. A classical method used for this is univariate and multivariate Cox proportional hazards analyses. Following such approach, several statistic [113, 114] and machine learning [115, 116, 117] methods conceived for significant features extraction have been applied to this problem. More recent approaches include the notion of pathways in this drug detection problem [118]. Such methods allow identifying the regulatory mechanisms related to the best drug targets [119] and this mechanistic information is valuable to understand the disease and the complexity of drug targeting. We have introduced in [120] the *caspo* method, which learns Boolean networks (BNs) from phosphoproteomic multiple perturbation data by using logic programming. Phophoproteomic data measures protein phosphorylation or protein abundance. It can be obtained in a high-throughput fashion for tens of proteins under different cases of perturbations (*e.g.* stimulations and inhibitions) of the biological system. This framework allows us to retrieve families of BNs having the best fit to the experimental data from exhaustive searches over a large-scale Prior Signaling Network. In Chapter 5 *caspo* methodology will be explained in detail. In this work we review a method that allows *caspo* to handle patients data. In fact, *caspo* needs as input data proteins measurements across multiple perturbations. While such datasets are possible to obtain for cell lines, they are impossible to obtain for patients. However, by preselecting partial measurements of the complete patients dataset, we may retrieve cases where the protein observations behave as if they were perturbed in a same way for different treatment response classes of patients. We discuss here how this approach is suitable to find the mechanisms differentiating the complete remission and primary resistant responses of Acute Myeloid Leukemia patients. The patients' data was obtained from the Acute Myeloid Leukemia DREAM 9 challenge.

### 4.4.2 Methods

#### Discovering Boolean networks distinguishing different classes of patients data

In this section we review a method [121] based on ASP and *caspo* to predict the Boolean models associated to patients holding separate diagnostics: complete remission

(CR) and primary resistant (PR). This method receives as input information a Prior Knowledge Network (PKN) and an experimental dataset consisting of protein measurements associated to several patients. It consists of four steps (see Fig. 4.4).

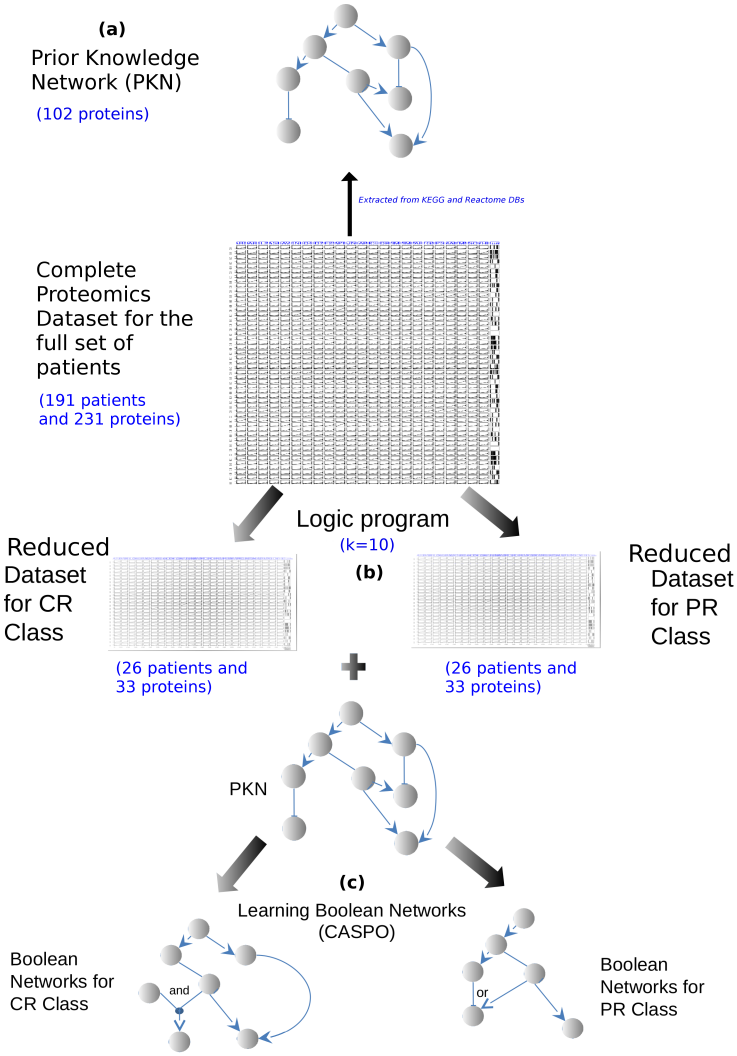



Figure 4.4: **Workflow of our method.** (a) PKN construction. In this step we pass the proteins present in our DREAM 9 dataset as input to the Cytoscape plug-in Reactome FI to construct the PKN. This plug-in finds all the paths between the input proteins across several databases, after that we select only relations coming from KEGG. (b) Protein and patient selection. This step consists on selecting $k$ proteins from the dataset for which there is a maximum number of pairs of patients that have identical values in the $k$ proteins but that belong to different response classes. (c) Learning. This step consists on finding the BNs for the two classes CR-PR corresponding to the two datasets obtained in step (b).

1. *Creation of a PKN.* We used public databases to connect the measured proteins. The PKN is composed of 3 types of nodes: stimuli are the entry of the network (nodes without predecessors), readouts are the output of the network (nodes without successors) and inhibitors are proteins in between the entry and output network layers.
2. *Protein and patient selection.* This step executes a logic program implemented in ASP that selects a group of $k$ stimuli and inhibitor proteins that maximize the number of pairs of patients for which the binarized values of their experimental measures matched in both classes (CR, PR) and where the difference of readouts measures for each class is maximal.
3. *BNs learning.* We used the dataset issued from step 2 to learn BNs with *caspo* [122]. This step produces two families of BNs, one for the CR class and the other for the PR class. Our objective here was to learn different families of BNs by using the identical stimuli-inhibitor cases and the maximal difference of readouts measures for each class and finally compare the structure and mechanisms between these BNs families.
4. *Classification.* The set of unseen patients was classified by using our learned logic models. For this we computed the Mean Square Error (MSE) between measured readouts and predicted readouts for each patient in the testing data based on the two families of the previously learned BNs. The given patient will be classified in the class with the lower MSE.

### 4.4.3 Case study - Acute Myeloid Leukemia

In 2014 the DREAM 9 challenge [3] was launched in order to predict the complete remission (CR) and primary resistant (PR) response to chemotherapy of 191 Acute Myeloide Leukemia (AML) patients from their proteomics data (231 measured proteins) and from 40 clinical data [123]. We describe here how we applied the method sketched in Fig. 4.4 to the DREAM 9 challenge dataset. First we create a PKN composed of 102 nodes (17 stimuli, 62 inhibitors and 23 readouts) connected by 294 edges. The second step of our method, allowed us to select a subset of $k = 10$ proteins extracted from the union of the stimuli and inhibitors present in the PKN (79 proteins), the chosen $k$ maximized the number of pairs of patients belonging to the CR and PR classes. Then we learned the 2

3. `https://www.synapse.org/#!Synapse:syn2455683/wiki/64007`

families of BNs using the reduced dataset from the previous step. The CR family had 10 BNs, while the PR one had 9. The size (number of logic clauses) of the optimal BNs for the CR case was of 24, while it was of 29 in the PR BNs (see Fig. 4.5). When comparing both networks, we can see that the normal growth factor - fibronectin - PI3K pathway in primary resistant patients is better connected to other network components (see yellow node in Fig.. 4.5). This finding suggests an important rewiring of the PI3K pathway in primary resistant patients compared to complete remission ones. This goes in agreement with previously published literature on AML treatment by targeting the PI3K pathway [124].

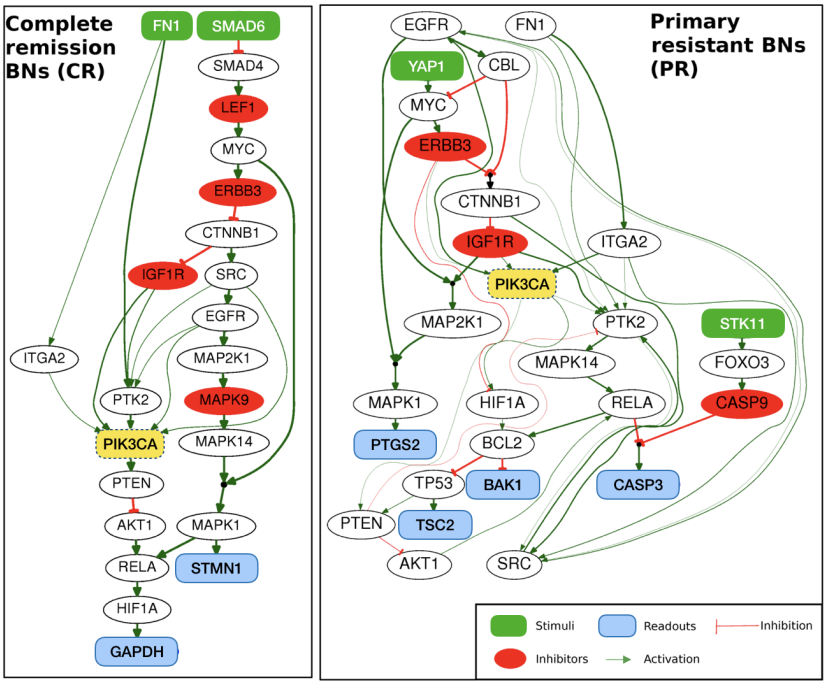



Figure 4.5: **Union of optimal BNs learned from the initial PKN and the reduced patients dataset from the complete remission (CR) and the primary resistant (PR) classes**. The thicker edges represent those that are the most frequent paths in the BN family. The association between a node and its predecessors is an AND gate if it is preceded by a filled black circle and an OR gate otherwise. Left: Boolean networks for CR patients. This BN can explain and predict the measurements of readouts STMN1 and GAPDH starting from the stimuli FN1 and SMAD6, passing by the inhibitors LEF1, ERBB3, IGF1R and MAPK9, and other intermediate proteins. Right: Boolean networks for PR patients. This BN explains and predicts the measurement of readouts PTGS2, TSC2, BAK1 and CASP3 starting from the stimuli FN1, YAP1 and STK11, passing by the inhibitors ERBB3, IGF1R and CASP9, and other intermediate proteins.

## 4.5 Conclusions

In this chapter we have presented the results of mainly 2 publications: the *perfect coloring* sign consistency approach [39] and a method to learn Boolean Networks distinguishing omics data of patients having different prognostics [121]. The results from these studies destined to elucidate computational models from patients datasets. Our goal was to make sense, from a mechanistic perspective, of the underlying differences among different classes of patients. Rather than using statistical and machine learning methods applied only to the experimental proteomics or transcriptomic datasets, we have included a general Prior Knowledge Network component by using publicly available repositories. Adding PKN information, allowed our methods to propose specific (patient risk or prognostic) outputs associated to molecular species, as subgraphs or Boolean models. We believe that this mechanistic information is a powerful predictor of disease, complementary and comparable to bioclinical markers as we could proof in [47] for Myelome Multiple patients. All of the proposed methods are based on logic programming, mainly on Answer Set Programming. These methods are publicly available and we have referenced through out this chapter the git repositories where the related softwares are available.

Chapter 5

# Learning Boolean Networks

*"You don't have to understand Life's nature,*
*then it becomes a grand affair.*
*Let every day just of itself occur*
*like a child walks away from every wind*
*and happens upon the gift of many flowers.*
*To collect and the blossoms spare,*
*that never enters the child's mind.*
*She gently unties them from her hair,*
*where they were kept captive with such delight,*
*and the hands of the loving, youthful years*
*reach out to embrace the new."*
— Reiner Maria Rilke.

## 5.1 Introduction

The following chapter summarizes a method we have proposed to learn Boolean networks from phosphoproteomics experiments, where the protein expression is observed upon multiple perturbations of the system. This work [125] conducted us to propose *caspo*. The *caspo* system, conceived in Answer Set Programming, was inspired by *CellNOpt* [126], where the search of optimal model was conceived using genetic algorithms. *caspo* has inspired other works we have conducted later, such as [33], [127], or [121] (see Chapter 4). We have presented a newer version of it in [128] integrating multiple functionalites. *caspo* continues inspiring similar works such as [38, 17, 129]. Applications of this system naturally present to us when elucidating the hidden biological mechanisms in a mass of genome-scale data. This is currently the case with the thesis research subject of Mathieu Bolteau, a PhD student that I currently co-supervise. His research subject is in the domain of human embryo development.

The following chapter is organized following mostly the results we presented in [46] (see Annex A.3). *caspo* research, has been possible thanks to the effort of all co-authors on

our *caspo* publications. I may name, specially, because of the impact of their contribution: Santiago Videla, Anne Siegel, Julio Saez-Rodriguez and Irina Konokotina.

## 5.2 Motivation and background

Predictive models of biological networks are a main component of systems biology. For a certain system of interest, if enough information is available about the biomolecules that constitute it and their interactions, one can convert this prior knowledge into a mathematical model (*e.g.* a set of differential equations or logic rules) that can be simulated. If experimental data is available, the model can be fitted (trained) to the data. That is, one determines the model parameters (for example, kinetic constants in a biochemical model) to obtain the most plausible model given the data. This is normally achieved by defining an objective function which describes the goodness of the model based on the data that is subsequently optimized [130].

This training process is not a trivial task due to factors including experimental error, limitations in the amount of data available, incompleteness of our prior knowledge, and inherent mathematical properties of the models. Thus, in general, there is no single solution but rather multiple models that describe the data equally (or similarly) well. In those cases, the model is said to be non-identifiable [131, 132].

In some cases, deterministic methods that guarantee the identification of the optimal models can be applied, but these methods are often limited by the exponential growth of the search space. Thus, usually one needs to use stochastic methods that may identify the optimum or at least exhibit sub-optimal models [130]. However, an incomplete characterization of the set of plausible models limits significantly the insight that can be gained about the underlying molecular mechanisms.

In this Chapter, we investigate this issue in the context of logic modeling of signaling networks. These models have been applied to analyze signal transduction in a variety of contexts [133, 134]. In particular, given a network encoding our knowledge of signal transduction and a dataset measuring the activation of proteins in this network upon various perturbations, one can derive from the network (Boolean) logic models fitted to the data. Models are simulated assuming that the network reaches a pseudo steady-state at a certain time upon stimulation, and the identification of the network that best fits the data is posed as an optimization problem. This problem can be solved using meta-heuristics (*e.g.* a genetic algorithm), and their application suggests that there are

multiple alternative models that explain the data [29]. However, stochastic search methods cannot characterize the models precisely: they are intrinsically unable not just to provide a complete set of solutions, but also to guarantee that an optimal solution is found. To overcome this limitation, approaches based on Integer Linear Programming (ILP) [135, 136] and ASP [137] have been applied, providing a proof of concept that a global optimum can be identified.

In this chapter we present *caspo*, a free open-source tool to learn (Boolean) logic models of signal transduction in a complete and global fashion. *caspo* uses CellNOpt pre- and post-processing routines [138]. It can handle feedback loops in the prior knowledge network, numerical datasets, and tolerance in the score due to experimental uncertainty. We use *caspo* to exhaustively explore the space of optimal and sub-optimal models for a real case describing pro-growth and inflammatory pathways in a liver cancer cell. We find that, even with small tolerance, thousands of models can be compatible with the data and use ASP's flexibility to further analyze them: we categorize them according to their input-output behavior and identify subsets of modules that are interchangeable with respect to the score. The multiple possible combinations of these modules are responsible for the large number of models found.

## 5.3 Methods

### 5.3.1 Learning Boolean logic models

Our prior knowledge about signal transduction can be described as a set of causal interactions among the biomolecules involved (mostly proteins) that can be mathematically formulated as a signed and directed graph. We call this graph the PKN. In such a graph, one can denote as *input* nodes those that can be stimulated or inhibited experimentally. When the system is perturbed by fixing the state of such nodes, one can measure the activity of each *output* node being observed. Such measurements are typically given by *phospho-proteomics datasets* consisting of measurements over $m$ proteins under $n$ experimental conditions. With $\theta_{ij} \in [0, 1]$, we denote the activity of a protein $j$ under the experimental condition $i$, where $0 \leq i \leq n$ and $0 \leq j \leq m$. In agreement with experimental errors, we used a discretization procedure so that $\theta_{ij} \in \{0, \frac{1}{100}, \ldots, \frac{99}{100}, 1\}$.

The state of nodes after a perturbation of the system cannot be predicted using only graph theory. However a simple framework is given by Boolean logic models [19]. In a

logic model, activation of nodes is defined by a set of operators. We use the representation known as sum of products (SOP; also called disjunctive normal form) which uses only AND ($\wedge$), OR ($\vee$), and NOT ($\neg$) operators. A simple form to encode logic models based on the SOP formalism is using hypergraphs [19]. A directed and signed *hypergraph* $H = (V, E)$ is a generalization of a directed and signed graph $G = (V, A)$, where $V$ is the set of nodes and $E$ the set of *hyperedges*. While edges in $A$ connect pairs of nodes $a, b \in V$, *hyperedges* in $E$ connect pairs of *sets of nodes* $S, T \subseteq V$. To describe a logic model as a hypergraph, each SOP expression is mapped to a set of hyperdges.

The PKN is first compressed to simplify the structure [29]. Then, since the exact logic gates are often not known, we perform an *expansion* to generate all possible gates compatible with the PKN. Mathematically, we derive a hypergraph $H = (V, E)$ from a graph $G = (V, A)$, so that for every signed hyperedge $(S, \{t\}) \in E$ and every $s \in S$, there exists an edge $(s, t) \in A$ having the corresponding sign.

Let $H$ be a hypergraph describing a logic model and $(\theta_{ij})_{i \leq n, j \leq m}$ be a phosphoproteomics dataset. For each experimental condition $i$, we can compute the Boolean prediction $\rho_{ij} \in \{0, 1\}$ of the state of a protein $j$ by using the logic formulas described by $H$. This corresponds to computing the (quasi) steady-state of the system. These simulated values at a quasi-steady state are considered an approximation of the state of the cell immediately after a perturbation and can be thus compared to experimental values obtained at early times after stimulation [19].

Then, the *fitness of the logic model* to the experimental dataset is obtained by comparing experimental observations, normalized between 0 and 1, to Boolean predictions based on the MSE as follows: $\frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} (\rho_{ij} - \theta_{ij})^2$.

**Combinatorial optimization problem.**

The problem of learning Boolean logic models that we address in this work consists of finding minimal hypergraphs derived from the PKN that minimize the MSE where the size of a hypergraph $H$ is the sum of cardinalities of each hyperedge source (*i.e.*, the sum of the number of inputs): $\sum_{(S,T) \in E} |S|$. Thus, the problem can be formulated as a lexicographic multi-objective optimization where the first objective is to minimize MSE, and the second objective is to minimize size. Our prior assumption that $\theta_{ij}$ belongs to a finite set of values implies that this problem is of discrete nature. Further, the optimization can be relaxed by using different degrees of tolerance over the optimum for each objective, *i.e.*, MSE and size.

**Global Truth Tables.**

Inspired by truth tables in propositional logics, we introduce the concept of GTT as a way of describing the input-output behavior of a Boolean logic model. For a given logic model, we can compute its predictions on observable *output* nodes in response to every possible experimental condition on *input* nodes. Comparing GTT allows one to decide whether two logic models, regardless of their structures, are experimentally distinguishable or not. Furthermore, GTT provide a way of grouping a large number of logic models according to their input-output behavior to facilitate the analysis.

### 5.3.2 Learning Boolean logic models with ASP

Recall that ASP is a declarative problem solving paradigm from the field of Logic Programming combining several computer science areas [37, 41]. As a full declarative paradigm, instead of telling a computer *how to solve the problem*, with ASP one defines *what the problem is* and leaves its solution to the solver. These solvers are based on Boolean constraint solving technology, and they can solve hard discrete combinatorial search problems, with comparable results to ILP.

The distinct feature of ASP is its rich modeling language, making it popular as a tool for declarative problem solving. Sophisticated preprocessing techniques (*grounding*) are required for dealing with this rich language. Thanks to the development of an ASP language standard, its expressiveness and powerful solvers, ASP has been widely used in many fields of computer science for a decade. Quite recently, the capability of solvers has increased such that ASP started to be applied to solve hard combinatorial problems arising in bioinformatics and systems biology. Applications include expanding metabolic networks [139], repairing inconsistencies in gene regulatory networks [14], modeling the dynamics of regulatory networks [140], inferring functional dependencies from time-series data, [141], integrating gene expression with pathway information [142], and analyzing the dynamics of reactions networks [143].

We used the freely-available ASP grounder *gringo* and solver *clasp*, both included in the Potsdam Answer Set Solving Collection [1]. Importantly, we relied on the capability of the solvers to handle multi-criteria optimization in order to guarantee the global optimum by reasoning over the complete space of solutions. Several reasoning modes (enumeration, union and intersection) were also necessary to complete the combinatorial study of the

1. http://potassco.sourceforge.net/

the family of feasible solutions.

### 5.3.3 Software: caspo

We have implemented *caspo: Cell ASP Optimizer*, a Python package which combines PyASP [2] and CellNOpt [3] to provide an easy to use software for learning Boolean logic models (Fig 5.1). The software is freely-available for download and also as a web service through the Mobyle framework [144]. PyASP encapsulates the main ASP tools, *gringo* and *clasp*, into Python objects. These objects can be fed with logic programs describing different tasks, be launched with dedicated parameter settings, and return the ASP results for further processing. CellNOpt ([138]) is a software for training logic models using different formalisms (Boolean, Fuzzy or ODE). The software allows us to import and preprocess a PKN, normalize experimental data, train logic models to data using heuristic methods, and postprocess and visualise the resulting models. CellNOpt is written as a set of R packages available on Bioconductor and as a Cytoscape plugin (CytoCopter), and it can used within Python using the package *cellnopt.wrapper*.

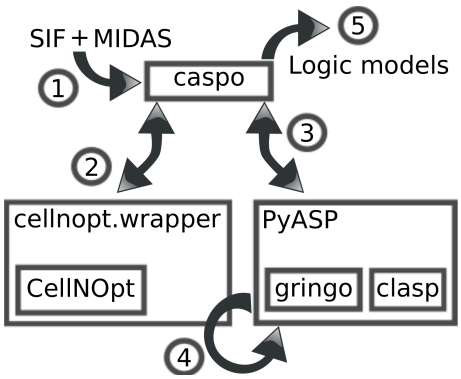



Figure 5.1: **High-level design of caspo.** (1) Inputs file are a PKN in Cytoscape's SIF format, and a dataset as a CSV file in the MIDAS format. (2) Preprocessing routines by CellNOpt. (3) Finds an optimum model. (4) Finds all models within the tolerance. (5) Outputs all models found.

2. http://pypi.python.org/pypi/pyasp
3. http://www.cellnopt.org/

## 5.4 Results

To illustrate the use of *caspo*, we use a pro-growth and pro-inflammatory model in liver cells. The model is trained to phospho-proteomics data generated in the liver cancer cell line HepG2. Data is generated upon perturbation with combination of ligands and small-molecule inhibitors blocking the activities of specific kinases [145]. The dataset contains measurements using the Luminex technology of 15 species under 64 experimental conditions. This model was introduced in [29] and here we use a variation from [146]. In this case, there are 130 possible hyperedges and thus, the number of possible logic models (*i.e.* search space of the combinatorial optimization) is given by $2^{130}$.

### 5.4.1 Family of Optimal Models

We first used *caspo* to compute all global optimum solutions to the optimization over our case-study. We found 16 Boolean logic models with minimal score in 0.36 seconds (see Fig. 5.2). All models having the same fitness to data (MSE=0.0499) and size (28). Moreover, the same 16 logic models were found (0.5 seconds) using an extended PKN with feedback loops from [138]. Cross validation analysis showed no significant difference in the optimum MSE with respect to the complete dataset.

The 16 different models arise due to four pairs of sub-models (modules) equivalent in terms of score. These modules represent alternative ways to activate specific nodes and are independent from each other. For each pair, only one of the modules appears in a given model; that is, they are *mutually-exclusive*. Thus, selecting either member of each pair provides an optimal model and all possible combinations give rise to the $2^4 = 16$ models. To elucidate the differences between the 16 models from their responses to all possible experimental conditions, we computed and compared their GTTs (Global Truth Tables; see Section 5.3.1). Interestingly, they all have the same GTT. That is, for any combination of input nodes (stimuli and inhibitors), the same values are predicted for all the readouts by the 16 models. Therefore, the optimization reports a single solution in terms of input-output behavior, despite the fact that this solution can take the form of any of the 16 models. In order to distinguish among these models (and thus determine which of the mutually-exclusive modules are functional), we would require a different experimental setup, *i.e.*, new species have to be either perturbed or measured.

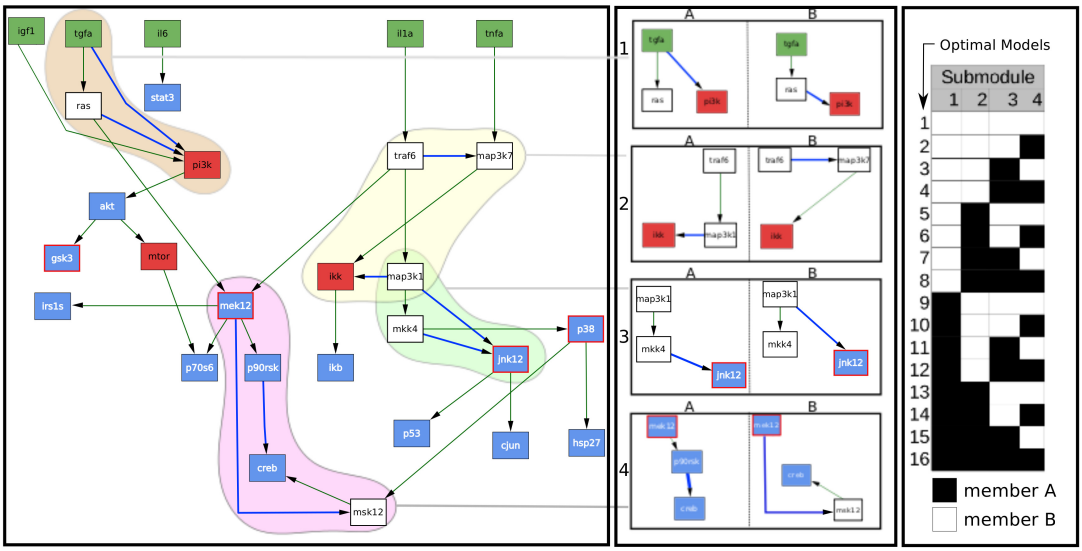



Figure 5.2: **Optimal logic models for HepG2 cells computed by *caspo*. (A)** Network of the union of 16 optimal models. Green nodes represent ligands that are experimentally stimulated. Red (or red-bordered) nodes represent those species that are inhibited with an small molecule inhibitor (drug). Blue nodes represent species that were measured using the Luminex technology. White nodes are neither measured nor perturbed. Green links represent hyperedges present in all models, and blue hyperedges present in some of the models. All models have logical gates of type OR. The colored regions highlight four pairs of mutually-exclusive modules. The network was generated with Cytocopter. **(B)** Mutually-exclusive modules. For the gates activating the nodes pi3k, ikk, jnk12, and creb, there are two alternative modules that are equivalent in terms of the value of the scoring function, denoted as A and B. **(C)** Distribution of the mutually-exclusive modules across the family of 16 optimal models. White or black boxes refer to member A or B in panel B.

### 5.4.2 Sub-optimal Models: Enumeration and Structure

Experimental error is inherent in biochemical data. Therefore, one needs to consider models whose predictions deviate from those of the optimal one by an amount within the experimental error [29]. Considering that the optimization minimize MSE and size, we defined as *sub-optimal models* those solutions having MSE within a 10% of tolerance with respect to the MSE of optimal models (a conservative approximation to the real experimental error), and maximal size of 28 (the size of the optimal models, see Section 5.4.1). From these settings, *caspo* found 11,700 sub-optimal models (Fig. 5.3) with sizes 28, 27, 26 and 25 whose MSE spanned from 0.0499 to 0.0546. We observed that the number of

models decreases exponentially with the tolerance over the MSE (*e.g.* 8% - 7,378 models, 6% - 6,048 models, 2% - 192 models). Allowing also a tolerance over the size would generate a much larger number of models by the addition of spurious links to those of size 28 (*e.g.* size 29 - 51,480 models, size 30 - 189,364 models). We therefore limited, for simplicity of this study, the size to 28.

The complete computation of sub-optimal models allows a precise characterization of the distribution of hyperedges, and, therefore, of logical gates in the potential models. When we evaluated the distribution of the 130 possible hyperedges (*i.e.* those that are included in the hypergraph derived from the original PKN) across the 11,700 models, we found that 14 hyperedges are *present in all* sub-optimal models, and we thus expect them to be functional in HepG2 cells. 59 hyperedges are *absent from all models*, thus suggesting that they are not functional in these cells. Finally, 57 hyperedges are present in only a subset of the models; their frequency ranges from 0.99 to 0.0003, showing a large variability. Therefore, for the given experimental data, these hyperedges are not identifiable as it is not possible to determine whether they are functional or not in HepG2 cells.

Analogously to the set of optimal models, we investigated the combinatorics within the family of sub-optimal models. We found four mutually-exclusive pairs of modules (Fig. 5.3B). Replacing a module of each pair by the other has no effect on the MSE for two of the pairs (1,2 in Fig. 5.3B). However, for the pairs 3 and 4 there is a difference; 32% and 26.8%, respectively, of the sub-optimal models differ in the output for a range from 8 to 15% of the experimental conditions. All modules were constituted by a single hyperedge, except 1A, which is set by two hyperedges: $\{(ras \wedge \neg akt \rightarrow mek), (ras \wedge pi3k \rightarrow mek)\}$ (Fig. 5.3, module 1A). These two hyperdges were therefore always either both present or both absent (*mutually-inclusive*). As expected, there is a clear difference between the frequencies in each pair of exclusive patterns where smaller or simpler hyperedges are always more abundant. Importantly, the mutually-exclusive modules for the family of sub-optimal models are not the same as those present when only optimal models are considered. This indicates that the combinatorics exhibited within optimal models is not so important when considering experimental error, probably due to the larger variability among sub-optimal models.

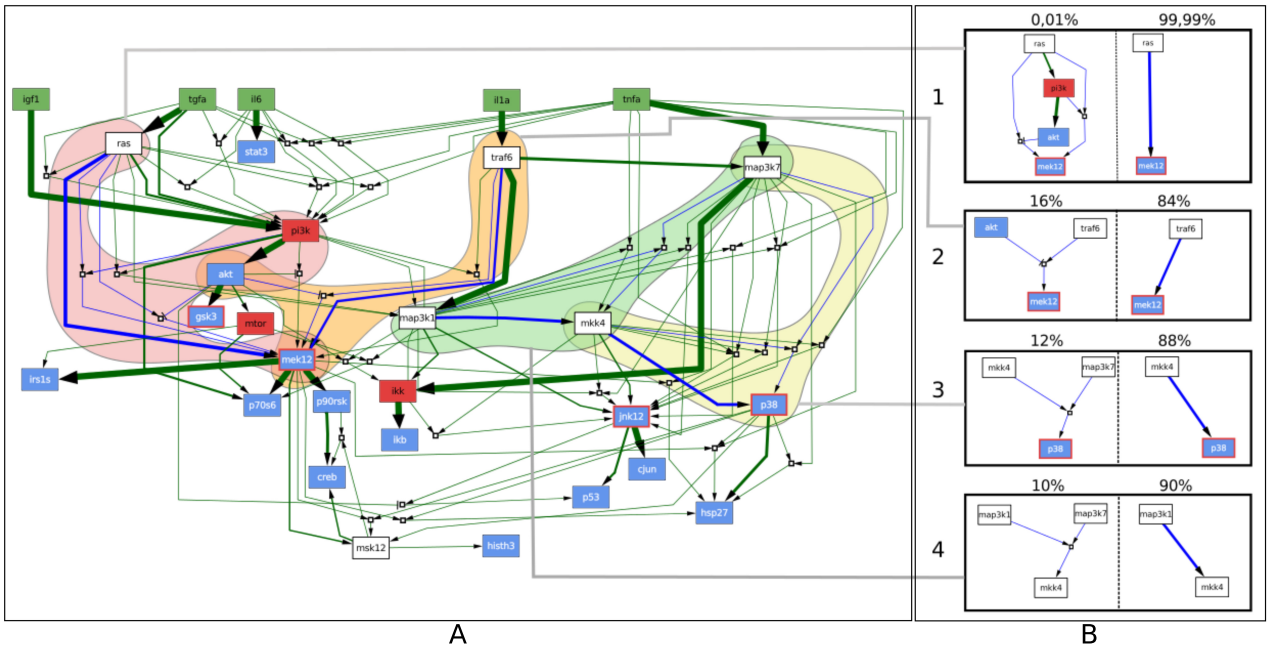



Figure 5.3: **Sub-optimal models generated with *caspo* with 10% error tolerance.** (**A**) Network of the union of 11,700 sub-optimal models. Green nodes represent ligands that are experimentally stimulated. Red (or red-bordered) nodes represent those species that are inhibited with an small molecule inhibitor (drug). Blue nodes represent species that were measured using the Luminex technology. White nodes are neither measured nor perturbed. AND gates in the models are represented by empty boxes. The thickness of the hyperedges correspond to their frequencies among the 11,700 sub-models. (**B**) Four pairs of mutually-exclusive modules (blue hyperedges in **A**) and their corresponding frequencies on top. These modules determine the behavior of three nodes in the network: mek12, mkk4, and p38.

### 5.4.3 Input-output behavior

To further characterize the family of sub-optimal models, we next studied its input-output behavior as expressed by its GTT. Using *caspo*, we found that the 11,700 sub-optimal models correspond to 91 different GTTs. In these 91 GTTs, the predicted values are the same for 30% (4915 out of 16384) of all the possible experimental conditions (*i.e.* $2^{14}$ combinations of the 14 inputs of the model). Therefore, such predictions can be seen as the "core" predictions of the system behavior independently from experimental noise. Considering the remaining 70% of experimental conditions, we found that at least 7 experiments are needed in order to discriminate among all GTT. By performing such experiments, one would be able to generate at least 1 different output prediction between

every pair of GTT. Among the 11,700 sub-optimal models there are only 13 different MSEs. The distribution of such MSEs is very inhomogeneous, and two MSE (0.0519 and 0.0542) gather 71% of sub-optimal models (Fig. 5.4).

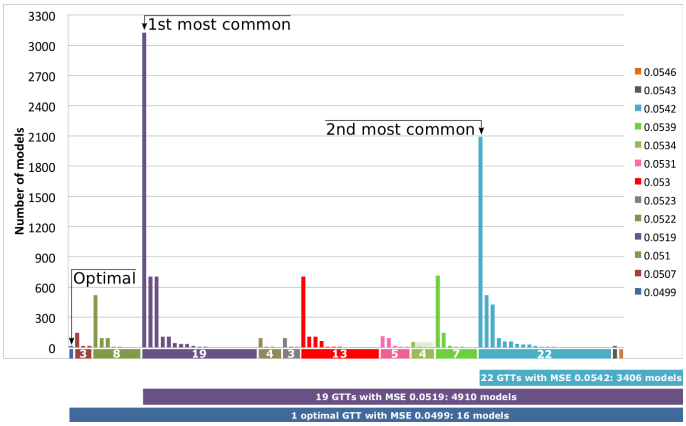



Figure 5.4: **Distribution of sub-optimal models.** The sub-optimal models are ordered (from left to right) first according to their MSE, and then according to their 91 GTT. The number of different models leading to the same GTT is plotted in vertical bars. GTT are ordered and colored by their MSE. The 16 optimal models correspond to MSE 0.0499. The 2 most common GTT describe the response of 3126 and 2090 models.

For both most frequent MSE, a GTT is much more common than all the others: the first GTT, at MSE 0.0519, is shared by 3126 (27%) sub-optimal models while the second most common GTT, at MSE 0.0542, is shared by 2090 (18%) models. In contrast, the minimal GTT, at MSE 0.0499, was shared by only the 16 minimal models. This analysis suggests that the single optimal GTT at MSE 0.0499 is far from being representative over the 11,700 sub-optimal models (0.1%). The two most common GTT are arguably much more relevant. Interestingly, a hierarchical clustering reveals that these two most common GTT cluster quite separately and that the GTT representing 27% of all sub-optimal models is very close to the optimal one (Fig. 5.5).

## 5.5 Conclusion

A useful approach to model large-scale signaling networks consists on training Boolean logic models from prior knowledge and dedicated experimental data. The problem of training these models is an optimization task that can be solved with stochastic search methods [29], which have the important limitation that they do not guarantee global

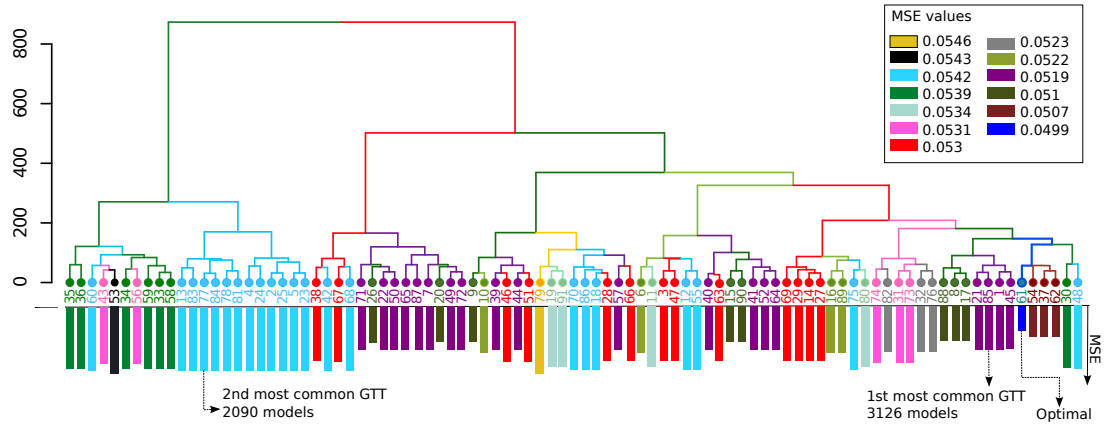



Figure 5.5: **Hierarchical clustering of GTTs.** Hierarchical clustering of the 91 GTT based on their predictions for the readouts across all experimental conditions. Bars length on the leafs represents the corresponding MSE value for each GTT. The optimal GTT (61) is highlighted, as well as the two most common ones (85 and 77). The most common GTT is very close to the optimal one, whereas the second most common GTT has a quite different behavior.

optimality nor an exhaustive solution. In this paper, we show how recasting this problem in a highly declarative language allows us to identify the complete family of feasible models and query them to obtain insight into model degeneracy.

In a real-case study we have seen that there is a family of feasible models with a deep combinatorial structure: several combinations of internal submodules, with equal or similar scores, can equivalently explain the observed behavior of the system. This leads to a rapid growth of the family of sub-optimal models. Taking into account the inherent noise in data, we showed that 11,700 different models can be considered as plausible representations of the prior knowledge network and an experimental phospho-proteomics dataset. Thanks to our exhaustive characterization of these models, we could determine unambiguously which hyperedges (biological links) are functional or not, based on their distributions across the models and determine whether groups of hyperedges are exclusive from each other.

To further characterize this family of models, we introduced the concept of GTT and used it to explore their input-output behavior. Compared to the model topologies, the variability is much lower; the 11,700 models can be grouped in 91 GTTs, and for 30% of the 16384 possible perturbations all models gave the same predictions. Interestingly, the distribution of models among GTT is far from being equidistributed, and two GTTs comprise almost half of the models, while the GTT corresponding to the optimal score is

very specific (0.1% of the models). While the most common GTT is similar to the GTT with optimal score, the second most common GTT is very different. However, a single experiment is able to discriminate these models.

These results underscore the importance of exploring exhaustively the family of models and take into account experimental error to obtain an adequate picture of the feasible model solutions. Our formal approach based on ASP allows a precise characterization of the information that can be inferred from the confrontation of prior knowledge with experimental observations over protein signaling networks. It also permits the study of the internal combinatorics leading to the variability of the system functioning and provides a tool towards experimental design. Due to the complexity of signaling networks and the limitations of existing experimental technologies (in terms of which nodes can be measures and/or perturbed), models typically show an important lack of identifiability. This is a general limitation of models in systems biology [132]. In the context of Boolean models, we expect that further development of experimental design [136], in intimate co-ordination with advances in experimental techniques will allow us to tackle this issue. We approached this perspective later with a publication on 2015 [33].

This work opens the way to several prospective tracks. First, it would be useful to evaluate our ASP formulation and those based on ILP from [135] and [136] in order to understand their strengths and complementary features. In contrast to ILP, ASP is a relatively new tool for problem solving in biology. ASP, having its roots in knowledge representation and reasoning, has proven to be very well suited to address highly combinatorial search and discrete optimization problems, with at least comparable performance to well established ILP solvers. On the other hand, ILP as a mathematical programming framework may be more suitable to study problems based on calculus over large domains of integer or rational numbers. Therefore, combining the expressiveness and power of several solving technologies instead of selecting one of them seems a promising option for the future [147, 148].

Second, the perspective of studying the extension of our approach to time-series data, although switching from a steady-state to a dynamical viewpoint implies a growth of the search space. Fitting models whose steady-states evolve between clearly separated time-scales [138] should be of similar complexity to the problem studied in this paper. Fitting to the actual time-courses of a Boolean model has a higher level of complexity, as it requires to adjust the time-step of the Boolean model to the real time of the measurements. This perspective was later investigated, implemented and applied (see Chapter 6).

More generally, we need to develop a rigorous framework to study models of biological networks as a family of plausible realizations, not of single networks. A first approximation could be to compare experimental data (ideally a distribution across individual cells) to a distribution of simulated results across a family of *single* logical models. The comparison of the distribution of feasible models to single cell data emerges as longer-term follow-up of this work that should provide deep insight into the cell-to-cell heterogeneity of signal transduction [149]. We are currently applying *caspo* to single-cell data related to Human embryonic development in the thesis project of Mathieu Bolteau, which I co-supervise.

Altogether, we have implemented an open-source tool based on ASP providing a powerful framework to analyze networks models in systems biology. Further, several prospective tracks will certainly lead to future developments in order to extend and improve the functionalities of *caspo*.

Chapter 6

# Learning Dynamical Boolean Networks

*"Nothing binds you except your thoughts;*
*nothing limits you except your fear;*
*and nothing controls you except your beliefs."*
— Marianne Williamson.

## 6.1 Introduction

In this last chapter it is presented the problem of inferring Boolean Networks from time-series expression data under perturbation. This work involved the interaction of many previous and new collaborators in the context of a challenging but beautiful doctoral thesis research project, in which I co-supervised the work of Misbah Razzaq.

The study presented in this chapter was published in [150] (see Annex A.4), the results presented here were taken from that publication. The full story of this research did not started or finished on this paper, previous (published in [151, 152]), and forward collaborations (published in [150]) were made and had to be considered to understand the whole of this contribution. The contribution of all co-authors of these related publications was important for this reserach to appear. I specially mention two colleagues, Max Ostrowski and Loïc Paulevé, who made essential contributions to this work.

## 6.2 Motivation and background

Protein signaling networks are static views of dynamic processes since they respond to stimuli and perturbation. They constitute complex regulatory systems controlled by crosstalk and feedback mechanisms. Because these networks are often altered in diseases, discovering the precise mechanisms of signal transduction may provide a better fundamen-

tal understanding of disease behavior. For instance, a main difficulty in cancer treatment is the fact that cell populations specialize upon treatment and therefore patient responses may be heterogeneous. Computational models of signaling control for different patient groups could guide cancer research towards a better drug targeting system. In this work, we propose a methodological framework to discriminate among the regulatory mechanisms of four breast cancer cell lines by building predictive computational models.

Several formalisms have been used widely to model interaction networks. Models built using differential equations require explicit specifications of kinetic parameters of the system and work well for small-scale systems. While being a useful tool, mathematical modeling becomes computationally intensive as networks become larger [153, 154, 155]. Stochastic modeling is suitable for problems of a random nature but also fails to scale well with large scale systems [153]. The Boolean Network (BN) formalism [156] is a powerful approach to model signaling and regulatory networks [157]. Various BN learning frameworks exist focusing on varying levels of details [153, 158]. As compared to the extensive literature on Boolean frameworks, BN modeling of signaling networks is quite recent.

In this work, we have used the caspo time series (*caspo-ts*) [151, 152] method to learn BNs from multiple perturbation phosphoproteomic time series data given a Prior Knowledge Network (PKN, refer to Section 5.3.1). We have improved and adapted *caspo-ts* to deal with a midscale Prior Knowledge Network (PKN) with 64 nodes and 178 edges in order to learn the BNs of four breast cancer cell lines (BT20, BT549, MCF7, UACC812) from their time series phosphoproteomic datasets. Importantly, the PKN did not contain any information about the temporal changes or dynamic properties of the proteins. This information was learned from a dataset describing the dynamics of signaling processes for those breast cancer cell lines as part of the HPN-DREAM challenge. In comparison to the current methods that learn signaling networks as Boolean models using static measurements[159, 160], and one-time point measurements across multiple perturbations [46, 120, 136, 135], our method allows us to handle time series data. A further advantage is the guarantee of discovering optimal BNs, where the distance between original and over-approximated time series is minimal. This is achieved by using computational solvers such as Answer Set Programming (ASP) [41].

Our results show that the ASP component of our method allows us to filter the explosion of possible dynamical states inherent to this type of problem, and thanks to that filtering, the model-checking step allows us to provide BNs exactly reproducing the binarized time series data. These BNs are referred to as true positive (TP) BNs. Our results

point to measurements in the time series HPN-DREAM dataset that contradict the experimental setting and to perturbations that show contradictory dynamics. We observed that given the same PKN, the solving time was different for each cell line dataset. For cell lines BT20, BT549, MCF7 our method found TP BNs, while for the UACC812 cell line dataset it was impossible to find a TP BN within a time-frame of 7 days. This computation time difference is due to the different structure of the solution space among cell lines. This could point to the situation where the dataset is not explainable by the prior knowledge network, which may give valuable insights to experimentalists. For example, that the number of consistent experimental perturbations is not sufficient, and that the knowledge of the PKN is incomplete given this dataset. We also show that this method is capable of recovering time series measurements with a Root Mean Square Error (RMSE) of 0.31, the minimum achieved so far as compared to other participants of the HPN-DREAM challenge. Our method focuses on learning optimal BNs' structures. It does not predict time-series traces of the proteins from the learned BNs. However it detects the minimum distance that is possible to obtain from the proteins of the learned BNs in comparison to the time-series traces in the testing data. This is the main conceptual difference of our method compared to those proposed by the HPN-DREAM challenge. This difference needs to be considered when comparing the RMSE score. Based on a comparison with the canonical mTOR pathway, we show that the discovered context specific BNs have an average AUROC score of 0.77. We found 38% of the cell line specific interactions explaining the heterogeneity among these four cancer cell lines, which can be observed in different cell line specific networks. All in all, our results show that *caspo-ts* handles real (HPN-DREAM) datasets, where data points are incomplete and subject to experimental error. Our method is applicable to any kind (gene or protein expressions) of time series datasets measured upon different perturbations. We have proved here that *caspo-ts* handles a PKN size of 64 nodes and 170 edges; this is relevant since approaches modeling time usually only scale up to very small networks because of state graph explosion.

## 6.3 Materials and Methods

### 6.3.1 Data acquisition

The DREAM portal provides unrestricted access to complex, pre-tested data to encourage the development of computational methods. In this study, we are focused on the

HPN-DREAM challenge, which was motivated by the fact that the same perturbation may lead to different signaling behaviors in different backgrounds, making it necessary to build a model which can perform unseen predictions (absent from the learning data). The main goal of the HPN-DREAM [1] challenge is to learn signaling networks efficiently and effectively to predict the dynamics of breast cancer.

**Learning data**

Reverse Phase Protein Array (RPPA) quantitative proteomics technology was used for generating the dataset of this challenge. The measurements focus on short term changes on up to 45 proteins and their phosphorylation over 0 to 4 hours. The HPN-DREAM dataset includes temporal changes in phosphorylated proteins at seven different time points ($t_1$ = 0min, $t_2$ = 5min, $t_3$ = 15min, $t_4$ = 30min, $t_5$ = 60min, $t_6$ = 120min, $t_7$ = 240 min). The learning data consists of four cancer cell lines (BT20, BT549, MCF7 and UACC812) under different perturbations (≈8 stimuli and ≈3 inhibitors). The number of perturbations varies from 24 to 32 depending on the cell line. In each cancer cell line approximately 45 phosphorylated proteins are measured against different sets of perturbations over multiple time scales. After removing perturbations with inconsistent behaviors or incomplete time series, we had 15, 13, 13 and 18 perturbations for MCF7, BT20, BT549 and UACC812 cell lines respectively measuring 23 readouts.

**Testing data**

Test data is available for assessing the performance of networks learned from the learning data. The HPN-DREAM portal provides testing data for four cancer cell lines (BT20, BT549, MCF7 and UACC812) under different perturbations (8 stimuli and 1 inhibitor). They contain gold standard datasets of time series predictions of up to 45 proteins having the same time scale as learning data[161, 162]. The number of perturbations varies from 7 to 8 depending on the cell line. This data is used to test the quality of the BNs given by *caspo-ts*.

### 6.3.2 *Caspo-ts* modeling framework

The caspo time series (*caspo-ts*) method was proposed in [152]. This method learns BNs from multiple perturbation phosphoproteomic time series data given a PKN. *caspo-*

1. `https://www.synapse.org/#!Synapse:syn1720047/wiki/55342`

*ts* is based on ASP and a model-checking step is needed to detect true positive BNs. In [152] our approach was tested on synthetic data for a small PKN (≈17 nodes and ≈50 edges) [152]. A similar approach, based on genetic algorithms [163], was proposed to learn context specific networks given a PKN and experimental information about stable states and their transitions but it does not scale well with large networks and finding a global optimum is not guaranteed. In Fig. 6.1 we show the workflow of the *caspo-ts* method for the inference of BNs. This method was tailored to handle protein phosphoproteomic time series data. The input of the method consists of a PKN and normalized phosphoproteomic time series data under different perturbations to generate a family of BNs whose structure is compatible with the PKN and that can also reproduce the patterns observed in the experimental data. In the following, we will develop the main notions of this framework.

**Prior knowledge network.** Also defined in Section 5.3.1, it is one input of *caspo-ts* and it is modeled as a labeled (or colored) directed graph $(V, E, \sigma)$ with $V = \{v_1, v_2, \ldots, v_n\}$ the set of nodes, $E \subseteq V \times V$ the set of directed edges and $\sigma \subseteq E \times \{+1, -1\}$ the signs of edges. The set of nodes is denoted by $V = S \cup I \cup R \cup U$ where $S$ are stimuli, $I$ are inhibitors, $R$ are readouts, and $U$ are unobserved nodes. Stimuli, inhibitors, readouts, and unobserved nodes are encoded by different colors in the graphs presented in this case study. Stimuli are shown in green, inhibitors in red, readouts in blue, and unobserved nodes in white (Fig 6.1). Moreover, the subsets $S$, $I$, $R$, $U$ are all pairwise disjoint except for $I$ and $R$, because a protein can be inhibited as well as measured. Stimuli are used to bound the system and also serve as interaction points of the system, these nodes can be experimentally stimulated, *e.g.* cellular receptors. Inhibitors are those nodes which remain inactive or blocked over all time points of the experiment by small molecule inhibitors. Stimuli and inhibitor nodes take Boolean values $\{0, 1\}$ representing the fact that the node was stimulated (1) or inhibited (0). Readouts are experimentally measured given a combination of stimuli and inhibitors. They usually take continuous values in [0;1] after normalization. Unobserved nodes are neither measured nor experimentally manipulated. In this study, we use the term *perturbation* to refer to the combination of stimuli and inhibitors, similarly to other studies such as [161, 162].

**Phosphoproteomic time series data.** It is the second input of *caspo-ts* and it consists of temporal changes in phosphorylated proteins under a perturbation (Fig 6.1). Without loss of generality, we assume that the time series data are related to the observation of

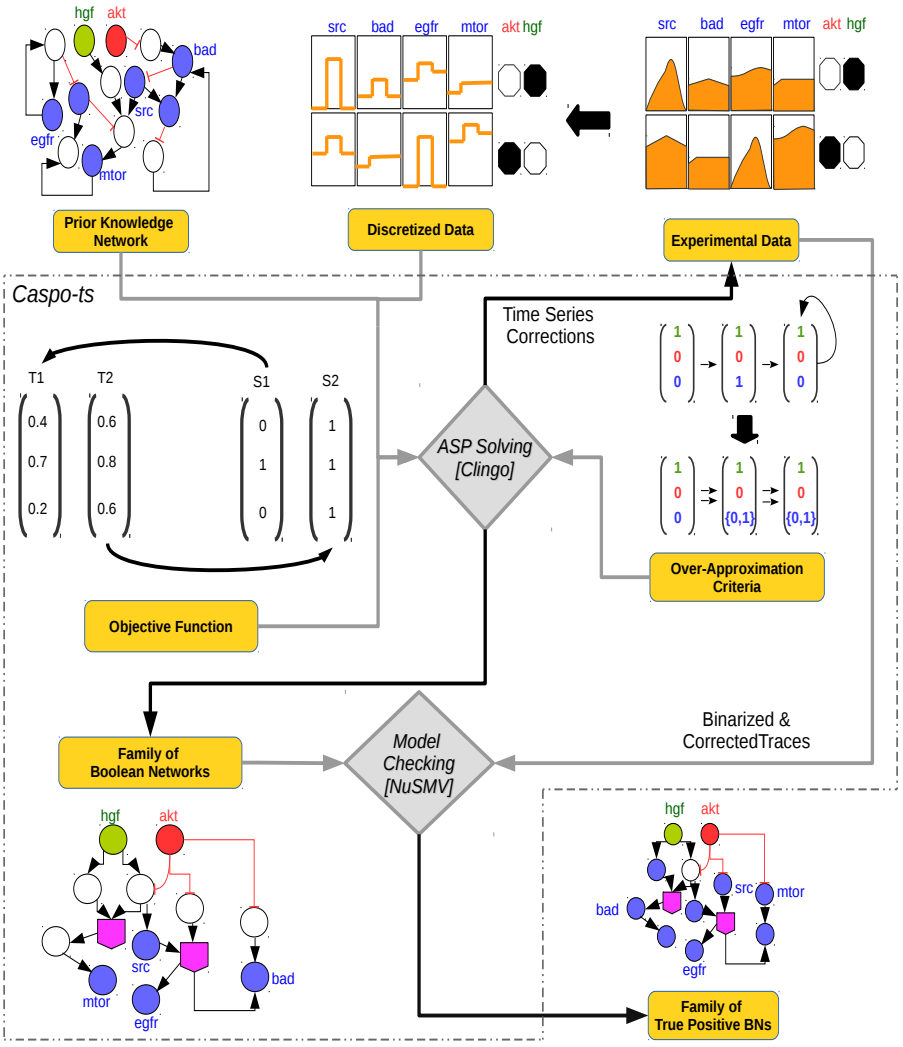



Figure 6.1: ***Caspo-ts* workflow.** *Caspo-ts* receives as input data a prior knowledge network (PKN) and a discretized phosphoproteomic dataset. In this example the phosphoproteomic data consists of two perturbations involving *akt* (inhibitor) and *hgf* (stimulus): 1) $akt = 0$, $hgf = 1$ and 2) $akt = 1$, $hgf = 0$. A black colored perturbation means the inhibitor or stimulus was perturbed (1) while white represents the opposite (0). Readouts are specified in blue and describe the time series under given perturbations. Using this input data, *caspo-ts*, performs two steps: ASP solving and model checking. In the ASP solving step: (i) a set of BNs, compatible with the PKN, is generated, (ii) afterwards an *over-approximation constraint* is imposed upon each candidate BN to filter out invalid BNs, that do not result in an over-approximation of the reachability between the Boolean states given by the phosphoproteomic dataset, and finally (iii) BNs are optimized using an objective function minimizing the distance to the experimental measures. The ASP step also introduces repairs in some data points of the time series that added penalties to the objective function. These corrected traces will be given to the model checker. In the model checking step, the exact reachability of all the (binarized and corrected) time series traces in the family of BNs is verified.

$m \leq n$ nodes for the nodes $\{v_1, \ldots, v_m\}$ (so the nodes $\{v_{m+1}, \ldots, v_n\}$ are not observed). The observations consist of normalized continuous values: a time series of $k$ data points is denoted by $T_P = (t_P^1, \ldots, t_P^k)$, where $P \subseteq S \cup I$ is a perturbation and $t^j \in [0; 1]^m$ for $1 \leq j \leq k$. This data will be discretized in order to link it with further BNs' discovery (ASP solving and model checking steps).

**Boolean Network.** It is the output of *caspo-ts*. A *Boolean Network (BN)* [164, 165] is defined as a pair $BN = (N, F)$, where

— $N = \{v_1, \ldots, v_n\}$ is a finite set of nodes (or variables/proteins/genes)

— $F = \{f_1, \ldots, f_n\}$ is a set of Boolean functions (regulatory functions) $f_i : \mathbb{B}^k \to \mathbb{B}$, with $\mathbb{B} = \{0, 1\}$, describing the evolution of variable $v_i$.

A vector (or *state*) $x = (x_1, \ldots, x_n)$ captures the values of all nodes in $N$ at a time step, where $x_i$ represents the value of the node $v_i$, and is either 1 or 0. There are up to $2^n$ possible distinct states for each time step. Next, we define the transition $x \to x'$ between two states of a BN. If there is no update for node $v_i$ then $x'_i = x_i$. If there is an update for node $v_i$ then the state of a node $v_i$ at the next time step is determined by $x'_i = f_i(x_1, \ldots, x_n)$. Note that usually only a subset of the nodes influence the evolution of node $v_i$. These nodes are called the *regulatory nodes* of $v_i$. The state of each node can be updated in a synchronous (parallel) or asynchronous fashion. In the synchronous update schedule, the states of all nodes are updated, while in asynchronous update schedule, the states of any number of nodes can be updated at a time. The work presented in this article is independent of the update schedule routine, hence any number of nodes can be updated at a time.

**Over-Approximation Criteria** The goal is to generate BNs that can reproduce the experimental data as well as possible. For this objective, the states have to be reachable from another. We use $x \to^* y$ to say that state $y$ can be reached from state $x$ with an arbitrary number of steps. Since this reachability is a computationally hard problem (PSPACE-complete) [166], we use an over-approximation for checking reachability resulting in false positive (FP) BNs [152, 151]. The meta-states have been introduced to check over-approximated reachability.

A meta-state $u = (u_1, u_2, \ldots, u_n)$ is a vector of dimension $n$ over non-empty subsets of $\mathbb{B}$, noted $\mathbb{M} = \{\{0\}, \{1\}, \{0, 1\}\}$; the set of meta-states is $\mathbb{M}^n$. Meta-states characterize a set of Boolean states: a state $x \in \mathbb{B}^n$ belongs to a meta-state $u$, written $x \in u$, iff each

Boolean component $x_i$ belongs to the set $u_i$. Given a state $x$, we use $\bar{x}$ for the corresponding meta-state $(\{x_1\}, \ldots, \{x_n\})$. We define the transition relation $u \rightrightarrows v$ between the meta-states $u$ and $v$ as follows: $u \neq v$ and $v = (u_1, \ldots, u_i \cup \{f_i(x) \mid x \in u\}, \ldots, u_n)$ for some $1 \leq i \leq n$.

In [152], it has been shown that if $y$ is reachable from $x$ ($x \rightarrow^* y$) then there exists a meta-state $u$ such that $y \in u$ and $\bar{x} \rightrightarrows^* u$. This definition is further refined to describe the necessary condition for reachability called support consistency. A state $x$ is support consistent with state $y$ denoted by $x \rightsquigarrow^* y$, if and only if there exists a meta-state $u$ with $\bar{x} \rightrightarrows^* u$ such that $y \in u$ and for all $1 \leq i \leq n$ either

— $y_i \neq x_i$, or

— $y_i = x_i$ and $u_i \neq \{0, 1\}$, or

— $y_i = x_i$, $u_i = \{0, 1\}$, and there exists $z \in u$ such that $f_i(z) = y_i$.

If state $y$ is reachable from state $x$ ($x \rightarrow^* y$) then $x \rightsquigarrow^* y$. Since we are using the over-approximation criteria, it is possible that some BNs may fail to reproduce the exact trajectories of the time series data. These BNs are called false positive (FP). To filter out the false positive BNs, exact model checking is applied.

**ASP solving.** Given a PKN and a phosphoproteomic dataset, a family of candidate BNs, compatible with this PKN, is exhaustively enumerated including the main nodes (the sets $S$,$I$,$R$) of the experimental data. We refer the reader to [120] for a detailed description of BN's compatibility with a PKN. Afterwards an *over-approximation constraint* is imposed upon each candidate BN to filter out invalid BNs [152], that do not result in an over-approximation of the reachability between the Boolean states given by the phosphoproteomic dataset. Finally, an optimization step is performed to select those BNs having a minimal distance between the actual time series $T_P$ and the over-approximated time series $Y_P$. We have adopted the Root Mean Square Error (RMSE) as the *objective function*:

$$RMSE = \sqrt{\frac{1}{m * k * |\mathcal{P}|} \sum_{i=1}^{m} \sum_{j=1}^{k} \sum_{P \in \mathcal{P}} ((t_P^j)_i - (y_P^j)_i)^2} \tag{6.1}$$

where $m$ is the number of observed nodes, $k$ is the number of time points, and $\mathcal{P}$ is the set of perturbations. In addition, the optimization step highlights the data points in the time series which added penalties to the RMSE. Such data points are automatically corrected before the model checking step.

All the analyses described in this step are performed using ASP, namely the `clingo` 4.5.4 solver. This solver guarantees finding optimal solutions, and all BNs outputted by the ASP solver step will be identically optimal. For the HPN-DREAM case study, the full enumeration of optimal BNs creates billions of BNs, and since the next (model checking) step can take days of computation depending on the verified BN we choose to limit this enumeration to a fixed number of BNs.

**Model checking and True Positive BNs.** From the ASP solving step, a set of optimal BNs that over-approximate the phosphoproteomic time series data is produced. This set of BNs is verified with an exact *model checking* to detect true positive (TP) BNs. *TP BNs* are guaranteed to reproduce all the (binarized) trajectories under all perturbations by verifying exact reachability in the BN state graph. For this, we have used computational tree logic (CTL) implemented in the `NuSMV` 2.6.0 [167], which is a symbolic model checker.

### 6.3.3 Graph similarity measure

We introduced a graph similarity measure in order to check the variability among the families of BNs generated by *caspo-ts*. We compare the reactions existing in the gold standard network ($A$) with the family of BNs (BNF). This measure is based on the Jaccard similarity coefficient which measures the similarity of these models.

#### Jaccard similarity coefficient

The Jaccard index between $A$ and $B_i$ can be defined as length of the intersection divided by the union:

$$J(A, B_i) = \frac{\mid A \cap B_i \mid}{\mid A \cup B_i \mid} = \frac{\mid A \cap B_i \mid}{\mid A \mid + \mid B_i \mid - \mid A \cap B_i \mid} \tag{6.2}$$

We apply the Jaccard Similarity Coefficient on $B_i$ (where $B_i \subset$ BNF) by taking A as being the gold standard.

## 6.4 Results

### 6.4.1 Prior knowledge network

The structure of the protein signaling network was generated by mapping the experimentally measured phosphorylated proteins (HPN-DREAM dataset) to their equivalents from literature-curated databases and connecting them together within one network. The reference network (Fig 6.2) was built using the ReactomeFIViz app (also called the ReactomeFIPlugIn or Reactome FI Cytoscape app) [168], which accesses the interactions existing in the Reactome and other databases [168, 169]. The PKN shown in Fig 6.2 consists of 64 nodes (7 stimuli, 3 inhibitors, and 23 readouts) and 178 edges.

### 6.4.2 Cell line specific Boolean networks

In this section, we show the generated BNs for each cell line. For this, we used *caspo-ts* to learn the BNs from the PKN (Fig 6.2) and the phosphoproteomic data of four breast cancer cell lines - BT20, BT549, MCF7, and UACC812. We inferred a family of cell line specific BNs for each cancer cell line.

*caspo-ts* produces BNs fulfilling two criteria, (i) satisfaction of the over-approximation criteria and (ii) optimality with respect to the RMSE objective function. ASP-optimal solutions were fast to collect, their computation time ranged from 21 seconds to 3 minutes depending on the cell line. Afterwards, these ASP-optimal BNs were given to the model-checker for further verification. This second step is more complex and we put a restriction for the computation time of 7 days for each cell line. The number of verified BNs varies from one cell line to another, depending on a number of factors such as the number of perturbations, the order of answer sets in the solutions space, and the perturbation order. The total number of verified ASP-optimal BNs within the 7 days time-frame were 231, 52, 188 and 150 for the BT549, MCF7, BT20 and UACC812 cell lines respectively. We obtained 191, 21, and 72 true positive BNs for BT549, MCF7, and BT20 cell lines respectively with an optimal fit to the data. For the UACC812 cell line, we were unable to obtain true positive BNs within the 7 day time limit for verification. Hence, we kept the first 20 BNs from the 150 ASP-optimal BNs for the UACC812 cell line. The *caspo-ts* method uses the ASP solver (clingo), which is able to exhaustively enumerate all solutions. The clingo solver by default uses an enumeration scheme, in which, once a solution is found, it backtracks to the first point from where the next solution can be found. This

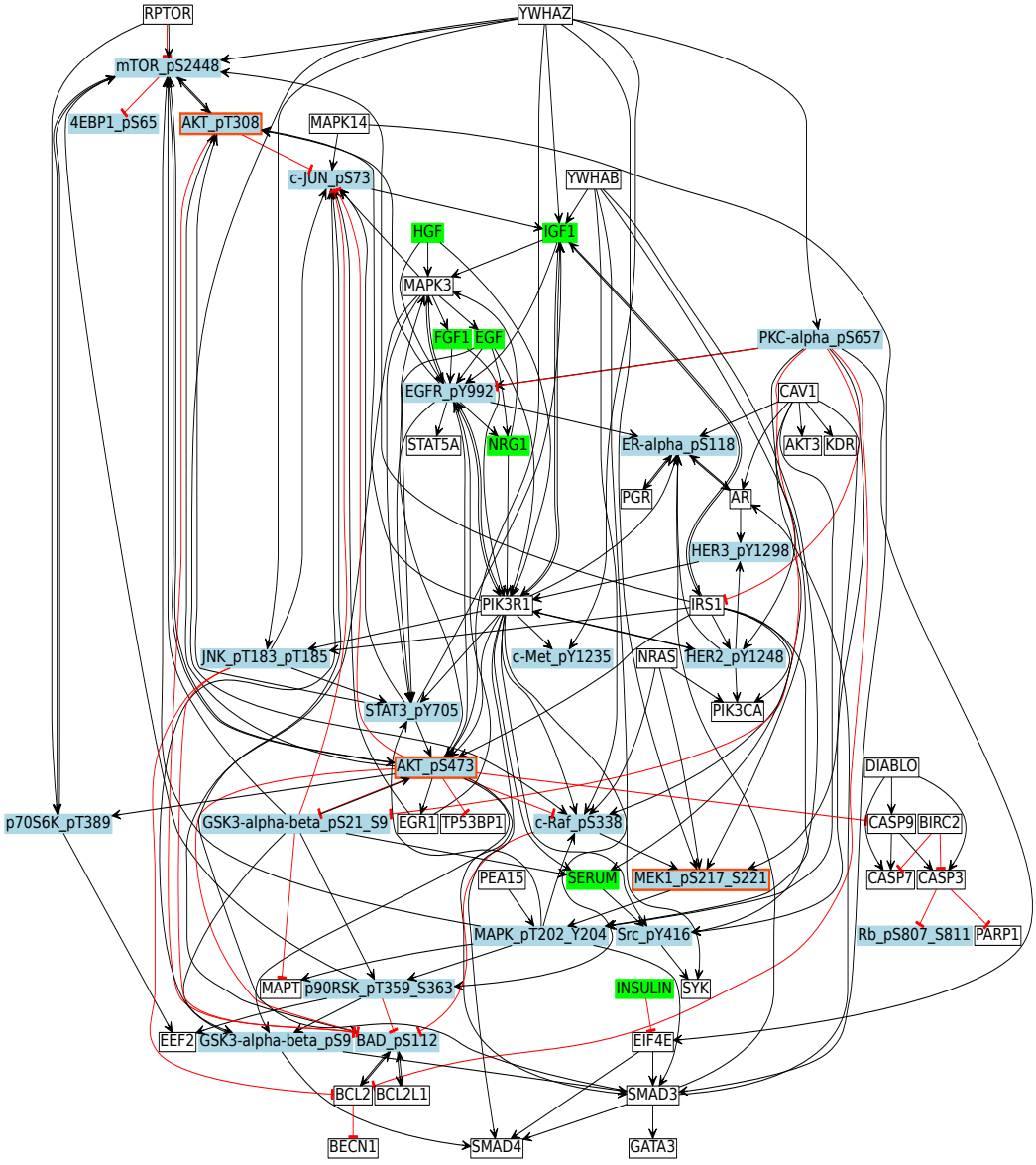



Figure 6.2: **Breast cancer signaling pathway.** This figure shows the reconstructed signaling network from a combination of databases. An arrow shows the positive regulatory relationship between two proteins, while a T shaped arrow indicates inhibition. Green nodes are stimuli, blue nodes are readouts, white nodes are unmeasured or unobserved, and blue nodes with a red border represent inhibitors and readouts at the same time. Please note that in the node labels, we have added the phosphorylation sites to the protein names in order to connect them to the experimental measurements.

typically leads to the situation where successive solutions only change in a small part. As a result, *caspo-ts* may enter a solution space where BNs are clustered together. We have

observed that given the size of the PKN and the small number of perturbations in the experimental data, the solution space of the *caspo-ts* can be rather very large containing billions of BNs making it difficult to enumerate true positive BNs (because of the model checking overhead) in a reasonable time if it gets stuck in a cluster of false positive BNs.

An aggregated network was built (Fig 6.3) by combining the BN families (with 191, 21, 72, and 20 BNs for BT549, MCF7, BT20, and UACC812 cell lines respectively) obtained for the four cell lines by keeping the hyper-edges (Boolean functions) having a frequency higher than 0.3 within each BN family. The frequency is calculated by counting the number of common Boolean functions and dividing it by the total number of Boolean functions within the BN family of each cell line. This aggregated network contains 34 nodes and 74 Boolean functions involving 36 AND gates. As compared to the PKN (Fig 6.2), the inferred networks are highly specific to each cell line. In Fig 6.3, all cell lines share only 4% of Boolean functions which are shown in thick black colored edges. This shows that the inferred BNs of these four breast cancer cell lines are very diverse and different from each other.

To measure cell lines similarity, we calculated the similarity score (see Section 6.3.3) on the family of BNs (with 191, 21, 72, and 20 BNs for BT549, MCF7, BT20, and UACC812 cell lines, respectively). This algorithm receives two parameters as input: (1) one gold standard BN and (2) a family of BNs. It outputs a score in $[0; 1]$, measuring the average of the similarity scores between each BN in the family and the gold standard BN. In our case, the gold standard BN is the aggregation of one family of BNs. The similarity scores between all pairs of breast cancer cell lines are shown in Table 6.1. Fig 6.3 agrees with the results presented in Table 6.1 as we can see the clear discrepancies among the four cell lines. It can be seen that 23% of the Boolean functions are shared among BT549 and MCF7, and also between BT20 and UACC812. BT20 shares the least number of Boolean functions (15%) with BT549. This table revealed pronounced differences among different cell lines of breast cancer. We also analyzed the diversity of Boolean functions among the family of BNs within the same cell line. The similarity among Boolean functions from BT20 (0.73) and MCF7 (0.63) is higher than the ones from BT549 (0.43) and UACC812 (0.46) cell lines.

### 6.4.3 Evaluation

The performance of the *caspo-ts* method is evaluated using three criteria: 1) RMSE calculation using a typical learning and testing data approach, 2) random data comparison,

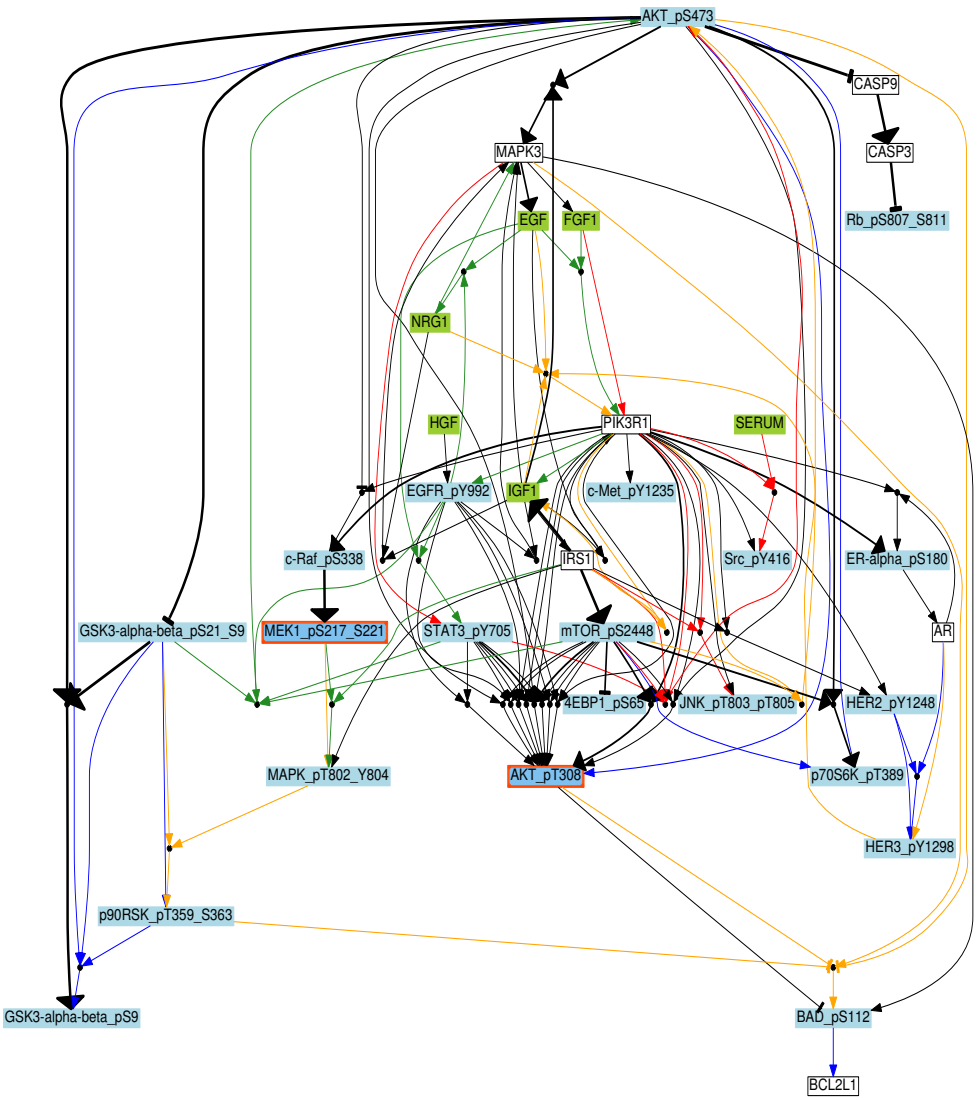



Figure 6.3: **Boolean network of breast cancer cell lines.** The aggregated graph for all cell lines. Blue, red, green and orange edges are used for each cell line BT20, BT549, MCF7 and UACC812, respectively. The nodes are connected by logic gates (AND or OR) to their direct predecessors. Edges are used to show influences ($\rightarrow$ for positive and $\dashv$ for negative). An AND gate is depicted by a small black circle where the incoming edges correspond to the inputs of the gate. An OR gate is depicted by multiple incoming edges to the node. A different color scheme is used to represent different types of nodes. The green color is for stimuli, the red for inhibitors, the blue for readouts, and the white for unobserved nodes. Black edges denote common hyper-edges across cell lines; the thickness of the black hyper-edge denotes the number of cell lines sharing this hyper-edge.

3) AUROC (Area Under the Operating Curve) score.

The BNs are learned using the learning dataset (see Section 6.3.1) only. The prediction

Table 6.1: **Similarity scores among breast cancer cell lines.**

| **Cell Lines** | **Size of BNs' family** | **Similarity Score** | | | |
|---|---|---|---|---|---|
| | | **BT20** | **BT549** | **MCF7** | **UACC812** |
| $BT20$ | 72 | 0.73 | 0.15 | 0.17 | 0.23 |
| $BT549$ | 191 | ** | 0.43 | 0.23 | 0.20 |
| $MCF7$ | 21 | ** | ** | 0.63 | 0.21 |
| $UACC812$ | 20 | ** | ** | ** | 0.46 |

accuracy is evaluated by comparing the RMSE of trajectories in the testing dataset with those recovered by the learned networks (see Equation 6.1). There are two types of RMSE - discrete and model. The *discrete RMSE* is imposed by the discretization of the method. Since we use a discrete learning approach, our recovered traces will be in {0,1} and this introduces an error with respect to continuous measurements in [0;1]. The *model RMSE* refers to the learned BN error with respect to the normalized time series data; that is, the model RMSE is at least as large as the discrete RMSE. When the difference between these two is zero then the inferred BNs are able to recover the discrete trajectories without any error. If the model RMSE is greater than the discrete RMSE then the inferred BNs have some errors in the recoverability of the discrete time series data. To check how our method performs in case of random time series, we have calculated the *RMSE* score for random data and compared it with learning and testing data. Next, the validity of these networks is verified by comparing them with the canonical MTOR signaling pathway using two parameters, *i.e.,* true positive rate (TPR) and false positive rate (FPR).

### Validation using root mean square error criteria

The goal was not only to infer optimal BNs but also to verify that these BNs are able to recover trajectories that do not exist in the learning data. For this purpose, we use experimental testing data to check the specificity of the trajectories of the proposed networks. This testing data is provided by the HPN-DREAM challenge (see Section 6.3.1). Table 6.2 shows the corresponding RMSE in case of learning and testing data. It can be seen that the inferred BNs are able to produce the trajectories without any error in the learning dataset for all cell lines. It is encouraging to see that the inferred BNs are able to recover the discrete testing trajectories without any error in MCF7, and with a minimal error of 0.0009, 0.0106, and 0.0094 in BT20, BT549, and UACC812, respectively.

We also compared the RMSE score with the top two best performers of the HPN-DREAM challenge. We got the top position with an RMSE score of 0.31 as compared to their RMSE scores of 0.47 and 0.50. Notice that in comparison to other HPN-DREAM challenge methods based on Bayesian inference, Regression, and Granger Causality among others, *caspo-ts* does not make new predictions but it checks the recoverability of the testing trajectories with the inferred BNs.

Table 6.2: **Root mean square error.** This table summarizes the RMSE results for each cell line. We have calculated the discrete RMSE (error related to the discretization of the data) and the model RMSE (*caspo-ts* error). The Delta column shows the difference between model and discrete RMSE.

| **Cell Line** | **Learning** | | | **Testing** | | |
|---|---|---|---|---|---|---|
| | **Discrete** | **Model** | **Delta** | **Discrete** | **Model** | **Delta** |
| $BT20$ | 0.3464 | 0.3464 | 0 | 0.3293 | 0.3302 | 0.0009 |
| $BT549$ | 0.3498 | 0.3498 | 0 | 0.3007 | 0.3113 | 0.0106 |
| $MCF7$ | 0.3207 | 0.3207 | 0 | 0.2772 | 0.2772 | 0 |
| $UACC812$ | 0.3464 | 0.3464 | 0 | 0.3084 | 0.3178 | 0.0094 |

### Validation using random data samples

The objective of this analysis is to show that the BNs obtained with *caspo-ts* using the HPN-DREAM datasets for the four cell lines have a lower RMSE score with respect to random trajectories, and therefore are very specific to the HPN-DREAM datasets. For this purpose, we generated 100 random data samples per cell line. In each sample, we generated a random value in $[0; 1]$ for each readout protein in each time point without changing the perturbations. Then, we calculated the RMSE of these samples with respect to the inferred BNs of each cell line, and finally compared it with the learning and testing RMSE of these BNs. Fig 6.4 plots the RMSE ratio (see Equation (6.3)) of the inferred BNs with respect to the learning, testing and random data.

$$RMSE\ ratio = \frac{Discrete\ RMSE}{Model\ RMSE} \tag{6.3}$$

In Fig 6.4, the RMSE ratio for random datasets is displayed by red boxplots for each cell line, and the RMSE ratio for testing and learning datasets is shown as clear outliers in green and blue colors respectively. It is worth noting that the *caspo-ts* method has failed

to recover random data time series, hence proving the specificity of the learned networks with respect to the HPN-DREAM challenge dataset.

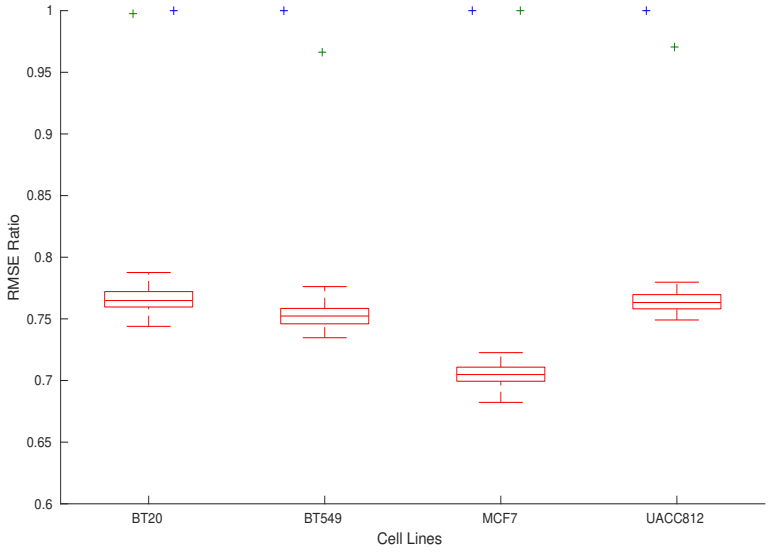



Figure 6.4: **Performance assessment with learning, testing and random datasets.** The x-axis shows the cell line and the y-axis shows the RMSE ratio (see Equation (6.3)) of the inferred BNs from the HPN-DREAM data for each cell line with respect to the three datasets. The three datasets are encoded by different color codes. The RMSE ratio with respect to the HPN-DREAM learning and testing datasets is shown in blue and green colors, respectively. The random dataset RMSE ratio distribution is shown as red boxplots.

Additionally, we computed the RMSE of the testing data by using a *leave one out* approach. For this we generated slightly modified samples, by selecting random values of 5% of the learning data points. The same experimental perturbations and readout proteins were kept. Our results show that the BNs learned from the 5% randomized data have an RMSE of 0.3113 with respect to the testing data, demonstrating *caspo-ts* robustness. For such 5% modified datasets, true positive BNs are difficult to obtain with the model checker; most candidates were false positive models. This highlights the complexity of this BN learning problem when few experimental perturbations are given because the space of candidate ASP-optimal BNs to verify is large and it is heavily populated with false positive Boolean models.

### Validation using MTOR canonical pathway

To perform the validation of the structure of the BNs, we calculated a set of *standard nodes* from our PKN which are downstream nodes of MTOR and belong to the canonical MTOR pathway. We then evaluated how many of these standard nodes are also downstream nodes of MTOR in the learned BNs. In the following, the set of downstream nodes of MTOR in the learned BNs is referred to as *inferred set*. The *inferred set* is specific to each cell line. True positive rate (TPR) and false positive rate (FPR) are defined by Equations (6.4) and Equation (6.5) respectively:

$$TPR = \frac{TP}{TP + FN} \tag{6.4}$$

$$FPR = \frac{FP}{FP + TN} \tag{6.5}$$

Here, TP is the number of nodes in the intersection between standard and inferred sets, FP is the number of nodes in the inferred set but not in the standard set [2], FN is the number of nodes in the standard set but not in the inferred set and TN is the number of nodes which are not in the standard set nor the inferred set.

Fig 6.5 shows the Receiver Operating Characteristic (ROC) curve of each cell line. For BNs of each cell line, TPR and FPR was calculated using Equation (6.4) and (6.5). BT549 cell line models are the most accurate ones followed by MCF7 and BT20. We can observe the clear distinction between true positive and false positive BNs. The BNs inferred by *caspo-ts* have an average AUROC score of 0.77 which is comparable to the AUROC score of 0.78 of the top performing method of HPN-DREAM challenge. A number of assumptions made during the modeling phase may have influenced our ranking. First, since our method can pinpoint the noisy, incomplete and erroneous experiment, it allows us to use only the reliable experimental settings. Second, our method constrains its solutions space to the proteins existing in the PKN, anything outside the prior knowledge cannot be found. From Fig 6.5, we can see that the *caspo-ts* method shows promising results for the inferred true positive BNs.

2. Notice that TP and FP should not be confused with true and false positives from the over-approximation.

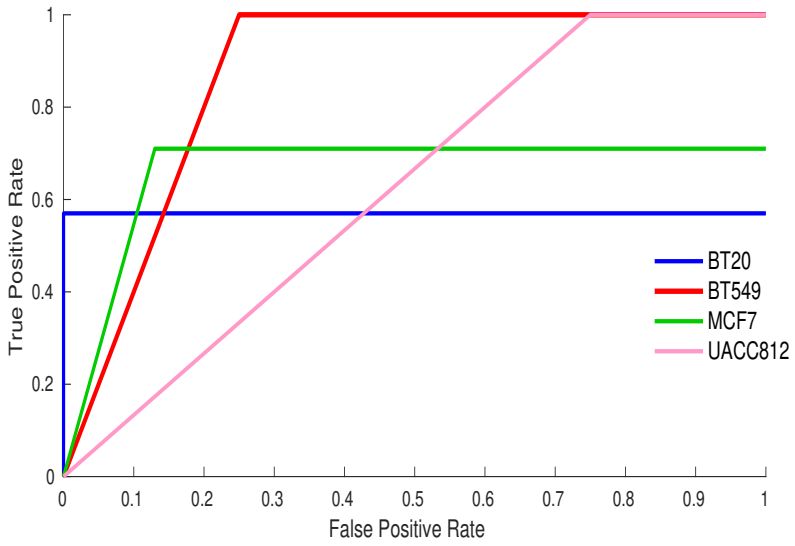



Figure 6.5: **ROC curve across all cell lines.** The x-axis shows the false positive rate and the y-axis denotes the true positive rate. These rates are calculated using equation (6.4) and (6.5). The average AUROC score is 0.77.

## 6.5 Conclusion

In this chapter we presented a study to build cell line specific signaling networks for the HPN-DREAM time series dataset of 4 breast cancer cell lines (BT20, BT549, MCF7, and UACC812) using *caspo-ts*. This method combines Answer Set Programming and Model Checking techniques to infer true positive BNs verifying the experimental data. *Caspo-ts* allowed us to handle a midscale PKN (64 nodes and 178 edges) and a real dataset subject to experimental error. *Caspo-ts* enabled us to learn key dynamic mechanisms within the BNs explaining the time series data. Our results suggest that the behavior of cell line specific signaling networks is highly variable even under the same perturbations, agreeing with the heterogeneity of breast cancer and specifically with previous analysis on this data [162]. The inferred Boolean models of each cell line were analyzed to identify commonalities as well as discrepancies. Moreover, these inferred models can be executed computationally to identify potential drug targets or to see the effect of unseen perturbations. The predictive power of these models can be increased with improvements in protein interaction databases and comprehensive experimental data.

We have discovered 38% of the cell line dependent behaviors as compared to the 33% of

the HPN-DREAM challenge winner [170]. We have implemented an algorithm to analyze the variability among cell lines and observed pairwise similarities among these cell lines. The similarity index varies from 15% (BT20 & BT549) to 23% (MCF7 & BT549, BT20 & MCF7). We have analyzed the similarity among the family of BNs of the same cell line as well, which varies from 43% to 73%. We have evaluated the accuracy of our method with RMSE and AUROC scores. The average RMSE of the inferred BNs was 0.31 placing *caspo-ts* in first place in comparison with the top scores reported in the HPN-DREAM challenge. Various choices made during this study may have an impact on the final score. The *caspo-ts* method allowed us to remove noisy and faulty experiments, leaving us with the reliable experimental settings only. Here, we made the choice to use only the reliable experiments of the learning dataset instead of using all experimental settings. Also, we did not observe all 45 proteins as we could not find connections in our PKN for all the studied proteins, leaving us with approximately 23 proteins for each cell line.

Nonetheless, the obtained results are quite promising, making *caspo-ts* a good candidate computational method for learning models given time series datasets and a prior knowledge network. In addition, *caspo-ts* can be used to pinpoint the errors in the experimental data. In particular, we discovered four experiments where the protein $AKT$ was inhibited and had a dynamic behavior as a readout protein. Our work therefore provides a novel approach to show erroneous experiments which is crucial and complementary to current approaches. Finally, the HPN-DREAM dataset contained some noisy readings of experiments. Noisy experimental data reduces the efficiency of computational methods by increasing the variability among constructed Boolean models. To overcome this, we suggest to build automated methods to filter out the noisy experiments. This approach provides a step forward in building context dependent networks in the case of phosphoproteomic data.

## 6.6 Perspective

A future direction of this work was to investigate several aspects of the *caspo-ts* method, such as (i) the order of the solution space of over-approximated Boolean models; (ii) the computational time for checking reachability; (iii) designing an efficient experimental design strategy and applying it prior to selecting the most informative experiments. Because *caspo-ts* uses an ASP solver to enumerate BNs, in the resulting sequence of solutions similar BNs are typically clustered together. This can be problematic for large scale

problems where we cannot explore the whole solution space in reasonable time. We have worked on sampling to randomly select BNs from the solution space. Further, we studied another technique, which allows for shuffling the order in which solutions are enumerated [171]. Our plan was to implement this by dynamically modifying the heuristic of the ASP solver at execution time. This perspective was finally implemented and discussed on this HPN-DREAM dataset in our publication [172].

Chapter 7

# Conclusion

*"Some changes look negative on the surface*
*but you will soon realize that space*
*is being created in your life*
*for something new to emerge."*
— Eckhart Tolle

This memoir has presented part of my research which was to conceive computational models from or for biological systems. The methods we proposed are mainly based on logic programming using the Answer Set Programming paradigm. They however interact with maching learning methods, statistical analyses, and biological expertise, in order to propose significant computational predictions answering to real biological problems. This chapter summarizes my research results and proposes straightforward perspectives.

## 7.1 Short summary of contributions

### 7.1.1 Biological insights

The biological questions that we have approached and answered with the methods reviewed in this manuscript are:

— The understanding of the regulatory mechanisms, in terms of genes or proteins interactions in Cancer. Mainly, of Multiple Myeloma [47] (see Chapter 3) and Hepatocellular carcinoma [48]. These researches were conducted with biologists, experts on the specif cancer domain. Our method's results were in accordance with biological literature or complementary to classical techniques to find markers.

— The learning of computational models from proteomics training datasets, which can effectively predict testing datasets. These data was provided by the Dialogue on Revearse Engineering Assessment and Methods (DREAM) Challenges. In par-

ticular to understand patient or cell-lines specificity. This type of research was conducted in the context of Acute Myeloide Leukemia [121] (see Chapter 4) and of Breast Cancer [150] (see Chapter 6). Our results were compared in detail with other methods that analyzed the same data. They have produced, on the context of the cited publications, concrete results on mechanisms not previously discovered and better prediction rates.

### 7.1.2 Bioinformatics - automatic recovery of interaction networks

Concerning the bioinformatics field (more specifically the methods that generate interaction networks from databases), even if not approached in this manuscript, it is important to mention the work we undertook in [11], which mines the knowledge inside public databases of genes and protein interactions in order to build causal interaction graphs automatically. This work was based on Semantic Web, and produced very interesting results on the graphs generated, when compared with state-of-the-art methods. It is important to mention that the interactions network, constitute what we called *Interaction Graphs* or *Prior Knowledge Networks*, both essential inputs to all methods proposed in this manuscript.

### 7.1.3 Modeling Biology

Finally, what has mostly occupied the space in my research career until now has been modeling biology. Writing a (logic) program that computes biological hypothesis captures my attention already since 15 years. The models we have proposed are in general very intuitive, since they test rules defining the connection between a biological species with those species that interact with it. However, since graphs connecting biological species are large (in some cases composed of thousands of species) and incomplete, analyzing these models needs to be supported with efficient programming. Also, considering the large mass of experimental data, automatic frameworks offer an excellent help here towards integration and understanding. Most of my research relies on two modeling structures: sign-consistent graphs and Boolean networks.

### Sign-consistent graphs

We have started exploring sign consistency back on 2009 and proposed an stable tool on [44] (see Chapter 2). Sign-consistent graphs, propose a computational way to integrate large-scale networks with genome-wide measurements, such as gene expression data. The uncertain nature of the sign-consistency rule has always questioned my PhD students, and Bertrand Miannay proposed another consistency rule, based on a *perfect coloring rule* [39] (see Chapter 4), which we had very few months to exploit and study its impact. Recently, in the PhD thesis of Sophie Le Bars, she has proposed another sign-consistency rule, based on the *majoritarian sign* (Le Bars et al. *BMC Bioinformatics* under revision). We have shown that this rule allows to constrain more the system, generating predictions which are more sensitive to small perturbations of the system and which are more discrete, and thus closer to the quantitative world, facilitating the integration of gene regulatory modeling with metabolic modeling.

### Boolean models

The modeling of Boolean networks appeared during my postdoctoral research, while being in contact with other type of data, named *phospho-proteomics*, which allows to measure much precisely the system, in terms of phosphorylated proteins. Inspired by previous works, we proposed an exhaustive method to propose a global-optimization, by using Answer Set Programming in order to learn Boolean networks from phospho-protemics datasets. This was the beginning of *caspo* [125] (see Chapter 5). We have proposed further works, in the continuation of *caspo*, such as proposing experimental designs [33] and finally we have extended it to use time-series phosphoproteomics data in *caspots* [150] (see Chapter 6). Finally, the logic of *caspo*, has inspired us to propose pre-processing methods [121] (see Chapter 4) to transform single patients data (having different treatment response) into pseudo-perturbations, in order to model them with *caspo*.

## 7.2 Perspectives

Whereas it is unpredictable for me to know in what type of research domain my energies will be focused on a future. Since 2012 I continue discovering the work of a *maître de conférences*, where lecturing and research responsabilites are combined. I have discovered that this amazing domain of work, subject to human relationships, is as unpredictable as

any biological system. Making a parallel to what I have observed in my research of modeling living systems, taking for granted approximations or trying to force the results by strong constraints, have not led to interesting results in terms of a human balanced aspect concerning my future research directions. Also, I have observed that the more balanced are my considerations of the human aspect, the more balanced the research projects and collaborations are for me. I see how naturally and sometimes in a surprising way, novel research scenarios appear. Similarly to the story of writing this HDR manuscript, without necessarily forcing it, it just comes to light in a precise moment. Having considered that, I would like to write here about three perspectives of continuation, that have naturally appeared, and seemed supported by current collaborations and previous results.

### 7.2.1 Modeling Human embryonic development

Back in 2014, I have came to know the work of Sara-Jane Dunn [173] concerning the discovery of perturbation-based computational models by using SAT Modulo Theories solvers applied to mouse embryonic development. This work was extremely interesting, since convincing the Systems Biology community that solvers on discrete domain problems, may have a chance to model biology, was a hard path for me back on 2014 [1]. Sara-Jane Dunn also knows our work on ASP solvers, she was invited on 2018 to be part of the thesis jury of Misbah Razzaq, a PhD student I co-supervised.

Four years ago, I have started contact with Laurent David, a biologist and bioinformatician, who works in the Center of Research in Transplantation and Translational Immunology (CR2T), affiliated to Nantes Université, studying Human embryonic development. He uses single cell RNA-Seq (scRNA-seq) data. On [174] he published a work in which they used scRNA-seq data to discover hypotheses of cell fate in the context of Human embryonic development. They have proposed key transcription factors controlling this process and we have started discussions with Laurent David concerning the possibility of building computational models, based on scRNA-seq data, explaining Human embryo development. Interestingly, Laurent David, has also approached Sara-Jane Dunn's work, but from the biological perspective. Thanks to the AIBy4 program [2], we have recently been granted a PhD funding, thanks to which we co-supervise Mathieu Bolteau together with Jérémie Bourdon. Our current objective is to pre-process scRNA-seq data in order to

1. By the way, the opposite is also true: convincing formal methods community to approach concrete and real biological data and questions is hard as well.
2. `https://aiby4.ls2n.fr/en/aiby4-english/`

build separate families of Boolean networks specific to each developmental stage. This idea is strongly inspired from the work published in [121], but instead of applying it to single patient proteomic data, we search to apply it to single cell embryo-developmental-stage data. Our first results have been submitted for consideration in the *ISMB/ECCB* 2023 conference. We show a concrete framework where specific Boolean networks are learned from scRNA-seq data of cells at two developmental stages.

### 7.2.2 Modeling biology with SAT and ASP hybrid solvers

In 2013, after finishing the *caspo* publication [125], we had the idea and motivation to collaborate with people in the Integer Linear Programming (ILP) domain, so that an effective comparison between both methods could be established, so we could propose potential users how to approach an specific systems biology problem. It was interesting since ASP [125] and ILP [135] frameworks to model a similar problem were proposed. Unfortunately, shared motivation and geographical circumstances were factors that discouraged at that time exploring this path.

Throughout all my research, it has been always clear, that ASP is just one tool, and not the only way to approach biological modeling. Thanks to our research, we have a clear view of the limitations of this paradigm on the context of modeling biological data. However, for the moment we are still delighted of the results we have obtained so far. Recently, on 2022, we have began a perspective collaboration with the team of Eric Monfroy, who works at the LERIA laboratory, of the University of Angers, in the context of a PhD thesis he currently supervises. The thesis project is to build an hybrid solver combining ASP and SAT solvers. This could benefit our exploration of other solvers of discrete problems in the context of modeling biology.

### 7.2.3 Modeling gene and metabolic networks' integration

In 2017 I have integrated the ComBi research team at the LS2N. We have since then started to collaborate with Jérémie Bourdon on the context of proposing an integrated framework to model gene regulatory network and metabolic networks integration. Flux Balance Analysis of metabolic network is one of the research domains in which Jérémie Bourdon is expert. On 2019 we were granted a PhD funding, in the context of the co-supervision of the PhD thesis of Sophie Le Bars, which research subject was the integration of gene and metabolic modeling. This ambitious project is hardly affordable in 3-years of

PhD studies. However, we are very happy of the results obtained so far, which point to a computational framework very sensible to perturbations on the gene regulatory network, which provides an easy integration to metabolic Flux Balance Analysis (Le Bars et al. *BMC Bioinformatics* under revision).

In 2016, I started a collaboration with Catherine Pellat-Deceunynck, INSERM CRCINA (Center for Integrated Research in Cancerology and Immunology Nantes Angers), to model the response of a group of Multiple Myeloma cell lines subject to multiple drugs, for which their viability was measured. Gene expression profiles of each cell line was also measured and made available. Our objective was to model this response. We had in that time resources for 1.5 year of postdoc. Despite of the fact that an interaction graph was proposed, our project did not lead to any concrete result due to multiple reasons. One of the reasons was that the predictions of Iggy (see Chapter 2) were little constraining the system after each drug perturbation, no difference was observed. This collaboration allowed us to propose a research project we subitted to the ANRJCJC call on 2019 (see Section 7.3.1), that was rejected. With the current methodological framework, proposed in the PhD thesis of Sophie Le Bars, revisiting this biological application appears as an interesting perspective to follow.

## 7.3 Research Project

In 2018, we answered to the call ANRJCJC 2019, in the section *Mathématique, informatique, automatique, traitement du signal pour répondre aux défis de la biologie et de la santé (CES 45)* with a research project named HOLDinG, which stands for: Hybrid constraint mOdeLing to bridge cell lines Drug response to Gene expression. In this section we describe the main objectives of this proposal.

### 7.3.1 Context, positioning and objectives

**HOLDinG** aims to explore cause-effect graph strategies for personalized medicine. In the last years biologists have produced huge amount of transcriptomic, proteomic, and metabolomics data, to measure patients' molecular species expression or activity at a given state for a specific cellular or tissue type in order to study the impact of a more personalized treatment. Statistical methods are often proposed [175] to detect a signature of gene-expression profiles that differentiates patient expression profiles and, more

importantly, classifies patients with different response to treatments. In particular, whole-genome tumor gene expression was shown to act as a proxy for unmeasured phenotypes and was used to predict patient response to drugs in cancer research. This approach has given interesting results in terms of highlighting genes correlated to functional response and as a way of classifying cell lines expression profiles into correct drug response categories [176]. Methods based in protein-protein networks have been used to understand gene-expression profiles heterogeneity [177], and signaling networks have been shown to be highly rewired in cancer [178]. In our proposal we want to model gene expression measurements through mechanistic network-based approaches to explain drug response. To this end, we will employ constraint-based methods mixing Logic Programming (LoP) and Linear Programming (LiP). With this hybrid framework we will model a multi-layer network, including signaling, gene-regulation, metabolic, and drug treatments information. This in-silico model will represent individual cell lines and will be used to understand cell lines heterogeneity in a mechanistic sense. The biological model for our approach will be composed of 36 human myeloma cell lines and 11 mantle cell lymphoma cell lines, derived from patients with multiple myeloma (MM) or mantle cell lymphoma (MCL), that were subject to 20 treatments, which include clinical drugs and pathways inhibitors.

In comparison to protein-protein networks approaches, we think that providing a computational model with its related automatic reasoning framework is much richer than extracting significant subgraphs which do not have a predictive power, and that do not take into account the logic (activations, inhibitions) of the molecular interactions. Compared to statistical models analyses such as ridge regression, our method exploits mechanistic constraints on the variables that may be more effective for restricting the number of predictive features. This was shown for LoP models in [47], where we identified prognosis markers from transcriptomic data which were as significant as state-of-the-art clinical markers for the same disease. In recent works we have proposed independent LoP frameworks to model experiments and networks integration. We proposed *caspo* to train signaling networks to experimental data in [125] and Iggy to model gene regulatory networks and data integration in [44]. In [151] and [150] we showed that dynamical behaviors can be approximated with LoP models and finally verified through model checkers. Interestingly our predictions [101] show less error than quantitative (ODE [3]-based) models for the same data (DREAM Challenge 8 [4]). For metabolic networks reconstruction a LoP framework was

3. Ordinary Differential Equations
4. `http://dreamchallenges.org/project/dream-8-hpn-dream-breast-cancer-network-inference-challenge/`

proposed in [179] and recently a hybrid approach integrating LiP solvers combined topological and stoichiometric constraints to build optimal metabolic network models [180]. In HOLDinG we want to combine these separate frameworks into a unique one, as sketched on [181]. Our integrated model will predict the quantitative outcome of cell viability upon a drug treatment. HOLDinG's objectives are:

**Objective 1 - Network construction from Microarray data and public repositories.** The biological networks to be studied are of two types: (RN) regulatory and signaling and (MN) metabolic. Both are meant to be generic networks and thus composed of mechanisms implied in different Human cell types. To construct (RN), we will automatically query public pathway repositories and build a network containing signaling, regulatory and drug-gene targets layers. For this, we will use the BRAvo tool [11]. BRAvo allows querying pathway databases represented in RDF 3 , such as Pathway Commons [182], using Semantic Web standards from a list of target genes. BRAvo's output is a signed and oriented network that contains the up-stream events of the target genes according to the knowledge in the queried databases. These target genes list will be composed of the differentially expressed genes of the Microarray experiments across all MM and MCL cell lines. To construct the metabolic network (MN), we plan to use the INIT methodology and extract a specific map of Human metabolic pathways using the target genes. The (MN) is the basis of the LiP model. The causal relations between the (RN) nodes and the metabolic genes of (MN) will be inferred using a method implemented within the BRAvo framework. This combined graph between (RN) and the causal relations previously described will be named hybrid network (HN).

**Objective 2 - Computational models.** The HN network nodes edges and their control type (*e.g.* activator or inhibitor) will be represented as predicates in the logic program, LoP model instance. The experimental qualitative measurements over specific cell line targetgenes (*e.g.* up-, down-regulation) will also be represented as predicates and added in the LoP model instance. Afterwards, a set constraints will be added to the LoP model in order to reason over this instance. Questions such as *are all the up-, down-regulation levels agreeing with the logic of the graph?*, or *what novel system behavior can be inferred from this agreement?* can be addressed using existing logic programs such as Iggy based on ASP. The idea of these constraints is to integrate experimental measures with generic regulatory knowledge allowing the specialization of the model behavior to a specific MM

or MCL cell line behavior These constraints need to be redefined and extended in order to: (1) include the signaling pathways reaction logic, and (2) predict a robust relation between a drug and a gene expression level

**Objective 3 - Model validation and prediction.** The LoP model inference of the *metabolic genes* will allow LoP and LiP models communication. In particular, for each cell line, the reactions catalyzed by the *metabolic genes* with a low LoP model inferred state (given a particular drug), will be removed from the LiP model and a quantitative metabolic (biomass) response of the system will be computed using the standard Flux Balance Analysis methodology by using the Cobra Toolbox, and studied for each drug of the panel. The domain of flux parameters, used for the LiP computation (after LoP model prediction), will be refined with the help of experimental observations of the drug effect on cell viability. That is, if the quantitative response (model prediction) does not agree with the experimental functional observation, the LoP-LiP models can be revised. Once a valid model is built for each cell line, two key questions will be answered by solving optimization problems over integer and real values: (i) optimal drug selection and (ii) cell-lines treatment response classification.

**Objective 4 - Proof computational models soundness: experimental validation.** In Objective 3 the in-silico analyses will reveal a list of mechanisms (molecular species or reactions) in the models that discriminate the behavior of different cell lines drug-responses. We plan the validation of these mechanisms by using:

— Sequencing data validation. We will use cell mutational status to proof if the predicted mechanisms correlate with sequence mutations.

— Wet lab validation. The discriminating molecular species will be validated by classical under and over expression as well as, if existing, by the use of specific pharmacological inhibitors. Validation of several isolated genes has been done previously using the cell line collection. Microarray data [183] allowed the identification of genes involved in clonogenic growth [184], as well as druggable genes related to patient's subgroup [185] or to mutations [186]. Second, combinations of drugs/inhibitors will be experimentally assessed in cell lines predicted to respond positively and negatively to the combinations. Calculation of antagonism, additivity and synergism will be performed for all predicted combinations using a large panel of cell lines. Third, all results will be assessed in primary tumor cells, which will be charac-

terized for the identified gene(s) expression, and/or for their belonging to a defined molecular subgroup (cytogenetic subgroup of patients, mutations).

### 7.3.2 Partnership

HOLDinG is a young investigator project coordinated by Carito Guziolowski, CG, 36 years old [5]. She is an Associate Professor in Computer Science, that obtained a CNRS excellence chair 2012-2017 in Bioinformatics, and she teaches at the Ecole Centrale de Nantes. CG's research is made within the ComBi team at the LS2N (*Laboratoire des Sciences du Numérique de Nantes*). CG is currently co- supervising 1 postdoc and 1 PhD thesis. She has previously co-supervised 2 PhD theses that were defended on Nov. 2016 and Dec. 2017, 1 Postdoc (Nov. 2016 - Nov. 2017), 1 Engineer (Sep. 2016 - Sep. 2017), and 2 Master internships. CG has co-authored 32 publications in peer reviewed journals and conferences in computational systems biology including PLoS Comp Biol, Scientific Reports, FEBS Journal, Bioinformatics, and Theo Comput. Sci.

This project consortium is composed of 3 partners: Prof. Jérémie Bourdon, JB, (LS2N), Dr. Catherine Pellat-Deceunynck, CP, (DR2 CNRS, INSERM, Nantes) and Prof. Torsten Schaub, TS, (Potsdam University, Germany). JB's expertise on constructing and modeling metabolic networks using LoP and LiP frameworks will be valuable for this project. The INSERM partner provides the biological and experimental validation expertise, which are precious elements to choose the modeling hypotheses and have fast feedback on modeling predictions. TS and colleagues have recently developed a new version of the clingo solver to conceive logic programs using linear constraints over real and integer domains. His expertise will be valuable to bridge the LoP models concerning signaling and gene regulatory networks with LiP models of metabolism behavior and drug response.

To achieve the objectives for the project we ask for a total amount of 250k€ for 48 months: 200k€ to fund two Master internships, 1 PhD position, and 1-year postdoctoral position that will work on Objectives 1-3; 40k€ for missions; and 10k€ for computer science equipment. Complementary to this funding we have a funding for 18 months engineer position until March 2018 to design and implement software that builds formal models from Pathways Databases (LS2N) via the SyMeTRIC Pays de la Loire regional project. The INSERM team will perform experimental validation of the computational predictions (Objective 4) by assigning 1 INSERM permanent member for each biological

5. This collaboration was set up in 2018, the information given for all members corresponds to that date

cell line type: MM, MCL.

# List of Tables

3.1 **Frequency score of Iggy's predictions for the NPC and MM subjects**. The references column lists the publications that agreed with our sign prediction. Connectivity refers to the ratio of genes connected to each predicted node. The OVE (observed variant expression) shows the number of variant gene expressions, using the best precision threshold, across all the GEPs. . . . . 41

3.2 **Node's perturbation results**. Each node was perturbed in two directions (column Dir): +, activation and -, inhibition. TPS represents the frequency with which perturbing a node in a specific direction was significant (*i.e.* it generated a high, 10% top, SCENFIT score) across the MM profiles ($TPS^{MM}$) or NPC profiles ($TPS^{NPC}$). The highlighted rows contain percentages which refer to perturbations that have a direction opposite to that of the predicted signs obtained with the frequency score (Table 3.1). P.val was obtained using a unilateral Fisher test. . . . . 43

3.3 **Parameters associated with MM overall survival**. HR stands for hazard ratio and CI, for confidence interval. . . . . 45

6.1 **Similarity scores among breast cancer cell lines.** . . . . . 91

6.2 **Root mean square error.** This table summarizes the RMSE results for each cell line. We have calculated the discrete RMSE (error related to the discretization of the data) and the model RMSE (*caspo-ts* error). The Delta column shows the difference between model and discrete RMSE. . . . . 92

# List of Figures

2.1 **Gene expression.** The sequence of steps needed to transform a double-stranded DNA molecule into a functional protein. . . . . 10

2.2 **Small transcriptional regulatory network.** Extract of the transcriptional network of genes and proteins in *Escherichia coli*. The names in capital letters correspond to TFs (proteins): HU and CRP, that can activate or repress other genes transcription. Arrows ending with "->" or "-|" imply that the initial product activates or, respectively, represses production of the product of arrival. . . . . 12

2.3 **RSTC (receptor - signaling - transcription - cellular process) network** This network was generated from the Pathway Interaction Database explaining the up-regulated genes induced after HGF (Hepatocyte Growth Factor) stimulation in Human cells. The graph legend is provided in the lower box. The seed nodes are composed of: HGF receptor nodes, the protein nodes where two-fold up-regulated genes could be overlayed, and the cell migration and proliferation nodes. These nodes were used to generate the RSTC network. . . . . 13

2.4 **Biological network represented as an influence graph** A regulatory network (A) mapped into an influence graph (B). Interactions among molecules create an influence graph. The arrows in the influence graph represent a positive (+) or negative (-) influence. . . . . 14

2.5 **Influence or interaction graph** IG with a positive feedback loop between E and F. . . . . 17

2.6 **Iggy : discretization of experimental observations** We assume the real values of observed experimental measurements on molecular species given in a continuous scale. When imposed a set of thresholds $t_i$, the real values are discretized into 5 discrete values. . . . . 18

2.7 **Iggy : summary of local reasoning constraints.** IGs with different labelings where green stands for increase, red for decrease, and blue for 0-change. All labelings satisfy the basis of Constraint 2 for node D, but only the labelings a-d satisfy also Constraint 3. . . . . 20
2.8 **Iggy : global sign consistency.** Example for an IG with partial labeling, which is locally consistent for A and B, but globally inconsistent because there exists no single total labeling satisfying Constraint 2 for A and B. . 21
2.9 **Iggy: checking for consistency.** Consistent total labelings of the example in Fig. 2.5 under different consistency notions. In this example, there is a partial labeling given over 3 nodes in the graph, and 43 possible coloring models or labelings consistent at least under one consistency notion (WP or SP) are fully displayed. Note that all possible labelings are $3^5$. A grey cell indicates that the labeling above is consistent and a white cell with "." means that it is not a consistent labeling. . . . . 22
2.10 **Iggy : minimal correction sets (MCOS) repair** Given the inconsistent labeling presented in the top-left IG, there are three alternative repair sets: repair set (a) adds a positive influence to $A$, repair set (b) includes a negative influence on $B$, and repair set (c) includes a positive influence on $A$ and a negative influence on $B$. Repair sets (a) and (b) are minimal (1 repair), while repair set (c) is not minimal (2 repairs). From the set of consistent labelings under minimal repairs, (a) and (b), the strong predictions are $pred(A) = +, pred(B) = -$, and the weak predictions is $pred(C) = \pm$. 25
2.11 **IG and expression profile.** Signed and oriented IG: red arrows refer to inhibitions, while red to activations. The partial labeling is given for two nodes: *rpsP* and *rpmC*, both up-regulated (observed as '+'). . . . . 27
2.12 **Answers sets.** Result from the execution with clingo v4 of the encoding in Listing 2.1. There is one answer set showing the different `labelV` chosen. 28

3.1 **Interaction graph explaining the differentially expressed genes in Multiple Myeloma.** Representation of the subgraph obtained from the PID-NCI database. CD40, IL6 and IGF1 (the nodes in the top portion of the graph) are the 3 queried pathways. The 529 genes that are differentially expressed across all profiles are merged for this representation in the node "Gene set". We used the same syntax for all nodes in this study. The edges from the "Gene set" node to proteins have been deleted for the sake of clarity. 38

3.2 **Decision tree** based on Iggy's predictions of MM subjects RNA expression data with respect to normal ones. . . . . 40

3.3 **Survival curve from Iggy's predictions applied to MM.** (Left) Gene expression of FOXM1 among MM datasets with or without the prediction (FOXM1*[c], -). (Right) Overall survival (OS) of patients with or without prediction (FOXM1*[c], -). . . . . 44

3.4 **Survival curves.** Overall survival (OS) in patients depending of inhibition of the *G1/S transition of mitotic cell cycle* node (left) and inhibition of *RB1/E2F1-3/DP[n]* (right). . . . . 45

4.1 **Overview of the perfect coloring modeling framework.** Arrows refer to processing steps and yellow boxes refer to input/output data. . . . . 53

4.2 **Components identification by perfect coloring approach**. The 15 components identified from all the perfect coloring models generated from the PID-NCI database (2269 nodes, 2683 edges). The components composed only of one gene are labeled with the Uniprot identifier. . . . . 57

4.3 **Distribution of similarity score from the perfect coloring approach across two expression profiles of two different patient cohorts (UAMS and HOVON) for the same graph component**. The perfect coloring method detected 30 components in the graph obtained from the Trrust database using the differentially expressed genes of the gene expression profiles (GEP) of 2 research centers (UAMS, HOVON). These GEPs were provided by the Multiple Myeloma DREAM challenge. The similarity scores of each patient with respect to the genes of the component are shown in the x-axis and represent how the perfect coloring values from the component match the continuous data of the GEPs provided by both independent platforms. . . . . 58

4.4 **Workflow of our method.** (a) PKN construction. In this step we pass the proteins present in our DREAM 9 dataset as input to the Cytoscape plug-in Reactome FI to construct the PKN. This plug-in finds all the paths between the input proteins across several databases, after that we select only relations coming from KEGG. (b) Protein and patient selection. This step consists on selecting $k$ proteins from the dataset for which there is a maximum number of pairs of patients that have identical values in the $k$ proteins but that belong to different response classes. (c) Learning. This step consists on finding the BNs for the two classes CR-PR corresponding to the two datasets obtained in step (b). . . . . 60

4.5 **Union of optimal BNs learned from the initial PKN and the reduced patients dataset from the complete remission (CR) and the primary resistant (PR) classes**. The thicker edges represent those that are the most frequent paths in the BN family. The association between a node and its predecessors is an AND gate if it is preceded by a filled black circle and an OR gate otherwise. Left: Boolean networks for CR patients. This BN can explain and predict the measurements of readouts STMN1 and GAPDH starting from the stimuli FN1 and SMAD6, passing by the inhibitors LEF1, ERBB3, IGF1R and MAPK9, and other intermediate proteins. Right: Boolean networks for PR patients. This BN explains and predicts the measurement of readouts PTGS2, TSC2, BAK1 and CASP3 starting from the stimuli FN1, YAP1 and STK11, passing by the inhibitors ERBB3, IGF1R and CASP9, and other intermediate proteins. . . . . 62

5.1 **High-level design of caspo.** (1) Inputs file are a PKN in Cytoscape's SIF format, and a dataset as a CSV file in the MIDAS format. (2) Preprocessing routines by CellNOpt. (3) Finds an optimum model. (4) Finds all models within the tolerance. (5) Outputs all models found. . . . . 69

5.2 Optimal logic models for HepG2 cells . . . . . 71

5.3 **Sub-optimal models generated with *caspo* with 10% error tolerance.** (**A**) Network of the union of 11,700 sub-optimal models. Green nodes represent ligands that are experimentally stimulated. Red (or red-bordered) nodes represent those species that are inhibited with an small molecule inhibitor (drug). Blue nodes represent species that were measured using the Luminex technology. White nodes are neither measured nor perturbed. AND gates in the models are represented by empty boxes. The thickness of the hyperedges correspond to their frequencies among the 11,700 sub-models. (**B**) Four pairs of mutually-exclusive modules (blue hyperedges in **A**) and their corresponding frequencies on top. These modules determine the behavior of three nodes in the network: mek12, mkk4, and p38. . . . . 73

5.4 **Distribution of sub-optimal models.** The sub-optimal models are ordered (from left to right) first according to their MSE, and then according to their 91 GTT. The number of different models leading to the same GTT is plotted in vertical bars. GTT are ordered and colored by their MSE. The 16 optimal models correspond to MSE 0.0499. The 2 most common GTT describe the response of 3126 and 2090 models. . . . . 74

5.5 **Hierarchical clustering of GTTs.** Hierarchical clustering of the 91 GTT based on their predictions for the readouts across all experimental conditions. Bars length on the leafs represents the corresponding MSE value for each GTT. The optimal GTT (61) is highlighted, as well as the two most common ones (85 and 77). The most common GTT is very close to the optimal one, whereas the second most common GTT has a quite different behavior. . . . . 75

6.1 ***Caspo-ts* workflow.** *Caspo-ts* receives as input data a prior knowledge network (PKN) and a discretized phosphoproteomic dataset. In this example the phosphoproteomic data consists of two perturbations involving *akt* (inhibitor) and *hgf* (stimulus): 1) $akt = 0$, $hgf = 1$ and 2) $akt = 1$, $hgf = 0$. A black colored perturbation means the inhibitor or stimulus was perturbed (1) while white represents the opposite (0). Readouts are specified in blue and describe the time series under given perturbations. Using this input data, *caspo-ts*, performs two steps: ASP solving and model checking. In the ASP solving step: (i) a set of BNs, compatible with the PKN, is generated, (ii) afterwards an *over-approximation constraint* is imposed upon each candidate BN to filter out invalid BNs, that do not result in an over-approximation of the reachability between the Boolean states given by the phosphoproteomic dataset, and finally (iii) BNs are optimized using an objective function minimizing the distance to the experimental measures. The ASP step also introduces repairs in some data points of the time series that added penalties to the objective function. These corrected traces will be given to the model checker. In the model checking step, the exact reachability of all the (binarized and corrected) time series traces in the family of BNs is verified. . . . . 83

6.2 **Breast cancer signaling pathway.** This figure shows the reconstructed signaling network from a combination of databases. An arrow shows the positive regulatory relationship between two proteins, while a T shaped arrow indicates inhibition. Green nodes are stimuli, blue nodes are readouts, white nodes are unmeasured or unobserved, and blue nodes with a red border represent inhibitors and readouts at the same time. Please note that in the node labels, we have added the phosphorylation sites to the protein names in order to connect them to the experimental measurements. . . . . 88

6.3 **Boolean network of breast cancer cell lines.** The aggregated graph for all cell lines. Blue, red, green and orange edges are used for each cell line BT20, BT549, MCF7 and UACC812, respectively. The nodes are connected by logic gates (AND or OR) to their direct predecessors. Edges are used to show influences ($\rightarrow$ for positive and $\dashv$ for negative). An AND gate is depicted by a small black circle where the incoming edges correspond to the inputs of the gate. An OR gate is depicted by multiple incoming edges to the node. A different color scheme is used to represent different types of nodes. The green color is for stimuli, the red for inhibitors, the blue for readouts, and the white for unobserved nodes. Black edges denote common hyper-edges across cell lines; the thickness of the black hyper-edge denotes the number of cell lines sharing this hyper-edge. 90

6.4 **Performance assessment with learning, testing and random datasets.** The x-axis shows the cell line and the y-axis shows the RMSE ratio (see Equation (6.3)) of the inferred BNs from the HPN-DREAM data for each cell line with respect to the three datasets. The three datasets are encoded by different color codes. The RMSE ratio with respect to the HPN-DREAM learning and testing datasets is shown in blue and green colors, respectively. The random dataset RMSE ratio distribution is shown as red boxplots. . . . . 93

6.5 **ROC curve across all cell lines.** The x-axis shows the false positive rate and the y-axis denotes the true positive rate. These rates are calculated using equation (6.4) and (6.5). The average AUROC score is 0.77. . . . . 95

Chapter 8

# CV

# Carito Guziolowski, Ph.D.

carito.guziolowski@ls2n.fr
https://sites.google.com/site/caritoguziolowski
(+33) 2 51 12 58 85


## Education

2006 - 2009 — Ph.D. in Computer Science, (*mention: très honorable*) at Université de Rennes 1, Rennes, France. Dissertation Topic: "Analysis of Large-Scale Biological Networks with Constraint-Based Approaches over Static Models". Advisor: Anne Siegel.

2005 - 2006 — Master in Bioinformatics (*mention: bien*) at Université de Rennes 1, Rennes, France.

2002 - 2005 — Engineer in Computer Science (graduated with high distinction) at Universidad de Chile, Santiago, Chile.

1999 - 2002 — Licentiate in Engineering Sciences mention Computer Science (graduated with distinction) at Universidad de Chile, Santiago, Chile.

## Professional Experience

since 2012 — *Maître de Conférences à l'Ecole Centrale de Nantes*, LS2N.

2009-2012 — Postdoctoral fellow at TIGA Center, Medical Systems Biology, Department of Medical Biometry and Informatics, University Hospital Heidelberg, Heidelberg, Germany.

2006-2009 — PhD candidate at INRIA Rennes, Symbiose team, Rennes, France. Funding: Conicyt-Ambasse de France scolarship.

2006 — Master Degree Internship at Symbiose Project - INRIA Bretagne Atlantique, Rennes, France. "Testing a new approach of qualitative modelling in *Escherichia coli* transcriptional regulatory network".

2005 — Research Engineer at Functional Genomics for the Nectarine Fruit - Universidad Andres Bello, Santiago, Chile. Developing a data management system for large gene sequencing and annotation projects.

Research Engineer at Centre of Mathematical Modelling - Universidad de Chile, Santiago, Chile. Installation of GenDB, a genome annotation system for prokaryotic genomes.

## Awards

2019-2023 — PEDR from the *Ministère de la Recherche et de l'Enseignement Supérieur*.

2012-2017 — Excellence CNRS chair in Bioinformatics.

2006-2009 — Franco-Chilean Scholarship for doctoral studies granted by CONICYT[1] and the French Embassy.

2005-2006 — Franco-Chilean Scholarship for an internship experience granted by CONICYT and INRIA[2].

[1] Chilean National Commission for Scientific and Technological Research
[2] French National Institute for Research in Computer Science and Control

## Publications

### Peer reviewed journal articles

2023 S Le Bars, M Bolteau, J Bourdon, and **C Guziolowski**. "Predicting weighted unobserved nodes in a regulatory network using Answer Set Programming" *BMC Bioinformatics* (accepted, to appear).

2021 Lefebvre M, Gaignard A, Folschette M, Bourdon J, **Guziolowski C.** "Large-scale regulatory and signaling network assembly through linked open data". *Database (Oxford).* 2021 Jan 18;2021:baaa113. doi: 10.1093/database/baaa113. PMID: 33459761.

2020 Folschette M, Legagneux V, Poret A, Chebouba L, **Guziolowski C.**, Théret N. "A pipeline to create predictive functional networks: application to the tumor progression of hepatocellular carcinoma". *BMC Bioinformatics.* 2020 Jan 14;21(1):18. doi: 10.1186/s12859-019-3316-1.

2018 Razzaq M, Paulevé L, Siegel A, Saez-Rodriguez J, Bourdon J, **Guziolowski C.** "Computational discovery of dynamic cell line specific Boolean networks from multiplex time-course data". *PLoS Comput Biol.* 2018 Oct 29;14(10):e100653.

Fourati S, Talla A, Mahmoudian M, Burkhart JG, Klén R, Henao R, Yu T, Aydın Z,Yeung KY, Ahsen ME, Almugbel R, Jahandideh S, Liang X, Nordling TEM, Shiga M, Stanescu A, Vogel R; Respiratory Viral DREAM Challen"A crowdsourced analysis to identify ab initio molecular signatures predictive of susceptibility to viral infection". *Nat Commun.* 2018 Oct 24;9(1):4418. doi: 10.1038/s41467-018-06735-8.

L Chebouba, D Boughaci, **C Guziolowski.** "Proteomics versus clinical data and stochastic local search based variable selection for Acute Myeloid Leukemia patients' classification". *Journal of Medical Systems* 42 (7) 2018.

B Miannay, S Minvielle, F Magrangeas, **C Guziolowski.** "Constraints on signaling networks logic reveal functional subgraphs on Multiple Myeloma OMIC data". *BMC Systems Biology BMC series – open*, 12(Suppl 3):32 2018.

L Chebouba, B Miannay, D Boughaci, **C Guziolowski**. "Discriminate the response of Acute Myeloid Leukemia patients to treatment by using proteomics data and Answer Set Programming". *BMC Bioinformatics* 2018 19(Suppl 2):59 doi: https://doi.org/10.1186/s12859-018-2034-4.

A. Poret, **C. Guziolowski**. "Therapeutic target discovery using Boolean network attractors: improvements of kali". *Royal Society Open Science* 5(2):171852 2018 DOI: 10.1098/rsos.171852.

2017 B Miannay, S Minvielle, O Roux, P Drouin, H Avet-Loiseau, C Guérin-Charbonnel, W Gouraud, M Attal, T Facon, N C Munshi, P Moreau, L Campion, F Magrangeas, **C Guziolowski**. "Logic programming reveals alteration of key transcription factors in multiple myeloma". *Scientific Reports* 2017. 7(1) 9257 doi:10.1038/s41598-017-09378-9.

L. Fippo Fitime, O. Roux, **C. Guziolowski***, L. Paulevé*. "Identification of Bifurcation Transitions in Biological Regulatory Networks using Answer-Set Programming", *Algorithms Mol Biol.* 2017 Jul 20;12:19. doi: 10.1186/s13015-017-0110-3.

S. Videla, J. Saez-Rodriguez, **C. Guziolowski**, A. Siegel, "caspo: a toolbox for automated reasoning on the response of logical signaling networks families", *Bioinformatics* 2017 Mar 15;33(6):947-950.

2016 M. Ostrowski, L. Paulevé, T. Schaub, A. Siegel, **C. Guziolowski**, "Boolean Network Identification from Perturbation Time Series Data combining Dynamics Abstraction and Logic Programming", *Biosystems*, 2016, Nov; 149:139-153.

## Publications (continued)

- V. Acuna, A. Aravena, **C. Guziolowski**, D. Eveillard, A. Siegel, A. Maass. "Deciphering transcriptional regulations coordinating the response to environmental changes". *BMC Bioinformatics*, 17:35 , 2016.
- A. Kittas, A. Delobelle, S. Schmitt, K. Breuhahn, **C. Guziolowski**, N. Grabe. "Directed random walks and constraint programming reveal active pathways in HGF signaling". *FEBS Journal*, 2016 Jan; 283(2):350-60.

2015
- S Thiele; L Cerone; J Saez-Rodriguez; A Siegel; **C Guziolowski***, S Klamt*. "Extended Notions of Sign Consistency to Relate Experimental Data to Signaling and Regulatory Network Topologies". *BMC Bioinformatics* 2015, 16:345.
- S. Videla, I. Konokotina, L. Alexopoulos, J. Saez-Rodriguez, T. Schaub, A. Siegel, **C. Guziolowski** "Designing experiments to discriminate families of logic models". *Frontiers in Bioengineering and Biotechnology*, 2015, DOI=10.3389/fbioe.20.

2014
- S Videla, **C Guziolowski**, F Eduati, S Thiele, M Gebser, J Nicolas, J Saez-Rodriguez, T Schaub, A Siegel. "Learning Boolean logic models of signaling networks with Answer Set Programming". *Theoretical Computer Science*, vol. 599 pp 79-101 2014.

2013
- **C. Guziolowski***, S. Videla*, F. Eduati, S. Thiele, T. Cokelaer, A. Siegel, J. Saez-Rodriguez. "Exhaustively characterizing feasible logic models of a signaling network using Answer Set Programming". *Bioinformatics* 2013; 29 (18) :2320-6. doi: 10.1093/bioinformatics/btt393.

2012
- **C. Guziolowski**, A. Kittas, F. Dittmann, N. Grabe. "Automatically generating causal networks linking growth factor stimuli to functional cell state changes". *FEBS Journal* 2012, 279, 18:3462-74.

2010
- **C Guziolowski**, S Blachon, T Baumuratova, G Stoll, O Radulescu, A Siegel. "Designing Logical Rules to Model the Response of Biomolecular Networks with Complex Interactions: An Application to Cancer Modeling". *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 8, no.5, pp. 1223-1234, 2010.

2009
- **C Guziolowski**, A Bourde, F Moreews, A Siegel. "BioQuali Cytoscape plugin: analysing the global consistency of regulatory networks". *BMC Genomics* 2009, 10:244 doi:10.1186/1471-2164-10-244.

2008
- P Veber, **C Guziolowski**, M Le Borgne, O Radulescu, and A Siegel. "Inferring the role of transcription factors in regulatory networks". *BMC Bioinformatics* 2008 9: 228 doi:10.1186/1471-2105-9-228.

2007
- **C Guziolowski**, P Veber, M Le Borgne, O Radulescu, and A Siegel. "Checking Consistency Between Expression Data and Large Scale Regulatory Networks: A Case Study" *Journal of Biological Physics and Chemistry* 7 2007 37-43
- G.Didier and **C.Guziolowski**. "Mapping Sequences by Parts" *Algorithms for Molecular Biology* 2007, 2:11doi:10.1186/1748-7188-2-11.

2006
- M.Latorre, H.Silva, J.Saba, **C.Guziolowski**, P.Vizoso, V.Martinez, J.Maldonado, A.Morales, R.Caroca, V.Cambiazo, R.Campos-Vargas, M.Gonzalez, A.Orellana, J.Retamales, L.A.Meisel "JUICE: a data management system that facilitates the analysis of large volumes of information in an EST project workflow" *BMC Bioinformatics* 2006,7:513 doi:10.1186/1471-2105-7-513.

### Conference proceedings

2023
- M Bolteau, J Bourdon, L David, and **C Guziolowski**. "Inferring Boolean networks from single-cell human embryo datasets: proof of concept with trophectoderm maturation". *CMSB 2023* (submitted).

## Publications (continued)

2020 Le Bars S., Bourdon J., **Guziolowski C.** "Comparing Probabilistic and Logic Programming Approaches to Predict the Effects of Enzymes in a Neurodegenerative Disease Model". In: Abate A., Petrov T., Wolf V. (eds) Computational Methods in Systems Biology. *CMSB 2020*. Lecture Notes in Computer Science, vol 12314. Springer, Cham.

2018 Razzaq M., Kaminiski R., Romero J., Schaub T., Bourdon J., **Guziolowski C.** "Computing Diverse Boolean Networks from Phosphoproteomic Time Series Data". In *CMSB 2018*, Brno, Czech Republic, September 2018. LNCS, vol 11095, pp 59-74

2017 Miannay B., Minvielle S., Roux O., Magrangeas F., **Guziolowski C.**. "Constraints On Signaling Networks Logic Reveal Functional Subgraphs On Multiple Myeloma OMIC Data". Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, *BCB 2017*, Boston, MA, USA, August 20-23, 2017, pp 768-69

2016 L Fitime, C Schuster, P Angel, O Roux, and **C Guziolowski**. "Integrating time-series data in large-scale discrete cell-based models". In A. Abate and D. Safranek (Eds.): *HSB 2015* - 4th International Workshop on Hybrid Systems Biology, Madrid, Spain. LNCS 9271, pp. 75–95, 2016.

2015 M Ostrowski*, L Paulevé*, T Schaub, A Siegel, and **C Guziolowski**. "Boolean Network Identification from Multiplex Time Series Data". In CMSB 2015 - 13th conference on Computational Methods for Systems Biology, Nantes, France, September 2015. LNBI 9308, pp 170-181.

2012 S. Videla*, **C. Guziolowski***, F. Eduati, S. Thiele, N. Grabe, J. Saez-Rodriguez, and A. Siegel. "Revisiting the Training of Logic Models of Protein Signaling Networks with a Formal Approach based on Answer Set Programming". LNCS 7605, Computational Methods in Systems Biology, pp 342-361, 2012.

2010 M Gebser, **C Guziolowski**, M Ivanchev, T Schaub, A Siegel, S Thiele, P Veber. "Repair and Prediction (under Inconsistency) in Large Biological Networks with Answer Set Programming". Proceeding of the Twelfth International Conference on Principles of Knowledge Representation and Reasoning (KR'10), Toronto, Canada, pp. 497-507, AIII Press, 2010.

2009 S Blachon, G Stoll, **C Guziolowski**, A Zinovyev, E Barillot, A Siegel, O Radulescu. "Method for relating inter-patient gene copy numbers variations with gene expression via gene influence networks", *BMIINT'09*: Biomedical Informatics and Intelligent Methods in the support of Genomic Medicine, Thessaloniki, Grece, AIAI, pp 72-87 2009.

2008 **C Guziolowski**, J Gruel, O Radulescu, A Siegel. "Curating a large-scale regulatory network by evaluating its consistency with expression datasets", *CIBB'08*: 5th International Conference on Bioinformatics and Biostatistics, Salerno, Italy 2008, Lecture Notes in Computer Science, Springer vol. 5488, pp. 144-155 2009.

2007 A Siegel, M Le Borgne, O Radulescu, **C Guziolowski**, P Veber. "Qualitative response of interaction networks: application to the validation of biological models", Contribution to minisymposium New Research in Bioinformatics. ICIAM'07: 6th International Congress on Industrial and Applied Mathematics, PAMM, vol. 7, no. 1, pp. 1121803-1121804, Zurich 2007.

## Publications (continued)

### Book Chapters

2019 Razzaq M., Chebouba L., Le Jeune P., Mhamdi H., **Guziolowski C.**, Bourdon J. "Logic and Linear Programs to Understand Cancer Response". In: Liò P., Zuliani P. (eds) Automated Reasoning for Systems Biology and Medicine. Computational Biology, vol 30. Springer, Cham

## Scientific supervision

### Ph.D. candidates supervision

2021-2024 M. Bolteau (50%), funding: ANR AIBY4, ANR BOOSTIVF. Director: Jérémie Bourdon. Title: "Logic programs to infer computational models of human embryonic development".

2019-2022 S. Le Bars (50%), funding: Bourse MENRT, Ministère de la Recherche. Director: Jérémie Bourdon. Title: "Hybrid, logical and linear modelling to predict in silico the effect of disturbances on metabolism".

2015-2018 M. Razzaq (50%), funding: chaire CNRS - Centrale Nantes. Thesis Director: Jérémie Bourdon. Thesis title: "Integrating Phosphoproteomic Time Series Data into Prior Knowledge Networks". Current position: Chargée de Recherche at INRAE Center, Val De Loire, France.

2014-2017 B. Miannay (40%), funding: Projet Régional GRIOTE. Thesis Director: Olivier Roux. Thesis title: "Regulatory networks analysis with graph coloring approaches applied to multiple myeloma". Current position: Data Engineer at Resilience, digital oncology, Paris, France.

2013-2016 L. Fitime (50%), funding: CNRS-Région Pays de la Loire. Thesis Director: Olivier Roux. Thesis title: "Hybride modelling, analysis and quantitative verification of large biological regulatory networks". Current position: Teacher-researcher at National Advanced School of Engineering of University of Yaoundé I, Cameroun.

### Other

2017-2018 M. Folschette, funding: UBL, *Université Bretagne Loire*. Postdoc.

2016-2017 A. Poret, funding: chaire CNRS, *Centrale Nantes*, *Cancéropole Grand Ouest*. Postdoc.

M. Lefebvre, funding: *Projet Régional* SYMETRIC. Engineer.

L. Chebouba, funding: Profas B+, *Programme Algérie - France*. Doctoral internship.

2015 I. Konokotina, funding: *Master Automatique Robotique et Systèmes de Production*. Master student.

## Participation to research projects

- ANR project JCJC "BOOSTIFV", coordinated by Thomas Fréour (2020-2024).
- ANR project "AiBy4", coordinated by Harold Mouchère and Diana Mateus (2020-2025).
- Regional project "GRIOTE", coordinated by Jérémie Bourdon, Richard Redon, Dominique Tessier (2013-2017).

## Participation to research projects (continued)

- Regional project “SyMeTRIC”, coordinated by Jérémie Bourdon, Richard Redon (2014-2018).

## Scientific responsabilities

### Ph.D. theses jury

- 09/2022 S. Chevalier, *Université Paris-Saclay*, Gif-sur-Yvette, France, *examinatrice*.
- 09/2021 F. Gouveia, *INESC-ID / Instituto Superieur Técnico, Universidad de Lisboa*, Lisbon, Portugal, *rapporteure*.
- 06/2019 N. Sella, *Sorbonne Université*, Paris, France *examinatrice*.
- 09/2018 S. Neaves, *King's College London*, England, *rapporteure*.
- 12/2017 J. Coquet, *Université de Rennes*, France, *examinatrice*.
- 02/2017 R. Rozanski, University of Manchester, England, *rapporteure*.

### Invitations

- 2017 University of Manchester, 11-12 july, Professor Ross King.

### Animation

- 2023 Organisation and program committe of *Journée "Filles, maths et informatique : une équation lumineuse"* at Nantes, 7 February.
- 2019 Organisation of the working group *GT BIOSS (Biologie Systémique Symbolique)* on the topic *Médecine Personalisée* at Nantes.
- Organisation and program committee of the JOBIM *Journées Ouvertes en Biologie, Informatique et Mathématiques* at Nantes.
- 2017 Program commitee of the JOBIM conference.
- 2016 Participation at the *maître de Conférences* recruitment committee at LIF, Marseille.
- 2015 Organisation committee of CMSB conference “Computational Methods in Systems Biology”, Nantes, France, 16-18 September.
- 2010 Organisation committee of "2nd European Workshop on Tissue Imaging and Analysis", Heidelberg, Germany, 25-26 June.

### Reviewer

- Journals Bioinformatics, ISMB ECCB, Biosystems, BMC Systems Biology, BMC Bioinformatics, Molecular Genetics and Genomics, IET Control, Applications, ICECCS, CIBB.
- Research Projects FST (Luxembourg), Fondecyt (Chile), BSF (United States - Israel).

### Other responsabilities

- since 03/2022 Equality referent at the LS2N.
- since 2022 Co-responsible of the “Equality and Diversity Mission” at the LS2N.
- 2019 Member of the “Equality Mission” at the LS2N.

## Academic Activities

### Teaching

since 2012 — At *Ecole Centrale de Nantes*, France: "Algorithmics and Programming Languages", (60 hours[3]), 1st year students; "Equality and violence mechanisms in humans" (8 hours), 1st year students; option "Soft skills"; "Logic Programming" (40 hours), 2nd-3rd year students, option "Artificial Intelligence"; Supervision of students projects (13h), option "Artificial Intelligence". "Software engineering" and "Computer systems development techniques" (31 hours), 2nd-3rd year students, option "Computer Science".

2010-2011 — At Heidelberg University, Germany "Theoretical Basics in Bioinformatics", Master of Medical Informatics, (6 hours); "Introduction to Bioinformatics", Bachelor of Medical Informatics, (3 hours).

2007-2009 — At *Ecole Nationale de la Statistique et de l'Analyse de l'Information*, Bruz, France. Exercise and practical sessions on "UML and Java programming"(90 hours).

2002 — At *Universidad de Chile*, Santiago, Chile. Exercise and practical sessions on "Introduction to Algorithmic and Programming Languages".

### Committees participation

since 2021 — Member of the Disciplinary Commission of Users at *Ecole Centrale de Nantes*.

Member of the "Equality and Diversity" commission at *Ecole Centrale de Nantes*.

## Career development and conditions of exercise

since 09/2019 — Working at partial time (80%).

2017 — Research team reconfiguration within the LS2N: from Meforbio team to Combi.

2013 — Maternity leave (3 months)

[3] hours during an academic year

Appendix A

# Appendix

## A.1 Sign consistency

Thiele *et al. BMC Bioinformatics* (2015) 16:345
DOI 10.1186/s12859-015-0733-7

BMC Bioinformatics

METHODOLOGY ARTICLE Open Access

# Extended notions of sign consistency to relate experimental data to signaling and regulatory network topologies

Sven Thiele[1], Luca Cerone[2], Julio Saez-Rodriguez[2], Anne Siegel[3,4], Carito Guziołowski[5*] and Steffen Klamt[1*]

## Abstract

**Background:** A rapidly growing amount of knowledge about signaling and gene regulatory networks is available in databases such as KEGG, Reactome, or RegulonDB. There is an increasing need to relate this knowledge to high-throughput data in order to (in)validate network topologies or to decide which interactions are present or inactive in a given cell type under a particular environmental condition. Interaction graphs provide a suitable representation of cellular networks with information flows and methods based on sign consistency approaches have been shown to be valuable tools to (i) predict qualitative responses, (ii) to test the consistency of network topologies and experimental data, and (iii) to apply repair operations to the network model suggesting missing or wrong interactions.

**Results:** We present a framework to unify different notions of sign consistency and propose a refined method for data discretization that considers uncertainties in experimental profiles. We furthermore introduce a new constraint to filter undesired model behaviors induced by positive feedback loops. Finally, we generalize the way predictions can be made by the sign consistency approach. In particular, we distinguish strong predictions (e.g. increase of a node level) and weak predictions (e.g., node level increases or remains unchanged) enlarging the overall predictive power of the approach. We then demonstrate the applicability of our framework by confronting a large-scale gene regulatory network model of *Escherichia coli* with high-throughput transcriptomic measurements.

**Conclusion:** Overall, our work enhances the flexibility and power of the sign consistency approach for the prediction of the behavior of signaling and gene regulatory networks and, more generally, for the validation and inference of these networks

**Keywords:** E. coli, Gene regulation, Interaction graphs, Sign consistency, Uncertainty, Logic modeling, Answer Set Programming (ASP)

## Background

The advancements of measurement technologies and high-throughput methods in molecular biology have led to a tremendous increase in the availability of factual biological knowledge as well as of data capturing the response of biological systems to experimental conditions. Knowledge about metabolic, signaling, and gene regulatory interactions and networks is available in databases such as KEGG, Regulon DB, PID, or Reactome which can be used as a starting point to build causal models of biomolecular networks [1]. Specifically, signaling and gene regulatory networks carrying signal and information flows can be represented as interaction (or influence) graphs [2–6], Bayesian networks [7], some form of logic (including Boolean or constrained fuzzy logic) modeling [4, 8, 9], or ordinary differential equations [10–12]. However, there is an increasing need to relate large-scale network models to high-throughput data in order to (in)validate network topologies or to decide which regulatory or signaling interactions are present in a particular biological system, cell type, environmental condition etc.

*Correspondence: carito.guziolowski@irccyn.ec-nantes.fr; klamt@mpi-magdeburg.mpg.de
[5] École Centrale de Nantes, IRCCyN UMR 6597, 1 rue de la Noë, 44321 Nantes, France
[1] Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstraße 1, 39106 Magdeburg, Germany
Full list of author information is available at the end of the article

Significant work has been published on this subject, attempting to detect inconsistencies among measured high-throughput data and signaling and regulatory networks and to subsequently identify missing or inactive interactions such that the optimized network structure maximizes consistency with experimental data [2, 4, 13–18]. Some of these approaches use signed directed graphs, also called interaction or influence graphs (IG), as underlying model where edges indicate either positive or negative effect of one node upon another. Although these models are qualitative and simple, they have frequently been used to study signal flows in a wide range of biological systems. Moreover, the fact that every Boolean and every ODE model has an underlying interaction graph renders their analysis directly relevant for other modeling formalisms and it has been shown that some important global properties of Boolean or ODE models are determined by the structure of their associated IG [6, 19, 20]. IG have also been used for qualitative reasoning, to describe physical systems where a detailed quantitative description is unavailable [21]. In fact, this has been one motivation for using IG in the context of biological systems [20] where knowledge and data are usually uncertain.

One important class of methods relating IG with experimental data is based on the notion of *sign consistency*. The key idea here is to represent the potential network behaviors resulting from steady-state shift experiments (such as upregulation or downregulation of node activation levels after network perturbations) by certain kinds of discrete constraints. A first approach based on sign consistency was introduced in [2]. There, experimentally measured changes in node activities were represented by two labels (increase, decrease) on the IG nodes. Constraints relating nodes labels and IG are introduced to model the propagation of regulatory effects. Later, in [3, 22], Answer Set Programming (ASP) [23] was used to find admissible node labelings adhering to the posed constraints, and optimal repairs to restore sign-consistency were proposed. A related formalism was presented in [17]. Major differences to previous studies were (i) consideration of three node labels (increase, decrease, 0-change), (ii) the representation of the constraints as an integer linear programming (ILP) problem, and (iii) the introduction of new repair operations minimizing inconsistencies between the IG structure and the experiments.

The goal of this study is fourfold. First, we aim at unifying existing approaches into a general framework. We show that different notions of sign consistency mainly differ in the way zero changes are modeled. Then, we propose a refined method for data discretization allowing one to express uncertainties during the discretization step. In addition, we introduce a new constraint to filter undesired self-fulfilled explanations which result from positive feedback loops. Finally, we introduce an extended prediction method that allows not only strong (e.g., "increase") but also weak predictions (e.g., "increase or 0-change"), enlarging the predictive power of the approach. We applied the extended framework to a realistic case study where we analyze high-throughput transcriptomic measurements of *Escherichia coli* in the context of a large-scale gene regulatory network model obtained from RegulonDB. Taken together, we demonstrate that these extensions increase the applicability and flexibility of the approach significantly.

## Methods

### Definitions

An *influence or interaction graph* (IG) is a signed directed graph $(V, E, \sigma)$, where $V$ is a set of nodes, $E$ a set of edges, and $\sigma : E \to \{+, -\}$ a labeling of the edges. Every node in $V$ represents a species in the modeled system and an edge $j \to i$ means that the change of $j$ in time influences the level of $i$. Every edge $j \to i$ of an IG can be labeled with a sign, either $+$ or $-$, denoted by $\sigma(j, i)$, where $+$ $(-)$ indicates that $j$ tends to increase (decrease) $i$. An example IG is given in Fig. 1.

In this framework, we confront the IG with *experimental profiles*. In our approach, the experimental profiles are supposed to come from steady-state shift experiments where, initially, the system is at steady-state, then externally perturbed in certain nodes, and settles eventually into another steady-state. For some species $S \subseteq V$ (genes, proteins, or metabolites) concentrations are measured in the initial and final state. The raw data is given by a real value $obs(s)$ for every measured species $s \in S$ specifying

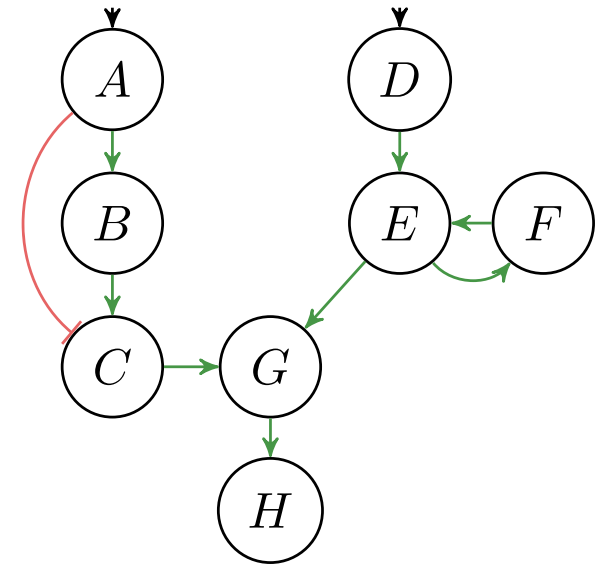



**Fig. 1** Interaction graph with a positive feedback loop between *E* and *F*

the difference of the node states at the beginning and in the new steady state. As defined below, we determine for these nodes whether the concentration has increased, decreased or not significantly changed.

### Data discretization

We propose a refined method to discretize the measurements using four (condition-dependent) thresholds $t_1 \leq t_2 < 0 < t_3 \leq t_4$, allowing one to consider uncertainties in the discretization process. As illustrated in Fig. 2, these thresholds define a mapping $\mu : S \to \{-, \triangledown, 0, \triangle, +\}$ as follows:

$$\mu(s) = \begin{cases} - & | \quad obs(s) \leq t_1, \\ \triangledown & | \quad t_1 < obs(s) \leq t_2, \\ 0 & | \quad t_2 < obs(s) < t_3, \\ \triangle & | \quad t_3 \leq obs(s) < t_4, \\ + & | \quad t_4 \leq obs(s). \end{cases}$$

We consider measurements which are smaller than $t_1$, bigger than $t_4$, and between $t_2$ and $t_3$ as certain (decrease -, increase +, no-change 0) while measurements that are between $t_1$ and $t_2$ (resp. $t_3$ and $t_4$) are uncertain (uncertain-decrease $\triangledown$, uncertain-increase $\triangle$) and not exactly classifiable. With that, an experimental profile $(S, I, \mu)$ is defined by the set of measured species $S$, the set of *input nodes* $I \subseteq S$ (the experimentally perturbed species) whose changes are trivially explained, and the mapping $\mu$ as defined above.

### Local consistency rules

Given an IG $(V, E, \sigma)$ and an experimental profile $(S, I, \mu)$ one can describe the rules that relate both. For this purpose we look for total labelings $\mu^t : V \to \{-, 0, +\}$ that satisfy the local constraints defined below. It is important to notice that $\mu^t$ will define a *total* labeling using the *three* labels $\{-, 0, +\}$ whereas $\mu$ defines a *partial* labeling (only measured nodes are labeled) based on the *five* labels $\{-, \triangledown, 0, \triangle, +\}$ representing the discretized measurements.

With the first constraint, we look for total labelings $\mu^t$ that satisfy the observed measurements captured in the partial node labeling given by $\mu$:

**Constraint 1** (satisfy observations)**.** *Let $(V, E, \sigma)$ be an IG, $(S, I, \mu)$ an experimental profile, $\mu^t : V \to \{+, -, 0\}$ be a total labeling, and let $i \in V$ be a node with $\mu^t(i) \in \{+, 0, -\}$.*

*Then $\mu^t$ satisfies Constraint 1 for node $i$ iff $i \notin S$, or $\mu^t(i) = +$ and $\mu(i) \in \{+, \triangle\}$, or $\mu^t(i) = 0$ and $\mu(i) \in \{\triangle, 0, \triangledown\}$, or $\mu^t(i) = -$ and $\mu(i) \in \{\triangledown, -\}$.*

Note, uncertain measurements restrict the labeling of a node to two out of the three values $\{+, -, 0\}$, while measurements with high certainty fix a node's label to exactly one value.

Next we demand for every non-input node $i$, that its change $\mu^t(i)$ ought to be explained by the total influence of its predecessors in the IG. The *influence* of $j$ on $i$ is given by the product $\mu^t(j)\sigma(j, i) \in \{+, -, 0\}$.

**Constraint 2** (change must be justified by a change in a predecessor)**.** *Let $(V, E, \sigma)$ be an IG, $(S, I, \mu)$ an experimental profile, $\mu^t : V \to \{+, -, 0\}$ be a total labeling, and let $i \in V \setminus I$ be a non-input node with $\mu^t(i) \in \{+, -\}$.*

*Then $\mu^t$ satisfies Constraint 2 for node $i$ if there is some edge $j \to i$ in $E$ such that $\mu^t(i) = \mu^t(j)\sigma(j, i)$.*

Constraint 2 is consistent with the propagation rule used in [2, 3] which demands that increases and decreases must be explained by predecessor nodes while 0-changes are unconstrained, that is 0-changes can always occur irrespective of the state of the predecessor nodes (note that 0-changes were not considered in [2, 3]). One argument for this reasoning is that it is often impossible to estimate the strength of the influences and the thresholds at which a downstream effect occurs are unknown. Hence, we cannot guarantee that an influence really has an effect and therefore allow 0-change. On the other hand, the constraint still enforces explanations for observed changes in node activation levels; each change must be explainable by an influence (with proper sign) of at least one predecessor.

Melas et al. [17] suggested also to demand proper explanations for 0-changes using the following constraint:

**Constraint 3** (0-change must be justified)**.** *Let $(V, E, \sigma)$ be an IG, $(S, I, \mu)$ an experimental profile, $\mu^t : V \to \{+, -, 0\}$ be a total labeling, and let $i \in V \setminus I$ be a non-input node with $\mu^t(i) = 0$. Then $\mu^t$ satisfies Constraint 3*

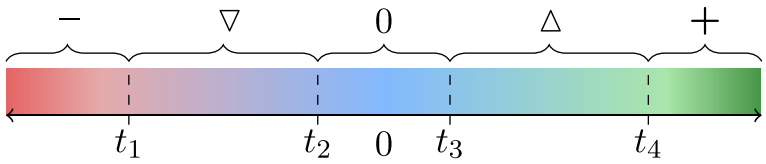



**Fig. 2** Discretization of observed changes into sign constraints

*for node $i$ if there is either no edge $j \to i$ in $E$ such that $\mu^t(j)\sigma(j,i) \in \{+,-\}$ or there exist at least two edges $j_1 \to i$ and $j_2 \to i$ in $E$ such that $\mu^t(j_1)\sigma(j_1,i) + \mu^t(j_2)\sigma(j_2,i) = 0$*

Constraint 3 restricts the occurrence of 0-changes. A node is only allowed to show 0-change if it receives either no influence or contradictory influences. This constraint thus assumes that each influence has indeed an effect and only contradictory influences can cancel each other out.

In Fig. 3, we illustrate IGs with different labelings where green stands for increase, red for decrease and blue for 0-change. Notice, that Constraint 2 intentionally allows situations like in labeling $g$ and $h$, where $D$ is labeled as 0-change even if the predecessor $B$ is showing an increase resp. decrease. On the other hand, Constraint 2 forbids $D$ to increase or decrease, if all predecessors are labeled as 0-change.

### From local to global reasoning

While there might exist several total labelings that satisfy the local constraints for *some* nodes we are interested in checking global consistency, where a total labeling exists such that the local constraints are satisfied for *all* nodes. In Fig. 4, we illustrate an IG together with a partial labeling which is locally consistent but globally inconsistent. In other words, there exist two total labelings such that the *local consistency rules* (Constraints 1, 2 and 3) are satisfied, for either $A$ or $B$, but there exists no single total labeling that satisfies these constraints for all nodes.

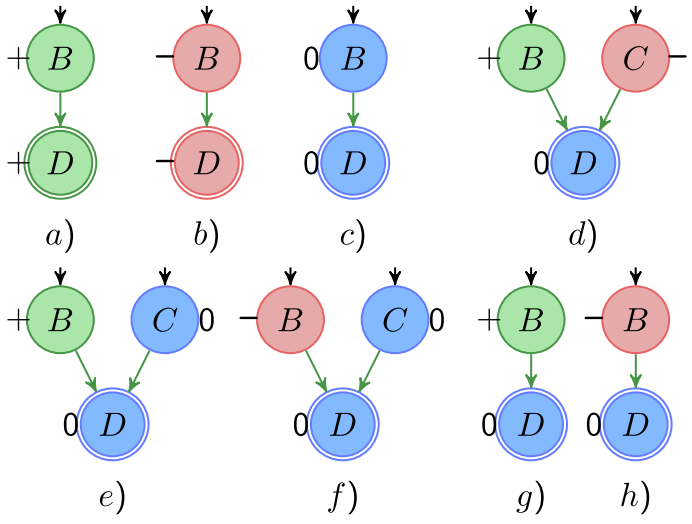
We use the previously defined constraints to define the following global consistency notions.

**Consistency Notion 1** (weak propagation, WP)**.** *We call an IG and an experimental profile $(S, I, \mu)$ consistent under weak propagation (WP), iff there exists a total labeling $\mu^t$ such that Constraint 1 and 2 are satisfied for all nodes.*

**Consistency Notion 2** (strong propagation, SP)**.** *We call an IG and an experimental profile $(S, I, \mu)$ consistent under strong propagation (SP), iff there exists a total labeling $\mu^t$ such that Constraints 1, 2 and 3 are satisfied for all nodes.*

Further, we introduce here a new *global constraint* to ensure that every node change is justified by a chain of influences that can be traced back to an (perturbed) input node. This natural constraint is especially useful to forbid self-justification of changes via positive feedback loops (see Fig. 5).

**Constraint 4** (a change must be founded in an input)**.** *Let $(V, E, \sigma)$ be an IG, $(S, I, \mu)$ an experimental profile, $\mu^t$ :*




| | $a$ | $b$ | $c$ | $d$ | $e$ | $f$ | $g$ | $h$ |
|---|---|---|---|---|---|---|---|---|
| Constraint 2 | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ |
| Constraint 3 | ✓ | ✓ | ✓ | ✓ | × | × | × | × |

**Fig. 3** IGs with different labelings where green stands for increase, red for decrease, and blue for 0-change. All labelings satisfy the basis Constraint 2 for node $D$, but only the labelings *a-d* satisfy also Constraint 3. Examples with uncertain measurements are shown in the Additional file 1

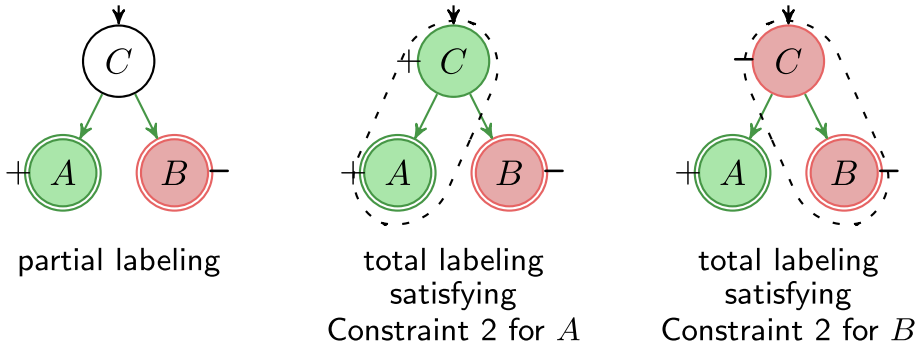



**Fig. 4** Example for an influence graph with partial labeling, which is locally consistent for *A* and *B*, but globally inconsistent because there exist no single total labeling satisfying Constraint 2 for *A* and *B*

$V \to \{+,-,0\}$ *be a total labeling, and* $i \in V$ *a node with* $\mu^t(i) \in \{+,-\}$.

*Then* $\mu^t$ *satisfies Constraint 4 for node i if either i is an input node* $i \in I$*, or there exist a path* $(v_0,\ldots,v_k)$ *in E with* $v_0 \in I$, $v_k = i$ *and* $\mu^t(v_{n-1})\sigma(v_{n-1},v_n) = \mu^t(v_n)$ *for all* $n = 1\ldots k$.

In Fig. 5, we illustrate an IG with a partial labeling (left) and two total labelings (middle and right) derived from the partial one. Both total labelings satisfy the local propagation rules (Constraints 2, 3), but only the second total labeling satisfies the global propagation rule (Constraint 4). While the first labeling suggests a self-sustained increase in *B* and *C* as explanation for the increase in *D*, the second labeling hints to an increase in the input node *A*. Using Constraint 4 we can avoid manual removal of positive feedback loops as done in previous studies [17].

We combine the new constraint with previously defined constraints into the following consistency notions.

**Consistency Notion 3** (founded weak propagation, FWP)**.** *We call an IG and an experimental profile* $(S,I,\mu)$ *consistent under founded weak propagation (FWP), iff there exists a total labeling* $\mu^t$ *such that Constraints 1, 2 and 4 are satisfied for all nodes.*

**Consistency Notion 4** (founded strong propagation, FSP)**.** *We call an IG and an experimental profile* $(S,I,\mu)$ *consistent under founded strong propagation (FSP), iff there exists a total labeling* $\mu^t$ *such that Constraints 1, 2, 3, and 4 are satisfied for all nodes.*

### Consistency checking

We can now apply the previously defined consistency notions to enumerate consistent total labelings and to verify the consistency of network and observation data for a given experimental profile. We consider an IG consistent with an experimental profile $(S,I,C)$ if there exists at least one consistent total labeling (consistent with respect to the chosen Notion WP, SP, FWP or FSP). Consider Fig. 6 which shows the total labelings of the IG in Fig. 1 consistent with an example experimental profile (*A* and *D* were increased resulting in a measured 0-change in *H*) under the different consistency notions. Note that the notions become more strict, accepting less labelings as consistent and therefore excluding certain system behaviors. The set of admissible labeling under SP is a subset of the

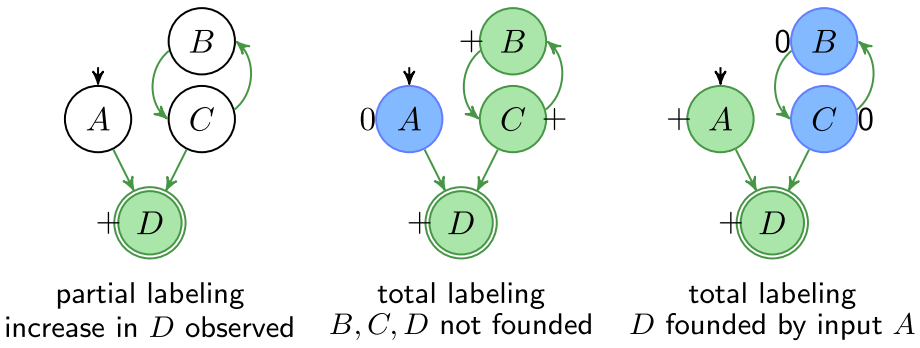



**Fig. 5** Example for an influence graph with partial labeling, which can be explained either by self activation in *B* and *C* or by the input node *A*

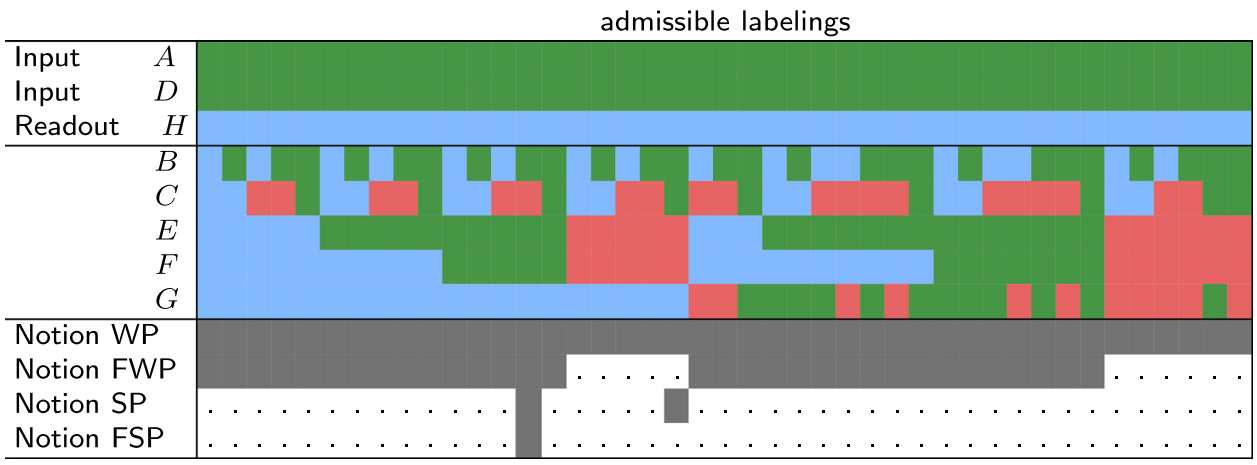



**Fig. 6** Consistent total labelings of the example in Fig. 1 under different consistency notions. The consistent labelings under all four consistency notions are fully displayed, a grey cell indicates that the labeling above is consistent and a white cell with "." means that it is not a consistent labeling

admissible labelings under WP and the set of admissible labeling under FSP is a subset of the admissible labelings under SP. Further, one can see that Constraint 4 excludes all labelings where $E$ and $F$ decrease. This behavior does not satisfy Constraint 4, as it is only possible by mutual inhibition using the positive loop between $E$ and $F$, which is not founded in an input.

#### *Predictions under consistency*

The consistency check of network and experimental data is the first analysis that is performed with the sign consistency approach. If network and data are consistent the sign consistency approach can be used to predict the behavior of unmeasured entities in the network. This can also be used to predict the outcome of a planed experiment and reversely to plan an experiment that should result in a specific desired behavior. In the sign consistency approach, each consistent labeling represents an admissible behavior of the system. We call a statement that holds in all admissible behaviors under the given consistency notion a *prediction*. If parts of the system act the same in all admissible behaviors this can be predicted. We can predict the following types of behaviors in our systems. We predict that a species *increases* $+$ (resp. *decreases* $-$, does *not change* 0) if it increases (resp. decreases, does not change) in all admissible labelings. We call these strong predictions, because the possible behaviors of a species are reduced to exactly one. Further, we can predict that a species does not increase (resp. does not decrease, does change) if it does not increase (resp. not decrease, does change) in all admissible labelings. Therefore, we can also predict *weak increase* $\oplus$, when a species does not decrease, but increases in at least one admissible behavior, and does not change in another admissible behavior. Likewise, we predict *weak decrease* $\ominus$ when a species does not increase, but decreases in at least one admissible behavior, and does not change in another. Finally, we predict *change* $\pm$ when a species does always change, it increases in at least one admissible behavior and decreases in another. We call $\oplus$, $\ominus$, and $\pm$ weak predictions because one possible behavior is excluded while one degree of freedom is left.

Formally, for a set $V$ of nodes in our network and the set $M$ of labelings consistent with our experimental profile, we define the prediction function $pred : V \to \{+, -, 0, \oplus, \ominus, \pm, \text{no}\}$ as follows:

$$pred(x) = \begin{cases} + \mid \forall \mu \in M : \mu(x) = +, \\ - \mid \forall \mu \in M : \mu(x) = -, \\ 0 \mid \forall \mu \in M : \mu(x) = 0, \\ \\ \oplus \mid \forall \mu \in M : \mu(x) \neq -, \ \exists \mu \in M : \mu(x) = +, \\ \qquad\qquad\qquad\qquad\quad \exists \mu \in M : \mu(x) = 0, \\ \ominus \mid \forall \mu \in M : \mu(x) \neq +, \ \exists \mu \in M : \mu(x) = -, \\ \qquad\qquad\qquad\qquad\quad \exists \mu \in M : \mu(x) = 0, \\ \pm \mid \forall \mu \in M : \mu(x) \neq 0, \ \exists \mu \in M : \mu(x) = +, \\ \qquad\qquad\qquad\qquad\quad \exists \mu \in M : \mu(x) = -, \\ \text{no} \mid \textit{else}. \end{cases}$$

#### *Recovery rate and precision*

In Table 1, we show the predictions for the example given in Fig. 1. One can see that the more constrained consistency notions yield smaller sets of admissible labelings and a higher *recovery rate* (for how many unmeasured species can predictions be obtained). In the systematic comparison of the consistency notions based on real experimental data we not only consider recovery rate but also prediction *precision* (true positives/(true positives + false positives)). A strong prediction ($+/-/0$) will be a true positive if it has

**Table 1** Predictions for the example in Fig. 1 derived from the admissible behaviors in Fig. 6

| | *B* | *C* | *E* | *F* | *G* |
|---|---|---|---|---|---|
| Notion WP | ⊕ | no | no | no | no |
| Notion FWP | ⊕ | no | ⊕ | ⊕ | no |
| Notion SP | + | ± | ± | ± | 0 |
| Notion FSP | + | + | + | + | 0 |

The **no** means that the node can have any value of {+, 0, −} which means that pratically no prediction is possible

a certain measurement with the same value (+/−/0). A weak prediction ⊕ (resp. ⊖ and ±) will be a true positive if it has a certain measurement + or 0 (resp. − or 0 and + or −). Reversely, a prediction will be a false positive if has a certain measurement value with a contradictory value +.

### Repairing inconsistent networks and data

If network and data are inconsistent the natural question arising is how to repair networks and/or data, that is, how to modify network and/or data in order to re-establish their mutual consistency. A major challenge lies in the range of possible repair operations, since an inconsistency can be explained by missing interactions or inaccurate information in a network as well as by measurement errors. The sign consistency approach can be used to determine a set of repair operations that are suitable to restore consistency. Typically, plenty of suitable repair operations are possible, in particular, if multiple repair operations are admitted. However, one usually is only interested in repairs that make few changes on the model and/or data. These minimal repair sets cannot only be used for hypotheses generation (e.g., which data might be questionable or which edges might be missing or inactive) but as a quantitative measure for the fitness of model and data. Also note that once consistency is re-established, network and data can again be used for predicting behaviors of unmeasured entities.

In [17], four repair operations were introduced; two of them for single experiments (SCEN-FIT, Minimal Correction Sets (MCoS)) and two for multiple experiments (OPT-SUBGRAPH, OPT-GRAPH). The latter two are computationally more demanding as they seek to optimize the whole network structure based on many perturbation experiments. SCEN-FIT, as explained in detail in the Additional file 1, seeks to find a consistent node labeling that is closest to the given measurements and can thus help to identify inconsistencies between network and dataset. Herein we will focus on MCoS and thus deal with analysis of single experiments. This is motivated by our application example where we indeed have multiple experiments (105) but where the number of experiments is low compared to the number of edges and nodes in the network (1646) disabling a meaningful network structure optimization. However, we note here that our extended notion FSP can be straighforwardly applied to these repair operations as well.

#### *Minimal Correction Sets (MCoS)*

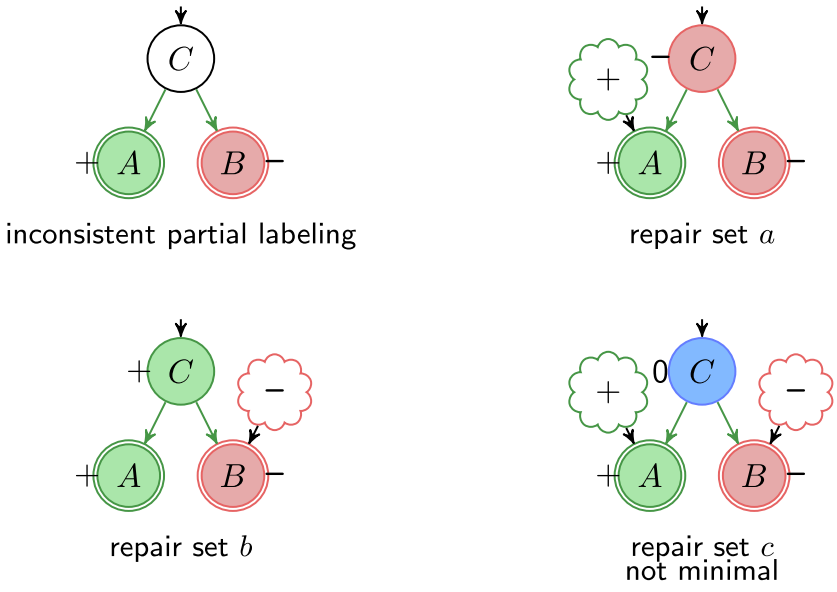
To resolve inconsistencies one may add new influences to the model if the later is considered to be potentially incomplete (which is often the case in practice). Adding an influence can be used to indicate missing (unknown) regulations or oscillations of regulators that would explain the (topology-inconsistent) measurements. We use minimal correction sets (MCoS) as defined in [17] as minimal sets of new signed (positive or negative) input influences that restore consistency of model and data. MCoS are defined as signed influences and are specific for a single experiment; they might be incompatible with other experiments. Note that every inconsistency can be repaired by adding a new influence. Therefore, adding influences is always suited to restore consistency. Also the MCoS can be interpreted as a measure of consistency of model and data. Compared to SCEN-FIT, MCoS yields always a smaller or equal number of repairs. Therefore we define the inconsistency-index of a network with respect to data as (MCoS/number of observations in the experiment). Figure 7 illustrates how repair through addition of influences works.

#### *Prediction under minimal repair*

Due to the capability of repairing, the sign consistency approach enables prediction even if model and data are mutually inconsistent. Predictions under minimal repair are obtained from the identification of consequences shared by all consistent labelings under all possible minimal repairs. Note that this approach although it confines to minimal repairs following the law of parsimony, does not favor any of the possible minimal repairs but only considers a statement a prediction if it holds under every minimal repair.

### Software

The different consistency notions as well as the methods for consistency checking and quantification, prediction, and all data and network repair operations were implemented in an open source application `iggy` [24]. `iggy` uses ASP [23] as logical modeling and constraint solving paradigm, it is part of the BioASP software collection and can easily be installed via the python package index (PyPI). ASP is used to model problems from NP and provides state-of-the-art solvers. In particular, we use the solver `clasp` [25] via the `pyasp` [26] package. On an AMD Opteron 6168 1.9 GHz with 96 GB RAM, given a network with 1646 nodes and 4277 edges our software needs $\approx$ 20




| Repair set | $a$ | $b$ | $c$ |
|---|---|---|---|
| Number of repairs | 1 | 1 | 2 |
| WP | ✓ | ✓ | ✓ |
| FWP | ✓ | ✓ | ✓ |
| SP | ✓ | ✓ | ✓ |
| FSP | ✓ | ✓ | ✓ |
| Minimality | ✓ | ✓ | × |

| Node | $A$ | $B$ | $C$ |
|---|---|---|---|
| repair set $a$ | $+$ | $-$ | $-$ |
| repair set $b$ | $+$ | $-$ | $+$ |
| prediction under minimal repair | $+$ | $-$ | $\pm$ |

**Fig. 7** Repair by adding signed influences example (Minimal Correction Sets - MCoS). There exist three alternative repair sets: repair set *a* adds a positive influence to *A* and repair set *b* includes a negative influence on *B*, repair set *c* includes a positive influence on *A* and a negative influence on *B*. Repair sets *a* and *b* are minimal containing only one repair, repair set *c* is not minimal having two repairs. Looking at the intersection of the labelings under minimal repairs, we can conclude that *C* is either responsible for an increase in *A* or a decrease in *B*. We can therefore exclude a labeling of *C* with 0, we can predict: $pred(C) = \pm$

min to compute the predictions under minimal repair (MCos) for the unmeasured species of 105 experiment data sets each containing 1392 measurements. For further information visit http://bioasp.github.io/iggy.

## Results and discussion

To investigate the suitability of the different consistency notions, we used the gene regulatory network of *Escherichia coli* and confronted it with Microarray data. The network was obtained from RegulonDB [27], version 8.3 in october 2013, and we focused on its biggest weakly connected component which is composed of 1646 nodes and 4277 edges and covers 94 % of the nodes of the full RegulonDB network. Unsigned edges are treated as two parallel edges with opposite signs. The data refers to the microarray log ratio expression of 3607 genes measured under 240 different stress conditions in *E. coli* published in [28]. We chose 105 of 240 experiments which can be interpreted as steady state shift experiments and 1392 of the 3607 genes which occur in the RegulonDB network. Since the input nodes for the stress condition experiments are unknown, we simply defined all nodes without predecessors as inputs.

The GEO/GSE codes for the used experiments are listed in the Additional file 1. The microarray data was discretized as described in the Methods Section using the typical thresholds: $t_1 = -2$, $t_2 = -0.01$, $t_3 = 0.01$, $t_4 = 2$, to generate the constraints that restrict the labeling $\mu$ for the nodes measured in the experimental profile.

To evaluate the influence of the minimal correction sets (MCoS) and to investigate the suitability of the different consistency notions to predict the behavior of unobserved entities in a regulatory network, we performed a cross-validation using the *E. coli* data.

### Quality of regulatory network when confronted to the expression profiles

As a first step, we assess the quality of network and data by comparing it to randomized data. We generated 100 randomized datasets for each real experiment by

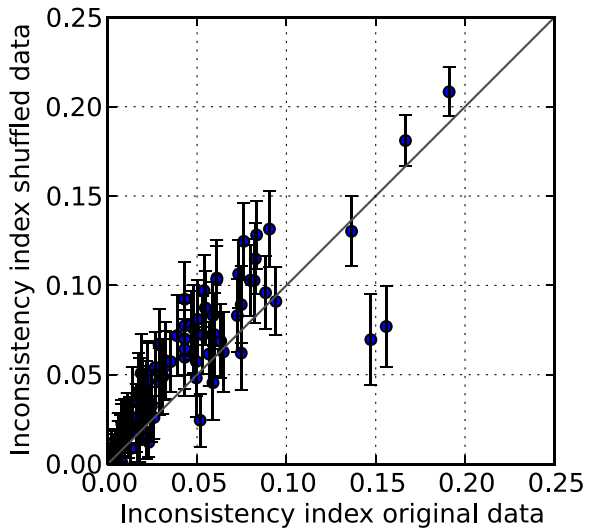



**Fig. 8** Inconsistency index of the 105 experiments. The x-axis shows the inconsistency index of the original experiments and the y-axis the average inconsistency of the shuffled experiments. The error bars indicate the standard deviation of the inconsistency index among 100 shuffled samples

shuffling the observed signs among the observed nodes; but preserving the sign distribution for each dataset. We then computed for real and randomized data the inconsistency index which is defined as the quotient of the number of minimal corrections (MCoS) to restore consistency (under notion FSP) divided by the number of observations in the experiment. Then we computed the Wilcoxon signed-rank test to assess whether the population means of the two samples differ. The obtained p-value of $2.0497e\text{-}11$ indicates a highly significant difference of real and randomized data, suggesting that the real data are more (sign-) consistent with the network topology than random data. Figure 8 shows the *inconsistency index* for real and randomized data for each experiment. We can see that the real *E. coli* dataset exhibits a significantly lower inconsistency index than the randomized data.

Figure 9 shows the distribution of the measured signs in the experimental data revealing that the data tends to be less consistent if more +/- are contained.

### Predictions under the different consistency notions

To investigate the suitability of the different consistency notions to predict the behavior of unobserved entities in a regulatory network, we performed a cross-validation using the *E. coli* data. While other validation methods exist, we decided to use cross-validation as a model validation technique because it allows us to assess how the results of the approach will generalize to independent datasets. To set up cross-validation, we created for each experiment 100 samples each containing a random 10 % share of the measurements. We then confronted the *E. coli* network with these samples, determined the minimal corrections necessary to restore consistency, and computed the predictions that hold under all minimal correction sets.

In Table 2 one can see the distribution of the $+$, $-$, 0 and weak predictions as well as how the precision of the different types of predictions varies among the different notions (WP is similar to FWP see Additional file 1). With the different consistency notions we were on average able to compute behavior predictions for up to 69 % of the remaining nodes in the network, for which no measurement was given. One can observe that the share of nodes with predictions increases drastically with notion

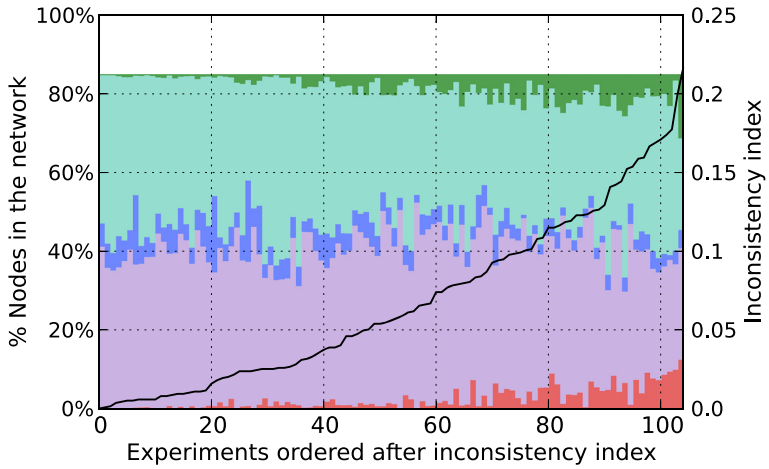



**Fig. 9** Distribution of observed signs in the experimental data. The x-axis shows the 105 experiments ordered after their inconsistency index represented by the black line, getting less consistent from left to right. The left y-axis quantifies the cumulative percentage of nodes in the network which were measured. The signs are represented by the following colors: $-$ (red), $\nabla$ (purple), 0 (blue), $\Delta$ (turquoise), and $+$ (green)

**Table 2** Average % of unobserved nodes that have predictions of a particular behavior, their information gain and the precision for these predictions under the different notions giving 10 % or 50 % of the *E. coli* expression measurements as input. In the last major column ("all predictions") the rows "% of unobserved nodes" quantify the overall recovery rates

| Prediction | +/− | | | 0 | | | Weak predictions ⊕/⊖/± | | | All predictions | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Notion | FWP | SP | FSP | FWP | SP | FSP | FWP | SP | FSP | FWP | SP | FSP |
| Obtained using 10 % of the measurements as input. | | | | | | | | | | | | |
| % of unobserved nodes | 0.13 % | 2.60 % | 2.91 % | 1.19 % | 46.43 % | 53.64 % | 7.74 % | 12.99 % | 13.14 % | 9.06 % | 62.03 % | 69.70 % |
| Information gain | 0.13 % | 2.60 % | 2.91 % | 1.19 % | 46.43 % | 53.64 % | 2.86 % | 4.80 % | 4.85 % | 4.17 % | 53.83 % | 61.41 % |
| Precision of prediction | 29.02 % | 56.28 % | 54.50 % | 82.16 % | 70.69 % | 71.66 % | 80.94 % | 85.19 % | 85.14 % | 80.03 % | 72.97 % | 73.24 % |
| ***P*-value** | | | | | | | | | | **0.2791** | **1.0389e-08** | **8.0897e-10** |
| Obtained using 50 % of the measurements as input. | | | | | | | | | | | | |
| % of unobserved nodes | 0.48 % | 3.06 % | 3.08 % | 4.84 % | 67.93 % | 72.30 % | 28.99 % | 7.27 % | 7.07 % | 34.31 % | 78.26 % | 82.45 % |
| Information gain | 0.48 % | 3.06 % | 3.08 % | 4.84 % | 67.93 % | 72.30 % | 10.70 % | 2.68 % | 2.61 % | 16.02 % | 73.67 % | 77.99 % |
| Precision of prediction | 24.10 % | 62.16 % | 62.14 % | 82.79 % | 70.22 % | 70.97 % | 82.44 % | 86.75 % | 86.80 % | 81.33 % | 71.10 % | 71.57 % |
| ***P*-value** | | | | | | | | | | **0.5** | **2.9782e-09** | **2.0785e-10** |

SP and even further with FSP, mainly through an increased prediction of 0-change behaviors.

The different types of predictions contain different amount of informations. A weak prediction gives less information than a strong prediction because it discards only one out of three possible labels. Hence, the 69 % of nodes with prediction does not equal 69 % of information gained. Therefore, we also computed the information gain given by these predictions. For $n$ unconstrained nodes, for which no measurements are taken into account, $3^n$ possible behaviors exist, for $k$ nodes with strong predictions the possible behaviors can be restricted to just 1, and for $l$ nodes with weak predictions remain still $2^l$ possible behaviors, for $m$ nodes without predictions remain still $3^m$ possible behaviors, and the overall information gain can then be expressed as $(log(3^n) - log(1^k + 2^l + 3^m)/log(3^n))$. In our experiments we observed an average information gain up to 61 % for the nodes for which no measurements had been taken into account. For more information on how to compute the information gain we refer to the Additional file 1.

To validate the quality of the predictions (obtained from 10 % of the data), we compared them with the validation data (the remaining 90 % of measurements). For the nodes where a prediction and validation data was available, we compared both. We obtained on average precisions that range from 73 % to 80 %. Overall, SP and FSP allow us to make predictions for a much bigger part of the network, resulting in a much higher information gain with only a slightly decreased precision, and for + and − predictions with a significant higher precision than FWP. In Section 5 of the Additional file 1 we plot the detailed recovery and precisions per experiment for notions FWP and FSP.

To test the influence of the number of measurements on recovery rate and precision, we also created a dataset with 50 % and 75 % (see Additional file 1: Table S3) of the measurements. Compared to the results with 10 % the overall recovery rate increases up to 82 % (FSP). This is due to the fact that the increased amount of data helps to put more constraints on the systems behavior. For notion SP and FSP the number of weak predictions drops slightly because many of them become strong predictions. The precision of +, − and weak predictions benefits from the richer datasets under notion SP and FSP, while the precision of 0-change decreases only slightly.

Weak predictions easily have higher precisions, because they have a bigger chance to be true positives. To validate that the precisions obtained in our test case are indeed meaningful, we tested our approach on a randomized dataset. We could verify that the predictions from randomized data have less precision than the predictions obtained from the real data (see Additional file 1: Table S3),especially for notions SP and FSP. Accordingly the p-values shown in Table 2 indicate a high significance that the predictions made by SP and, even more pronounced, by FSP are better than random.

These results show that the strong-propagation notions (SP and FSP) are the most pertinent choice to explain gene expression shifts within the *E. coli* transcriptional network. Using FSP we predict with high precision

Prediction given 10%, Notion FWP

| | − | 0 | + | ⊖ | ± | ⊕ | no |
|---|---|---|---|---|---|---|---|
| − | 0 | 0 | 0 | 1 | 0 | 1 | 19 |
| ▽ | 0 | 6 | 0 | 20 | 0 | 20 | 573 |
| 0 | 0 | 1 | 0 | 2 | 0 | 2 | 71 |
| △ | 0 | 6 | 0 | 18 | 0 | 20 | 529 |
| + | 0 | 0 | 0 | 1 | 0 | 2 | 27 |
| no | 0 | 5 | 0 | 14 | 0 | 15 | 293 |

Prediction given 50%, Notion FWP

| | − | 0 | + | ⊖ | ± | ⊕ | no |
|---|---|---|---|---|---|---|---|
| − | 0 | 0 | 0 | 2 | 0 | 1 | 18 |
| ▽ | 1 | 15 | 1 | 43 | 0 | 43 | 518 |
| 0 | 0 | 2 | 0 | 5 | 0 | 5 | 64 |
| △ | 1 | 14 | 1 | 39 | 0 | 42 | 478 |
| + | 0 | 0 | 0 | 2 | 0 | 3 | 25 |
| no | 0 | 17 | 0 | 49 | 0 | 53 | 207 |

Notion FSP

| | − | 0 | + | ⊖ | ± | ⊕ | no |
|---|---|---|---|---|---|---|---|
| − | 1 | 6 | 0 | 1 | 0 | 1 | 11 |
| ▽ | 8 | 290 | 6 | 42 | 0 | 32 | 241 |
| 0 | 1 | 38 | 1 | 5 | 1 | 5 | 27 |
| △ | 6 | 276 | 8 | 32 | 1 | 39 | 212 |
| + | 1 | 9 | 2 | 1 | 0 | 2 | 15 |
| no | 5 | 195 | 5 | 19 | 1 | 17 | 86 |

Notion FSP

| | − | 0 | + | ⊖ | ± | ⊕ | no |
|---|---|---|---|---|---|---|---|
| − | 1 | 5 | 0 | 1 | 0 | 0 | 14 |
| ▽ | 5 | 224 | 3 | 13 | 0 | 8 | 366 |
| 0 | 0 | 29 | 0 | 1 | 1 | 1 | 43 |
| △ | 3 | 211 | 5 | 8 | 1 | 12 | 335 |
| + | 0 | 7 | 2 | 0 | 0 | 1 | 19 |
| no | 4 | 239 | 5 | 12 | 0 | 10 | 56 |

**Fig. 10** Confusion matrices for predicted behaviors under each notion given 10% (left) 50% (right) of the data. The columns indicate the predicted behavior and the rows the measured behavior. Given are the average numbers of nodes predicted to change as indicated by the column and measured in the experiment as indicated by the row with respect to the total number of measured predictions. Predictions that are consistent with the measurement are in green and confusions in red. Gray fields denote either predictions that could not be verified because no validation data was available or nodes for which validation data was available but no predictions had been made

that 53 % to 72 % of the network remains unaltered (0-change). Understanding the differentially expressed network regions becomes more delicate, since the precision remains on average 54 % to 62 % which, however, is still significantly higher than for notion FWP. Nevertheless, 48 % of the experiments had a precision above 75 % for up- or down-regulation (strong) predictions when considering a dataset with 50 % of the measures. Note, that the notion of precision changes its conclusiveness when applied to incompletely determined predictions. Thus, we use confusion matrices as an alternative representation to illustrates the performance of our prediction method. Here one can see that for uncertain observations, relatively few strong predictions are confused (see Fig. 10). Therefore, wrong predictions may be related to the choice of the discretization thresholds and that a single threshold was chosen for all genes.

## Conclusion

We presented a unified framework to express different notions of sign consistency on interaction graphs. A refined methodology for data discretization into five values allows the consideration of uncertainties in experimental profiles. Within this framework we introduced a new constraint to filter undesired self-fulfilled regulations that result from positive feedback loops. Finally, our extended prediction method considers not only strong (unique value) but additionally weak (multiple admissible values) predictions, enlarging the predictive power of the approach.

We evaluated our framework by confronting the full RegulonDB network with 105 experimental gene-expression profiles. Our cross-validation results obtained when choosing 10 % of the initial dataset show that the overall precision of the methods ranges from 72 % to 80 %. The precision of the FSP notion has a much higher and significant p-value. With its increased precision and recovery, FSP appears to be the superior notion.

We expect that the information gain is in general higher for datasets from (typically smaller) signaling networks (see e.g. [17]). This might be due to the fact that in the stress experiments considered here the (perturbed) inputs of the gene regulatory network were unknown which poses less constraints than in signaling networks with normally well-defined signal inputs (given by the applied ligands, inhibitors etc.).

Our method requires a careful selection of discretization thresholds. Therefore, we performed a detailed sensitivity analysis on a wide range of the discretization thresholds (see Additional file 1: Section 4). The analysis shows that there is a relatively small sensitivity of the results (precision, information gain) w.r.t. the chosen thresholds. We also discuss further aspects of threshold selection in the Additional file 1.

There is a relationship between the concept of sign consistency and the dependency matrix (discussed in more detail in [17]). The notion of the dependency matrix was originally introduced in [4] and has been used in several studies for checking consistency between signaling network topologies and experimental data from stimulus-response experiments, (e.g., [5, 29]). In fact, the dependency matrix can be seen as another sign consistency notion which is more relaxed than SP or FSP (what might still be useful, e.g. when analyzing transient instead of steady-state responses). Since additional propagation rules are straightforward to implement in the framework presented herein, other sign consistency notions, including the dependency matrix or those that pose different constraints for 0-changes, could be considered as well. Overall, our work enhances the flexibility and power of the sign consistency approach for the prediction of the behavior of signaling and gene regulatory networks and, more generally, for the validation and inference of these networks.

## Additional file

**Additional file 1: Supplementary.** Contains the following supplementary material. Explanation of SCEN-FIT. Explanation of uncertain observations. Information gain by predictions in the sign consistency approach. Sensitivity analysis - Choosing the thresholds for discretization. Recovery and precision for *E. coli* cross-validation experiments. GEO/GSEcodes for the experiments used. (PDF 1218 kb)

### Abbreviations

IG: Influence graph; ILP: Integer linear programming; ASP: Answer set programming; WP: Weak propagation; SP: Strong propagation; FWP: Founded weak propagation; FSP: Founded strong propagation.

### Competing interests

The authors declare that they have no competing interests.

### Authors' contributions

ST, SK, AS, and CG conceived and supervised the study. LC and JSR contributed to the investigation of different sign consistency notions. ST implemented iggy. ST and CG calculated results for the E. coli case study. All authors discussed results of data analysis. ST, CG, and SK drafted the manuscript. All authors read and approved the final manuscript.

### Acknowledgements

This work was funded in part by the German Federal Ministry of Education and Research within the "Virtual Liver Network" (grant 0315744) and "JAK-Sys" (grant 0316167B).

### Author details

[1] Max Planck Institute for Dynamics of Complex Technical Systems, Sandtorstraße 1, 39106 Magdeburg, Germany. [2] European Molecular Biology Laboratory, European Bioinformatics Institute, Hinxton CB101SD, UK. [3] CNRS, UMR 6074 IRISA, Campus de Beaulieu, 35042 Rennes, France. [4] INRIA, Dyliss project, Campus de Beaulieu, 35042 Rennes, France. [5] École Centrale de Nantes, IRCCyN UMR 6597, 1 rue de la Noë, 44321 Nantes, France.

Received: 30 April 2015 Accepted: 9 September 2015
Published online: 28 October 2015

**References**


1. Catlett NL, Bargnesi AJ, Ungerer S, Seagaran T, Ladd W, Elliston KO, et al. Reverse causal reasoning: applying qualitative causal knowledge to the interpretation of high-throughput data. BMC Bioinforma. 2013;14:340.
2. Guziołowski C, Bourde A, Moreews F, Siegel A. BioQuali Cytoscape plugin: analysing the global consistency of regulatory networks. BMC Genomics. 2009;10(1):244. doi:10.1186/1471-2164-10-244.
3. Gebser M, Schaub T, Thiele S, Veber P. Detecting inconsistencies in large biological networks with answer set programming. Theory Prac Logic Program. 2011;11(2–3):323–60.
4. Klamt S, Saez-Rodriguez J, Lindquist J, Simeoni L, Gilles E. A methodology for the structural and functional analysis of signaling and regulatory networks. BMC Bioinforma. 2006;7(1):56. doi:10.1186/1471-2105-7-56.
5. Samaga R, Saez-Rodriguez J, Alexopoulos LG, Sorger PK, Klamt S. The Logic of EGFR/ErbB Signaling: Theoretical Properties and Analysis of High-Throughput Data. PLoS Comput Biol. 2009;5(8):1000438. doi:10.1371/journal.pcbi.1000438.
6. Thieffry D. Dynamical roles of biological regulatory circuits. Brief. Bioinforma. 2007;8(4):220–5. doi:10.1093/bib/bbm028 http://bib.oxfordjournals.org/content/8/4/220.full.pdf+html.
7. Sachs K, Perez O, Pe'er D, Lauffenburger DA, Nolan GP. Causal Protein-Signaling Networks Derived from Multiparameter Single-Cell Data. Science. 2005;308(5721):523–9. doi:10.1126/science.1105809.
8. Morris MK, Saez-Rodriguez J, Sorger PK, Lauffenburger DA. Logic-based models for the analysis of cell signaling networks. Biochemistry. 2010;49(15):3216–24.
9. Wang RS, Saadatpour A, Albert R. Boolean modeling in systems biology: an overview of methodology and applications. Phys Biol. 2012;9(5): 055001.
10. Schoeberl B, Eichler-Jonsson C, Gilles ED, Müller G. Computational modeling of the dynamics of the map kinase cascade activated by surface and internalized egf receptors. Nat Biotechnol. 2002;20(4):370–5.
11. Quach M, Brunel N, d'Alché-Buc F. Estimating parameters and hidden variables in non-linear state-space models based on odes for biological networks inference. Bioinforma. 2007;23(23):3209–16.
12. Karlebach G, Shamir R. Modelling and analysis of gene regulatory networks. Nat Rev Mol Cell Biol. 2008;9(10):770–80.
13. Ideker TE, Thorsson V, Karp RM. Discovery of Regulatory Interactions Through Perturbation: Inference and Experimental Design. In: Proceedings of the Pacific Symposium on Biocomputing. Seattle, USA: World Scientific Press; 2000.
14. Saez-Rodriguez J, Alexopoulos LG, Epperlein J, Samaga R, Lauffenburger DA, Klamt S, et al. Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction. Mol Syst Biol. 2009;5(1):331.
15. Sharan R, Karp R. Reconstructing boolean models of signaling In: Chor B, editor. Research in Computational Molecular Biology. Lecture Notes in Computer Science. Springer; 2012. p. 261–71. doi:10.1007/978-3-642-29 627-7_28. http://dx.doi.org/10.1007/978-3-642-29627-7_28.
16. Terfve C, Cokelaer T, Henriques D, MacNamara A, Goncalves E, Morris M, et al. Cellnoptr: a flexible toolkit to train protein signaling networks to data using multiple logic formalisms. BMC Syst Biol. 2012;6(1):133. doi:10.1186/1752-0509-6-133.
17. Melas IN, Samaga R, Alexopoulos LG, Klamt S. Detecting and Removing Inconsistencies between Experimental Data and Signaling Network Topologies Using Integer Linear Programming on Interaction Graphs. PLoS Comput Biol. 2013;9(9):1003204. doi:10.1371/journal.pcbi.1003204.
18. Videla S, Guziołowski C, Eduati F, Thiele S, Gebser M, Nicolas J, et al. Learning Boolean logic models of signaling networks with ASP. Theoretical Computer Science. 2015;599:79–101. Advances in Computational Methods in Systems Biology, doi:10.1016/j.tcs.2014.06.022, http://www.sciencedirect.com/science/article/pii/S0304397514004587.
19. Radde N, Bar NS, Banaji M. Graphical methods for analysing feedback in biological networks - a survey. Int J Syst Sci. 2010;41(1):35–46. doi:10.1080/00207720903151326. http://dx.doi.org/10.1080/ 00207720903151326.
20. Samaga R, Klamt S. Modeling approaches for qualitative and semi-quantitative analysis of cellular signaling networks. Cell Commun Signal. 2013;11(1):43. doi:10.1186/1478-811X-11-43.
21. Kuipers B. Qualitative reasoning: Modeling and simulation with incomplete knowledge. Automatica. 1989;25(4):571–85. doi:10.1016/0005-1098(89)90099-X.
22. Gebser M, Guziołowski C, Ivanchev M, Schaub T, Siegel A, Thiele S, et al. Repair and prediction (under inconsistency) in large biological networks with answer set programming In: Lin F, Sattler U, Truszczynski M, editors. Proceedings of the Twelfth International Conference on the Principles of Knowledge Representation and Reasoning (KR'10). Menlo Park, CA: AAAI Press; 2010.
23. Baral C. Knowledge Representation, Reasoning and Declarative Problem Solving. Cambridge: Cambridge University Press; 2003.
24. Thiele S. Iggy-1.2: A tool for consistency based analysis of influence graphs and observed systems behavior. zenodo.org. 2015. doi:10.5281/zenodo.19042. http://dx.doi.org/10.5281/zenodo.19042.
25. Gebser M, Kaminski R, Kaufmann B, Ostrowski M, Schaub T, Thiele S. A User's Guide to `gringo`, `clasp`, `clingo`, and `iclingo`. 2010. http://potassco.sourceforge.net. Accessed 10 Oct 2015.
26. Thiele S. PyASP 1.4.1 - A convenience wrapper for the ASP tools gringo, gringo4 and clasp. 2015. doi:10.5281/zenodo.22968. http://dx.doi.org/10. 5281/zenodo.22968.
27. Salgado H, Gama-Castro S, Peralta-Gil M, Diaz-Peredo E, Sanchez-Solano F, Santos-Zavaleta A, et al. RegulonDB (version 5.0): Escherichia coli K-12 transcriptional regulatory network, operon organization, and growth conditions. Nucleic Acids Res. 2006;34(Database issue):394–7.
28. Sangurdekar DP, Srienc F, Khodursky AB. A classification based framework for quantitative description of large-scale microarray data. Genome Biol. 2006;7(4):32.
29. Ryll A, Samaga R, Schaper F, Alexopoulos LG, Klamt S. Large-scale network models of il-1 and il-6 signalling and their hepatocellular specification. Mol BioSyst. 2011;7:3253–270. doi:10.1039/C1MB05261F.

## A.2 Multiple Myeloma sign-consistency modelling

SCIENTIFIC REPORTS

OPEN

# Logic programming reveals alteration of key transcription factors in multiple myeloma

Received: 29 November 2016
Accepted: 25 July 2017
Published online: 23 August 2017

Bertrand Miannay[1,2], Stéphane Minvielle[2,3], Olivier Roux[1], Pierre Drouin[1], Hervé Avet-Loiseau[4], Catherine Guérin-Charbonnel[2,5], Wilfried Gouraud[2,5], Michel Attal[6], Thierry Facon[7], Nikhil C Munshi[8,9], Philippe Moreau[2,3], Loïc Campion[2,5], Florence Magrangeas[2,3] & Carito Guziolowski[1]

**Innovative approaches combining regulatory networks (RN) and genomic data are needed to extract biological information for a better understanding of diseases, such as cancer, by improving the identification of entities and thereby leading to potential new therapeutic avenues. In this study, we confronted an automatically generated RN with gene expression profiles (GEP) from a cohort of multiple myeloma (MM) patients and normal individuals using global reasoning on the RN causality to identify key-nodes. We modeled each patient by his or her GEP, the RN and the possible automatically detected repairs needed to establish a coherent flow of the information that explains the logic of the GEP. These repairs could represent cancer mutations leading to GEP variability. With this reasoning, unmeasured protein states can be inferred, and we can simulate the impact of a protein perturbation on the RN behavior to identify therapeutic targets. We showed that JUN/FOS and FOXM1 activities are altered in almost all MM patients and identified two survival markers for MM patients. Our results suggest that JUN/FOS-activation has a strong impact on the RN in view of the whole GEP, whereas FOXM1-activation could be an interesting way to perturb an MM subgroup identified by our method.**

Multiple myeloma (MM) is a neoplasm of plasma cells with an incidence rate of approximatively 5/100,000 in Europe. The median survival of MM patients has improved substantially over the past decade. Owing to the establishment of high-dose therapy followed by autologous stem cell transplantation as a routine procedure, significant improvements in supportive care strategies, and the introduction and widespread use of the immunomodulatory drugs thalidomide and lenalidomide, and the proteasome inhibitor bortezomib. Nevertheless, almost all MM patients ultimately relapse, and new drugs and new combinations for the treatment of MM are warranted. MM is a heterogeneous disease at both the clinical and molecular levels. Recent large scale genomics analysis based on the landscape of copy-number alterations and on whole exome sequencing have revealed the hallmarks of genetic changes in MM such as hyperdiploidy, translocations involving the IgH locus, and mutations in the RAS/MAP and NF-kB pathways and in TP53[1]. These genetic changes as well as gene-expression profiling (GEP) have been widely used in the molecular classification of newly diagnosed patients to define diagnostic entities and identify promising new therapeutic targets[2–7]. However, at present a standard of classification based on subgroups that could be targeted therapeutically is still being debated. Clearly, there is a need for innovative tools to improve the identification of the prognostically relevant entities, clinically and biologically, in newly diagnosed MM patients. It is tempting to use the mutational spectrum based on whole-exome sequencing as a gold standard; however we have previously shown that a large number of exome mutant alleles are not expressed clinically or biologically[8]. In addition, exome sequencing may miss potential driver mutations in the non coding regulatory elements known to affect enhancer activity, which thereby affect the transcriptional program[9]; therefore GEP remains a tool of choice.

[1]LS2N, UMR 6004, École Centrale de Nantes, Nantes, France. [2]CRCINA, INSERM, CNRS, Université d'Angers, Université de Nantes, Nantes, France. [3]CHU de Nantes, Nantes, France. [4]Unit for Genomics in Myeloma, IUC-Oncopole; and, CRCT INSERM 1037, Toulouse, France. [5]Institut de Cancérologie de l'Ouest, Nantes, France. [6]Department of Hematology, IUC, Toulouse, France. [7]Department of Hematology, CHU, Lille, France. [8]Lebow Institute of Myeloma Therapeutics and Jerome Lipper Multiple Myeloma Center, Dana-Farber Cancer Institute, Harvard Medical School, Boston, MA, 02115, USA. [9]Boston Veterans Administration Healthcare System, West Roxbury, MA, 02132, USA. Correspondence and requests for materials should be addressed to F.M. (email: Florence.Magrangeas@univ-nantes.fr) or C.G. (email: Carito.Guziolowski@ls2n.fr)

However, GEP alone is limited and must be integrated with innovative approaches that use biological regulatory networks to extract biological information relative to gene expression datasets to provide significant clues about the etiology of myeloma.

During the past decade, many methods of so-called *pathway analysis* or *active pathways detection* have been developed. These methods use as a knowledge base a biological pathway or regulatory network, that compiles a series of molecular phenomena that lead to activation (or inhibition) of gene expression, a cell product such as a hormone, or a physical modification of the cell. Regulatory network information is currently available through databases such as Gene Ontology (GO)[10], the Kyoto Encyclopedia of Genes and Genomes (KEGG)[11], the Pathway Interaction Database (PID)[12], Wikipathway[13], Transfac[14], and Causal Biological Networks (CBN)[15]. The main objective of pathway analysis methods is to confront or integrate GEP data with regulatory networks or pathways to distinguish two or more classes of cells (e.g. healthy vs ill) from GEP data by inferring a specific signature for each class. We can identify three principal categories of approaches that have been used to associate GEP with specific pathways[16].

The Over-Representation Analysis (ORA) group of approaches[17,18] includes approaches that are based on differentially expressed (DE) genes. These approaches score single pathways based on the proportion of DE genes (identified with statistical tests or with a threshold) contained in each pathway. In most cases, these methods use a hyper-geometric test[17] to score each pathway. Moreover, the majority of ORA approaches that use functional annotation (GO) or pathway maps (KEGG) consider the consequences of the DE genes (leading to the differentially expression of proteins) in the associations between gene and pathway. Martin *et al.*[19] called this type of reasoning *forward assumption* compared to the *backward assumption*[18], which considers the causes of those DE genes in the gene-pathway association.

The Functional Class Scoring (FCS) group of approaches uses the full datasets without any pre-selection, allowing integration of the effects of low gene expression variations in the identification of the pathways involved. FCS approaches can use forward[20,21] or backward[22,23] reasoning. Although these methods improve the problem of genes selection, the pathways in which individual genes are involved are still studied independently. Moreover, the position of the genes in the topology is not used in the analysis.

The Pathway Topology (PT) approaches are very similar to the FCS approaches, but in addition, they score genes according to the pathways to which they belong. Whereas some of these approaches only include interactions between genes[24–27], others consider different types of relationships between genes[19,28], generally activation and inhibition. The majority of methods study each pathway independently. Within this group, we can also identify methods that use both forward[24–27] and backward[19,28] reasoning.

In this work, we propose to integrate the GEPs obtained from myeloma cells (MC) of 602 MM patients and from normal plasma cells (NPC) of 9 healthy donors with the whole compendium of the PID-NCI public pathway repository so as to better understand the mechanisms of plasma cell carcinogenesis. To integrate this data, we first automatically build a directed (and labeled) graph using the whole compendium of the PID-NCI public pathway repository. This graph connects signaling pathways to the transcription of the genes in the GEP dataset. We then integrate the graph with the expression data by reasoning on its logic using IGGY[29], a tool based on logic programming (Answer Set Programming) that confronts a node coloring (GEP) with labeled and directed graphs. Our combined approach could be considered to fall within the PT category since it takes into account the causality and activation/inhibition logic of graph edges. However, unlike previous methods, it uses a global logic to analyze experimental and pathway data. In this formalism, both forward and backward modes are included as reasoning modes (causes-consequences). IGGY allows us to check the consistency of the information and to generate predictions based upon automatic repairs for upstream non-measured species. It uses DE data as well as the identically expressed genes across classes (invariant genes) in its analysis. The proposed method does not correlate protein activation with gene expression; the two entities are identified separately in the graph. The non-measured protein activations necessary to satisfy the GEP according to the entire pathway database topology are used later to propose a signature for each dataset profile. This global signature can be used to characterize the dataset classes. Moreover, our model also allows us to *in silico* quantify the effect of perturbations on this global pathway for each single patient. We show how this type of method, which combines large-scale information in terms of number of patients, the complete GEP, and the entire compendium database, can be applied to identify new specificities of MM disease compared to normal cells. As a result, we inferred information on the states of specific proteins in the cell that may cause these disorders, and we identified specific markers of MC compared to NPC that can be used to identify survival markers. Furthermore, these markers can be studied as therapeutic targets because of their over-representation and their impact on the involved pathways.

## Materials and Methods

**Data.** *Experimental Procedures.* Plasma cells were isolated from the bone marrow of 602 newly diagnosed cases of MM. The samples were obtained during standard diagnostic procedures conducted at the Intergroupe Francophone du Myélome (IFM) centers. The subjects included patients younger than 65 years of age who were enrolled in either the IFM 2005–01 trial (n = 311) or the IFM-2007-02 (n = 128) trial, older patients enrolled in the IFM-2007-01/Multiple Myeloma 020 trial (n = 76) and 9 normal donors. The experiments were undertaken with the understanding and written informed consent of each subject. Plasma cell purification was performed as previously described[30]. Purified plasma cells were frozen at −80 °C in lysis buffer. Approval for this study was obtained from the University Hospital of Nantes. The study fulfilled the requirements of the Declaration of Helsinki.

*Gene expression profiling.* RNA was extracted using the AllPrep DNA/RNA MiniKit or the RNeasy Micro kit (QIAGEN, Valencia, CA, USA) in accordance with the manufacturer's instructions. RNA quality and quantity were assessed using Agilent 2100 Bioanalyzer (Agilent, Palo Alto, CA, USA) and a Nanodrop Spectrophotometer

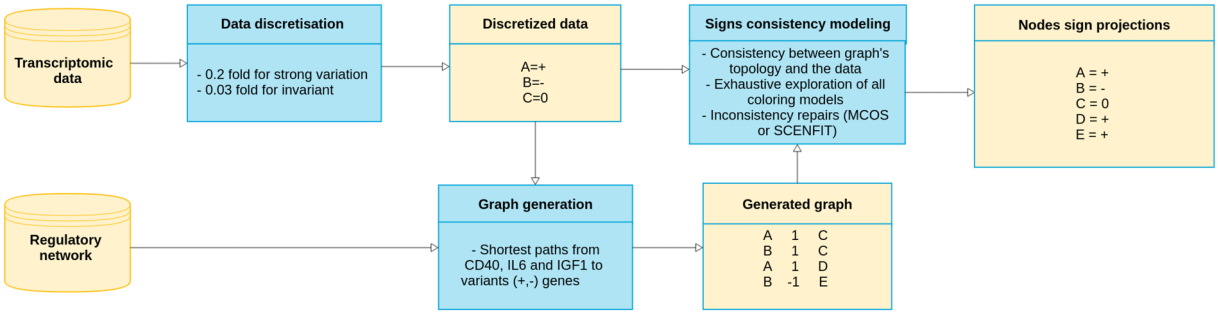



**Figure 1.** Overview of the sign consistency modeling framework.

(NanoDrop Technologies, DE, USA), respectively. MM samples from which 50 ng of total RNA was available were processed according to manufacturer's instructions (NuGEN, San Carlos, CA, USA) before labeling and hybridization onto an Affymetrix Human Exon1.0 chip according to the manufacturer's instructions (Affymetrix, Santa Clara, CA). Data were analyzed and $log_2$ normalized with Expression Console Affymetrix software v1.1 using an RMA algorithm.

*Data discretization.* Since our *graph coloring model* requires invariant genes, classical discretization methods cannot be used in our study. To identify over-/underexpressed and invariant gene expression for each profile, we used two thresholds: $k_1$ for invariant genes and $k_2$ for variant genes. For each gene $g$, we computed a vector, $p^g$, composed of its differential expression, $p_i^g$, in each dataset expression profile $i$. A profile $i$ refers to each of the 611 cells of type MC or NPC considered in this study. $p_i^g$ values were computed by subtracting the mean expression of g in the NPC set from its gene expression level in MC (Supplementary Material, Figure S1). We then discretized the values of $p_i^g$ using two thresholds $k_1$ and $k_2$.

If $p_i^g > k_2$, g was considered over-expressed for $i$;
if $p_i^g < -k_2$, g was considered under-expressed for $i$;
and if $-k_1 < p_i^g < k_1$, g was considered invariant for $i$.

By choosing different combinations of values for $k_1$ and $k_2$ (see Supplementary Material), we obtained 150 sets of vectors that contain the discrete overexpressed (+), underexpressed (−), or invariant (0) values for all the genes expressed in each dataset. We discarded combinations of values leading to $p^g$ vectors with a sign proportion greater than 50%. Each $k_1 - k_2$ combination was used to test the precision of our approach. We did this 100 times by using 50% of the discretized {+, −, 0} genes' expression to predict the other 50% for each MC dataset, after which we comparing the measured and predicted data using a precision matrix (Supplementary Material, Table S1). The thresholds leading to the best precision of 43% (IC 95%: ±3%) were $k_1 = 0.03$ and $k_2 = 0.2$; these thresholds were used in the remainder of the study to select the variant and invariant genes. In the Supplementary Material, Figure S2, we show the precision obtained for all of the threshold combinations that were selected. Since our discretization method fixes the same thresholds for all genes across all profiles, we also used K-means to discover gene-specific thresholds. However, the precision of K-means methods (for $k = 3$) was lower than that obtained using the selected thresholds (see Supplementary Material, Figure S3). In the same way, to demonstrate the interest of using invariant genes, we computed the precision of recovering 50% of the data using a two-signs model that receives input data and predicts only over- and underexpressed values. The computed precision was 48%. Note that the two-signs model has a precision closer to a random precision distribution (50%), whereas the precision obtained using a three-signs model is farther from the random precision (33%).

**Graph generation.** We used the 2012 version of the complete pathways database PID-NCI (Pathway Interaction Database)[12] and downloaded it in PID-XML format. This database is specialized to include regulatory pathways involved in cancer. The complete graph contains 17,932 nodes (proteins, complexes, genes, transcription or protein modification events) and 27,976 edges (activation or inhibition). To orient our analysis to the expression profiles and to the biological problem at hand, we built a subgraph with signed edges by extracting the downstream events from three signaling pathways (IL6/IL6-R, IGF1/IGF1-R and CD40), all of which are known to include cellular receptors involved in MM[31], to the over- and underexpressed variant genes from all datasets by the shortest paths. This cycled, directed subgraph was then filtered by deleting all nodes that are not observed and with one predecessor or one successor[32]. This filtering step involves no loss of information with respect to the graph coloring model and allowed us to reduce the complexity of the analysis while maintaining the dependencies between the nodes.

**Sign consistency modeling framework.** In Fig. 1, we illustrate the input (network and transcriptomic data) and output (sign projections) information obtained when the sign consistency modeling is applied. In the following sections, we describe in detail the main modeling steps of this framework.

*Graph coloring model.* Assuming a directed graph G(V, E, $\alpha$) in which V is the set of nodes, E is the set of edges and $\alpha$ is a function labeling the edges as $\alpha: E \rightarrow \{+, -\}$, let $\beta$ be a set of observed data with $\beta: V \rightarrow \{+, -, 0\}$. In our case, $\beta$ is obtained from GEP and labels only the nodes that are preceded by a "transcription" event as reported by the PID-NCI database. Thus, there are nodes in V, such as proteins or complexes, that remain unlabeled. Our reasoning framework expresses that there is at least one state or *coloring model* of this biological system. A coloring model is an assignment $\mu: V \rightarrow \{+, -, 0\}$ of each node in V to a sign in $\{+, -, 0\}$. Let us denote by S the set of all possible coloring models; note that $|S| = 3^{|V|}$. When imposing the restrictions of $\beta$ to S, we reduce the size of all possible coloring models ($S^{\star}$) to $3^{|V|-|\beta|}$.

*Sign consistency.* The sign consistency imposes a reasoning mode over a graph $G$ and a labeling $\beta$ (Supplementary Material, Figure S4). This reasoning imposes that each $\{+, -\}$ variation (in a given coloring model) associated with a node $n$ in $V$ is explained by the variations in the direct predecessors of $n$ in $G$. This notion can be implemented with the following consistency rules, all of which can be verified automatically:

1. All the nodes fixed to be *inputs* are consistent. Usually, these nodes have no predecessors.
2. Each $\{+, -\}$ variation associated with a node $n$ in a given coloring model has to be explained by a direct predecessor of n. That is, each variant node associated with a sign in $\{+, -\}$ that is not an input needs at least one activator (inhibitor) with the same (opposite) sign.
3. Each invariant node $m$ (associated with sign 0) has to be explained either by the fact that (i) all direct predecessors of $m$ are associated with an invariant sign, or (ii) at least two direct predecessors of $m$ are associated with opposite variant signs $\{+, -\}$.

When a graph G is consistent with $\beta$, then a set $\overline{S} \subseteq S^{\star}$ of consistent coloring models can be built. A consistent solution in $\overline{S}$ will be a coloring model in which all the nodes of the graph are colored with respect to $\beta$ and respect the consistency rules.

*Repairs.* When a consistent solution does not exist, the graph topology of $G$ is not able to explain the labeling $\beta$ according to the three previously explained consistency rules. In this study, we used two approaches to restore the consistency.

**MCOS-repair:** This repair mode corrects the graph topology. Considering that the graph is not complete (missing information, generation method, etc.), we can suppose that some inconsistencies are caused by events that are missing from the graph. It is possible to correct the graph by adding a set of artificial influences (Supplementary Material, Figure S5). In this case, we use the *cardinal minimal correction set* (MCOS) of artificial influences that can be added to restore the consistency. The MCOS is in general not unique.

**SCENFIT-repair:** This repair mode corrects $\beta$ by considering wrong information in the observed data. $\beta$ will be corrected by switching the sign of the observed nodes so as to minimize the number of switches. The switch of an observed node is quantified by a cost, as described below. The set of possible minimal SCENFIT-repairs is not unique.

1. Changing a variant sign $(+, -)$ to the opposite variant sign will have a cost of 2.
2. Changing an invariant (respectively variant) into a variant (respectively invariant) sign will have a cost of 1.

*Sign projection.* After applying a repair operation, the set of consistent coloring models will be the *union* of the consistent coloring models under each minimal repair. Usually, the number of consistent coloring models is very large, and we use a projection of these models to deduce and propose insights from the graph-observations confrontation. We distinguish 7 sign projections classes:

1. 3 classes are *strong*, meaning that the node has the same sign in all consistent solutions $\{+, -, 0\}$.
2. 3 classes are *weak*, meaning that the node has 2 signs in the consistent solutions: *Not*+ $(-, 0)$, *Not*− $(+, 0)$, *change* $(+, -)$.
3. The last class means that a node has three signs in the consistent solutions:? $(+, -, 0)$.

**Key nodes identification.** In this analysis, we used the sign projections computed after restoring the consistency using the MCOS-repairs. To compare predictions across individuals using statistical and machine-learning approaches, we decomposed each sign projection result over a node $i$ in $V$ into a triplet of boolean values. The boolean value expresses whether the couple $(i, s)$, where $s \in \{+, -, 0\}$, belongs to the sign-projection result. Since we only focused on sign-projections, the nodes observed in $\beta$ were not considered. To reduce the number of variables, we excluded the boolean value that refers to invariant couples (nodes coupled with "0"). In this way, we represent the sign-projections obtained for each GEP as a boolean matrix $M$ of size $2 \times m \times (N^{MC} + N^{NPC})$, where m represents the number of nodes in G that were never observed in any GEP and $N^{MC}$ (respectively $N^{NPC}$) represents the number of profiles in class MC (respectively *NPC*). $M_{ij}$ stands for the decomposed prediction of node $i$ under profile $j$; note that $M_{ij}$ can be separated into $M_{ij}^{+}$ and $M_{ij}^{-}$, where $M_{ij}^{s} = 1$ expresses that node $i$ is predicted to be of sign $s$ in profile $j$. To identify specific markers of MM, we analyzed $M$ and looked for overrepresented values when comparing the vectors belonging to MC with those belonging to NPC. For this, we used two approaches, a machine-learning approach based on supervised learning and a

statistical approach based on frequency classification. For the supervised learning, we used a decision tree[33] and a random forest classification[34]. Due to the underrepresentation of the NPC, we increased the weight of each NPC by 67 so as to have the same order of population in each group (9 NPC and 602 MC). For the frequency approach, we calculated the frequency score (FS) for each group (MC or NPC) and for each assignment ($i$, $s$) as follows:

$$FS_{i,s}^{C} = \frac{1}{N^{C}} \sum_{j=1}^{N^{C}} M_{ij}^{s}, \tag{1}$$

where $C$ represents the class MC or NPC and $s$ represents the $\{+, -\}$ sign assigned to $i$. We then sorted our results based on a Fisher test between the proportions for NPC and MC to determine the most specific node assignments for the MC datasets.

**Nodes perturbation.** In this analysis, we quantified the effectiveness of a node perturbation (Supplementary Material, Figure S6) to simulate *in silico* the activation or inhibition of a protein. The quantification of these *in silico* perturbations was performed in two steps. First, we considered the set of assignments in $M$ (see previous section), where $M_{ij}^{s} = 1$ expresses that node $i$ is predicted to be of sign s in profile j, with $s \in \{+, -\}$. For each assignment $M_{ij}^{s}$, we generated a new dataset of observations $\beta_{ij}^{s}$ identical to the original dataset of profile $j$ ($\beta_j$) except that we added an observation on node $i$ fixed to $s \in \{+, -\}$. We then computed the SCENFIT score $SF_{ij}^{s}$ between the graph $G$ and $\beta_{ij}^{s}$. The second step consisted of computing the Top Perturbation Score (TPS) for each assignment ($i$, $s$) according to its $SF_{ij}^{s}$ across all GEP $j$, as follows:

$$TPS_{i,s}^{C} = \frac{1}{N^{C}} \sum_{j=1}^{N^{C}} f(i, s, j),$$

where

$$f(i, s, j) = \begin{cases} 1, & \text{if } SF_{ij}^{s} \geq top(SF_{kj}^{s}),\ \forall\ k \in V \backslash Dom(\beta_{ij}^{s}). \\ 0, & \text{otherwise.} \end{cases}$$

In these equations, $C$ represents the class MC orv NPC, and $s$ represents the $\{+, -\}$ sign assigned to $i$. The function $top(SF_{kj}^{s})$ will compute the threshold score that separates the 10% top-ranked SCENFIT scores of profile $j$, that is, those perturbations that generate the highest number of SCENFIT repairs.

**Software and tools.** For the sign consistency analysis, we used IGGY[29], which makes use of an ASP[35] description of the consistency problem. The graph generation and the mapping of predictions to the couples node-sign were implemented with Python 2.7 using the package NetworkX[36]. The learning and statistical analysis was conducted using R[37]. The computation associated with testing the consistency of the 611 GEP required 5 minutes on a standard machine. All the calculations of nodes perturbations were conducted using the BIRD infrastructure (www.pf-bird.univ-nantes.fr) with 320 nodes and 1.3To RAM.

**Graphs availability.** All graphs used in this study are available online using cynetshare. The subgraphs of NCI-PID before (goo.gl/upfzwC) and after compaction (goo.gl/SfNSv4). The subgraph from Fig. 2 is available at goo.gl/YgHvtQ. The cytoscape session containing all graphs is available at goo.gl/V1Rno5.

## Results

**Data discretization and graph generation.** The NCI-PID integration allowed us to find 634 genes (a protein preceded by a transcription event). Independently, our discretization method (Supplementary Material, Figure S1) proposed observations $\{+, -, 0\}$ on microarray probes corresponding to 15,418 proteins identified in Uniprot. Merging both lists allowed us to identify 557 genes present in the NCI-PID and observed as over/underexpressed or invariant in our GEPs. Variant and invariant genes are distributed across the MC and NPC datasets (Table 1). By extracting the downstream events from three signaling pathways (IL6/IL6-R, IGF1/IGF1-R and CD40)[31] to the variant genes, we generated an induced subgraph from NCI-PID containing 2,269 nodes, 2,683 edges and connecting 529 variant genes. This graph was then compacted to a new graph with 596 nodes and 960 edges (Fig. 2) and composed of 529 observed nodes (genes) and 67 unobserved nodes, including 23 proteins, 33 complexes, 2 biological processes, 9 proteins reactions (translocation, phosphorylation, etc.).

**Validation of predictions.** The confrontation between the data and the graph topology allowed us to predict the node signs for each dataset (Table 1). To validate our predictions, we compared the precision of the predictions with that of the randomized data. In this case, we used 50% of the measured genes $\{+, -, 0\}$ to predict the other half of the genes for each sample; we performed the same process after randomizing the data and repeated this computation up to 1000 times. We obtained two sets of precisions (Fig. 3; Supplementary Material, Table S1). A two-tailed t-test yielded a p-value lower than 2.2e–16. This shows the efficiency of our prediction method in comparison with random precision.

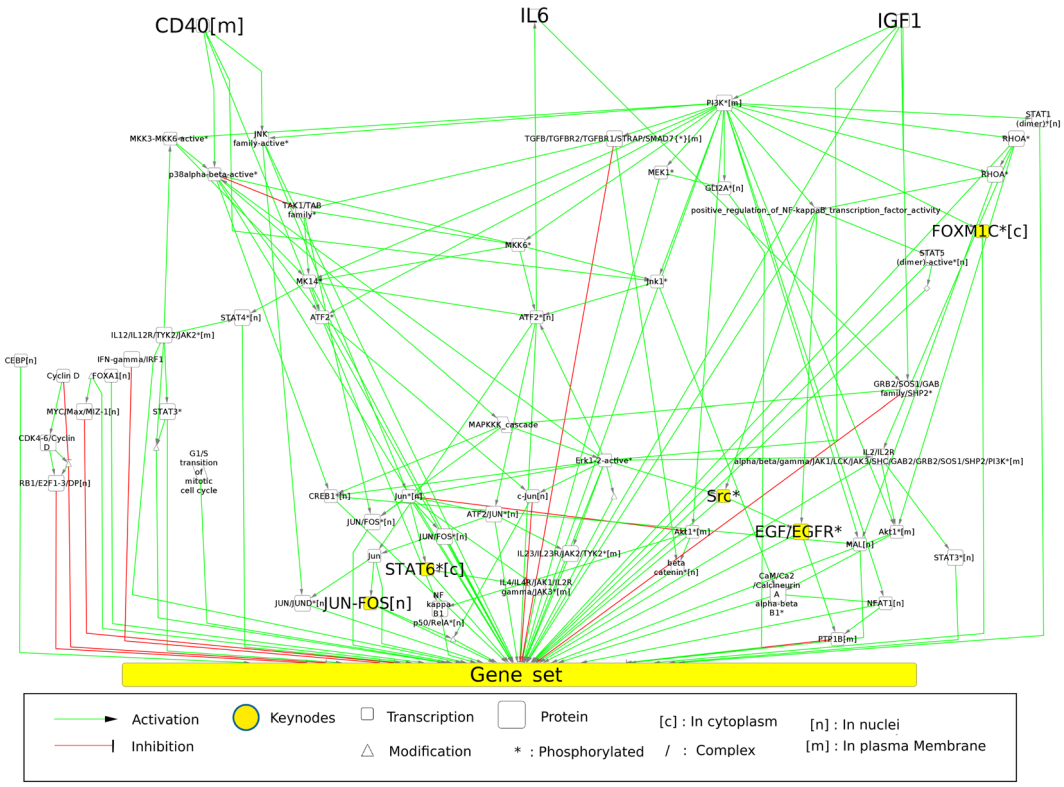



**Figure 2.** Representation of the subgraph obtained from the PID-NCI database. CD40, IL6 and IGF1 (the nodes in the top portion of the graph) are the 3 queried pathways. The 529 genes that are differentially expressed across all profiles are merged for this representation in the node "Gene set". The 5 top-ranked nodes according to their FS are labeled in bold type and colored in yellow. We used the same syntax for all nodes in this study. The edges from the "Gene set" node to proteins have been deleted for the sake of clarity.

| | Observed data | | Predicted data | |
|---|---|---|---|---|
| **Signs** | **NPC** | **MC** | **NPC** | **MC** |
| + | 34% | 38% | 30% | 31% |
| − | 34% | 51% | 29% | 36% |
| 0 | 32% | 11% | 14% | 3% |
| change | — | — | 7% | 6% |
| Not+ | — | — | 2% | 1% |
| Not− | — | — | 3% | 1% |
| ? | — | — | 15% | 22% |
| Total | 2085 | 210975 | 3279 | 153181 |

**Table 1.** Observed and predicted data repartition between NPC and MC. Observed data are the data extracted from the gene expression profiles. Predicted data are the sign projections predicted after the confrontation between the observed data and the PID-NCI graph. In the last row, we show the total observations and predictions across all profiles.

**Identification of specific node assignments for MC.** To identify MC subgroups, we applied a decision tree algorithm to the presence/absence value of a sign prediction (see Methods section). This result is illustrated in Fig. 4. It shows that the combination of the assignments (JUN/FOS[n], −) and (FOXM1*[c], −) is associated with the majority of MC (73%) and that the method can distinguish MC from NPC. JUN/FOS[n] represents the protein complex composed of JUN and FOS, which is located in the nucleus, whereas FOXM1*[c] represents the FOXM1 protein, which is phosphorylated and located in the cytoplasm. The full node syntax is given in Fig. 2. Moreover, we can identify another important group of MC (13%) that is characterized by the presence of (JUN/FOS[n], −) and the absence of (FOXM1*[c], −) and (SRC*, −). Similar results were obtained using a random forest classification (Supplementary Material, Figure S7).

To characterize the shared specificity for all MC, we computed the frequency scores (FS) for our predictions (see Methods section). The complete list of the FS obtained is shown in Table S2 of the Supplementary Material. In Table 2, we show the 5 best p-values associated with a Fisher test with $FS_{MC} > FS_{NPC}$. For these

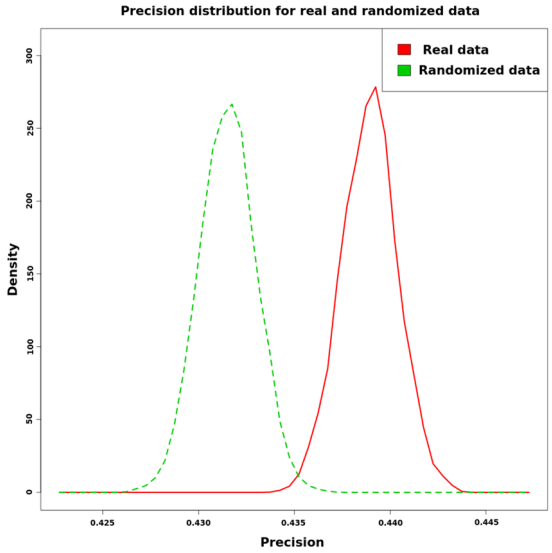



**Figure 3.** Precision distribution of our method with real observed and randomized data.

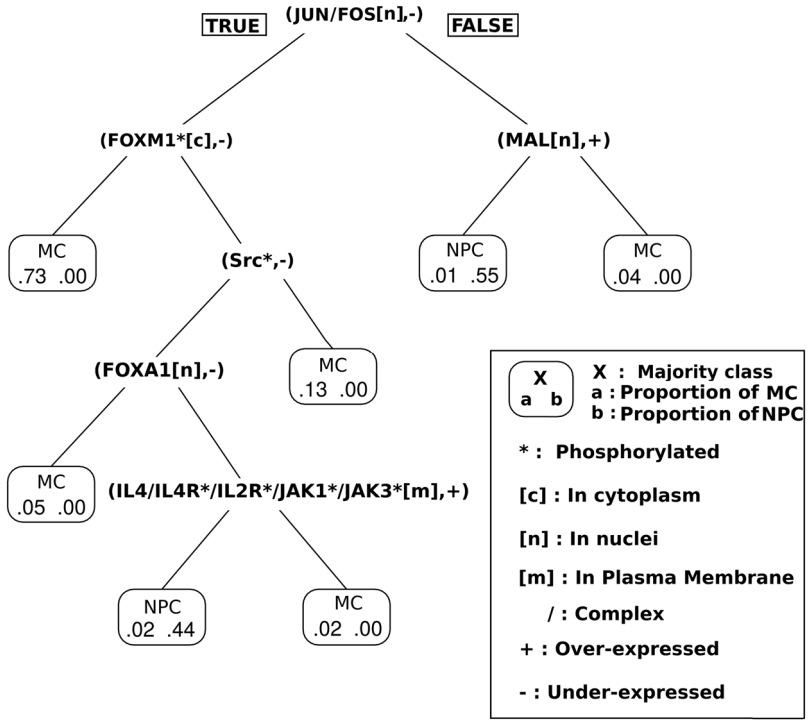



**Figure 4.** Decision tree based on predicted node-sign assignments.

assignments, we checked the number of input/output variant genes connected to each node in the graph (Table 2, column connectivity). We observe that inhibition of the complex JUN/FOS[n] is predicted for 95.6% of MC. The activity levels of FOXM1*[c] and STAT6*[c] were predicted to decrease. This decrease, in terms of protein activity, is correlated with the level of gene expression in 76% and 93%, respectively, of the MC datasets. The FS classification identified the presence of (Src*, +) as an interesting marker for MC datasets. Interestingly, the decision tree approach identified the absence of (Src*, −) as distinguish MC datasets that were previously characterized by (JUN/FOS[n], −) and (FOXM1*[c], −). Both the machine-learning and statistical methods identified (JUN/FOS[n], −) and (FOXM1*[c], −) as important markers of MC datasets. In Fig. 2, we show (marked as yellow nodes) how these 5 main proteins or protein complexes appear connected following the PID-NCI representation.

| Predicted node | Sign | $FS^{NPC}$ | $FS^{MC}$ | p.val (Fisher) | References | Connectivity | OVE | |
|---|---|---|---|---|---|---|---|---|
| | | | | | | | + | − |
| JUN/FOS[n] | − | 0.444 | 0.956 | 2.65E-005 | 38–42 | 8/529 | 373 | 137 |
| FOXM1*[c] | − | 0.222 | 0.774 | 7.97E-004 | 43, 44 | 529/529 | 85 | 265 |
| STAT6*[c] | − | 0.222 | 0.764 | 1.05E-003 | ∅ | 8/529 | 30 | 429 |
| EGF/EGFR*[m] | + | 0.556 | 0.935 | 2.08E-003 | 45–47 | 529/529 | 79 | 4 |
| Src* | + | 0.556 | 0.935 | 2.08E-003 | 48–50 | 529/529 | 110 | 48 |

**Table 2.** 5 top-ranked results for the frequency analysis for MC signatures. $FS^{NPC}$ and $FS^{MC}$ show the frequency scores for NPC and MC, respectively. The references column lists the publications that agreed with our sign prediction. Connectivity refers to the ratio of genes connected to each predicted node. The OVE (observed variant expression) shows the repartition of variant gene expression using the best precision threshold without considering graph information.

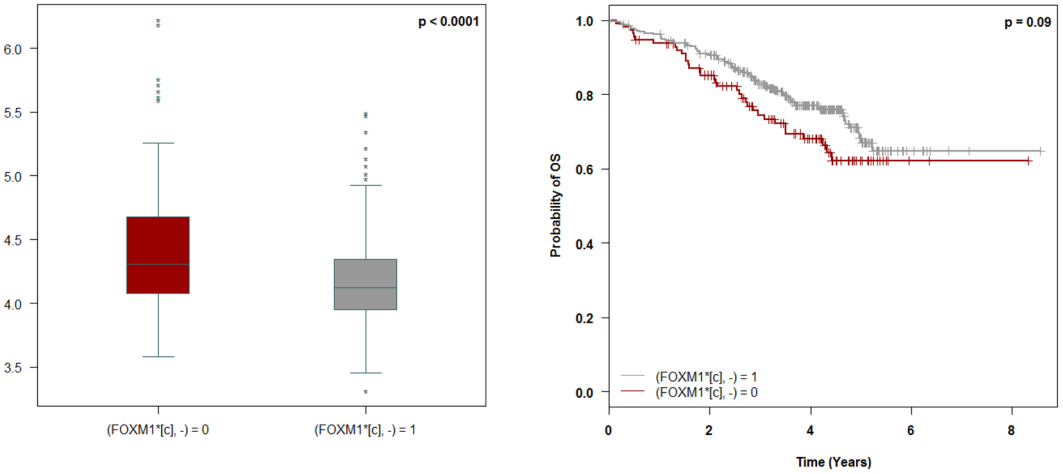



**Figure 5.** (Left) Gene expression of FOXM1 among MC datasets with or without the prediction (FOXM1*[c], −). (Right) Overall survival (OS) of patients with prediction (FOXM1* [c], −) or without prediction (FOXM1* [c], −).

**JUN/FOS activity as specific marker.** The FOS and JUN proteins form a heterodimer complex that is responsible for AP-1 activity. This activity is known to play a role in tumorigenesis because it has been implicated in the induction of apoptosis, in the promotion of cell survival and in proliferation. The classification methods showed that (JUN/FOS[n], −) is the best assignment to distinguish MC from NPC and revealed that AP-1 activity is lower in almost all MM patients than in normal controls. Inspection of individual patients' subgraphs showed predominantly underexpression (65% of the observed expression in MC) of the proapoptotic protein BIM (Supplementary Material, Figure S8). These results are in agreement with the results of *in vitro* studies demonstrating that in myeloma cell lines IL6 protects against apoptosis via AP-1 inactivation[39].

**FOXM1 activity as survival marker.** FOXM1, a transcriptional factor known to be associated with MM, has been studied as a therapeutic target[44]. Based on the graph reduction and on our reasoning model, FOXM1*[c] is equivalent to FOXM1*[n] and is representative of the FOXM1 transcriptional activity. Firstly, we analyzed *FOXM1* gene expression in the MC groups in which FOXM1 activity was predicted. We found that decreased FOXM1 activity is associated with reduced expression of the *FOXM1* gene (Fig. 5, left). Since our model identified a subgroup of patients with decreased activity of FOXM1 and since decreased expression of *FOXM1* is associated with superior survival (Supplementary Material, Figure S9), we wanted to know whether FOXM1 activity could impact survival. We compared overall survival (OS) in both predicted groups in the larger cohort of patients that received comparable treatment (Velcade-dexamethasone induction followed by high-dose melphalan and autologous stem cell transplantation; n = 450) (Fig. 5, right). A log-rank test between these groups yielded a p-value of $<0.1$, allowing us to conclude that low FOXM1 activity is associated with a trend towards better survival.

**Improvement of the current prognostic model in MM using node variables.** Univariate and multivariate Cox proportional hazards analyses were performed on the cohort of 450 MM patients who received comparable treatment to determine the relative prognostic values of the 201 couples combining unobserved nodes and all signs (+, −, 0) and the three strongest known prognostic variables in MM (Table 3); these were the translocation of chromosomes 4 and 14 (t(4;14)), the deletion in the short arm of chromosome 17 (del(17p)) and serum 2-microglobulin $\geq$5.5 mg/L ($\beta_2$-microglobulin) for OS determination[51]. In the multivariate analysis,

| Parameters | Univariate analysis | | | Multivariate analysis | | |
|---|---|---|---|---|---|---|
| | HR | 95%CI | P.value | HR | 95%CI | P.value |
| $\beta_2$-microglobulin, mg/L $\geq$5.5 *v* <5.5 | 2.03 | 1.35–3.05 | 0.001 | 1.53 | 0.99–2.35 | 0.056 |
| t(4,14), Yes *v* no | 3.19 | 2.08–4.89 | <0.01 | 2.41 | 1.49–3.90 | <0.01 |
| del17p > 60 *v* $\leq$60 | 4.16 | 2.53–6.83 | <0.01 | 3.16 | 1.80–5.56 | <0.01 |
| (G1/S transition of mitotic cell cycle, −), yes *v* no | 0.33 | 0.22–0.47 | <0.01 | 0.47 | 0.30–0.72 | <0.01 |
| (RB1/E2F1–3/DP[n], +), yes *v* no | 0.49 | 0.33–0.75 | 0.001 | 0.58 | 0.36–0.93 | 0.025 |

**Table 3.** Parameters Associated With Overall Survival.

| Node | Dir. | $TPS^{NPC}$ | $TPS^{MC}$ | p.val |
|---|---|---|---|---|
| JUN/FOS[n] | + | **22.2%** | **74.6%** | 0.001 |
| | − | 44.4% | 0.5% | 1 |
| FOXM1*[c] | + | **11.1%** | **36.4%** | 0.107 |
| | − | 55.6% | 19.1% | 0.997 |
| STAT6*[c] | + | **33.3%** | **55.0%** | 0.169 |
| | − | 44.4% | 21.9% | 0.970 |
| EGF/EGFR*[m] | + | 0.0% | 0.3% | 0.971 |
| | − | **0.0%** | **3.5%** | 0.728 |
| Src* | + | 0.0% | 1.3% | 0.887 |
| | − | **11.1%** | **33.4%** | 0.150 |

**Table 4.** Top perturbation score for nodes identified with the FS method. Dir stands for the direction of the perturbation (+, activation and −, inhibition). TPS represents the frequency with which perturbing a node in a specific direction was significant (i.e. it generated a high, 10% top, SCENFIT score) across the MC profiles ($TPS^{MC}$) or NPC profiles ($TPS^{NPC}$). The bold percentages refer to perturbations that have a direction opposite to that of the predicted signs obtained with the frequency score (Table 3). P.val was obtained using a unilateral Fisher test.

estimation of hazard ratios for death indicates that both (G1/S transition of mitotic cell cycle, −) and (RB1/E2F1-3/DP[n], +) were independent powerful prognostic factors (Supplementary Material, Figure S10).

The multivariate model with the known prognostic parameters shows that these factors increase the log-likelihood from −515.16 (null model) to −496.62 (3 parameter model), with p-significance $<10^{-7}$ (null model vs 3 parameter model) whereas the parameters (G1/S transition of mitotic cell cycle, −) and (RB1/E2F1-3/DP[n], +) increase the log-likelihood from −496.62 (3-parameter model: AIC3p = 999.2) to −486.90 (5-parameter model: AIC5p = 983.8) with p-significance $<10^{-4}$ (3-parameter model vs 5-parameter model) and $<10^{-10}$ (null model vs 5-parameter model). Therefore, we can conclude that the 5-parameter model provides more prognostic information than the 3-parameter model (AIC5p < AIC3p and p5p vs 3p $<10^{-4}$). In term of the global increase in the log-likelihood between the null model and the 5-parameter model, the specific impact of the selected pairs represents more than 34% of the total.

**Node perturbation.** From the computation of all *in silico* node perturbations (see Methods section), we evaluated the impact of perturbing the key nodes found with the FS method (Table 4). A unilateral Fisher test allowed us to evaluate the significance of each perturbation compared to the NPC datasets. We can see that the activation of JUN/FOS generates a top-ranked (10% top) score of conflicts and therefore repairs 74.6% of the MC datasets, whereas it repairs only 22.2% of the NPC datasets. Interestingly, *in vitro* JUN overexpression in MM cell lines results in cell death and growth inhibition[41]. A similar tendency (more conflicts in MC than in NPC) is observed when FOXM1 is activated, but the difference cannot be considered significant. Nonetheless, we note that of the 36.4% of profiles in which the activation of FOXM1 is top-ranked, 96.8% correspond to patient profiles with the prediction (FOXM1*[c], −) (Supplementary Material, Table S3). For the other proteins and complexes, we can see that the difference between MM and NPC is not significant. It is worth noting that the p-value of a perturbation that goes in the opposite direction of the prediction shown in Table 2 is in all cases lower than the one of a perturbation which goes in the same direction of the prediction.

## Discussion

**Data discretization and graph generation.** Our method incorporates both differential and similar expressions in its reasoning. All *pathway analysis* methods reviewed in the Introduction use the difference in gene expression between the two classes of subjects to extract the specific signatures. The similarity of expression between classes is not used in the ORA and FCS approaches because these methods base their analyses on the differential expression of genes. The PT approaches reviewed use only differential gene expression in their reasoning. We believe that adding information on similar expression enables us to better capture cellular behavior.

The results of the precision analysis using a two-sign coloring model tend to support this hypothesis. Our method differs from classic pathway analysis methods in that it incorporates the notion of automatic reasoning. Within the context of MM, we are able to automatically detect repairs. These repairs are specific for each GEP and could represent cancer mutations, regulatory network incompleteness or experimental errors. The graph used in this study contains 529 genes; we can therefore observe the strong connectivity that exists among PID pathways since the total number of genes in the PID is 634. This strong connectivity is important for methods such as ours that are able to reason on the information content of the whole database. We observe, however, that the number of genes connected to cancer pathways in PID is far below the total number of human genes. This underrepresentation of regulatory knowledge is an important limitation of PID. Apart from this fact, PID-NCI includes important modeling information that identifies *transcription events*. This information allows us to separate gene expression from protein activity. These two parameters are not necessarily correlated, especially in cases involving phosphorylated proteins or complexes such as JUN/FOS.

**Key nodes identification.** Our analysis of the predictions made by the method allowed us to identify nodes associated with a sign specific to MC compared to NPC datasets. Among these assignments, we found the inhibition of JUN/FOS[n] and FOXM1*[c]. These proteins are known to be involved in cancer in general[52,53] and in hematological malignancies in particular[43,44]. In the case of FOXM1, we showed that this transcription factor can represent a survival marker when its activity decreases. We can draw a parallel with the bibliography, which identifies the activation of the FOXM1 pathway as a risk factor. For JUN/FOS, our analysis identified this pathway as a potential therapeutic target but not as a survival marker. We observed that inhibition of the associated pathways has been already identified in MM patients[39,41] and in patients with other cancers. Moreover, this pathway is targeted in some therapeutic approaches[38,40]. We identified two couples that improve classical prognostic models. In the case of the first couple, (G1/S transition of the mitotic cell cycle, −), we can associate this node with the proliferation pathway. The computed prognostic model showed that the prediction of inhibited proliferation can be a protective factor for MM patients. The second node, (RB1/E2F1-3/DP[n], +), was also identified as a protective factor by the 5-parameter model. This complex is known to be involved in the RB pathway, which influences cell growth pathways by regulating the initiation of DNA replication. This pathway is usually altered in cancer, leading to a loss of function[54], and current therapeutic approaches aim to activate this pathway[55].

Using the *in silico* node perturbation method, we were able to estimate the effect of perturbing a node within a particular dataset (*i.e.* single patient cancer cell). This method represents a powerful tool for analyzing the consequences of perturbations of oncogenic pathways in a given patient, especially as *in-vitro* experiments are limited due to the small amount of viable myeloma cells that are obtained after bone marrow aspiration. The results of this *in silico* analysis show that activation of JUN/FOS[n] had a significant impact on 75% of MC profiles; all of these JUN/FOS[n] = "+" sensible MC profiles had the prediction (JUN/FOS[n], −). In addition, activating FOXM1*[c] had a significant impact on 36.4% of the profiles; 96.8% of the FOXM1*[c] = "+" sensible MC profiles had the prediction (FOXM1*[c], −). The difference in the percentages of JUN/FOS[n] and FOXM1*[c] can be explained by the graph topology and the connectivity of the individual nodes. JUN/FOS[n] is connected to eight genes through a distance of 1 molecular species (Supplementary Material, Figure S8); therefore, perturbing JUN/FOS[n] will impact these genes directly since they are strongly constrained by the sign of JUN/FOS[n]. On the other hand, FOXM1 is connected to 529 genes through longer paths through distances of from 4 to 77 molecular species. These genes may have other predecessors that are independent of FOXM1; this could explain why activation of FOXM1 has a strong effect on only 37% of the MC profiles. Overall, we think that this *in silico* method could be used to reinforce the choice of a therapeutic target for a specific patient profile.

## Conclusion

In this study, we used a specific approach to study and understand the heterogeneous gene expression profiles of approximately 600 multiple myeloma (MM) patients. Our primary goal was to provide mechanistic scenarios by identifying protein activity states of molecules that may be central to the diversity of gene expression. Our approach relies heavily on reasoning based on graphs and on changes in gene expression in the form of logical programs that combine these two types of information. The method proposed here can be summarized in the following steps. First, we obtained a directed graph, allowing us to connect significantly up-/down-regulated genes to upstream MM-related cellular receptors. Second, we confronted this graph to transcriptomic data with IGGY, which is a tool that reasons on the logic of the graph and on shifts of expression in the data so as to predict (*node*, *sign*) assignments representing the specific states of biological entities. Using two approaches of classification, we were able to identify specific assignments for MC datasets compared to NPC datasets. Finally, taking advantage of our modeling framework, we studied the effect of performing single *in silico* perturbations.

One advantage of this method is that it makes it possible to infer information about protein states from transcriptomic data by using the causal nature of the interactions as documented in PID. This can be interesting when constructing biological models and, more specifically, when developing cancer models for which proteomic data are not always available and extractable, whereas transcriptomic data are easier to obtain. Moreover, compared to the previously presented classical pathway analysis methods, we identify not only the specific biological processes that are implicated in cancer profiles but also the mechanisms associated with those phenomena. After statistically testing the quality of the method's predictions, we proposed a set of five top-scoring proteins based on their respective changes in activity in MC compared with NPC. We found the AP-1 complex and the FOXM1 transcription factor to be concomitantly inactivated in a strong majority of patients regardless of treatment or age. Interestingly, this method identified a subgroup of MM patients with increased FOXM1 activity associated with poor survival. These findings allow us to validate the predictions of our approach and show that it is feasible to individualize or restrict the analysis of multiple expression profiles to identify markers within subgroups of

profiles and to identify parameters associated with survival in these subgroups. The 5-parameter model including the two predicted nodes improves the standard prognostic model in MM. In addition to its strong prognostic value, our model revealed two nodes, (G1/S transition of mitotic cell cycle, −) and (RB1/E2F1-3/DP[n], +), that are of potential biological interest in the understanding of the molecular mechanisms underlying resistance to treatment. Note that these nodes can only be predicted with the graph and coloring model, since they are a (logical) consequence of the GEP. Our results on *in silico* perturbations of a system are also encouraging because they show that changes in the activity of the predicted proteins can serve as input information for conducting efficient perturbations. In this work, we focused only on single perturbations, since they are more experimentally realistic. As a perspective of this work, we wish to deepen the graph vs. gene-expression confrontation analysis so as to understand the differences between MM subgroups based on age, prognosis and other criteria. In this context, one line of research would be to study minimal subsets of perturbations. Another possible line of research would be the classification of gene expression profiles based on plausible graph-coloring models.

## References


1. Morgan, G. J., Walker, B. A. & Davies, F. E. The genetic architecture of multiple myeloma. *Nature reviews. Cancer* **12**, 335–48 (2012).
2. Zhan, F. *et al*. The molecular classification of multiple myeloma. *Blood* **108**, 2020–2028 (2006).
3. Decaux, O. *et al*. Prediction of survival in multiple myeloma based on gene expression profiles reveals cell cycle and chromosomal instability signatures in high-risk patients and hyperdiploid signatures in low-risk patients: A study of the Intergroupe Francophone du Myélome. *Journal of Clinical Oncology* **26**, 4798–4805 (2008).
4. Shaughnessy, J. D. *et al*. A validated gene expression model of high-risk multiple myeloma is defined by deregulated expression of genes mapping to chromosome 1. *Blood* **109**, 2276–84 (2007).
5. Avet-Loiseau, H. *et al*. Prognostic significance of copy-number alterations in multiple myeloma. *Journal of clinical oncology: official journal of the American Society of Clinical Oncology* **27**, 4585–90 (2009).
6. Broyl, A. *et al*. Gene expression profiling for molecular classification of multiple myeloma in newly diagnosed patients. *Blood* **116**, 2543–53 (2010).
7. Walker, B. A. *et al*. Mutational Spectrum, Copy Number Changes, and Outcome: Results of a Sequencing Study of Patients With Newly Diagnosed Myeloma. *Journal of clinical oncology: official journal of the American Society of Clinical Oncology* **33**, 3911–20 (2015).
8. Rashid, N. U. *et al*. Differential and limited expression of mutant alleles in multiple myeloma. *Blood* **124**, 3110–7 (2014).
9. Mansour, M. R. *et al*. Oncogene regulation. An oncogenic super-enhancer formed through somatic mutation of a noncoding intergenic element. *Science (New York, NY)* **346**, 1373–7 (2014).
10. The Gene Ontology Consortium. Gene Ontology: tool for the unification of biology. *Nature Genetics* **25**, 25–29 (2000).
11. Kanehisa, M. & Goto, S. KEGG: kyoto encyclopedia of genes and genomes. *Nucleic acids research* **28**, 27–30 (2000).
12. Schaefer, C. F. *et al*. PID: the Pathway Interaction Database. *Nucleic acids research* **37**, D674–9 (2009).
13. Kelder, T. *et al*. WikiPathways: building research communities on biological pathways. *Nucleic acids research* **40**, D1301–7 (2012).
14. Wingender, E. The TRANSFAC project as an example of framework technology that supports the analysis of genomic regulation. *Briefings in Bioinformatics* **9**, 326–332 (2008).
15. Boué, S. *et al*. Causal biological network database: a comprehensive platform of causal biological network models focused on the pulmonary and vascular systems. *Database: the journal of biological databases and curation* **2015**, bav030 (2015).
16. Khatri, P., Sirota, M. & Butte, A. J. Ten years of pathway analysis: current approaches and outstanding challenges. *PLoS computational biology* **8**, e1002375 (2012).
17. Bindea, G. *et al*. ClueGO: a Cytoscape plug-in to decipher functionally grouped gene ontology and pathway annotation networks. *Bioinformatics (Oxford, England)* **25**, 1091–3 (2009).
18. Catlett, N. L. *et al*. Reverse causal reasoning: applying qualitative causal knowledge to the interpretation of high-throughput data. *BMC bioinformatics* **14**, 340 (2013).
19. Martin, F. *et al*. Quantification of biological network perturbations for mechanistic insight and diagnostics using two-layer causal models. *BMC bioinformatics* **15**, 238 (2014).
20. Subramanian, A. *et al*. Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America* **102**, 15545–50 (2005).
21. Backes, C. *et al*. GeneTrail–advanced gene set enrichment analysis. *Nucleic acids research* **35**, W186–92 (2007).
22. Lefebvre, C. *et al*. A human B-cell interactome identifies MYB and FOXM1 as master regulators of proliferation in germinal centers. *Molecular systems biology* **6**, 377 (2010).
23. Kong, S. W., Pu, W. T. & Park, P. J. A multivariate approach for integrating genome-wide expression data and biological knowledge. *Bioinformatics (Oxford, England)* **22**, 2373–80 (2006).
24. Ideker, T., Ozier, O., Schwikowski, B. & Siegel, A. F. Discovering regulatory and signalling circuits in molecular interaction networks. *Bioinformatics (Oxford, England)* **18**(Suppl 1), S233–40 (2002).
25. Komurov, K., Dursun, S., Erdin, S. & Ram, P. T. NetWalker: a contextual network analysis tool for functional genomics. *BMC genomics* **13**, 282 (2012).
26. Liu, W. *et al*. Topologically inferring risk-active pathways toward precise cancer classification by directed random walk. *Bioinformatics (Oxford, England)* **29**, 2169–77 (2013).
27. Yaveroğlu, Ö. N., Milenković, T. & Pržulj, N. Proper evaluation of alignment-free network comparison methods. *Bioinformatics* **31**, 2697–2704 (2015).
28. Draghici, S. *et al*. A systems biology approach for pathway level analysis. *Genome research* **17**, 1537–45 (2007).
29. S, T. *et al*. Extended notions of sign consistency to relate experimental data to signaling and regulatory network topologies. *BMC Bioinformatics* **16**, 345 (2015).
30. Avet-Loiseau, H. *et al*. Genetic abnormalities and survival in multiple myeloma: the experience of the Intergroupe Francophone du Myélome. *Blood* **109**, 3489–95 (2007).
31. Klein, B. Positioning NK-kappaB in multiple myeloma. *Blood* **115**, 3422–4 (2010).
32. Saez-Rodriguez, J. *et al*. Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction. *Molecular systems biology* **5**, 331 (2009).
33. Quinlan, J. Simplifying decision trees. *International Journal of Man-Machine Studies* **27**, 221–234 (1987).
34. Breiman, L. Random Forests. *Machine Learning* **45**, 5–32 (2001).
35. Baral, C. *Knowledge Representation, Reasoning and Declarative Problem Solving* (Cambridge University Press, 2003).
36. Hagberg, A. A., Schult, D. A. & Swart, P. J. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, 11–15 (2008).
37. R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria (2015).

38. Podar, K. *et al*. Up-regulation of c-Jun inhibits proliferation and induces apoptosis via caspase-triggered c-Abl cleavage in human multiple myeloma. *Cancer research* **67**, 1680–8 (2007).
39. Xu, F. H. *et al*. Interleukin-6-induced inhibition of multiple myeloma cell apoptosis: support for the hypothesis that protection is mediated via inhibition of the JNK/SAPK pathway. *Blood* **92**, 241–251 (1998).
40. Saha, M. N. *et al*. Targeting p53 via JNK pathway: a novel role of RITA for apoptotic signaling in multiple myeloma. *PloS one* **7**, e30215 (2012).
41. Chen, L. *et al*. Identification of early growth response protein 1 (EGR-1) as a novel target for JUN-induced apoptosis in multiple myeloma. *Blood* **115**, 61–70 (2010).
42. Fan, F. *et al*. Targeting Mcl-1 for multiple myeloma (MM) therapy: drug-induced generation of Mcl-1 fragment Mcl-1(128-350) triggers MM cell death via c-Jun upregulation. *Cancer letters* **343**, 286–94 (2014).
43. Uddin, S. *et al*. Overexpression of FoxM1 offers a promising therapeutic target in diffuse large B-cell lymphoma. *Haematologica* **97**, 1092–100 (2012).
44. Gu, C. *et al*. FOXM1 is a therapeutic target for high-risk multiple myeloma. *Leukemia* **30**, 873–882 (2016).
45. Mahtouk, K. *et al*. An inhibitor of the EGF receptor family blocks myeloma cell growth factor activity of HB-EGF and potentiates dexamethasone or anti-IL-6 antibody-induced apoptosis. *Blood* **103**, 1829–37 (2004).
46. Mahtouk, K. *et al*. Expression of EGF-family receptors and amphiregulin in multiple myeloma. Amphiregulin is a growth factor for myeloma cells. *Oncogene* **24**, 3512–3524 (2005).
47. Johnston, J. B. *et al*. Targeting the EGFR pathway for cancer therapy. *Current medicinal chemistry* **13**, 3483–3492 (2006).
48. Hallek, M. *et al*. Signal transduction of interleukin-6 involves tyrosine phosphorylation of multiple cytosolic proteins and activation of Src-family kinases Fyn, Hck, and Lyn in multiple myeloma cell lines. *Experimental hematology* **25**, 1367–77 (1997).
49. Coluccia, A. M. L. *et al*. Validation of PDGFRbeta and c-Src tyrosine kinases as tumor/vessel targets in patients with multiple myeloma: preclinical efficacy of the novel, orally available inhibitor dasatinib. *Blood* **112**, 1346–56 (2008).
50. Ishikawa, H. Requirements of src family kinase activity associated with CD45 for myeloma cell proliferation by interleukin-6. *Blood* **99**, 2172–2178 (2002).
51. Avet-Loiseau, H. *et al*. Combining fluorescent *in situ* hybridization data with ISS staging improves risk assessment in myeloma: an International Myeloma Working Group collaborative project. *Leukemia* **27**, 711–717 (2013).
52. Eferl, R. & Wagner, E. F. AP-1: a double-edged sword in tumorigenesis. *Nature reviews. Cancer* **3**, 859–68 (2003).
53. Shaulian, E. & Karin, M. AP-1 as a regulator of cell life and death. *Nature Cell Biology* **4**, E131–E136 (2002).
54. Nevins, J. R. The Rb/E2F pathway and cancer. *Human molecular genetics* **10**, 699–703 (2001).
55. Knudsen, E. S. & Wang, J. Y. J. Targeting the RB-pathway in cancer therapy. *Clinical cancer research: an official journal of the American Association for Cancer Research* **16**, 1094–9 (2010).


## Acknowledgements

This study was supported by Intergroupe Francophone du Myélome and by a French Institute National du Cancer Grant EVACAMM PROG/09/10 (to H.A.L., S.M.), a National Institutes of Health Grant PO1CA155258-01 (to S.M., H.A.L., N.C.M.), and a research grant from Celgene. B.M.'s PhD scholarship was funded by GRIOTE project. We would like to thank Elise Douillard, Magali Devic, Emilie Morenton and Nathalie Roi for excellent technical assistance. We thank Jérémie Bourdon, Nathalie Theret and Sophia Tsoka for suggestions and critical reading of the manuscript. We are most grateful to the bioinformatics core facility of Nantes (BiRD - Biogenouest) for its technical support.

## Author Contributions

B.M. implemented the predictions analysis and the perturbations methods, performed the computational analysis and wrote the paper. B.M., S.M., F.M. and C.G. conceived and supervised the study and drafted the manuscript. B.M., S.M., O.R., F.M. and C.G. discussed the results of the data analysis. P.D. performed the k-mean discretization method and its comparison and implemented the predictions validation method. W.G. performed the microarray analysis. L.C. performed the survival models comparison with and without predictions. C.G.C. performed the statistical analysis. H.A.L., M.A., T.F., N.C.M., and P.M. provided samples and clinical data. All authors read and approved the final manuscript.

## Additional Information

**Supplementary information** accompanies this paper at doi:10.1038/s41598-017-09378-9

**Competing Interests:** The authors declare that they have no competing interests.

**Accession codes** Minimum Information About a Microarray Experiment-compliant data has been deposited at: Gene Expression Omnibus with accession number GSE83503.

**Publisher's note:** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

## A.3 Learning Boolean Networks

BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 18 2013, pages 2320–2326
doi:10.1093/bioinformatics/btt393

Systems biology

Advance Access publication July 12, 2013

# Exhaustively characterizing feasible logic models of a signaling network using Answer Set Programming

Carito Guziolowski[1,†], Santiago Videla[2,3,4,†], Federica Eduati[5], Sven Thiele[2,3], Thomas Cokelaer[5], Anne Siegel[2,3,*] and Julio Saez-Rodriguez[5,*]

[1]École Centrale de Nantes, IRCCyN UMR CNRS 6597, 44321, Nantes, France, [2]CNRS, UMR 6074 IRISA, Campus de Beaulieu, 35042 Rennes, France, [3]INRIA, Dyliss project, Campus de Beaulieu, 35042 Rennes, France, [4]Universität Potsdam, Institut für Informatik, D-14482 Potsdam, Germany and [5]European Molecular Biology Laboratory, European Bioinformatics Institute, Hinxton CB10 1SD, UK

Associate Editor: Martin Bishop

## ABSTRACT

**Motivation:** Logic modeling is a useful tool to study signal transduction across multiple pathways. Logic models can be generated by training a network containing the prior knowledge to phospho-proteomics data. The training can be performed using stochastic optimization procedures, but these are unable to guarantee a global optima or to report the complete family of feasible models. This, however, is essential to provide precise insight in the mechanisms underlaying signal transduction and generate reliable predictions.

**Results:** We propose the use of Answer Set Programming to explore exhaustively the space of feasible logic models. Toward this end, we have developed *caspo*, an open-source Python package that provides a powerful platform to learn and characterize logic models by leveraging the rich modeling language and solving technologies of Answer Set Programming. We illustrate the usefulness of *caspo* by revisiting a model of pro-growth and inflammatory pathways in liver cells. We show that, if experimental error is taken into account, there are thousands (11 700) of models compatible with the data. Despite the large number, we can extract structural features from the models, such as links that are always (or never) present or modules that appear in a mutual exclusive fashion. To further characterize this family of models, we investigate the input–output behavior of the models. We find 91 behaviors across the 11 700 models and we suggest new experiments to discriminate among them. Our results underscore the importance of characterizing in a global and exhaustive manner the family of feasible models, with important implications for experimental design.

**Availability:** *caspo* is freely available for download (license GPLv3) and as a web service at http://caspo.genouest.org/.

**Supplementary information:** Supplementary materials are available at *Bioinformatics* online.

**Contact:** anne.siegel@irisa.fr or saezrodriguez@ebi.ac.uk

Received on March 20, 2013; revised on June 17, 2013; accepted on July 4, 2013

## 1 INTRODUCTION

Predictive models of biological networks are a main component of systems biology. For a certain system of interest, if enough information is available about the biomolecules that constitute it and their interactions, one can convert this prior knowledge into a mathematical model (e.g. a set of differential equations or logic rules) that can be simulated. If experimental data are available, the model can be fitted (trained) to the data. That is, one determines the model parameters (for example, kinetic constants in a biochemical model) to obtain the most plausible model given the data. This is normally achieved by defining an objective function that describes the goodness of the model based on the data that is subsequently optimized (Banga, 2008).

This training process is not a trivial task owing to factors including experimental error, limitations in the amount of data available, incompleteness of our prior knowledge and inherent mathematical properties of the models. Thus, in general, there is no single solution but rather multiple models that describe the data equally (or similarly) well. In those cases, the model is said to be non-identifiable (Kreutz and Timmer, 2009; Walter and Pronzato, 1996).

In some cases, deterministic methods that guarantee the identification of the optimal models can be applied, but these methods are often limited by the exponential growth of the search space. Thus, usually one needs to use stochastic methods that may identify the optimum or at least exhibit suboptimal models (Banga, 2008). However, an incomplete characterization of the set of plausible models limits significantly the insight that can be gained about the underlying molecular mechanisms.

In this article, we investigate this issue in the context of logic modeling of signaling networks. These models have been applied recently to analyze signal transduction in a variety of contexts (Calzone *et al.*, 2010; Wang *et al.*, 2012). In particular, given a network encoding our knowledge of signal transduction and a dataset measuring the activation of proteins in this network on various perturbations, one can derive from the network (Boolean) logic models fitted to the data. Models are simulated assuming that the network reaches a pseudo steady state at a certain time on stimulation, and the identification of the network that best fits the data is posed as an optimization problem. This problem can be solved using meta-heuristics (e.g. a genetic algorithm), and their application suggests that there are multiple alternative models that explain the data (Saez-Rodriguez *et al.*, 2009). However, stochastic search methods cannot characterize the models precisely: they are intrinsically unable not just to

*To whom correspondence should be addressed.
†The authors wish it to be known that, in their opinion, the first two authors should be regarded as joint First Authors.

provide a complete set of solutions, but also to guarantee that an optimal solution is found. To overcome this limitation, approaches based on Integer Linear Programming (ILP) (Mitsos *et al*., 2009; Sharan and Karp, 2012) and Answer Set Programming (ASP) (Videla *et al*., 2012) have been applied, providing a proof of concept that a global optimum can be identified.

Here we present *caspo*, a free open-source tool to learn (Boolean) logic models of signal transduction in a complete and global fashion. *caspo* uses CellNOpt pre- and post-processing routines [Terfve *et al*. (2012)]. It can handle feedback loops in the prior knowledge network (PKN), numerical datasets and tolerance in the score owing to experimental uncertainty. We use *caspo* to exhaustively explore the space of optimal and suboptimal models for a real case describing pro-growth and inflammatory pathways in a liver cancer cell. We find that, even with small tolerance, thousands of models can be compatible with the data and use ASP's flexibility to further analyze them: we categorize them according to their input–output behavior and identify subsets of modules that are interchangeable with respect to the score. The multiple possible combinations of these modules are responsible for the large number of models found.

## 2 METHODS

### 2.1 Learning Boolean logic models

Our prior knowledge about signal transduction can be described as a set of causal interactions among the biomolecules involved (mostly proteins) that can be mathematically formulated as a signed and directed graph. We call this graph the PKN. In such a graph, one can denote as *input* nodes those that can be stimulated or inhibited experimentally. When the system is perturbed by fixing the state of such nodes, one can measure the activity of each *output* node being observed. Such measurements are typically given by *phospho-proteomics datasets* consisting of measurements over $m$ proteins under $n$ experimental conditions. With $\theta_{ij} \in [0,1]$, we denote the activity of a protein $j$ under the experimental condition $i$, where $0 \leq i \leq n$ and $0 \leq j \leq m$. In agreement with experimental errors, we used a discretization procedure so that $\theta_{ij} \in \{0, \frac{1}{100}, \ldots, \frac{99}{100}, 1\}$.

The state of nodes after a perturbation of the system cannot be predicted using only graph theory. However, a simple framework is given by Boolean logic models (Klamt *et al*., 2006). In a logic model, activation of nodes is defined by a set of operators. We use the representation known as sum of products (SOP; also called disjunctive normal form), which uses only AND ($\wedge$), OR ($\vee$) and NOT ($\neg$) operators. A simple form to encode logic models based on the SOP formalism is using hypergraphs (Klamt *et al*., 2006). A directed and signed *hypergraph* $H = (V, E)$ is a generalization of a directed and signed graph $G = (V, A)$, where $V$ is the set of nodes and $E$ the set of *hyperedges*. While edges in $A$ connect pairs of nodes $a, b \in V$, *hyperedges* in $E$ connect pairs of *sets of nodes* $S, T \subseteq V$. To describe a logic model as a hypergraph, each SOP expression is mapped to a set of hyperedges.

The PKN is first compressed to simplify the structure (Saez-Rodriguez *et al*., 2009). Then, because the exact logic gates are often not known, we perform an *expansion* to generate all possible gates compatible with the PKN. Mathematically, we derive a hypergraph $H = (V, E)$ from a graph $G = (V, A)$, so that for every signed hyperedge $(S, \{t\}) \in E$ and every $s \in S$, there exists an edge $(s, t) \in A$ having the corresponding sign.

Let $H$ be a hypergraph describing a logic model and $(\theta_{ij})_{i \leq n, j \leq m}$ be a phospho-proteomics dataset. For each experimental condition $i$, we can compute the Boolean prediction $\rho_{ij} \in \{0, 1\}$ of the state of a protein $j$ by using the logic formulas described by $H$. This corresponds to computing the (quasi) steady state of the system. These simulated values at a quasi steady state are considered an approximation of the state of the cell immediately after a perturbation and can be thus compared with experimental values obtained at early times after stimulation (Klamt *et al*., 2006).

Then, the *fitness of the logic model* to the experimental dataset is obtained by comparing experimental observations, normalized between 0 and 1, with Boolean predictions based on the mean square error (MSE) as follows: $\frac{1}{mn} \sum_{i=1}^{n} \sum_{j=1}^{m} (\rho_{ij} - \theta_{ij})^2$.

*Combinatorial optimization problem*. The problem of learning Boolean logic models that we address in this work consists of finding minimal hypergraphs derived from the PKN that minimize the MSE where the size of a hypergraph $H$ is the sum of cardinalities of each hyperedge source (i.e. the sum of the number of inputs): $\sum_{(S,T) \in E} |S|$. Thus, the problem can be formulated as a lexicographic multi-objective optimization where the first objective is to minimize MSE, and the second objective is to minimize size. Our prior assumption that $\theta_{ij}$ belongs to a finite set of values implies that this problem is of discrete nature. Further, the optimization can be relaxed by using different degrees of tolerance over the optimum for each objective, i.e. MSE and size.

*Global Truth Tables*. Inspired by truth tables in propositional logics, we introduce the concept of Global Truth Tables (GTTs) as a way of describing the input–output behavior of a Boolean logic model. For a given logic model, we can compute its predictions on observable *output* nodes in response to every possible experimental condition on *input* nodes. Comparing GTTs allows one to decide whether two logic models, regardless of their structures, are experimentally distinguishable. Furthermore, GTTs provide a way of grouping a large number of logic models according to their input–output behavior to facilitate the analysis.

### 2.2 Learning Boolean logic models with ASP

ASP is a declarative problem-solving paradigm from the field of Logic Programming combining several computer science areas (Baral, 2003; Gebser *et al*., 2013). As a full declarative paradigm, instead of telling a computer *how to solve the problem*, with ASP one defines *what the problem is* and leaves its solution to the solver. These solvers are based on Boolean constraint solving technology, and they can solve hard discrete combinatorial search problems, with comparable results with ILP.

The distinct feature of ASP is its rich modeling language, making it popular as a tool for declarative problem solving. Sophisticated pre-processing techniques (*grounding*) are required for dealing with this rich language. Thanks to the development of an ASP language standard, its expressiveness and powerful solvers, ASP has been widely used in many fields of computer science for a decade. Recently, the capability of solvers has increased such that ASP started to be applied to solve hard combinatorial problems arising in bioinformatics and systems biology. Applications include expanding metabolic networks (Schaub and Thiele, 2009), repairing inconsistencies in gene regulatory networks (Gebser *et al*., 2010), modeling the dynamics of regulatory networks (Fayruzov *et al*., 2009), inferring functional dependencies from time-series data, (Durzinsky *et al*., 2011), integrating gene expression with pathway information (Papatheodorou *et al*., 2012) and analyzing the dynamics of reactions networks (Ray and Soh, 2012).

We used the freely available ASP grounder *gringo* and solver *clasp*, both included in the Potsdam Answer Set Solving Collection (http://potassco.sourceforge.net/). Importantly, we relied on the capability of the solvers to handle multi-criteria optimization to guarantee the global optimum by reasoning over the complete space of solutions. Several reasoning modes (enumeration, union and intersection) were also necessary to complete the combinatorial study of the family of feasible solutions. We refer the reader to the Supplementary Material for more details.

### 2.3 Software: caspo

We have implemented *caspo*: *Cell ASP Optimizer*, a Python package that combines PyASP (http://pypi.python.org/pypi/pyasp) and CellNOpt (http://www.cellnopt.org/) to provide an easy -to-use software for learning Boolean logic models (Fig. 1). The software is freely available for download and also as a web service through the Mobyle framework (Néron *et al*., 2009). PyASP encapsulates the main ASP tools, *gringo* and *clasp*, into Python objects. These objects can be fed with logic programs describing different tasks, be launched with dedicated parameter settings and return the ASP results for further processing. CellNOpt [Terfve *et al*. (2012)] is a software for training logic models using different formalisms (Boolean, Fuzzy or Ordinary Differential Equations). The software allows us to import and pre-process a PKN, normalize experimental data, train logic models to data using heuristic methods and post-process and visualize the resulting models. CellNOpt is written as a set of R packages available on Bioconductor and as a Cytoscape plugin (CytoCopter), and it can used within Python using the package *cellnopt.wrapper*.

## 3 RESULTS

To illustrate the use of *caspo*, we use a model of pro-growth and pro-inflammatory model in liver cells. The model is trained to phospho-proteomics data generated in the liver cancer cell line HepG2. Data are generated on perturbation with combination of ligands and small-molecule inhibitors blocking the activities of specific kinases (Alexopoulos *et al*., 2010). The dataset contains measurements using the Luminex technology of 15 species under 64 experimental conditions. This model was introduced in (Saez-Rodriguez *et al*., 2009) and here we use a variation from (Morris *et al*., 2011). In this case, there are 130 possible hyperedges and thus, the number of possible logic models (i.e. search space of the combinatorial optimization) is given by $2^{130}$.

### 3.1 Family of optimal models

We first used *caspo* to compute all global optimum solutions to the optimization over our case study. We found 16 Boolean logic models (Supplementary Fig. S1) with minimal score (0.36 s), all models having the same fitness to data (MSE = 0.0499) and size (28). Moreover, the same 16 logic models were found (0.5 s) using an extended PKN with feedback loops from Terfve *et al*. (2012). Cross validation analysis showed no significant difference in the optimum MSE with respect to the complete dataset (Supplementary Fig. S2).

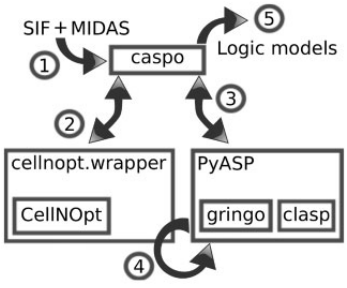



**Fig. 1.** High-level design of caspo. (1) Input files are a PKN in Cytoscape's SIF format, and a dataset as a CSV file in the MIDAS format (Supplementary Material). (2) Pre-processing routines by CellNOpt. (3) Finds an optimum model. (4) Finds all models within the tolerance. (5) Outputs all models found

The 16 different models arise owing to four pairs of submodels (modules) equivalent in terms of score. These modules represent alternative ways to activate specific nodes and are independent from each other. For each pair, only one of the modules appears in a given model; that is, they are *mutually exclusive*. Thus, selecting either member of each pair provides an optimal model and all possible combinations give rise to the $2^4 = 16$ models. To elucidate the differences between the 16 models from their responses to all possible experimental conditions, we computed and compared their GTTs (Section 2.1). Interestingly, they all have the same GTT. That is, for any combination of input nodes (stimuli and inhibitors), the same values are predicted for all the readouts by the 16 models. Therefore, the optimization reports a single solution in terms of input–output behavior, despite the fact that this solution can take the form of any of the 16 models. To distinguish among these models (and thus determine which of the mutually exclusive modules are functional), we would require a different experimental setup, i.e. new species have to be either perturbed or measured.

### 3.2 Suboptimal Models: Enumeration and Structure

Experimental error is inherent in biochemical data. Therefore, one needs to consider models whose predictions deviate from those of the optimal one by an amount within the experimental error (Saez-Rodriguez *et al*., 2009). Considering that the optimization minimize MSE and size, we defined as *suboptimal models* those solutions having MSE within a 10% of tolerance with respect to the MSE of optimal models (a conservative approximation to the real experimental error), and maximal size of 28 (the size of the optimal models; Section 3.1). From these settings, *caspo* found 11 700 suboptimal models (Fig. 2) with sizes 28, 27, 26 and 25 whose MSE spanned from 0.0499 to 0.0546. We observed that the number of models decreases exponentially with the tolerance over the MSE (e.g. 8%—7378 models, 6%—6048 models, 2%—192 models). Allowing also a tolerance over the size would generate a much larger number of models by the addition of spurious links to those of size 28 (e.g. size 29–51 480 models, size 30–189 364 models). We therefore limited, for simplicity of this study, the size to 28.

The complete computation of suboptimal models allows a precise characterization of the distribution of hyperedges, and, therefore, of logical gates in the potential models. When we evaluated the distribution of the 130 possible hyperedges (i.e. those that are included in the hypergraph derived from the original PKN) across the 11 700 models, we found that 14 hyperedges are *present in all* suboptimal models, and we thus expect them to be functional in HepG2 cells. Fifty-nine hyperedges are *absent from all models*, thus suggesting that they are not functional in these cells. Finally, 57 hyperedges are present in only a subset of the models; their frequency ranges from 0.99 to 0.0003, showing a large variability (Fig. 3). Therefore, for the given experimental data, these hyperedges are not identifiable, as it is not possible to determine whether they are functional in HepG2 cells.

Analogously to the set of optimal models, we investigated the combinatorics within the family of suboptimal models. We found four mutually exclusive pairs of modules (Fig. 2B). Replacing a

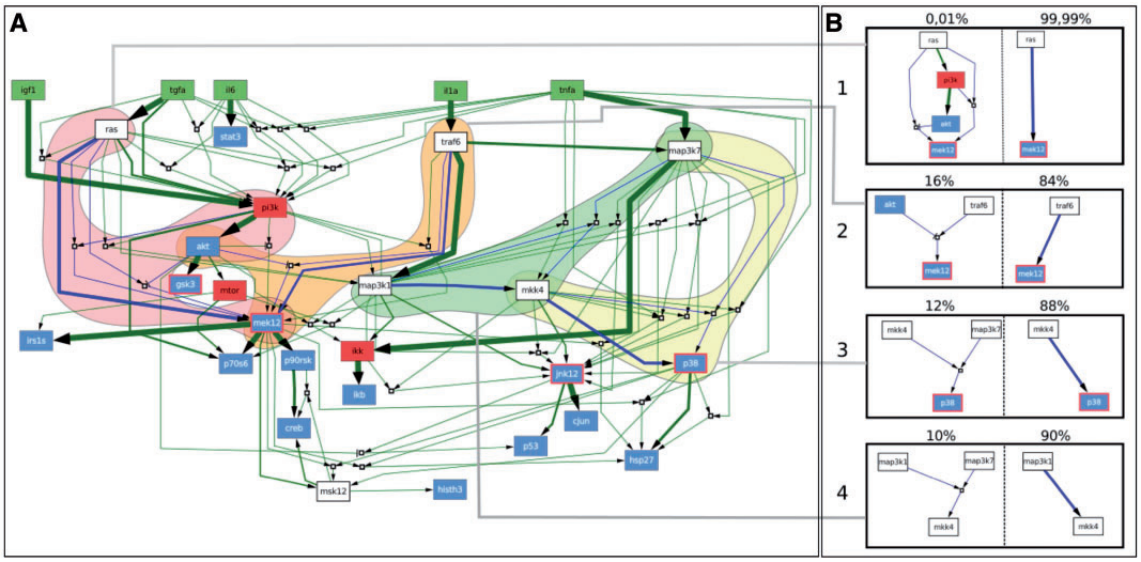



**Fig. 2.** Suboptimal models generated with *caspo* with 10% error tolerance. (**A**) Network of the union of 11 700 suboptimal models. Green nodes represent ligands that are experimentally stimulated. Red (or red-bordered) nodes represent those species that are inhibited with a small molecule inhibitor (drug). Blue nodes represent species that were measured using the Luminex technology. White nodes are neither measured nor perturbed. AND gates in the models are represented by empty boxes. The thickness of the hyperedges correspond to their frequencies among the 11 700 submodels. (**B**) Four pairs of mutually exclusive modules (blue hyperedges in A) and their corresponding frequencies on top. These modules determine the behavior of three nodes in the network: mek12, mkk4 and p38

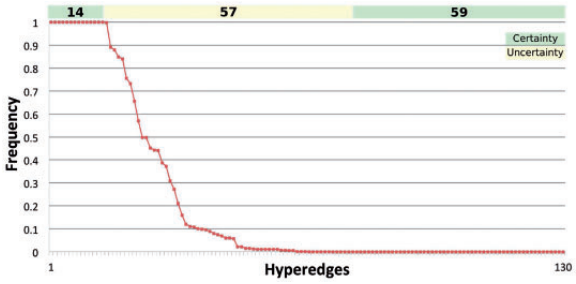



**Fig. 3.** Frequencies of hyperedges over 11 700 suboptimal models within 10% tolerance. Among the 130 possible hyperedges, 14 were always present, 59 were always absent and 57 were present in some but not all models

module of each pair by the other has no effect on the MSE for two of the pairs (1, 2 in Fig. 2B). However, for the pairs 3 and 4 there is a difference; 32 and 26.8%, respectively, of the suboptimal models differ in the output for a range from 8 to 15% of the experimental conditions. All modules were constituted by a single hyperedge, except 1A, which is set by two hyperedges: $\{(ras \wedge \neg akt \rightarrow mek), (ras \wedge pi3k \rightarrow mek)\}$ (Fig. 2, module 1A). These two hyperedges were therefore always either both present or both absent (*mutually inclusive*). As expected, there is a clear difference between the frequencies in each pair of exclusive patterns where smaller or simpler hyperedges are always more abundant. Importantly, the mutually exclusive modules for the family of suboptimal models are not the same as those present when only optimal models are considered. This indicates that the combinatorics exhibited within optimal models are not so important when considering experimental error, probably owing to the larger variability among suboptimal models.

### 3.3 Input–output behavior

To further characterize the family of suboptimal models, we next studied its input–output behavior as expressed by its GTTs. Using *caspo*, we found that the 11 700 suboptimal models correspond to 91 different GTTs. In these 91 GTTs, the predicted values are the same for 30% (4915 out of 16 384) of all the possible experimental conditions (i.e. $2^{14}$ combinations of the 14 inputs of the model). Therefore, such predictions can be seen as the 'core' predictions of the system behavior independently from experimental noise. Considering the remaining 70% of experimental conditions, we found that at least seven experiments are needed to discriminate among all GTTs (Table S4). By performing such experiments, one would be able to generate at least one different output prediction between every pair of GTTs.

Among the 11 700 suboptimal models, there are only 13 different MSEs. The distribution of such MSEs is inhomogeneous, and two MSEs (0.0519 and 0.0542) gather 71% of suboptimal models (Fig. 4). For both most frequent MSEs, a GTT is much more common than all the others: the first GTT, at MSE 0.0519, is shared by 3126 (27%) suboptimal models, while the second most common GTT, at MSE 0.0542, is shared by 2090 (18%) models. In contrast, the minimal GTT, at MSE 0.0499, was shared by only the 16 minimal models. This analysis suggests that the single optimal GTT at MSE 0.0499 is far from being representative over the 11 700 suboptimal models (0.1%). The two most common GTTs are arguably much more relevant. Interestingly, a hierarchical clustering reveals that these two

most common GTTs cluster separately and that the GTT representing 27% of all suboptimal models is close to the optimal one (Fig. 5).

Finally, we have investigated the space of experiments to identify the simplest ones (i.e. minimal number of stimulations and inhibitions), which maximize the pairwise differences between the optimal and the two most common GTTs. These three GTTs differ pairwise in either one or two readouts among *p70s6*, *creb*, *p53*, and only 192 experiments generate two differences. Out of these 192 experiments, we identified eight experiments with minimal number of stimulations, and among them, we selected the ones with minimal number of inhibitions (Fig. 6). We noted that the two experiments found generate the same output over the readouts. Thus, in contrast to the seven experiments needed to discriminate among all GTTs, only one experiment is required to discriminate between the optimal and the two most common GTTs.

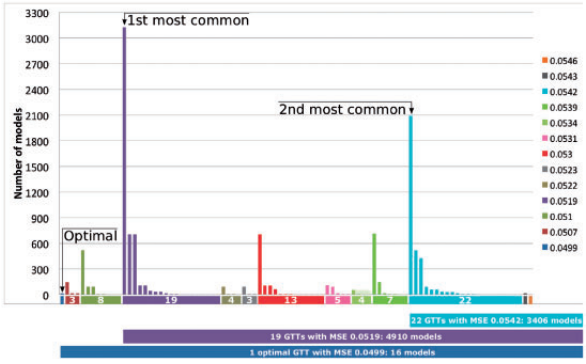



**Fig. 4.** Distribution of suboptimal models. The suboptimal models are ordered (from left to right) first according to their MSEs, and then according to their 91 GTTs. The number of different models leading to the same GTT is plotted in vertical bars. GTTs are ordered and colored by their MSE. The 16 optimal models correspond to MSE 0.0499. The two most common GTTs describe the response of 3126 and 2090 models

### 3.4 Comparison with an stochastic optimization

We compared *caspo* with CellNOpt (Terfve *et al.*, 2012), the existing tool to solve the same problem, but using a genetic algorithm. Stochastic search methods, such as genetic algorithms, are intrinsically unable not just to provide a complete set of solutions, but also to guarantee that an optimal solution is found. Typically, one needs to combine solutions from multiple runs to increase the confidence. Thus, to illustrate the value of *caspo* in comparison with CellNOpt, we have performed multiple runs of it over the same case study.

From multiple independent runs of CellNOpt (1000 runs with an average of 1000 s per run), we found 4706 suboptimal models out of the 11 700 models found using *caspo* (70 s). The MSEs of models found with CellNOpt spanned from 0.0499 to 0.0543 (Supplementary Fig. S3). This family of models was found combining 20% of the runs, whereas in the other 80% all models found were out of the allowed tolerance range. Notably, the 16 optimal models (MSE = 0.0499) were found by CellNOpt. Concerning GTTs, the 4706 models exhibit 51 input–output behaviors out of the 91 we found with *caspo* (Supplementary Fig. S4). The genetic algorithm retrieved all the GTTs in both extremes of the hierarchical cluster, while the rest of the cluster was not completely explored (Supplementary Fig. S5). Thus, plausible behaviors away from the most common ones appear less likely to be found. These results show the relevance of a software tool like *caspo*, which allows us to explore exhaustively the space of feasible solutions in short time.

## 4 CONCLUSION

A useful approach to model large-scale signaling networks consists on training Boolean logic models from prior knowledge and dedicated experimental data. The problem of training these models is an optimization task that can be solved with stochastic search methods (Saez-Rodriguez *et al.*, 2009), which have the important limitation that they do not guarantee global optimality nor an exhaustive solution. In this article, we show how recasting this problem in a highly declarative language allows us to

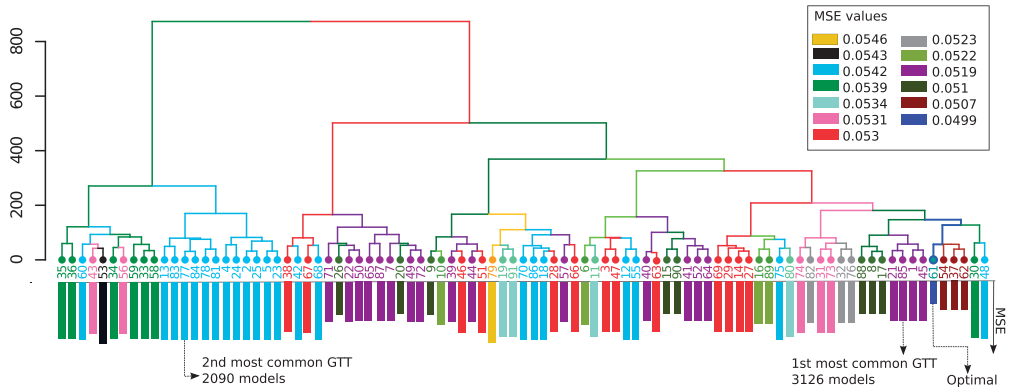



**Fig. 5.** Hierarchical clustering of GTTs. Hierarchical clustering of the 91 GTTs based on their predictions for the readouts across all experimental conditions. Bars length on the leafs represents the corresponding MSE value for each GTT. The optimal GTT (61) is highlighted, as well as the two most common ones (85 and 77). The most common GTT is close to the optimal one, whereas the second most common GTT has a different behavior

| | Stimuli | | | Inhibitors | | Readouts | | | |
|---|---|---|---|---|---|---|---|---|---|
| | TFGa | IGF1 | IL1a | mTORi | MEK12i | p70s6 | CREB | p53 | GTT |
| Experiment 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | Optimal |
| | | | | | | 1 | 0 | 1 | 1° most common |
| Experiment 2 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 2° most common |

**Fig. 6.** Experiments to discriminate more relevant GTTs. Both experiments generate the same output in each GTT. Stimuli not shown are inactive, inhibitors not shown are absent and readouts not shown have the same value inthree GTTs

identify the complete family of feasible models and query them to obtain insight into model degeneracy.

In a real-case study, we have seen that there is a family of feasible models with a deep combinatorial structure: several combinations of internal submodules, with equal or similar scores, can equivalently explain the observed behavior of the system. This leads to a rapid growth of the family of suboptimal models. Taking into account the inherent noise in data, we showed that 11 700 different models can be considered as plausible representations of the PKN and an experimental phosphoproteomics dataset. Thanks to our exhaustive characterization of these models, we could determine unambiguously which hyperedges (biological links) are functional, based on their distributions across the models and determine whether groups of hyperedges are exclusive from each other.

To further characterize this family of models, we introduced the concept of GTTs and used it to explore their input–output behavior. Compared with the model topologies, the variability is much lower; the 11 700 models can be grouped in 91 GTTs, and for 30% of the 16 384 possible perturbations, all models gave the same predictions. Interestingly, the distribution of models among GTTs is far from being equidistributed, and two GTTs comprise almost half of the models, while the GTT corresponding to the optimal score is specific (0.1% of the models). While the most common GTT is similar to the GTT with optimal score, the second most common GTT is different. However, a single experiment is able to discriminate these models.

These results underscore the importance of exploring exhaustively the family of models and take into account experimental error to obtain an adequate picture of the feasible model solutions. Our formal approach based on ASP allows a precise characterization of the information that can be inferred from the confrontation of prior knowledge with experimental observations over protein signaling networks. It also permits the study of the internal combinatorics leading to the variability of the system functioning and provides a tool toward experimental design. Owing to the complexity of signaling networks and the limitations of existing experimental technologies (in terms of which nodes can be measures and/or perturbed), models typically show an important lack of identifiability. This is a general limitation of models in systems biology (Kreutz and Timmer, 2009). In the context of Boolean models, we expect that further development of experimental design (Sharan and Karp, 2012), in intimate coordination with advances in experimental techniques will allow us to tackle this issue.

This work opens the way to several prospective tracks. First, it would be useful to evaluate our ASP formulation and those based on ILP from (Mitsos *et al*., 2009) and (Sharan and Karp, 2012) to understand their strengths and complementary features. In contrast to ILP, ASP is a relatively new tool for problem solving in biology. ASP, having its roots in knowledge representation and reasoning, has proven to be well suited to address highly combinatorial search and discrete optimization problems, with at least comparable performance with well established ILP solvers. On the other hand, ILP as a mathematical programming framework may be more suitable to study problems based on calculus over large domains of integer or rational numbers. Therefore, combining the expressiveness and power of several solving technologies instead of selecting one of them seems a promising option for the future (Liu *et al*., 2012; Ostrowski and Schaub, 2012).

Second, we plan to study the extension of our approach to time-series data, although switching from a steady state to a dynamical viewpoint implies a growth of the search space. Fitting models whose steady states evolve between clearly separated time-scales (Terfve *et al*., 2012) should be of similar complexity to the problem studied in this article. Fitting to the actual time-courses of a Boolean model has a higher level of complexity, as it requires to adjust the time-step of the Boolean model to the real time of the measurements.

More generally, we need to develop a rigorous framework to study models of biological networks as a family of plausible realizations, not of single networks. A first approximation could be to compare experimental data (ideally a distribution across individual cells) with a distribution of simulated results across a family of *single* logical models. The comparison of the distribution of feasible models with single cell data emerges as longer-term follow-up of this work that should provide deep insight into the cell-to-cell heterogeneity of signal transduction (Kolitz and Lauffenburger, 2012).

Altogether, we have implemented an open-source tool based on ASP providing a powerful framework to analyze networks models in systems biology. Further, several prospective tracks will certainly lead to future developments to extend and improve the functionalities of *caspo*.

## ACKNOWLEDGEMENTS

Thanks to E. Gonçalves for help plotting the networks and A. MacNamara and C. Chancellor for reading the manuscript.

*Funding*: EU through project 'BioPreDyn' (ECFP7-KBBE-2011-5 Grant number 289434); French National Agency for Research (ANR-10-BLANC-0218); Federal Ministry of Education and Research (BMBF 'MEDSYS' project 0315401B).

*Conflict of Interest*: none declared.

## REFERENCES

Alexopoulos,L.G. *et al.* (2010) Networks inferred from biochemical data reveal profound differences in toll-like receptor and inflammatory signaling between normal and transformed hepatocytes. *Mol. Cell Proteomics*, **9**, 1849–1865.

Banga,J.R. (2008) Optimization in computational systems biology. *BMC Syst. Biol.*, **2**, 47.

Baral,C. (2003) *Knowledge Representation, Reasoning and Declarative Problem Solving*. Cambridge University Press, Cambridge, UK.

Calzone,L. *et al.* (2010) Mathematical modelling of cell-fate decision in response to death receptor engagement. *PLoS Comput. Biol.*, **6**, e1000702.

Durzinsky,M. *et al.* (2011) Automatic network reconstruction using ASP. *Theory Pract. Logic Program.*, **11**, 749–766.

Fayruzov,T. *et al*. (2009) Modeling Protein Interaction Networks with Answer Set Programming. In: *International Conference on Bioinformatics and Biomedicine, 2009*. pp. 99–104.

Gebser,M. *et al*. (2010) Repair and prediction (under inconsistency) in large biological networks with answer set programming. In: *12th International Conference on Principles of Knowledge Representation and Reasoning*. AAAI Press, pp. 497–507.

Gebser,M. *et al*. (2013) *Answer Set Solving in Practice*. volume 19 of *Synthesis Lectures on Artificial Intelligence and Machine Learning*. Morgan and Claypool Publishers.

Klamt,S. *et al*. (2006) A methodology for the structural and functional analysis of signaling and regulatory networks. *BMC Bioinformatics*, **7**, 56.

Kolitz,S.E. and Lauffenburger,D.A. (2012) Measurement and Modeling of Signaling at a Single-Cell Level. *Biochemistry*, **51**, 7433–7443.

Kreutz,C. and Timmer,J. (2009) Systems biology: experimental design. *FEBS J*., **276**, 923–942.

Liu,G. *et al*. (2012) Answer set programming via mixed integer programming. In: *13th Int. Conf. on Principles of Knowledge Representation and Reasoning*. AAAI Press.

Mitsos,A. *et al*. (2009) Identifying drug effects via pathway alterations using an integer linear programming optimization formulation on phosphoproteomic data. *PLoS Comp. Biol*., **5**, e1000591.

Morris,M.K. *et al*. (2011) Training signaling pathway maps to biochemical data with constrained fuzzy logic: quantitative analysis of liver cell responses to inflammatory stimuli. *PLoS Comp. Biol*., **7**, e1001099.

Néron,B. *et al*. (2009) Mobyle: a new full web bioinformatics framework. *Bioinformatics*, **25**, 3005–3011.

Ostrowski,M. and Schaub,T. (2012) ASP modulo CSP: The clingcon system. *Theory Pract. Logic Program*., **12**, 485–503.

Papatheodorou,I. *et al*. (2012) Using Answer Set Programming to Integrate RNA Expression with Signalling Pathway Information to Infer How Mutations Affect Ageing. *PLoS One*, **7**, e50881.

Ray,O. and Soh,T. (2012) Analyzing pathways using ASP-based approaches. *Algebr. Numeric Biol*., **6479**, 167–183.

Saez-Rodriguez,J. *et al*. (2009) Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction. *Mol. Syst. Biol*., **5**, 331.

Schaub,T. and Thiele,S. (2009) Metabolic network expansion with answer set programming. In: *25th International Conference on Logic Programming*. volume 5649 of *LNCS*. Springer.

Sharan,R. and Karp,R.M. (2012) Reconstructing Boolean models of signaling. In: *Research in Computational Molecular Biology*. Springer Berlin Heidelberg, pp. 261–271.

Terfve,C.D. *et al*. (2012) CellNOptR: a flexible toolkit to train protein signaling networks to data using multiple logic formalisms. *BMC Syst. Biol*., **6**, 133.

Videla,S. *et al*. (2012) Revisiting the Training of Logic Models of Protein Signaling Networks with ASP. In: *10th International Conference on Computational Methods in Systems Biology*. LNCS. Springer, pp. 342–361.

Walter,E. and Pronzato,L. (1996) On the identifiability and distinguishability of nonlinear parametric models. *Math. Comput. Simul*., **42**, 125–134.

Wang,R.S. *et al*. (2012) Boolean modeling in systems biology: an overview of methodology and applications. *Phys. Biol*., **9**, 055001.

## A.4 Learning Dynamical Boolean Networks

PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

# Computational discovery of dynamic cell line specific Boolean networks from multiplex time-course data

**Misbah Razzaq**[1], **Loïc Paulevé**[2,3], **Anne Siegel**[4], **Julio Saez-Rodriguez**[5,6], **Jérémie Bourdon**[1], **Carito Guziolowski**[1]*

**1** Université de Nantes, Centrale Nantes, CNRS, Laboratoire des Sciences du Numérique de Nantes (LS2N UMR 6004), F-44000, Nantes, France, **2** LRI UMR8623, Université Paris-Sud, CNRS, Université Paris-Saclay, F-91400 Orsay, France, **3** Université Bordeaux, Bordeaux INP, CNRS, LaBRI, UMR5800, F-33400 Talence, France, **4** Institut de Recherche en Informatique et Systèmes Aléatoires, Rennes, France, **5** RWTH-Aachen University, Faculty of Medicine, Joint Research Center for Computational Biomedicine, Aachen, Germany, **6** European Molecular Biology Laboratory, European Bioinformatics Institute (EMBL-EBI), Cambridgeshire, UK

* Carito.Guziolowski@ls2n.fr





**Citation:** Razzaq M, Paulevé L, Siegel A, Saez-Rodriguez J, Bourdon J, Guziolowski C (2018) Computational discovery of dynamic cell line specific Boolean networks from multiplex time-course data. PLoS Comput Biol 14(10): e1006538. https://doi.org/10.1371/journal.pcbi.1006538

**Editor:** Joerg Stelling, ETH Zurich, SWITZERLAND

**Received:** January 25, 2018

**Accepted:** October 2, 2018

**Published:** October 29, 2018



**Data Availability Statement:** All the data can be downloaded from DREAM8 web portal. All the processed files for this publication can be found here: https://github.com/misbahch6/caspo-ts.

**Funding:** This work was supported by CG's CNRS chair of excellence, by Ecole Centrale de Nantes, and by the Pays de la Loire French Regional GRIOTE project. LP acknowledges support from French Agence Nationale pour la Recherche (ANR) in the context of the ANR-FNR project AlgoReCell [ANR-16-CE12-0034]. The funders had no role in

## Abstract

Protein signaling networks are static views of dynamic processes where proteins go through many biochemical modifications such as ubiquitination and phosphorylation to propagate signals that regulate cells and can act as feed-back systems. Understanding the precise mechanisms underlying protein interactions can elucidate how signaling and cell cycle progression occur within cells in different diseases such as cancer. Large-scale protein signaling networks contain an important number of experimentally verified protein relations but lack the capability to predict the outcomes of the system, and therefore to be trained with respect to experimental measurements. Boolean Networks (BNs) are a simple yet powerful framework to study and model the dynamics of the protein signaling networks. While many BN approaches exist to model biological systems, they focus mainly on system properties, and few exist to integrate experimental data in them. In this work, we show an application of a method conceived to integrate time series phosphoproteomic data into protein signaling networks. We use a large-scale real case study from the HPN-DREAM Breast Cancer challenge. Our efficient and parameter-free method combines logic programming and model-checking to infer a family of BNs from multiple perturbation time series data of four breast cancer cell lines given a prior protein signaling network. Because each predicted BN family is cell line specific, our method highlights commonalities and discrepancies between the four cell lines. Our models have a Root Mean Square Error (RMSE) of 0.31 with respect to the testing data, while the best performant method of this HPN-DREAM challenge had a RMSE of 0.47. To further validate our results, BNs are compared with the canonical mTOR pathway showing a comparable AUROC score (0.77) to the top performing HPN-DREAM teams. In addition, our approach can also be used as a complementary method to identify erroneous experiments. These results prove our methodology as an efficient dynamic model discovery method in multiple perturbation time course experimental data of large-

study design, data collection and analysis, decision to publish, or preparation of the manuscript.

**Competing interests:** The authors have declared that no competing interests exist.

scale signaling networks. The software and data are publicly available at https://github.com/misbahch6/caspo-ts.

## Author summary

Traditional canonical signaling pathways help to understand overall signaling processes inside the cell. Large scale phosphoproteomic data provide insight into alterations among different proteins under different experimental settings. Our goal is to combine the traditional signaling networks with complex phosphoproteomic time-series data in order to unravel cell specific signaling networks. In this study, we have applied the caspo time series (*caspo-ts*) approach which is a combination of logic programming and model checking, over the time series phosphoproteomic dataset of the HPN-DREAM challenge to learn cell specific BNs. The learned BNs can be used to identify the cell specific topology. Our analysis suggests that *caspo-ts* scales to real datasets, outputting networks that are not random with a lower fitness error than the models used by the 178 methods which participated in the HPN-DREAM challenge. On the biological side, we identified the cell specific and common mechanisms (logical gates) of the cell lines.

## Introduction

Protein signaling networks are static views of dynamic processes since they respond to stimuli and perturbation. They constitute complex regulatory systems controlled by crosstalk and feedback mechanism. Because these networks are often altered in diseases, discovering the precise mechanisms of signal transduction may provide a better fundamental understanding of disease behavior. For instance, a main difficulty in cancer treatment is that different signaling networks fact that cell populations specialize upon treatment and therefore patient responses may be heterogeneous. Computational models of signaling control for different patient groups could guide cancer research towards a better drug targeting system. In this work, we propose a methodological framework to discriminate among the regulatory mechanisms of four breast cancer cell lines by building predictive computational models.

Several formalisms have been used widely to model interaction networks. Models built using differential equations require explicit specifications of kinetic parameters of the system and work well for small-scale systems. While being a useful tool, mathematical modeling becomes computationally intensive as networks become larger [1–3]. Stochastic modeling is suitable for problems of a random nature but also fails to scale well with large scale systems [1].

The Boolean Network (BN) formalism [4] is a powerful approach to model signaling and regulatory networks [5]. Various BN learning frameworks exist focusing on varying levels of details [1, 6]. As compared to the extensive literature on Boolean frameworks, BN modeling of signaling networks is quite recent.

In this work, we have used the caspo time series (*caspo-ts*) [7, 8] method to learn BNs from multiple perturbation phosphoproteomic time series data given a Prior Knowledge Network (PKN). We have improved and adapted *caspo-ts* to deal with a midscale Prior Knowledge Network (PKN) with 64 nodes and 178 edges in order to learn the BNs of four breast cancer cell lines (BT20, BT549, MCF7, UACC812) from their time series phosphoproteomic datasets. Importantly, the PKN did not contain any information about the temporal changes or

dynamic properties of the proteins. This information was learned from a dataset describing the dynamics of signaling processes for those breast cancer cell lines as part of the HPN-DREAM challenge. In comparison to the current methods that learn signaling networks as Boolean models using static measurements[9, 10], and one-time point measurements across multiple perturbations [11–14], our method allows us to handle time series data. A further advantage is the guarantee of discovering optimal BNs, where the distance between original and over-approximated time series is minimal. This is achieved by using computational solvers such as Answer Set Programming (ASP) [15].

Our results show that the ASP component of our method allows us to filter the explosion of possible dynamical states inherent to this type of problem, and thanks to that filtering, the model-checking step allows us to provide BNs exactly reproducing the binarized time series data. These BNs are referred to as true positive (TP) BNs. Our results point to measurements in the time series HPN-DREAM dataset that contradict the experimental setting and to perturbations that show contradictory dynamics. We observed that given the same PKN, the solving time was different for each cell line dataset. For cell lines BT20, BT549, MCF7 our method found TP BNs, while for the UACC812 cell line dataset it was impossible to find a TP BN within a time-frame of 7 days. This computation time difference is due to the different structure of the solution space among cell lines. This could point to the situation where the dataset is not explainable by the prior knowledge network, which may give valuable insights to experimentalists. For example, that the number of consistent experimental perturbations is not sufficient, and that the knowledge of the PKN is incomplete given this dataset. We also show that this method is capable of recovering time series measurements with a Root Mean Square Error (RMSE) of 0.31, the minimum achieved so far as compared to other participants of the HPN-DREAM challenge. Our method focuses on learning optimal BNs' structures. It does not predict time-series traces of the proteins from the learned BNs. However it detects the minimum distance that is possible to obtain from the proteins of the learned BNs in comparison to the time-series traces in the testing data. This is the main conceptual difference of our method compared to those proposed by the HPN-DREAM challenge. This difference needs to be considered when comparing the RMSE score. Based on a comparison with the canonical mTOR pathway, we show that the discovered context specific BNs have an average AUROC score of 0.77. We found 38% of the cell line specific interactions explaining the heterogeneity among these four cancer cell lines, which can be observed in different cell line specific networks, shown in S1, S2, S3 and S4 Figs. All in all, our results show that *caspo-ts* handles real (HPN-DREAM) datasets, where data points are incomplete and subject to experimental error. Our method is applicable to any kind (gene or protein expressions) of time series datasets measured upon different perturbations. We have proved here that *caspo-ts* handles a PKN size of 64 nodes and 170 edges; this is relevant since approaches modeling time usually only scale up to very small networks because of state graph explosion.

## Related work

Regarding the training of BNs with respect to multiple perturbation datasets, CellNOpT (CNO)[16] assembles BNs from a Prior Knowledge Network (PKN) and phosphoproteomic datasets. Their tool has been implemented using stochastic search algorithms (more precisely, a genetic algorithm), to suggest multiple BNs explaining the data [17]. However, stochastic search methods cannot generate a complete set of solutions, hence they cannot guarantee a global optimal solution. In [11, 12], the authors overcome this problem by proposing *caspo*, an approach based on ASP to infer BNs explaining the underlying protein signaling network.

This approach can generate all possible optimal Boolean models as compared to the CellNOpt approach. The authors in [14], presented a framework based on integer linear programming (ILP) to learn the subset of interactions best fitted to the experimental data. Recently, another approach based on ILP has been proposed to reconstruct BNs from experimental data. Their learning approach do not require the information about the activation/repression properties of the network's edges [13].

The methods mentioned above are very useful but restrain themselves to learn from only two time points, assuming the system has reached an early steady-state when the measurements are performed. This assumption prevents us from capturing interesting characteristics like loops [3]. To overcome this issue, the caspo time series (*caspo-ts*) method was proposed in [8]. This method learns BNs from multiple perturbation phosphoproteomic time series data given a PKN. The proposed method is based on ASP and a model-checking step is needed to detect true positive BNs. They tested their approach on synthetic data for a small PKN (≈17 nodes and ≈50 edges) [8]. More recently, an approach based on genetic algorithms was proposed to learn context specific networks given a PKN and experimental information about stable states and their transitions but it does not scale well with large networks and finding a global optimum is not guaranteed [18].

***Caspo-ts* modeling framework.** We chose the *caspo-ts* method [7, 8] for the inference of BNs. This method was tailored to handle protein phosphoproteomic time series data. The input of the method consists of a PKN and normalized phosphoproteomic time series data under different perturbations to generate a family of BNs whose structure is compatible with the PKN and that can also reproduce the patterns observed in the experimental data. In the following, we will develop the main notions of this framework.

**Prior knowledge network.** It is one input of *caspo-ts* and it is modeled as a labeled (or colored) directed graph $(V, E, \sigma)$ with $V = \{v_1, v_2, \ldots, v_n\}$ the set of nodes, $E \subseteq V \times V$ the set of directed edges and $\sigma \subseteq E \times \{+1, -1\}$ the signs of edges. The set of nodes is denoted by $V = S \cup I \cup R \cup U$ where $S$ are stimuli, $I$ are inhibitors, $R$ are readouts, and $U$ are unobserved nodes. Stimuli, inhibitors, readouts, and unobserved nodes are encoded by different colors in the graphs presented in this case study. Stimuli are shown in green, inhibitors in red, readouts in blue, and unobserved nodes in white (Fig 1). Moreover, the subsets $S$, $I$, $R$, $U$ are all pairwise disjoint except for $I$ and $R$, because a protein can be inhibited as well as measured. Stimuli are used to bound the system and also serve as interaction points of the system, these nodes can be experimentally stimulated, *e.g.* cellular receptors. Inhibitors are those nodes which remain inactive or blocked over all time points of the experiment by small molecule inhibitors. Stimuli and inhibitor nodes take Boolean values {0, 1} representing the fact that the node was stimulated (1) or inhibited (0). Readouts are experimentally measured given a combination of stimuli and inhibitors. They usually take continuous values in [0;1] after normalization. Unobserved nodes are neither measured nor experimentally manipulated. In this study, we use the term *perturbation* to refer to the combination of stimuli and inhibitors, similarly to other studies such as [19–21].

**Phosphoproteomic time series data.** It is the second input of *caspo-ts* and it consists of temporal changes in phosphorylated proteins under a perturbation (Fig 1). Without loss of generality, we assume that the time series data are related to the observation of $m \leq n$ nodes for the nodes $\{v_1, \ldots, v_m\}$ (so the nodes $\{v_{m+1}, \ldots, v_n\}$ are not observed). The observations consist of normalized continuous values: a time series of k data points is denoted by $T_P = (t_P^1, \ldots, t_P^k)$, where $P \subseteq S \cup I$ is a perturbation and $t^j \in [0; 1]^m$ for $1 \leq j \leq k$. This data will be discretized in order to link it with further BNs' discovery (ASP solving and model checking steps).

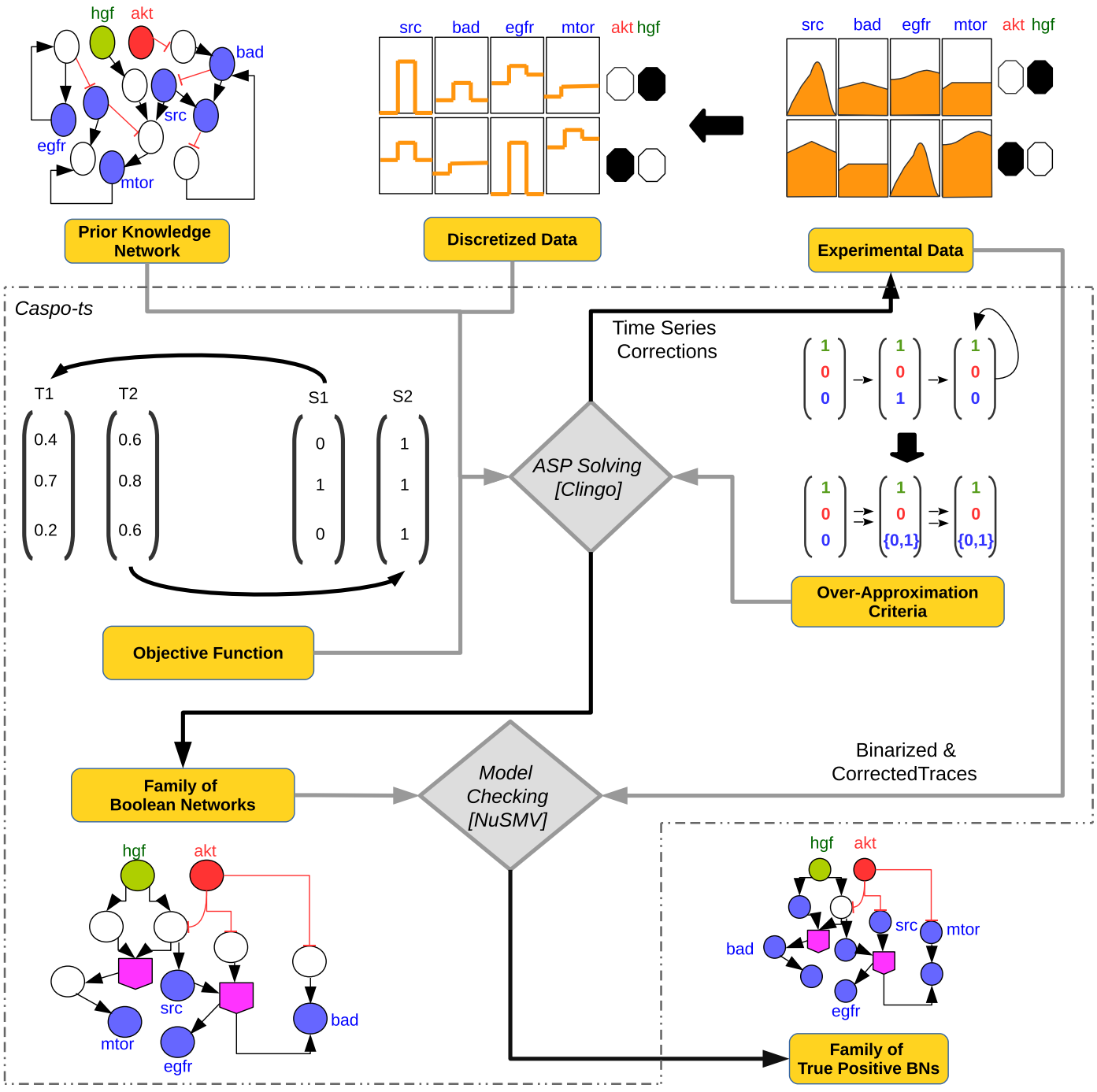



**Fig 1. *Caspo-ts* workflow.** *Caspo-ts* receives as input data a prior knowledge network (PKN) and a discretized phosphoproteomic dataset. In this example the phosphoproteomic data consists of two perturbations involving *akt* (inhibitor) and *hgf* (stimulus): 1) *akt* = 0, *hgf* = 1 and 2) *akt* = 1, *hgf* = 0. A black colored perturbation means the inhibitor or stimulus was perturbed (1) while white represents the opposite (0). Readouts are specified in blue and describe the time series under given perturbations. Using this input data, *caspo-ts*, performs two steps: ASP solving and model checking. In the ASP solving step: (i) a set of BNs, compatible with the PKN, is generated, (ii) afterwards an *over-approximation constraint* is imposed upon each candidate BN to filter out invalid BNs, that do not result in an over-approximation of the reachability between the Boolean states given by the phosphoproteomic dataset, and finally (iii) BNs are optimized using an objective function minimizing the distance to the experimental measures. The ASP step also introduces repairs in some data points of the time series that added penalties to the objective function. These corrected traces will be given to the model checker. In the model checking step, the exact reachability of all the (binarized and corrected) time series traces in the family of BNs is verified.

https://doi.org/10.1371/journal.pcbi.1006538.g001

**Boolean Network.** It is the output of *caspo-ts*. A *Boolean Network (BN)* [22, 23] is defined as a pair $B = (N, F)$, where

- $N = \{v_1, \ldots, v_n\}$ is a finite set of nodes (or variables/proteins/genes)
- $F = \{f_1, \ldots, f_n\}$ is a set of Boolean functions (regulatory functions) $f_i : \mathbb{B}^k \rightarrow \mathbb{B}$, with $\mathbb{B} = \{0, 1\}$, describing the evolution of variable $v_i$.

A vector (or *state*) $x = (x_1, \ldots, x_n)$ captures the values of all nodes $N$ at a time step, where $x_i$ represents the value of the node $v_i$, and is either 1 or 0. There are up to $2^n$ possible distinct states for each time step. Next, we define the transition $x \rightarrow x'$ between two states of a BN. If there is no update for node $v_i$ then $x'_i = x_i$. If there is an update for node $v_i$ then the state of a node $v_i$ at the next time step is determined by $x'_i = f_i(x_1, \ldots, x_n)$. Note that usually only a subset of the nodes influence the evolution of node $v_i$. These nodes are called the *regulatory nodes* of $v_i$. The state of each node can be updated in a synchronous (parallel) or asynchronous fashion. In the synchronous update schedule, the states of all nodes are updated, while in asynchronous update schedule, the state of one node is updated at a time. The work presented in this article is independent of the update schedule routine, hence any number of nodes can be updated at a time.

**ASP solving.** Given a PKN and a phosphoproteomic dataset, a family of candidate BNs, compatible with this PKN, is exhaustively enumerated including the main nodes (the sets *S*,*I*,*R*) of the experimental data. We refer the reader to [12] for a detailed description of BN's compatibility with a PKN. Afterwards an *over-approximation constraint* (see Materials and methods) is imposed upon each candidate BN to filter out invalid BNs [8], that do not result in an over-approximation of the reachability between the Boolean states given by the phosphoproteomic dataset. Finally, an optimization step is performed to select those BNs having a minimal distance between the actual time series $T_P$ and the over-approximated time series $Y_P$. We have adopted the Root Mean Square Error (RMSE) as the *objective function*:

$$RMSE = \sqrt{\frac{1}{m * k * |\mathcal{P}|} \sum_{i=1}^{m} \sum_{j=1}^{k} \sum_{P \in \mathcal{P}} ((t_P^j)_i - (y_P^j)_i)^2} \quad (1)$$

where $m$ is the number of observed nodes, $k$ is the number of time points, and $\mathcal{P}$ is the set of perturbations. In addition, the optimization step highlights the data points in the time series which added penalties to the RMSE. Such data points are automatically corrected before the model checking step.

All the analyses described in this step are performed using ASP, namely the `clingo` 4.5.4 solver [15]. This solver guarantees finding optimal solutions, and all BNs outputted by the ASP solver step will be identically optimal. For the HPN-DREAM case study, the full enumeration of optimal BNs creates billions of BNs, and since the next (model checking) step can take days of computation depending on the verified BN we choose to limit this enumeration to a fixed number of BNs.

**Model checking and true positive BNs.** From the ASP solving step, a set of optimal BNs that over-approximate the phosphoproteomic time series data is produced. This set of BNs is verified with an exact *model checking* to detect true positive (TP) BNs. *TP BNs* are guaranteed to reproduce all the (binarized) trajectories under all perturbations by verifying exact reachability in the BN state graph. For this, we have used computational tree logic (CTL) implemented in the `NuSMV` 2.6.0 [24], which is a symbolic model checker.

**Caspo-ts workflow.** The *caspo-ts* workflow is shown in Fig 1. It consists of two main steps, ASP solving and model checking, as described previously.

# Results

## Prior knowledge network

The structure of the protein signaling network was generated by mapping the experimentally measured phosphorylated proteins (HPN-DREAM dataset) to their equivalents from literature-curated databases and connecting them together within one network (see Materials and methods). The reference network (Fig 2) was built using the ReactomeFIViz app (also called the ReactomeFIPlugIn or Reactome FI Cytoscape app) [25], which accesses the interactions existing in the Reactome and other databases [25, 26]. The PKN shown in Fig 2 consists of 64 nodes (7 stimuli, 3 inhibitors, and 23 readouts) and 178 edges.

## Data processing

The learning and testing datasets used in this study were extracted from the HPN-DREAM challenge and correspond to time series protein measurements upon different perturbations of four breast cancer cell lines—UACC812, BT20, BT549, and MCF7 [20, 21] (see Materials and methods). Since readout signals are measured on variable ranges depending on the protein, a normalization step was necessary. The learning dataset had a few noisy, inconsistent and incomplete time series data points. The *caspo-ts* system identified these inconsistencies existing in the time series data. The recurrent experimental inconsistency observed was an oscillation in the protein signal upon experimental inhibition of the same protein.

To resolve the above mentioned issues, we performed the following data processing steps on the learning dataset:

1. Set the protein values between a common scale, *i.e.*, 0 and 1, using a maximum-value-based normalization scheme (see Materials and methods).
2. For time point 0 the expression of some readout proteins under some perturbations was not available. Thus, control experimental readings have been used as the time point 0 for such proteins.
3. In some cases readout measurements were duplicated for the same time point, to solve this noise issue we have chosen one time point arbitrarily.
4. We have removed inconsistent perturbations where the protein AKT was inhibited and was having a dynamic behavior as a readout protein.
5. We have considered only perturbations with complete time series data, since guessing the missing time points automatically with *caspo-ts* for this case study will be computationally expensive.

The experimental errors pointed in steps 2-5 were raised as warning or exceptions by *caspo-ts*. Steps 1 to 5 were applied on the learning dataset. Only step 1 was applied on the testing dataset.

## Cell line specific Boolean Networks

In this section, we show the generated BNs for each cell line. For this, we used *caspo-ts* to learn the BNs from the PKN (Fig 2) and the phosphoproteomic data of four breast cancer cell lines—BT20, BT549, MCF7, and UACC812. We inferred a family of cell line specific BNs for each cancer cell line and they are shown in the Supplementary Figures (S1, S2, S3 and S4 Figs).

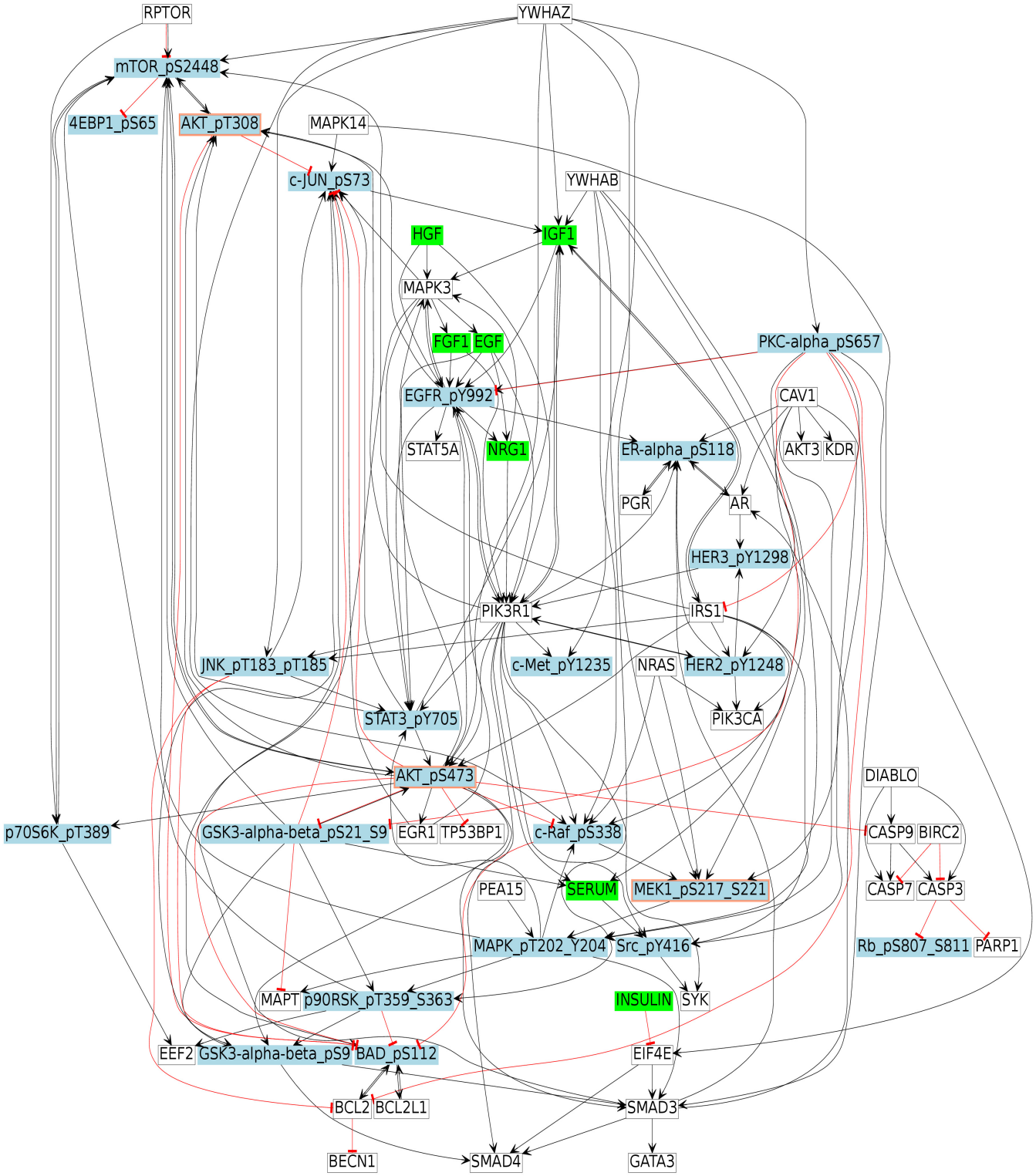



**Fig 2. Breast cancer signaling pathway.** This figure shows the reconstructed signaling network from a combination of databases. An arrow shows the positive regulatory relationship between two proteins, while a T shaped arrow indicates inhibition. Green nodes are stimuli, blue nodes are readouts, white nodes are unmeasured or unobserved, and blue nodes with a red border represent inhibitors and readouts at the same time. Please note that in the node labels, we have added the phosphorylation sites to the protein names in order to connect them to the experimental measurements.

https://doi.org/10.1371/journal.pcbi.1006538.g002

As explained in the *caspo-ts modeling framework* section, the *caspo-ts* method produces BNs fulfilling two criteria, (i) satisfaction of the over-approximation criteria (see Materials and methods) and (ii) optimality with respect to the RMSE objective function. ASP-optimal solutions were fast to collect, their computation time ranged from 36 seconds to 3 minutes depending on the cell line (S1 Table). Afterwards, these ASP-optimal BNs were given to the model-checker for further verification. This second step is more complex and we put a restriction for the computation time of 7 days for each cell line. The number of verified BNs varies from one cell line to another, depending on a number of factors such as the number of perturbations, the order of answer sets in the solutions space, and the perturbation order. The total number of verified ASP-optimal BNs within the 7 days time-frame were 231, 52, 188 and 150 for the BT549, MCF7, BT20 and UACC812 cell lines respectively. We obtained 191, 21, and 72 true positive BNs for BT549, MCF7, and BT20 cell lines respectively with an optimal fit to the data. For the UACC812 cell line, we were unable to obtain true positive BNs within the 7 day time limit for verification. Hence, we kept the first 20 BNs from the 150 ASP-optimal BNs for the UACC812 cell line. The *caspo-ts* method uses the ASP solver (clingo), which is able to exhaustively enumerate all solutions. The clingo solver by default uses an enumeration scheme, in which, once a solution is found, it backtracks to the first point from where the next solution can be found. This typically leads to the situation where successive solutions only change in a small part. As a result, *caspo-ts* may enter a solution space where BNs are clustered together. We have observed that given the size of the PKN and the small number of perturbations in the experimental data, the solution space of the *caspo-ts* can be rather very large containing billions of BNs making it difficult to enumerate true positive BNs (because of the model checking overhead) in a reasonable time if it gets stuck in a cluster of false positive BNs.

An aggregated network was built (Fig 3) by combining the BN families (with 191, 21, 72, and 20 BNs for BT549, MCF7, BT20, and UACC812 cell lines respectively) obtained for the four cell lines by keeping the hyper-edges (Boolean functions) having a frequency higher than 0.3 within each BN family. The frequency is calculated by counting the number of common Boolean functions and dividing it by the total number of Boolean functions within the BN family of each cell line. This aggregated network contains 34 nodes and 74 Boolean functions involving 36 AND gates. As compared to the PKN (Fig 2), the inferred networks are highly specific to each cell line. In Fig 3, all cell lines share only 4% of Boolean functions which are shown in thick black colored edges. This shows that the inferred BNs of these four breast cancer cell lines are very diverse and different from each other.

To measure cell lines similarity, we calculated the similarity score by applying the Graph Similarity Measure (see Materials and methods) on the family of BNs (with 191, 21, 72, and 20 BNs for BT549, MCF7, BT20, and UACC812 cell lines, respectively). This algorithm receives two parameters as input: (1) one gold standard BN and (2) a family of BNs. It outputs a score in [0; 1], measuring the average of the similarity scores between each BN in the family and the gold standard BN. In our case, the gold standard BN is the aggregation of one family of BNs. The similarity scores between all pairs of breast cancer cell lines are shown in Table 1. Fig 3 agrees with the results presented in Table 1 as we can see the clear discrepancies among the four cell lines. It can be seen that 23% of the Boolean functions are shared among BT549 and MCF7, and also between BT20 and UACC812. BT20 shares the least number of Boolean functions (15%) with BT549. This table revealed pronounced differences among different cell lines of breast cancer. We also analyzed the diversity of Boolean functions among the family of BNs within the same cell line. The similarity among Boolean functions from BT20 (0.73) and MCF7 (0.63) is higher than the ones from BT549 (0.43) and UACC812 (0.46) cell lines.

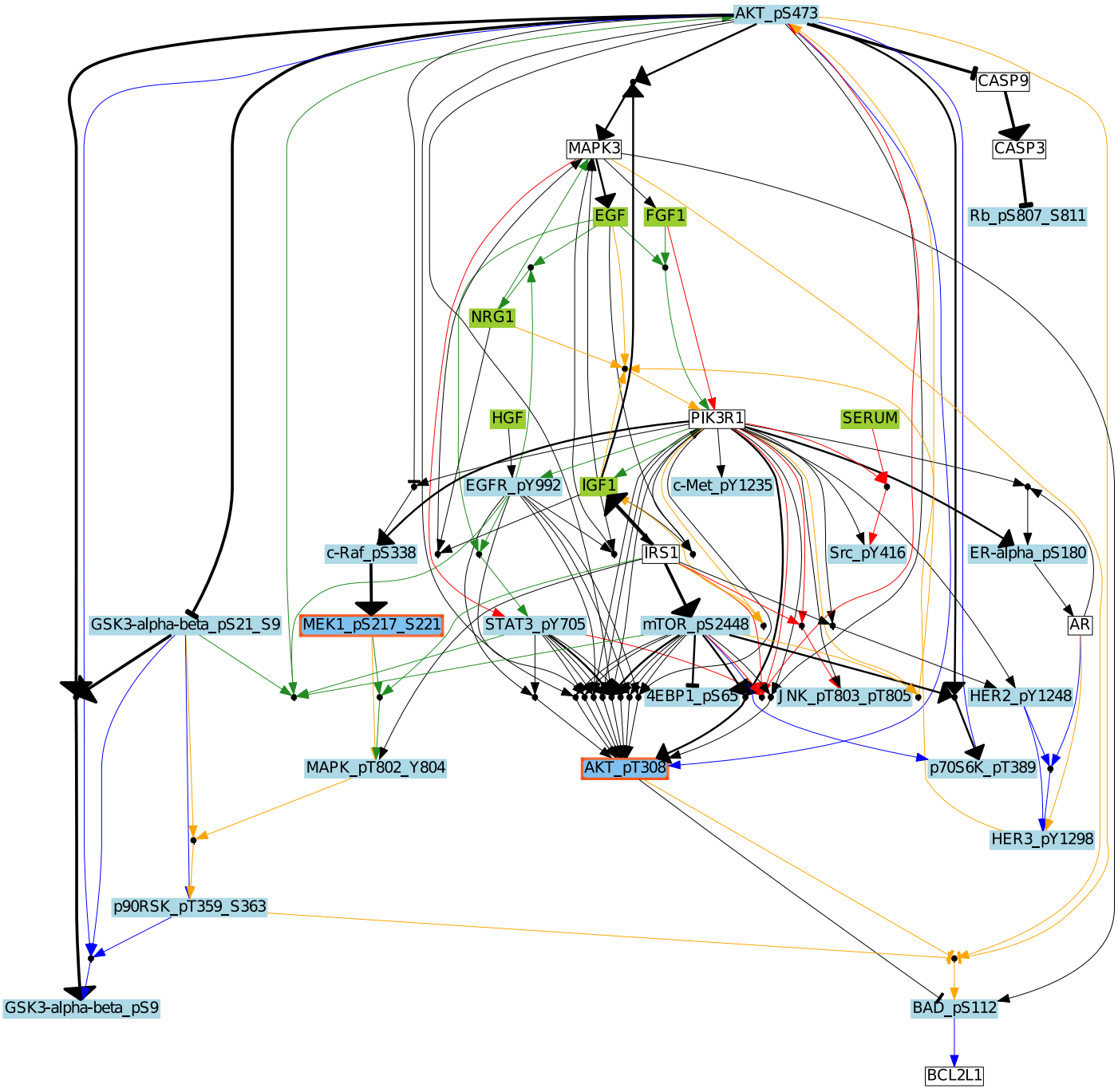



**Fig 3. Boolean Network of breast cancer cell lines.** The aggregated graph for all cell lines. Blue, red, green and orange edges are used for each cell line BT20, BT549, MCF7 and UACC812, respectively. The nodes are connected by logic gates (AND or OR) to their direct predecessors. Edges are used to show influences (→ for positive and ⊣ for negative). An AND gate is depicted by a small black circle where the incoming edges correspond to the inputs of the gate. An OR gate is depicted by multiple incoming edges to the node. A different color scheme is used to represent different types of nodes. The green color is for stimuli, the red for inhibitors, the blue for readouts, and the white for unobserved nodes. Black edges denote common hyper-edges across cell lines; the thickness of the black hyper-edge denotes the number of cell lines sharing this hyper-edge.

https://doi.org/10.1371/journal.pcbi.1006538.g003

**Table 1. Similarity scores among breast cancer cell lines.**

| Cell Lines | Size of BNs' family | Similarity Score | | | |
|---|---|---|---|---|---|
| | | BT20 | BT549 | MCF7 | UACC812 |
| *BT20* | 72 | 0.73 | 0.15 | 0.17 | 0.23 |
| *BT549* | 191 | ** | 0.43 | 0.23 | 0.20 |
| *MCF7* | 21 | ** | ** | 0.63 | 0.21 |
| *UACC812* | 20 | ** | ** | ** | 0.46 |

https://doi.org/10.1371/journal.pcbi.1006538.t001

## Heterogeneity among cell lines

There are a total of 69 distinct Boolean functions shown in Fig 4 along with their respective frequencies. It is interesting to note that the B549 and UACC812 cell lines have more distinct models among their family of BNs with a variable frequency range. This shows that these cell lines have different mechanisms agreeing with the results obtained through graph similarity measure given in Table 1.

Fig 5 shows the common Boolean functions along with their frequency in all BNs. Interestingly, only 4% of the Boolean functions are shared in all cell lines and 88% of these shared functions have the same frequency. In this figure, there is only one Boolean function which is frequent in 3 cell lines and has a lower frequency in BT20.

## Literature knowledge about Boolean functions discovered by *caspo-ts*

The union of the BNs learned for each cell line is displayed in the Supplementary Figures (S1, S2, S3 and S4 Figs). The *caspo-ts* method revealed that cell line specific reactions are clustered around the *AKT*, *MAPK3*, and *PIK3R1* proteins. *PI3K* is an important factor for cancer development in HER2 amplified cancers (UACC812) as compared to non-HER2 amplified (BT20, BT549 and MCF7) cancer cell lines. We can see from the Supplementary Figures (S1, S2, S3 and S4 Figs) that *PIK3R1* exists in all cell lines but is rather more connected in the UACC812 cell line with 10 incoming edges while in others with only 1 incoming edge. The *PIK3R1* node in UACC812 (S4 Fig) has a centrality measure of 0.37 while in the other three cell lines the centrality measure is less than 0.11. The centrality measure is used to quantify the most important node within a network i.e., the number of times a node has been used as a bridge (along the shortest path) to connect to other nodes in the network [27].

It has been established that *P1K3R1* (the regulatory unit of PI3K) plays an important role in suppressing tumors [28, 29]. Recently, it has been found that *PIK3R1* is mutated in 3% of breast cancer cell lines[30]. Nonetheless, it is worth studying the impact of the *PIK3R1* regulatory unit in breast cancer.

## Evaluation

The performance of the *caspo-ts* method is evaluated using three criteria: 1) RMSE calculation using a typical learning and testing data approach, 2) random data comparison, 3) AUROC (Area Under the Operating Curve) score.

The BNs are learned using the learning dataset (see Materials and methods) only. The prediction accuracy is evaluated by comparing the RMSE of trajectories in the testing dataset with those recovered by the learned networks (see Eq 1). There are two types of RMSE—discrete and model. The *discrete RMSE* is imposed by the discretization of the method. Since we use a discrete learning approach, our recovered traces will be in {0,1} and this introduces an error with respect to continuous measurements in [0;1]. The *model RMSE* refers to the learned BN

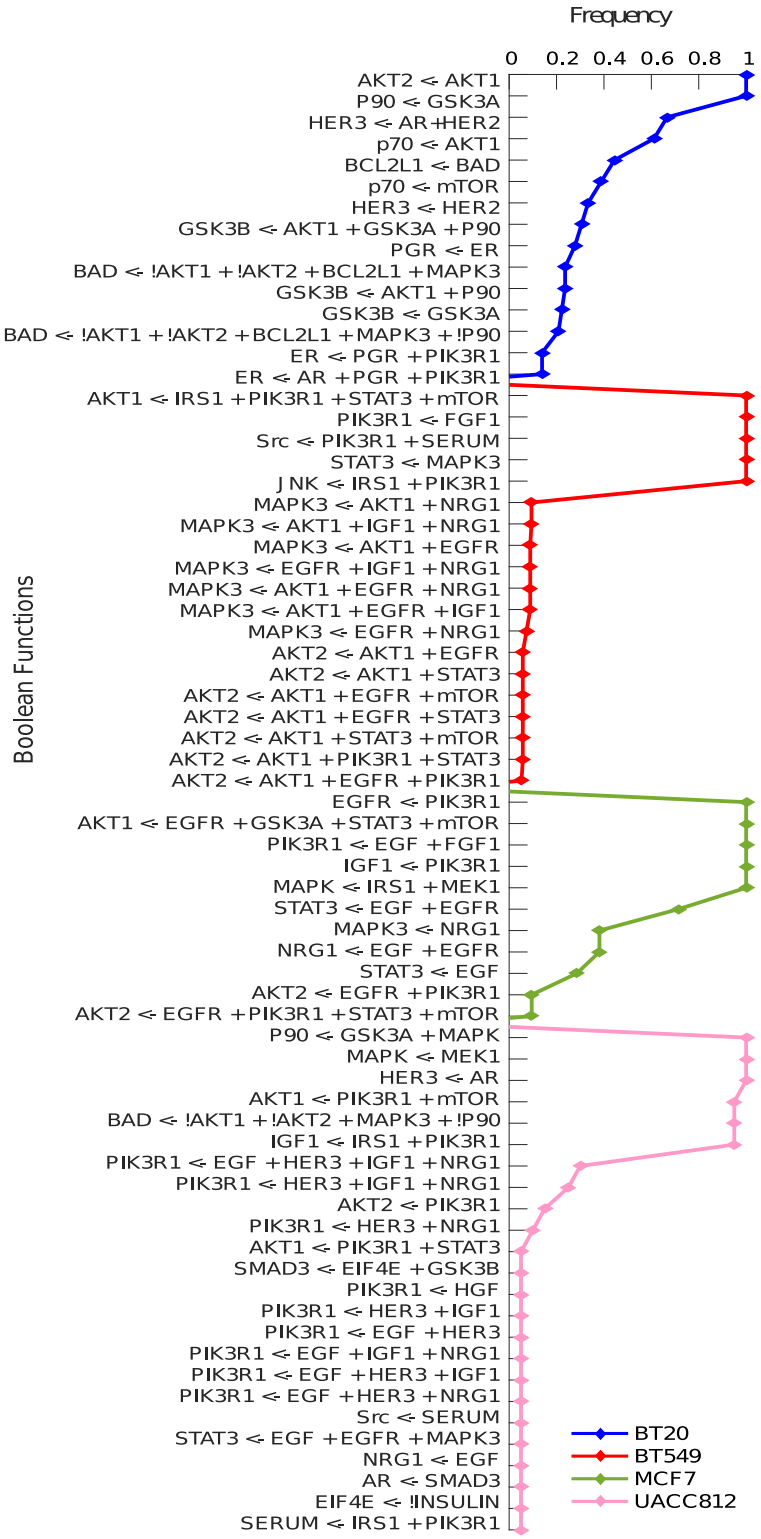



**Fig 4. Heterogeneous Boolean functions.** The Boolean functions are represented on the y-axis and the frequency of each Boolean function is shown on the x-axis. A Boolean function, or hyper-edge, is of the form *node* ← *expr*, where *node* is the receiver of the Boolean clause *expr* in the BN. In the Boolean clause, the *not* operator is represented by a "!" symbol and the AND operator by a "+" symbol. The disjunction of clauses is represented by multiple reactions upon the same receiver node.

https://doi.org/10.1371/journal.pcbi.1006538.g004

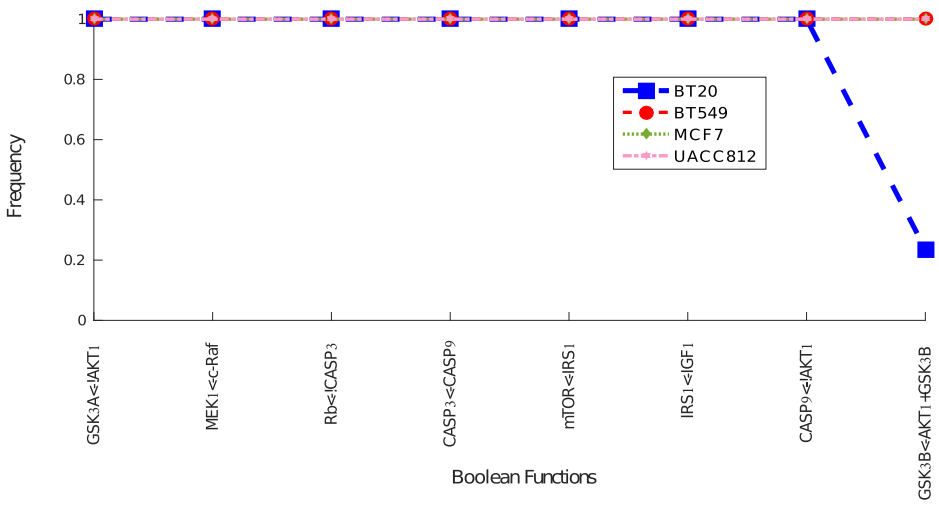



**Fig 5. Common Boolean functions across all four cell lines.** The Boolean functions are represented on the x-axis and the frequency of each Boolean function is shown on the y-axis.

https://doi.org/10.1371/journal.pcbi.1006538.g005

error with respect to the normalized time series data; that is, the model RMSE is at least as large as the discrete RMSE. When the difference between these two is zero then the inferred BNs are able to recover the discrete trajectories without any error. If the model RMSE is greater than the discrete RMSE then the inferred BNs have some errors in the recoverability of the discrete time series data. To check how our method performs in case of random time series, we have calculated the *RMSE* score for random data and compared it with learning and testing data. Next, the validity of these networks is verified by comparing them with the canonical MTOR signaling pathway using two parameters, *i.e.*, true positive rate (TPR) and false positive rate (FPR).

**Validation using root mean square error criteria.** The goal was not only to infer optimal BNs but also to verify that these BNs are able to recover trajectories that do not exist in the learning data. For this purpose, we use experimental testing data to check the specificity of the trajectories of the proposed networks. This testing data is provided by the HPN-DREAM challenge organizers (see Materials and methods). Table 2 shows the corresponding RMSE in case of learning and testing data. It can be seen that the inferred BNs are able to produce the trajectories without any error in the learning dataset for all cell lines. It is encouraging to see that the inferred BNs are able to recover the discrete testing trajectories without any error in MCF7, and with a minimal error of 0.0009, 0.0106, and 0.0094 in BT20, BT549, and UACC812, respectively.

We also compared the RMSE score with the top two best performers of the HPN-DREAM challenge. We got the top position with an RMSE score of 0.31 as compared to their RMSE

**Table 2. Root mean square error.** This table summarizes the RMSE results for each cell line. We have calculated the discrete RMSE (error related to the discretization of the data) and the model RMSE (*caspo-ts* error). The Delta column shows the difference between model and discrete RMSE.

| Cell Line | Learning | | | Testing | | |
|---|---|---|---|---|---|---|
| | Discrete | Model | Delta | Discrete | Model | Delta |
| *BT20* | 0.3464 | 0.3464 | 0 | 0.3293 | 0.3302 | 0.0009 |
| *BT549* | 0.3498 | 0.3498 | 0 | 0.3007 | 0.3113 | 0.0106 |
| *MCF7* | 0.3207 | 0.3207 | 0 | 0.2772 | 0.2772 | 0 |
| *UACC812* | 0.3464 | 0.3464 | 0 | 0.3084 | 0.3178 | 0.0094 |

https://doi.org/10.1371/journal.pcbi.1006538.t002

scores of 0.47 and 0.50. Notice that in comparison to other HPN-DREAM challenge methods based on Bayesian inference, Regression, and Granger Causality among others, *caspo-ts* does not make new predictions but it checks the recoverability of the testing trajectories with the inferred BNs.

**Validation using random data samples.** The objective of this analysis is to show that the BNs obtained with *caspo-ts* using the HPN-DREAM datasets for the four cell lines have a worse RMSE score with respect to random trajectories, and therefore are very specific to the HPN-DREAM datasets. For this purpose, we generated 100 random data samples per cell line. In each sample, we generated a random value in [0; 1] for each readout protein in each time point without changing the perturbations. Then, we calculated the RMSE of these samples with respect to the inferred BNs of each cell line, and finally compared it with the learning and testing RMSE of these BNs. Fig 6 plots the RMSE ratio (see Eq (2)) of the inferred BNs with respect to the learning, testing and random data.

$$RMSE\ ratio = \frac{Discrete\ RMSE}{Model\ RMSE} \tag{2}$$

In Fig 6, the RMSE ratio for random datasets is displayed by red boxplots for each cell line, and the RMSE ratio for testing and learning datasets is shown as clear outliers in green and

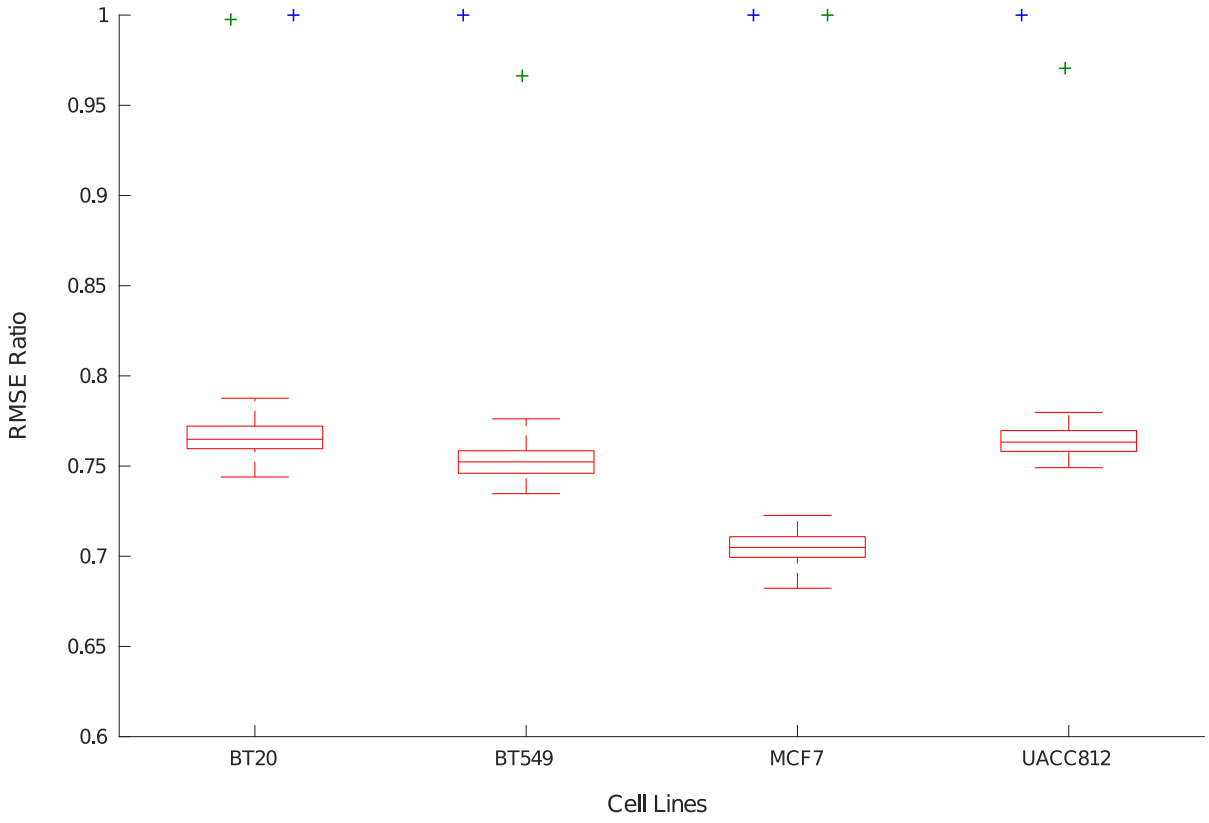



**Fig 6. Performance assessment with learning, testing and random datasets.** The x-axis shows the cell line and the y-axis shows the RMSE ratio (see Eq (2)) of the inferred BNs from the HPN-DREAM data for each cell line with respect to the three datasets. The three datasets are encoded by different color codes. The RMSE ratio with respect to the HPN-DREAM learning and testing datasets is shown in blue and green colors, respectively. The random dataset RMSE ratio distribution is shown as red boxplots.

https://doi.org/10.1371/journal.pcbi.1006538.g006

blue colors respectively. It is worth noting that the *caspo-ts* method has failed to recover random data time series, hence proving the specificity of the learned networks with respect to the HPN-DREAM challenge dataset.

Additionally, we computed the RMSE of the testing data by using a *leave one out* approach. For this we generated slightly modified samples, by selecting random values of 5% of the learning data points. The same experimental perturbations and readout proteins were kept. Our results show that the BNs learned from the 5% randomized data have an RMSE of 0.3113 with respect to the testing data, demonstrating *caspo-ts* robustness. For such 5% modified datasets, true positive BNs are difficult to obtain with the model checker; most candidates were false positive models. This highlights the complexity of this BN learning problem when few experimental perturbations are given because the space of candidate ASP-optimal BNs to verify is large and it is heavily populated with false positive Boolean models. Please refer to the supplementary information for details S2 Text.

**Validation using MTOR canonical pathway.** To perform the validation of the structure of the BNs, we calculated a set of *standard nodes* from our PKN which are downstream nodes of MTOR and belong to the canonical MTOR pathway. We then evaluated how many of these standard nodes are also downstream nodes of MTOR in the learned BNs. In the following, the set of downstream nodes of MTOR in the learned BNs is referred to as *inferred set*. The *inferred set* is specific to each cell line. True positive rate (TPR) and false positive rate (FPR) are defined by Eqs (3) and (4) respectively:

$$TPR = \frac{TP}{TP + FN} \tag{3}$$

$$FPR = \frac{FP}{FP + TN} \tag{4}$$

Here, TP is the number of nodes in the intersection between standard and inferred sets, FP is the number of nodes in the inferred set but not in the standard set, FN is the number of nodes in the standard set but not in the inferred set and TN is the number of nodes which are not in the standard set nor the inferred set. Note that TP and FP should not be confused with true and false positives from the over-approximation here.

Fig 7 shows the Receiver Operating Characteristic (ROC) curve of each cell line. For BNs of each cell line, TPR and FPR was calculated using Eqs (3) and (4). BT549 cell line models are the most accurate ones followed by MCF7 and BT20. We can observe the clear distinction between true positive and false positive BNs. The BNs inferred by *caspo-ts* have an average AUROC score of 0.77 which is comparable to the AUROC score of 0.78 of the top performing method of HPN-DREAM challenge. A number of assumptions made during the modeling phase may have influenced our ranking. First, since our method can pinpoint the noisy, incomplete and erroneous experiment, it allows us to use only the reliable experimental settings. Second, our method constrains its solutions space to the proteins existing in the PKN, anything outside the prior knowledge cannot be found. From Fig 7, we can see that the *caspo-ts* method shows promising results for the inferred true positive BNs.

## Discussion

In this paper, we built cell line specific signaling networks for the DREAM time series dataset of 4 breast cancer cell lines (BT20, BT549, MCF7, and UACC812) using *caspo-ts*. This method combines Answer Set Programming and Model Checking techniques to infer true positive BNs verifying the experimental data. *Caspo-ts* allowed us to handle a midscale PKN (64 nodes

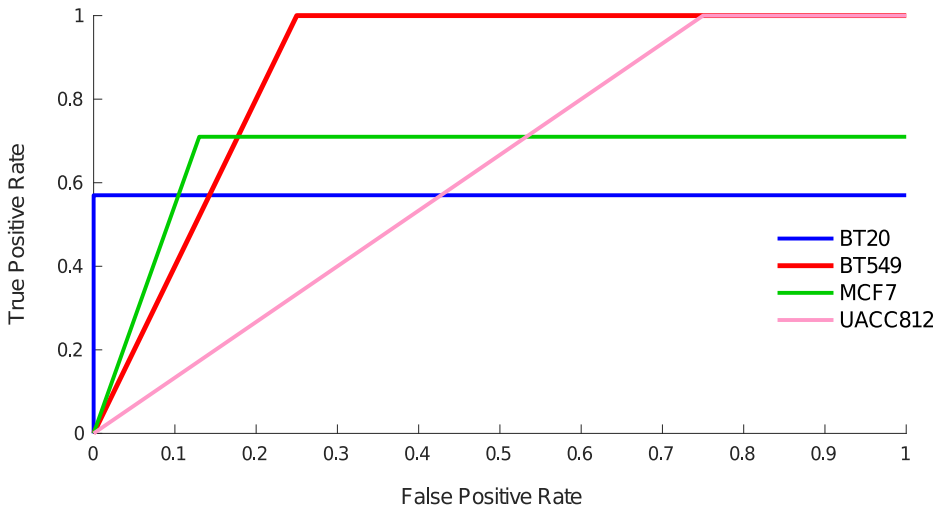



**Fig 7. ROC curve across all cell lines.** The x-axis shows the false positive rate and the y-axis denotes the true positive rate. These rates are calculated using Eqs (3) and (4). The average AUROC score is 0.77.

https://doi.org/10.1371/journal.pcbi.1006538.g007

and 178 edges) and a real dataset subject to experimental error. *Caspo-ts* enabled us to learn key dynamic mechanisms within the BNs explaining the time series data. Our results suggest that the behavior of cell line specific signaling networks is highly variable even under the same perturbations, agreeing with the heterogeneity of breast cancer and specifically with previous analysis on this data [21]. The inferred Boolean models of each cell line were analyzed to identify commonalities as well as discrepancies. Moreover, these inferred models can be executed computationally to identify potential drug targets or to see the effect of unseen perturbations. The predictive power of these models can be increased with improvements in protein interaction databases and comprehensive experimental data.

We have discovered 38% of the cell line dependent behaviors as compared to the 33% of the HPN-DREAM challenge winner [31]. We have implemented an algorithm to analyze the variability among cell lines and observed pairwise similarities among these cell lines. The similarity index varies from 15% (BT20 & BT549) to 23% (MCF7 & BT549, BT20 & MCF7). We have analyzed the similarity among the family of BNs of the same cell line as well, which varies from 43% to 73%. We have evaluated the accuracy of our method with RMSE and AUROC scores. The average RMSE of the inferred BNs was 0.31 placing *caspo-ts* in first place in comparison with the top scores reported in the HPN-DREAM challenge. Various choices made during this study may have an impact on the final score. The *caspo-ts* method allowed us to remove noisy and faulty experiments, leaving us with the reliable experimental settings only. Here, we made the choice to use only the reliable experiments of the learning dataset instead of using all experimental settings. Also, we did not observe all 45 proteins as we could not find connections in our PKN for all the studied proteins, leaving us with approximately 23 proteins for each cell line.

Nonetheless, the obtained results are quite promising, making *caspo-ts* a good candidate computational method for learning models given time series datasets and a prior knowledge network. In addition, *caspo-ts* can be used to pinpoint the errors in the experimental data. In particular, we discovered four experiments where the protein *AKT* was inhibited and had a dynamic behavior as a readout protein. Our work therefore provides a novel approach to show erroneous experiments which is crucial and complementary to current approaches. Finally, the HPN-DREAM dataset contained some noisy readings of experiments. Noisy experimental

data reduces the efficiency of computational methods by increasing the variability among constructed Boolean models. To overcome this, we suggest to build automated methods to filter out the noisy experiments. This approach provides a step forward in building context dependent networks in the case of phosphoproteomic data.

### Perspective

As a future direction, we are planning to investigate several aspects of the *caspo-ts* method, such as (i) the order of the solution space of over-approximated Boolean models; (ii) the computational time for checking reachability; (iii) designing an efficient experimental design strategy and applying it prior to selecting the most informative experiments. Because *caspo-ts* uses an ASP solver to enumerate BNs, in the resulting sequence of solutions similar BNs are typically clustered together. This can be problematic for large scale problems where we cannot explore the whole solution space in reasonable time. We are currently working on sampling to randomly select BNs from the solution space. Further, we are also studying another technique, which allows for shuffling the order in which solutions are enumerated [32]. We are planning to implement this by dynamically modifying the heuristic of the ASP solver at execution time. Finally, to reduce the false positive rate, we are planning to use multi-shot ASP solving [33] allowing us to customize the search and modify the underlying ASP program at runtime. In our case, we can call the model checker during solving to learn and add constraints to prune wrong BNs early on.

## Materials and methods

### Data acquisition

The DREAM portal provides unrestricted access to complex, pre-tested data to encourage the development of computational methods. In this study, we are focused on the HPN-DREAM challenge, which was motivated by the fact that the same perturbation may lead to different signaling behaviors in different backgrounds, making it necessary to build a model which can perform unseen predictions (absent from the learning data). The main goal of the HPN-DREAM challenge is to learn signaling networks efficiently and effectively to predict the dynamics of breast cancer [19].

**Learning data.** Reverse Phase Protein Array (RPPA) quantitative proteomics technology was used for generating the dataset of this challenge. The measurements focus on short term changes on up to 45 proteins and their phosphorylation over 0 to 4 hours. The HPN-DREAM dataset includes temporal changes in phosphorylated proteins at seven different time points ($t_1$ = 0min, $t_2$ = 5min, $t_3$ = 15min, $t_4$ = 30min, $t_5$ = 60min, $t_6$ = 120min, $t_7$ = 240 min). The learning data consists of four cancer cell lines (BT20, BT549, MCF7 and UACC812) under different perturbations (≈8 stimuli and ≈3 inhibitors). The number of perturbations varies from 24 to 32 depending on the cell line. In each cancer cell line approximately 45 phosphorylated proteins are measured against different sets of perturbations over multiple time scales. After removing perturbations with inconsistent behaviors or incomplete time series, we had 15, 13, 13 and 18 perturbations for MCF7, BT20, BT549 and UACC812 cell lines respectively measuring 23 readouts.

**Testing data.** Test data is available for assessing the performance of networks learned from the learning data. The HPN-DREAM portal provides testing data for four cancer cell lines (BT20, BT549, MCF7 and UACC812) under different perturbations (8 stimuli and 1 inhibitor). They contain gold standard datasets of time series predictions of up to 45 proteins having the same time scale as learning data [19–21]. The number of perturbations varies

from 7 to 8 depending on the cell line. This data is used to test the quality of the BNs given by *caspo-ts*.

**Normalization.** The protein measurements were ranging over variable ranges. Maximum value based normalization was used to set the measurements between a common scale, *i.e.*, 0 and 1 in order to assign activation or inactivation values to variables or species of the BN. Eq (5) describes the formula used for the normalization. Given time series $T^P$, we obtain time series $T'^P$:

$$(t'^P_j)_i = \frac{(t^P_j)_i}{\max\{(t^Q_l)_i \mid Q \in \mathcal{P}, 1 \leq l \leq k\}} \tag{5}$$

where $i \in \{1, \ldots, m\}$ are the observations, $j \in \{1, \ldots, k\}$ are time-points, and $P \in \mathcal{P}$ are the perturbations. Here $(t^P_j)_i$ represents the value of protein $i$ under perturbation $P$ at time-point $j$ and the denominator denotes the highest value of protein $i$ under all perturbations and time-points.

## Prior knowledge network derivation

PKNs are available in different databases such as Reactome, PID, and KEGG among others [26, 34–45]. We can construct a PKN through a tool such as ReactomeFIViz [25] which is available as a Cytoscape [46] plugin. A PKN alone cannot be used to build reliable dynamical models or to explain underlying biological behaviors [16, 47], especially in the case of multiple perturbations data because of the need of specificity. In order to overcome this issue, methods have been proposed which take into account both literature based knowledge and experimental data to build logic models [3, 7, 11, 12, 16].

## Learning Boolean Networks with *caspo-ts*

In the *caspo-ts* modeling framework section, we have given the formal definitions of the inputs and the output (BN) of the *caspo-ts*. Here, we formally describe the over-approximation criteria. Finally. we give pseudo encodings of the input of the ASP part of the *caspo-ts*.

**Over-approximation criteria.** The goal is to generate BNs that can reproduce the experimental data as well as possible. For this objective, the states have to be reachable from another. We use $x \rightarrow^* y$ to say that state $y$ can be reached from state $x$ with an arbitrary number of steps. Since this reachability is a computationally hard problem (PSPACE-complete) [48], we use an over-approximation for checking reachability resulting in false positive (FP) BNs [7, 8]. The meta-states have been introduced to check over-approximated reachability.

A meta-state $u = (u_1, u_2, \ldots, u_n)$ is a vector of dimension $n$ over non-empty subsets of $\mathbb{B}$, noted $\mathbb{M} = \{\{0\}, \{1\}, \{0, 1\}\}$; the set of meta-states is $\mathbb{M}^n$. Meta-states characterize a set of Boolean states: a state $x \in \mathbb{B}^n$ belongs to a meta-state $u$, written $x \in u$, iff each Boolean component $x_i$ belongs to the set $u_i$. Given a state $x$, we use $\bar{x}$ for the corresponding meta-state $(\{x_1\}, \ldots, \{x_n\})$. We define the transition relation $u \rightrightarrows v$ between the meta-states $u$ and $v$ as follows: $u \neq v$ and $v = (u_1, \ldots, u_i \cup \{f_i(x) \mid x \in u\}, \ldots, u_n)$ for some $1 \leq i \leq n$.

In [8], it has been shown that if $y$ is reachable from $x$ ($x \rightarrow^* y$) then there exists a meta-state $u$ such that $y \in u$ and $\bar{x} \rightrightarrows^* u$. This definition is further refined to describe the necessary condition for reachability called support consistency. A state $x$ is support consistent with state $y$ denoted by $x \twoheadrightarrow^* y$, if and only if there exists a meta-state $u$ with $\bar{x} \rightrightarrows^* u$ such that $y \in u$ and for all $1 \leq i \leq n$ either

- $y_i \neq x_i$, or
- $y_i = x_i$ and $u_i \neq \{0, 1\}$, or

- $y_i = x_i$, $u_i = \{0, 1\}$, and there exists $z \in u$ such that $f_i(z) = y_i$.

If state $y$ is reachable from state $x$ ($x \rightarrow^* y$) then $x \rightsquigarrow^* y$. Since we are using the over-approximation criteria, it is possible that some BNs may fail to reproduce the exact trajectories of the time series data. These BNs are called false positive (FP). To filter out the false positive BNs, exact model checking is applied.

**Input encodings.** Here, we provide the pseudo logic program to describe the input data given in the *caspo-ts* modeling framework Section. The logic program in written in the ASP language. ASP is a powerful declarative logic programming language for knowledge representation and reasoning [49]. The basic idea is to encode the problem using a non-monotonic logic program and then feed it into the ASP solver, which computes the solution of the problem in the form of models (also known as answer sets). Note that we provide encodings only for the input data here, please refer to supplementary information for details (S1 Text).

Facts, rules and constraints are the building blocks of ASP programs. Here we use facts to describe the inputs. The PKN $(V, E, \sigma)$ is described by the following facts:

$$node(v). \text{ for } v \in V$$
$$edge(u, v, s). \text{ for } (u, v) \in E \text{ and } ((u, v), s) \in \sigma$$

For each perturbation $P \in \mathcal{P}$ and phosphoproteomic time series $T^P$, we have the following facts:

$$clamped(P, v, 0). \text{ for } v \in P \cap I$$
$$clamped(P, v, 1). \text{ for } v \in P \cap S$$
$$obs(P, j, v_i, s). \text{ for } s = (t_j^P)_i, 1 \leq j \leq k \text{ and } 1 \leq i \leq m$$

**Available software.** The *caspo-ts* github repository contains the sources as well as detailed user guide with two examples at the following address: https://github.com/misbahch6/caspo-ts.

## Graph similarity measure

This work introduces the study of a graph similarity measure in order to check the variability among the families of BNs generated by *caspo-ts*. We compare the reactions existing in the gold standard network ($A$) with the family of BNs ($\mathcal{B}$) and is based on the Jaccard similarity coefficient which measures the similarity of these models.

**Jaccard similarity coefficient.** The Jaccard index between $A$ and $B_i$ can be defined as length of the intersection divided by the union:

$$J(A, B_i) = \frac{|A \cap B_i|}{|A \cup B_i|} = \frac{|A \cap B_i|}{|A| + |B_i| - |A \cap B_i|} \quad (6)$$

We apply the Jaccard Similarity Coefficient on $B_i$ (where $B_i \subset \mathcal{B}$) by taking A as being the gold standard.

## Supporting information

**S1 Fig. Union of BNs of BT20.** Here, we show the union of BNs for the cell line BT20. This network is generated by combining 72 true positive BNs. It contains 31 nodes and 41 boolean functions with 12 AND gates. There are 2 stimuli, 2 inhibitors and 21 readouts.
(PDF)

**S2 Fig. Union of BNs of BT549.** Here, we show the union of BNs for the cell line BT549. This network is generated by combining 191 true positive BNs. It contains 28 nodes and 53 boolean functions with 35 AND gates. There are 5 stimuli, 2 inhibitors and 17 readouts.
(PDF)

**S3 Fig. Union of BNs of MCF7.** Here, we show the union of BNs for the cell line MCF7. This network is generated by combining 21 true positive BNs. It contains 24 nodes and 37 boolean functions with 19 AND gates. There are 4 stimuli, 2 inhibitors and 15 readouts.
(PDF)

**S4 Fig. Union of BNs of UACC812.** Here, we show the union of BNs for the cell line UACC812. This network is generated by combining 20 BNs. It contains 33 nodes and 54 boolean functions with 29 AND gates. There are 6 stimuli, 2 inhibitors and 18 readouts.
(PDF)

**S1 Table. Computation summary.** Here, we show the number of verified solutions, true positive and false positive BNs, and their computation (ASP solving and Model Checking steps) time for each cell line. It is worth noting that we generated 32 true positive BNs for UACC812 cell line by allowing the model checker to run without bounding it to the 7 day time limit. The ASP solving was performed on a standard laptop machine. The model checking task was performed on a cluster with 560 cores and 1.9 Tb of RAM.
(PDF)

**S1 Text. ASP encodings.**
(PDF)

**S2 Text. Validation by introducing noise in the learning data.**
(PDF)

## Author Contributions

**Conceptualization:** Carito Guziolowski.

**Data curation:** Misbah Razzaq.

**Formal analysis:** Misbah Razzaq, Loïc Paulevé, Anne Siegel, Carito Guziolowski.

**Funding acquisition:** Jérémie Bourdon, Carito Guziolowski.

**Investigation:** Misbah Razzaq, Julio Saez-Rodriguez, Jérémie Bourdon, Carito Guziolowski.

**Methodology:** Misbah Razzaq.

**Project administration:** Carito Guziolowski.

**Resources:** Julio Saez-Rodriguez.

**Software:** Misbah Razzaq, Jérémie Bourdon, Carito Guziolowski.

**Supervision:** Jérémie Bourdon, Carito Guziolowski.

**Validation:** Misbah Razzaq, Jérémie Bourdon, Carito Guziolowski.

**Visualization:** Misbah Razzaq.

**Writing – original draft:** Misbah Razzaq, Carito Guziolowski.

**Writing – review & editing:** Misbah Razzaq, Loïc Paulevé, Anne Siegel, Julio Saez-Rodriguez, Jérémie Bourdon, Carito Guziolowski.

## References


1. Watterson S, Marshall S, Ghazal P. Logic models of pathway biology. Drug discovery today. 2008; 13 (9):447–456. https://doi.org/10.1016/j.drudis.2008.03.019
2. Samaga R, Klamt S. Modeling approaches for qualitative and semi-quantitative analysis of cellular signaling networks. Cell communication and signaling. 2013; 11(1):43. https://doi.org/10.1186/1478-811X-11-43
3. MacNamara A, Terfve C, Henriques D, Bernabé BP, Saez-Rodriguez J. State–time spectrum of signal transduction logic models. Physical biology. 2012; 9(4):045003.
4. Kauffman SA. The origins of order: Self-organization and selection in evolution. Oxford University Press, USA; 1993.
5. Thomas R. Laws for the dynamics of regulatory networks. International Journal of Developmental Biology. 2002; 42(3):479–485.
6. Wynn ML, Consul N, Merajver SD, Schnell S. Logic-based models in systems biology: a predictive and parameter-free network analysis method. Integrative biology. 2012; 4(11):1323–1337. https://doi.org/10.1039/c2ib20193c
7. Ostrowski M, Paulevé L, Schaub T, Siegel A, Guziolowski C. Boolean Network Identification from Multiplex Time Series Data. In: Computational Methods in Systems Biology. vol. 9308. Springer; 2015. p. 170–181.
8. Ostrowski M, Paulevé L, Schaub T, Siegel A, Guziolowski C. Boolean network identification from perturbation time series data combining dynamics abstraction and logic programming. Biosystems. 2016; 149:139–153. https://doi.org/10.1016/j.biosystems.2016.07.009
9. Almudevar A, McCall MN, McMurray H, Land H. Fitting Boolean networks from steady state perturbation data. Statistical applications in genetics and molecular biology. 2011; 10(1):47. https://doi.org/10.2202/1544-6115.1727
10. Zhu P, Aliabadi HM, Uludağ H, Han J. Identification of Potential Drug Targets in Cancer Signaling Pathways using Stochastic Logical Models. Scientific reports. 2016; 6.
11. Guziolowski C, Videla S, Eduati F, Thiele S, Cokelaer T, Siegel A, et al. Exhaustively characterizing feasible logic models of a signaling network using answer set programming. Bioinformatics. 2013; 29 (18):2320–2326. https://doi.org/10.1093/bioinformatics/btt393
12. Videla S, Guziolowski C, Eduati F, Thiele S, Grabe N, Saez-Rodriguez J, et al. Revisiting the training of logic models of protein signaling networks with ASP. In: Computational Methods in Systems Biology. Springer; 2012. p. 342–361.
13. Sharan R, Karp RM. Reconstructing Boolean models of signaling. Journal of Computational Biology. 2013; 20(3):249–257. https://doi.org/10.1089/cmb.2012.0241
14. Mitsos A, Melas IN, Siminelakis P, Chairakaki AD, Saez-Rodriguez J, Alexopoulos LG. Identifying drug effects via pathway alterations using an integer linear programming optimization formulation on phosphoproteomic data. PLoS computational biology. 2009; 5(12):e1000591. https://doi.org/10.1371/journal.pcbi.1000591
15. Gebser M, Kaminski R, Kaufmann B, Schaub T. Clingo = ASP+ control: Preliminary report. arXiv preprint arXiv:14053694. 2014;.
16. Saez-Rodriguez J, Alexopoulos LG, Epperlein J, Samaga R, Lauffenburger DA, Klamt S, et al. Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction. Molecular systems biology. 2009; 5(1).
17. Terfve C, Cokelaer T, Henriques D, MacNamara A, Goncalves E, Morris MK, et al. CellNOptR: a flexible toolkit to train protein signaling networks to data using multiple logic formalisms. BMC systems biology. 2012; 6(1):133. https://doi.org/10.1186/1752-0509-6-133
18. Dorier J, Crespo I, Niknejad A, Liechti R, Ebeling M, Xenarios I. Boolean regulatory network reconstruction using literature based knowledge with a genetic algorithm optimization method. BMC bioinformatics. 2016; 17(1):410. https://doi.org/10.1186/s12859-016-1287-z
19. Heiser L. HPN-DREAM breast cancer network inference challenge; 2016. Available from: https://www.synapse.org/#!Synapse:syn1720047/wiki/55342.
20. Hill SM, Heiser LM, Cokelaer T, Unger M, Nesser NK, Carlin DE, et al. Inferring causal molecular networks: empirical assessment through a community-based effort. Nature methods. 2016; 13(4):310–318. https://doi.org/10.1038/nmeth.3773
21. Hill SM, Nesser NK, Johnson-Camacho K, Jeffress M, Johnson A, Boniface C, et al. Context specificity in causal signaling networks revealed by phosphoprotein profiling. Cell systems. 2017; 4(1):73–83. https://doi.org/10.1016/j.cels.2016.11.013

22. Kauffman SA. Metabolic stability and epigenesis in randomly constructed genetic nets. Journal of theoretical biology. 1969; 22(3):437–467. https://doi.org/10.1016/0022-5193(69)90015-0
23. Inoue K. Logic Programming for Boolean Networks. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence—Volume Volume Two. vol. 22 of IJCAI'11. AAAI Press; 2011. p. 924–930.
24. Cimatti A, Clarke E, Giunchiglia E, Giunchiglia F, Pistore M, Roveri M, et al. Nusmv 2: An opensource tool for symbolic model checking. In: International Conference on Computer Aided Verification. Springer; 2002. p. 359–364.
25. Wu G, Dawson E, Duong A, Haw R, Stein L. ReactomeFIViz: a Cytoscape app for pathway and network-based data analysis. F1000Research. 2014; 3. https://doi.org/10.12688/f1000research.4431.2
26. Wu G, Dawson E, Duong A, Haw R, Stein L. ReactomeFIViz: a Cytoscape app for pathway and network-based data analysis. F1000Research. 2014; 3. https://doi.org/10.12688/f1000research.4431.2
27. Abboud A, Grandoni F, Williams VV. Subcubic equivalences between graph centrality problems, APSP and diameter. In: Proceedings of the twenty-sixth annual ACM-SIAM symposium on Discrete algorithms. SIAM; 2014. p. 1681–1697.
28. Shekar SC, Wu H, Fu Z, Yip SC, Cahill SM, Girvin ME, et al. Mechanism of constitutive phosphoinositide 3-kinase activation by oncogenic mutants of the p85 regulatory subunit. Journal of Biological Chemistry. 2005; 280(30):27850–27855. https://doi.org/10.1074/jbc.M506005200
29. Taniguchi CM, Winnay J, Kondo T, Bronson RT, Guimaraes AR, Alemán JO, et al. The phosphoinositide 3-kinase regulatory subunit p85α can exert tumor suppressor properties through negative regulation of growth factor signaling. Cancer research. 2010; 70(13):5305–5315. https://doi.org/10.1158/0008-5472.CAN-09-3399
30. Network CGA, et al. Comprehensive molecular portraits of human breast tumours. Nature 490, 61e70; 2012.
31. Carlin DE. Computational evaluation and derivation of biological networks in cancer and stem cells. University of California, Santa Cruz; 2014.
32. Romero J, Schaub T, Wanko P. Computing Diverse Optimal Stable Models. In: ICLP (Technical Communications). vol. 52 of OASICS. Schloss Dagstuhl—Leibniz-Zentrum fuer Informatik; 2016. p. 3:1–3:14.
33. Kaminski R, Schaub T, Wanko P. A tutorial on hybrid answer set solving with clingo. In: Reasoning Web International Summer School. Springer; 2017. p. 167–203.
34. Duan G, Walther D. The roles of post-translational modifications in the context of protein interaction networks. PLoS Comput Biol. 2015; 11(2):e1004049. https://doi.org/10.1371/journal.pcbi.1004049
35. Kanehisa M, Goto S. KEGG: kyoto encyclopedia of genes and genomes. Nucleic acids research. 2000; 28(1):27–30. https://doi.org/10.1093/nar/28.1.27
36. Consortium GO, et al. The Gene Ontology (GO) database and informatics resource. Nucleic acids research. 2004; 32(suppl 1):D258–D261. https://doi.org/10.1093/nar/gkh036
37. Kelder T, van Iersel MP, Hanspers K, Kutmon M, Conklin BR, Evelo CT, et al. WikiPathways: building research communities on biological pathways. Nucleic Acids Research. 2012; 40(D1):D1301. https://doi.org/10.1093/nar/gkr1074
38. Nishimura D. BioCarta. Biotech Software & Internet Report: The Computer Software Journal for Scient. 2001; 2(3):117–120. https://doi.org/10.1089/152791601750294344
39. Stark C, Breitkreutz BJ, Reguly T, Boucher L, Breitkreutz A, Tyers M. BioGRID: a general repository for interaction datasets. Nucleic acids research. 2006; 34(suppl_1):D535–D539. https://doi.org/10.1093/nar/gkj109
40. Szklarczyk D, Morris JH, Cook H, Kuhn M, Wyder S, Simonovic M, et al. The STRING database in 2017: quality-controlled protein–protein association networks, made broadly accessible. Nucleic acids research. 2017; 45(D1):D362–D368. https://doi.org/10.1093/nar/gkw937
41. Xenarios I, Rice DW, Salwinski L, Baron MK, Marcotte EM, Eisenberg D. DIP: the database of interacting proteins. Nucleic acids research. 2000; 28(1):289–291. https://doi.org/10.1093/nar/28.1.289
42. Peri S, Navarro JD, Kristiansen TZ, Amanchy R, Surendranath V, Muthusamy B, et al. Human protein reference database as a discovery resource for proteomics. Nucleic acids research. 2004; 32(suppl_1):D497–D501. https://doi.org/10.1093/nar/gkh070
43. Hermjakob H, Montecchi-Palazzi L, Lewington C, Mudali S, Kerrien S, Orchard S, et al. IntAct: an open source molecular interaction database. Nucleic acids research. 2004; 32(suppl_1):D452–D455. https://doi.org/10.1093/nar/gkh052
44. Zanzoni A, Montecchi-Palazzi L, Quondam M, Ausiello G, Helmer-Citterich M, Cesareni G. MINT: a Molecular INTeraction database. FEBS letters. 2002; 513(1):135–140. https://doi.org/10.1016/S0014-5793(01)03293-8

45. Razick S, Magklaras G, Donaldson IM. iRefIndex: a consolidated protein interaction database with provenance. BMC bioinformatics. 2008; 9(1):405. https://doi.org/10.1186/1471-2105-9-405

46. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, et al. Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome research. 2003; 13 (11):2498–2504. https://doi.org/10.1101/gr.1239303

47. Rodriguez A, Crespo I, Androsova G, del Sol A. Discrete Logic Modelling Optimization to Contextualize Prior Knowledge Networks Using PRUNET. PloS one. 2015; 10(6):e0127216. https://doi.org/10.1371/journal.pone.0127216

48. Cheng A, Esparza J, Palsberg J. Complexity results for 1-safe nets. Theoretical Computer Science. 1995; 147(1):117—136. https://doi.org/10.1016/0304-3975(94)00231-7

49. Lifschitz V. What Is Answer Set Programming? In: AAAI. AAAI Press; 2008. p. 1594–1597.

# BIBLIOGRAPHY


[1] A. Martinez-Antonio, S. C. Janga, H. Salgado, and J. Collado-Vides, "Internal-sensing machinery directs the activity of the regulatory network in Escherichia coli," *Trends Microbiol*, vol. 14, pp. 22–27, Jan 2006. 10

[2] H. Osaki, A. Sasaki, E. Nakazono-Nagaoka, N. Ota, and R. Nakaune, "Genome segments encoding capsid protein-like variants of Pyrus pyrifolia cryptic virus," *Virus Res*, vol. 240, pp. 64–68, 08 2017. 11

[3] G. S. Pall and A. J. Hamilton, "Improved northern blot method for enhanced detection of small RNA," *Nat Protoc*, vol. 3, no. 6, pp. 1077–1084, 2008. 11

[4] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown, "Quantitative monitoring of gene expression patterns with a complementary DNA microarray," *Science*, vol. 270, pp. 467–470, Oct 1995. 11

[5] D. J. Lockhart and E. A. Winzeler, "Genomics, gene expression and DNA arrays," *Nature*, vol. 405, pp. 827–836, Jun 2000. 11

[6] T. Lenoir and E. Giannella, "The emergence and diffusion of DNA microarray technology," *J Biomed Discov Collab*, vol. 1, p. 11, Aug 2006. 11

[7] C. M. Perou, S. S. Jeffrey, M. van de Rijn, C. A. Rees, M. B. Eisen, D. T. Ross, A. Pergamenschikov, C. F. Williams, S. X. Zhu, J. C. Lee, D. Lashkari, D. Shalon, P. O. Brown, and D. Botstein, "Distinctive gene expression patterns in human mammary epithelial cells and breast cancers," *Proc Natl Acad Sci U S A*, vol. 96, pp. 9212–9217, Aug 1999. 11

[8] J. DeRisi, L. Penland, P. O. Brown, M. L. Bittner, P. S. Meltzer, M. Ray, Y. Chen, Y. A. Su, and J. M. Trent, "Use of a cDNA microarray to analyse gene expression patterns in human cancer," *Nat Genet*, vol. 14, pp. 457–460, Dec 1996. 11

[9] H. Kitano, "Systems biology: a brief overview," *Science*, vol. 295, pp. 1662–1664, Mar 2002. 11

[10] U. Nagalakshmi, Z. Wang, K. Waern, C. Shou, D. Raha, M. Gerstein, and M. Snyder, "The transcriptional landscape of the yeast genome defined by RNA sequencing," *Science*, vol. 320, pp. 1344–1349, Jun 2008. 11

[11] M. Lefebvre, A. Gaignard, M. Folschette, J. Bourdon, and C. Guziolowski, "Large-scale regulatory and signaling network assembly through linked open data," *Database (Oxford)*, vol. 2021, 01 2021. 6, 12, 14, 16, 99, 105

[12] P. Veber, M. L. Borgne, A. Siegel, and O. Radulescu, "Complex qualitative models in biology: A new approach," *Complexus*, vol. 2, pp. 3–4, 2004/2005. 15

[13] C. Guziolowski, A. Bourdé, F. Moreews, and A. Siegel, "BioQuali Cytoscape plugin: analysing the global consistency of regulatory networks," *BMC Genomics*, vol. 10, p. 244, May 2009. 15, 16, 17, 29

[14] M. Gebser, C. Guziołowski, M. Ivanchev, T. Schaub, A. Siegel, and S. Thiele, *Repair and prediction (under inconsistency) in large biological networks with answer set programming*. Menlo Park, CA: AAAI Press, 2010. 15, 17, 25, 68

[15] M. Gebser, T. Schaub, S. Thiele, and P. Veber, "Detecting inconsistencies in large biological networks with answer set programming," *Theory Prac Logic Program*, vol. 11, 2011. 15, 16, 17, 24, 25

[16] S. Thiele, S. Heise, W. Hessenkemper, H. Bongartz, M. Fensky, F. Schaper, and S. Klamt, "Designing Optimal Experiments to Discriminate Interaction Graph Models," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 16, pp. 925–935, May 2019. 15

[17] F. Gouveia, I. Lynce, and P. T. Monteiro, "Revision of Boolean Models of Regulatory Networks Using Stable State Observations," *J Comput Biol*, Dec 2019. 4, 15, 31, 64

[18] N. L. Catlett, A. J. Bargnesi, S. Ungerer, T. Seagaran, W. Ladd, K. O. Elliston, and D. Pratt, "Reverse causal reasoning: applying qualitative causal knowledge to the interpretation of high-throughput data," *BMC Bioinformatics*, vol. 14, p. 340, Nov 2013. 16, 34

[19] S. Klamt, J. SaezRodriguez, J. A. Lindquist, L. Simeoni, and E. D. Gilles, "A methodology for the structural and functional analysis of signaling and regulatory networks," *BMC Bioinformatics*, vol. 7, p. 56, Feb 2006. 16, 66, 67

[20] R. Samaga, J. Saez-Rodriguez, L. G. Alexopoulos, P. K. Sorger, and S. Klamt, "The logic of EGFR/ErbB signaling: theoretical properties and analysis of high-throughput data," *PLoS Comput Biol*, vol. 5, p. e1000438, Aug 2009. 16

[21] D. Thieffry, "Dynamical roles of biological regulatory circuits," *Brief Bioinform*, vol. 8, pp. 220–225, Jul 2007. 16

[22] K. Sachs, O. Perez, D. Pe'er, D. A. Lauffenburger, and G. P. Nolan, "Causal protein-signaling networks derived from multiparameter single-cell data," *Science*, vol. 308, pp. 523–529, Apr 2005. 16

[23] M. K. Morris, J. Saez-Rodriguez, P. K. Sorger, and D. A. Lauffenburger, "Logic-based models for the analysis of cell signaling networks," *Biochemistry*, vol. 49, pp. 3216–3224, Apr 2010. 16

[24] R. S. Wang, A. Saadatpour, and R. Albert, "Boolean modeling in systems biology: an overview of methodology and applications," *Phys Biol*, vol. 9, p. 055001, Oct 2012. 16

[25] B. Schoeberl, C. Eichler-Jonsson, E. D. Gilles, and G. Müller, "Computational modeling of the dynamics of the MAP kinase cascade activated by surface and internalized EGF receptors," *Nat Biotechnol*, vol. 20, pp. 370–375, Apr 2002. 16

[26] M. Quach, N. Brunel, and F. d'Alché Buc, "Estimating parameters and hidden variables in non-linear state-space models based on ODEs for biological networks inference," *Bioinformatics*, vol. 23, pp. 3209–3216, Dec 2007. 16

[27] G. Karlebach and R. Shamir, "Modelling and analysis of gene regulatory networks," *Nat Rev Mol Cell Biol*, vol. 9, pp. 770–780, Oct 2008. 16

[28] T. E. Ideker, V. Thorsson, and R. M. Karp, *Discovery of Regulatory Interactions Through Perturbation: Inference and Experimental Design.* Seattle, USA: World Scientific Press, 2000. 16

[29] J. Saez-Rodriguez, L. G. Alexopoulos, J. Epperlein, R. Samaga, D. A. Lauffenburger, and S. Klamt, "Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction," *Mol Syst Biol*, vol. 5, 2009. 16, 36, 66, 67, 70, 71, 74

[30] R. Sharan and R. M. Karp, "Reconstructing boolean models of signaling," in *Research in Computational Molecular Biology* (B. Chor, ed.), (Berlin, Heidelberg), pp. 261–271, Springer Berlin Heidelberg, 2012. 16

[31] C. Terfve, T. Cokelaer, D. Henriques, A. MacNamara, E. Goncalves, and M. Morris, "Cellnoptr: a flexible toolkit to train protein signaling networks to data using multiple logic formalisms," *BMC Syst Biol*, vol. 6, 2012. 16

[32] I. N. Melas, R. Samaga, L. G. Alexopoulos, and S. Klamt, "Detecting and removing inconsistencies between experimental data and signaling network topologies using

integer linear programming on interaction graphs," *PLoS Comput Biol*, vol. 9, 2013. 16, 17, 24

[33] S. Videla, C. Guziolowski, F. Eduati, S. Thiele, M. Gebser, J. Nicolas, J. Saez-Rodriguez, T. Schaub, and A. Siegel, "Learning boolean logic models of signaling networks with asp," *Theoretical Computer Science*, vol. 599, pp. 79–101, 2015. Advances in Computational Methods in Systems Biology. 4, 16, 64, 76, 100

[34] N. Radde, N. S. Bar, and M. Banaji, "Graphical methods for analysing feedback in biological networks - a survey," *Int J Syst Sci*, vol. 41, 2010. 16

[35] R. Samaga and S. Klamt, "Modeling approaches for qualitative and semi-quantitative analysis of cellular signaling networks," *Cell Commun Signal*, vol. 11, 2013. 16

[36] B. Kuipers, "Qualitative reasoning: Modeling and simulation with incomplete knowledge," *Automatica*, vol. 25, 1989. 16

[37] C. Baral, *Knowledge Representation, Reasoning and Declarative Problem Solving*. Cambridge: Cambridge University Press, 2003. 17, 25, 51, 68

[38] S. Thiele, S. Heise, W. Hessenkemper, H. Bongartz, M. Fensky, F. Schaper, and S. Klamt, "Designing optimal experiments to discriminate interaction graph models," *IEEE/ACM Trans Comput Biol Bioinform*, Mar 2018. 4, 17, 64

[39] B. Miannay, S. Minvielle, F. Magrangeas, and C. Guziolowski, "Constraints on signaling network logic reveal functional subgraphs on Multiple Myeloma OMIC data," *BMC Syst Biol*, vol. 12, p. 32, 03 2018. 19, 30, 49, 52, 53, 63, 100

[40] M. Gelfond and Y. Kahl, *Knowledge Representation, Reasoning, and the Design of Intelligent Agents: The Answer-Set Programming Approach*. USA: Cambridge University Press, 2014. 25

[41] M. Gebser, R. Kaminski, B. Kaufmann, and T. Schaub, *Answer Set Solving in Practice*. Synthesis Lectures on Artificial Intelligence and Machine Learning, Morgan & Claypool Publishers, 2012. 25, 68, 79

[42] M. Gelfond and V. Lifschitz, "The stable model semantics for logic programming," pp. 1070–1080, MIT Press, 1988. 26

[43] M. Gebser, B. Kaufmann, and T. Schaub, "Conflict-driven answer set solving: From theory to practice," *Artif. Intell.*, vol. 187, pp. 52–89, 2012. 28, 29

[44] S. Thiele, L. Cerone, J. Saez-Rodriguez, A. Siegel, C. Guziolowski, and S. Klamt, "Extended notions of sign consistency to relate experimental data to signaling and regulatory network topologies," *BMC Bioinformatics*, vol. 16, p. 345, Oct 2015. 29, 52, 100, 104

[45] C. Guziolowski, S. Blachon, T. Baumuratova, G. Stoll, O. Radulescu, and A. Siegel, "Designing logical rules to model the response of biomolecular networks with complex interactions: an application to cancer modeling," *IEEE/ACM Trans Comput Biol Bioinform*, vol. 8, no. 5, pp. 1223–1234, 2011. 29

[46] C. Guziolowski, A. Kittas, F. Dittmann, and N. Grabe, "Automatic generation of causal networks linking growth factor stimuli to functional cell state changes," *FEBS J*, vol. 279, pp. 3462–3474, Sep 2012. 4, 29, 64, 79

[47] B. Miannay, S. Minvielle, O. Roux, P. Drouin, H. Avet-Loiseau, C. Guérin-Charbonnel, W. Gouraud, M. Attal, T. Facon, N. C. Munshi, P. Moreau, L. Campion, F. Magrangeas, and C. Guziolowski, "Logic programming reveals alteration of key transcription factors in multiple myeloma," *Sci Rep*, vol. 7, p. 9257, 08 2017. 2, 5, 29, 32, 63, 98, 104

[48] M. Folschette, V. Legagneux, A. Poret, L. Chebouba, C. Guziolowski, and N. Théret, "A pipeline to create predictive functional networks: application to the tumor progression of hepatocellular carcinoma," *BMC Bioinformatics*, vol. 21, p. 18, Jan 2020. 5, 29, 31, 98

[49] S. L. Bars, J. Bourdon, and C. Guziolowski, "Comparing probabilistic and logic programming approaches to predict the effects of enzymes in a neurodegenerative disease model," in *Computational Methods in Systems Biology - 18th International Conference, CMSB 2020, Konstanz, Germany, September 23-25, 2020, Proceedings* (A. Abate, T. Petrov, and V. Wolf, eds.), vol. 12314 of *Lecture Notes in Computer Science*, pp. 141–156, Springer, 2020. 29

[50] S. L. Bars, J. Bourdon, and C. Guziolowski, "Comparing probabilistic and logic programming approaches to predict the effects of enzymes in a neurodegenerative disease model," in *Computational Methods in Systems Biology - 18th International Conference, CMSB 2020, Konstanz, Germany, September 23-25, 2020, Proceedings* (A. Abate, T. Petrov, and V. Wolf, eds.), vol. 12314 of *Lecture Notes in Computer Science*, pp. 141–156, Springer, 2020. 30

[51] H. Yu and R. H. Blair, "Integration of probabilistic regulatory networks into constraint-based models of metabolism with applications to Alzheimer's disease," *BMC Bioinformatics*, vol. 20, p. 386, Jul 2019. 30

[52] G. J. Morgan, B. A. Walker, and F. E. Davies, "The genetic architecture of multiple myeloma," *Nature reviews. Cancer*, vol. 12, 2012. 33

[53] F. Zhan, "The molecular classification of multiple myeloma," *Blood*, vol. 108, 2006. 33

[54] O. Decaux, "Prediction of survival in multiple myeloma based on gene expression profiles reveals cell cycle and chromosomal instability signatures in high-risk patients and hyperdiploid signatures in low-risk patients: A study of the intergroupe francophone du myélome," *Journal of Clinical Oncology*, vol. 26, 2008. 33

[55] J. D. Shaughnessy, "A validated gene expression model of high-risk multiple myeloma is defined by deregulated expression of genes mapping to chromosome 1," *Blood*, vol. 109, 2007. 33

[56] H. Avet-Loiseau, "Prognostic significance of copy-number alterations in multiple myeloma," *Journal of clinical oncology: official journal of the American Society of Clinical Oncology*, vol. 27, 2009. 33

[57] A. Broyl, "Gene expression profiling for molecular classification of multiple myeloma in newly diagnosed patients," *Blood*, vol. 116, 2010. 33

[58] B. A. Walker, "Mutational spectrum, copy number changes, and outcome: Results of a sequencing study of patients with newly diagnosed myeloma," *Journal of clinical oncology: official journal of the American Society of Clinical Oncology*, vol. 33, 2015. 33

[59] N. U. Rashid, A. S. Sperling, N. Bolli, D. C. Wedge, P. Van Loo, Y. T. Tai, M. A. Shammas, M. Fulciniti, M. K. Samur, P. G. Richardson, F. Magrangeas, S. Minvielle, P. A. Futreal, K. C. Anderson, H. Avet-Loiseau, P. J. Campbell, G. Parmigiani, and N. C. Munshi, "Differential and limited expression of mutant alleles in multiple myeloma," *Blood*, vol. 124, pp. 3110–3117, Nov 2014. 33

[60] M. R. Mansour, "Oncogene regulation. an oncogenic super-enhancer formed through somatic mutation of a noncoding intergenic element," *Science (New York, N.Y.)*, vol. 346, 2014. 33

[61] M. Ashburner, C. A. Ball, J. A. Blake, D. Botstein, H. Butler, J. M. Cherry, A. P. Davis, K. Dolinski, S. S. Dwight, J. T. Eppig, M. A. Harris, D. P. Hill, L. Issel-

Tarver, A. Kasarskis, S. Lewis, J. C. Matese, J. E. Richardson, M. Ringwald, G. M. Rubin, and G. Sherlock, "Gene ontology: tool for the unification of biology. The Gene Ontology Consortium," *Nat Genet*, vol. 25, pp. 25–29, May 2000. 34

[62] M. Kanehisa and S. Goto, "Kegg: kyoto encyclopedia of genes and genomes," *Nucleic Acids Res*, vol. 28, 2000. 34

[63] C. F. Schaefer, K. Anthony, S. Krupa, J. Buchoff, M. Day, T. Hannay, and K. H. Buetow, "Pid: the pathway interaction database," *Nucleic acids research*, vol. 37, 2009. 34, 35

[64] T. Kelder, "Wikipathways: building research communities on biological pathways," *Nucleic acids research*, vol. 40, 2012. 34

[65] E. Wingender, "The transfac project as an example of framework technology that supports the analysis of genomic regulation," *Briefings in Bioinformatics*, vol. 9, 2008. 34

[66] S. Boué, "Causal biological network database: a comprehensive platform of causal biological network models focused on the pulmonary and vascular systems," *Database: the journal of biological databases and curation*, vol. 2015, 2015. 34

[67] P. Khatri, M. Sirota, and A. J. Butte, "Ten years of pathway analysis: current approaches and outstanding challenges," *PLoS computational biology*, vol. 8, 2012. 34

[68] G. Bindea, "Cluego: a cytoscape plug-in to decipher functionally grouped gene ontology and pathway annotation networks," *Bioinformatics (Oxford, England)*, vol. 25, 2009. 34

[69] F. Martin, "Quantification of biological network perturbations for mechanistic insight and diagnostics using two-layer causal models," *BMC bioinformatics*, vol. 15, 2014. 34

[70] A. Subramanian, "Gene set enrichment analysis: a knowledge-based approach for interpreting genome-wide expression profiles," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 102, 2005. 34

[71] C. Backes, "Genetrail–advanced gene set enrichment analysis," *Nucleic acids research*, vol. 35, 2007. 34

[72] C. Lefebvre, "A human b-cell interactome identifies myb and foxm1 as master regulators of proliferation in germinal centers," *Molecular systems biology*, vol. 6, 2010. 34

[73] S. W. Kong, W. T. Pu, and P. J. Park, "A multivariate approach for integrating genome-wide expression data and biological knowledge," *Bioinformatics (Oxford, England)*, vol. 22, 2006. 34

[74] T. Ideker, O. Ozier, B. Schwikowski, and A. F. Siegel, "Discovering regulatory and signalling circuits in molecular interaction networks," *Bioinformatics (Oxford, England)*, vol. 18, 2002. 34

[75] K. Komurov, S. Dursun, S. Erdin, and P. T. Ram, "Netwalker: a contextual network analysis tool for functional genomics," *BMC genomics*, vol. 13, 2012. 34

[76] W. Liu, C. Li, Y. Xu, H. Yang, Q. Yao, J. Han, D. Shang, C. Zhang, F. Su, X. Li, Y. Xiao, F. Zhang, and M. Dai, "Topologically inferring risk-active pathways toward precise cancer classification by directed random walk," *Bioinformatics (Oxford, England)*, vol. 29, 2013. 34, 52

[77] Ö. N. Yaveroğlu, T. Milenković, and N. Pržulj, "Proper evaluation of alignment-free network comparison methods," *Bioinformatics*, vol. 31, 2015. 34

[78] S. Draghici, "A systems biology approach for pathway level analysis," *Genome research*, vol. 17, 2007. 34

[79] B. Klein, "Positioning nk-kappab in multiple myeloma„" *Blood*, vol. 115, 2010. 36

[80] J. Quinlan, "Simplifying decision trees," *International Journal of Man-Machine Studies*, vol. 27, 1987. 39

[81] L. Breiman, "Random forests," *Machine Learning*, vol. 45, 2001. 39

[82] K. Podar, "Up-regulation of c-jun inhibits proliferation and induces apoptosis via caspase-triggered c-abl cleavage in human multiple myeloma," *Cancer research*, vol. 67, 2007. 41, 47

[83] F. H. Xu, "Interleukin-6-induced inhibition of multiple myeloma cell apoptosis: support for the hypothesis that protection is mediated via inhibition of the jnk/sapk pathway," *Blood*, vol. 92, 1998. 41, 42, 46

[84] M. N. Saha, "Targeting p53 via jnk pathway: a novel role of rita for apoptotic signaling in multiple myeloma," *PloS one*, vol. 7, 2012. 41, 47

[85] L. Chen, "Identification of early growth response protein 1 (egr-1) as a novel target for jun-induced apoptosis in multiple myeloma," *Blood*, vol. 115, 2010. 41, 42, 46

[86] F. Fan, "Targeting mcl-1 for multiple myeloma (mm) therapy: drug-induced generation of mcl-1 fragment mcl-1(128-350) triggers mm cell death via c-jun upregulation," *Cancer letters*, vol. 343, 2014. 41

[87] S. Uddin, "Overexpression of foxm1 offers a promising therapeutic target in diffuse large b-cell lymphoma," *Haematologica*, vol. 97, 2012. 41, 46

[88] C. Gu, "Foxm1 is a therapeutic target for high-risk multiple myeloma," *Leukemia*, vol. 30, 2016. 41, 43, 46

[89] K. Mahtouk, "An inhibitor of the egf receptor family blocks myeloma cell growth factor activity of hb-egf and potentiates dexamethasone or anti-il-6 antibody-induced apoptosis," *Blood*, vol. 103, 2004. 41

[90] K. Mahtouk, "Expression of egf-family receptors and amphiregulin in multiple myeloma. amphiregulin is a growth factor for myeloma cells," *Oncogene*, vol. 24, 2005. 41

[91] J. B. Johnston, "Targeting the egfr pathway for cancer therapy," *Current medicinal chemistry*, vol. 13, 2006. 41

[92] M. Hallek, "Signal transduction of interleukin-6 involves tyrosine phosphorylation of multiple cytosolic proteins and activation of src-family kinases fyn, hck, and lyn in multiple myeloma cell lines," *Experimental hematology*, vol. 25, 1997. 41

[93] A. M. L. Coluccia, "Validation of pdgfrbeta and c-src tyrosine kinases as tumor/vessel targets in patients with multiple myeloma: preclinical efficacy of the novel, orally available inhibitor dasatinib," *Blood*, vol. 112, 2008. 41

[94] H. Ishikawa, "Requirements of src family kinase activity associated with cd45 for myeloma cell proliferation by interleukin-6," *Blood*, vol. 99, 2002. 41

[95] D. R. Cox, "Regression models and life-tables," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 34, no. 2, pp. 187–220, 1972. 44

[96] H. Avet-Loiseau, "Combining fluorescent in situ hybridization data with iss staging improves risk assessment in myeloma: an international myeloma working group collaborative project," *Leukemia*, vol. 27, 2013. 45

[97] R. Eferl and E. F. Wagner, "Ap-1: a double-edged sword in tumorigenesis," *Nature reviews. Cancer*, vol. 3, 2003. 46

[98] E. Shaulian and M. Karin, "Ap-1 as a regulator of cell life and death," *Nature Cell Biology*, vol. 4, 2002. 46

[99] J. R. Nevins, "The rb/e2f pathway and cancer„" *Human molecular genetics*, vol. 10, 2001. 47, 52

[100] E. S. Knudsen and J. Y. J. Wang, "Targeting the rb-pathway in cancer therapy," *Clinical cancer research: an official journal of the American Association for Cancer Research*, vol. 16, 2010. 47

[101] M. Razzaq, L. Chebouba, P. Le Jeune, H. Mhamdi, C. Guziolowski, and J. Bourdon, *Logic and Linear Programs to Understand Cancer Response*, pp. 191–213. Cham: Springer International Publishing, 2019. 4, 50, 104

[102] V. Marx, "Biology: The big challenges of big data," *Nature*, vol. 498, 2013. 52

[103] M. Bentele, I. Lavrik, M. Ulrich, S. Stößer, D. W. Heermann, H. Kalthoff, P. H. Krammer, and R. Eils, "Mathematical modeling reveals threshold mechanism in cd95-induced apoptosis," *J Cell Biol*, vol. 166, 2004. 52

[104] O. Ates, "Systems biology of microbial exopolysaccharides production," *Front Bioeng Biotechnol*, vol. 3, 2015. 52

[105] K. Mitra, A.-. R. Carvunis, S. K. Ramesh, and T. Ideker, "Integrative approaches for finding modular structure in biological networks," *Nat Rev Genet*, vol. 14, 2013. 52

[106] S. V. Rajkumar, "Multiple myeloma: 2016 update on diagnosis, risk-stratification, and management," *Am J Hematol*, vol. 91, 2016. 52

[107] C. F. Schaefer, K. Anthony, S. Krupa, J. Buchoff, M. Day, T. Hannay, and K. H. Buetow, "PID: the Pathway Interaction Database.," *Nucleic acids research*, vol. 37, pp. D674–9, Jan. 2009. 53

[108] H. Han, J. W. Cho, S. Lee, A. Yun, H. Kim, D. Bae, S. Yang, C. Y. Kim, M. Lee, E. Kim, S. Lee, B. Kang, D. Jeong, Y. Kim, H. N. Jeon, H. Jung, S. Nam, M. Chung, J. H. Kim, and I. Lee, "TRRUST v2: an expanded reference database of human and mouse transcriptional regulatory interactions," *Nucleic Acids Res.*, vol. 46, pp. D380–D386, Jan 2018. 53

[109] B. Miannay, "Iggy-POC." `https://github.com/BertrandMiannay/Iggy-POC`, 2017. 56

[110] P. Le Jeune, J. Paris, J. Voinea, J. Liu, and K. Boulkenafet, "Iguana." `https://github.com/ipeter50/Iguana`, 2018. 56

[111] P. D. Thomas, A. Kejariwal, M. J. Campbell, H. Mi, K. Diemer, N. Guo, I. Ladunga, B. Ulitsky-Lazareva, A. Muruganujan, S. Rabkin, J. A. Vandergriff, and O. Doremieux, "PANTHER: a browsable database of gene products organized

by biological function, using curated protein family and subfamily classification," *Nucleic Acids Res.*, vol. 31, pp. 334–341, Jan 2003. 56

[112] H. Han, J. W. Cho, S. Lee, A. Yun, H. Kim, D. Bae, S. Yang, C. Y. Kim, M. Lee, E. Kim, S. Lee, B. Kang, D. Jeong, Y. Kim, H. N. Jeon, H. Jung, S. Nam, M. Chung, J. H. Kim, and I. Lee, "TRRUST v2: an expanded reference database of human and mouse transcriptional regulatory interactions," *Nucleic Acids Res*, vol. 46, pp. D380–D386, 01 2018. 57

[113] M. Kuhn, P. Yates, and C. Hyde, *Statistical Methods for Drug Discovery*, pp. 53–81. Cham: Springer International Publishing, 2016. 59

[114] Wang, Yuanyuan (Marcia), *Statistical Methods for High Throughput Screening Drug Discovery Data*. PhD thesis, 2005. 59

[115] A. N. Lima, E. A. Philot, G. H. G. Trossini, L. P. B. Scott, V. G. Maltarollo, and K. M. Honorio, "Use of machine learning approaches for novel drug discovery," *Expert Opinion on Drug Discovery*, vol. 11, no. 3, pp. 225–239, 2016. 59

[116] E. Gawehn, J. A. Hiss, and G. Schneider, "Deep learning in drug discovery," *Molecular Informatics*, vol. 35, no. 1, pp. 3–14, 2016. 59

[117] R. F. Murphy, "An active role for machine learning in drug development," *Nat Chem Biol*, vol. 7, pp. 327–330, 2011. 59

[118] G. Apic, T. Ignjatovic, S. Boyer, and R. B. Russell, "Illuminating drug discovery with biological pathways," *FEBS Lett.*, vol. 579, pp. 1872–1877, Mar 2005. 59

[119] A. Korkut, W. Wang, E. Demir, B. A. Aksoy, X. Jing, E. J. Molinelli, O. Babur, D. L. Bemis, S. Onur Sumer, D. B. Solit, C. A. Pratilas, and C. Sander, "Perturbation biology nominates upstream-downstream drug combinations in RAF inhibitor resistant melanoma cells," *Elife*, vol. 4, Aug 2015. 59

[120] S. Videla, C. Guziolowski, F. Eduati, S. Thiele, N. Grabe, J. Saez-Rodriguez, and A. Siegel, "Revisiting the training of logic models of protein signaling networks with asp," in *Computational Methods in Systems Biology*, pp. 342–361, Springer Berlin/Heidelberg, 2012. 59, 79, 85

[121] L. Chebouba, B. Miannay, D. Boughaci, and C. Guziolowski, "Discriminate the response of Acute Myeloid Leukemia patients to treatment by using proteomics data and Answer Set Programming," *BMC Bioinformatics*, vol. 19, p. 59, Mar 2018. 4, 6, 59, 63, 64, 99, 100, 102

[122] S. Videla, J. Saez-Rodriguez, C. Guziolowski, and A. Siegel, "caspo: a toolbox for automated reasoning on the response of logical signaling networks families," *Bioinformatics*, vol. 33, no. 6, pp. 947–950, 2017. 61

[123] D. Noren, B. Long, R. Norel, K. Rrhissorrakrai, K. Hess, C. Hu, A. Bisberg, A. Schultz, E. Engquist, L. Liu, X. Lin, G. Chen, H. Xie, G. Hunter, P. Boutros, O. Stepanov, T. Norman, S. Friend, G. Stolovitzky, S. Kornblau, A. Qutub, and DREAM 9 AML-OPC Consortium, "A crowdsourcing approach to developing and assessing prediction algorithms for aml prognosis," *PLoS Computational Biology*, vol. 12, 6 2016. 61

[124] D. Thomas, J. A. Powell, F. Vergez, D. H. Segal, N. Y. Nguyen, A. Baker, T. C. Teh, E. F. Barry, J. E. Sarry, E. M. Lee, T. L. Nero, A. M. Jabbour, G. Pomilio, B. D. Green, S. Manenti, S. P. Glaser, M. W. Parker, A. F. Lopez, P. G. Ekert, R. B. Lock, D. C. Huang, S. K. Nilsson, C. Recher, A. H. Wei, and M. A. Guthridge, "Targeting acute myeloid leukemia by dual inhibition of PI3K signaling and Cdk9-mediated Mcl-1 transcription," *Blood*, vol. 122, pp. 738–748, Aug 2013. 62

[125] C. Guziolowski, S. Videla, F. Eduati, S. Thiele, T. Cokelaer, A. Siegel, and J. Saez-Rodriguez, "Exhaustively characterizing feasible logic models of a signaling network using Answer Set Programming," *Bioinformatics*, vol. 29, pp. 2320–2326, Sep 2013. 4, 64, 100, 102, 104

[126] J. Saez-Rodriguez, L. G. Alexopoulos, J. Epperlein, R. Samaga, D. A. Lauffenburger, S. Klamt, and P. K. Sorger, "Discrete logic modelling as a means to link protein signalling networks with functional analysis of mammalian signal transduction," *Mol Syst Biol*, vol. 5, p. 331, 2009. 4, 64

[127] S. Videla, C. Guziolowski, F. Eduati, S. Thiele, M. Gebser, J. Nicolas, J. Saez-Rodriguez, T. Schaub, and A. Siegel, "Learning boolean logic models of signaling networks with asp," *Theoretical Computer Science*, vol. 599, pp. 79–101, 2015. Advances in Computational Methods in Systems Biology. 4, 64

[128] S. Videla, J. Saez-Rodriguez, C. Guziolowski, and A. Siegel, "caspo: a toolbox for automated reasoning on the response of logical signaling networks families," *Bioinformatics*, vol. 33, pp. 947–950, 03 2017. 4, 64

[129] A. Vaginay, T. Boukhobza, and M. Smaïl-Tabbone, "Automatic synthesis of boolean networks from biological knowledge and data," in *Optimization and Learning*

(B. Dorronsoro, L. Amodeo, M. Pavone, and P. Ruiz, eds.), (Cham), pp. 156–170, Springer International Publishing, 2021. 4, 64

[130] J. R. Banga, "Optimization in computational systems biology.," *BMC Systems Biology*, vol. 2, p. 47, 2008. 65

[131] E. Walter and L. Pronzato, "On the identifiability and distinguishability of nonlinear parametric models," *Mathematics and Computers in Simulation*, vol. 42, pp. 125–134, Oct. 1996. 65

[132] C. Kreutz and J. Timmer, "Systems biology: experimental design," *FEBS Journal*, vol. 276, pp. 923–942, Jan. 2009. 65, 76

[133] R.-S. R. Wang, A. A. Saadatpour, and R. R. Albert, "Boolean modeling in systems biology: an overview of methodology and applications.," *Physical biology*, vol. 9, Sept. 2012. 65

[134] L. Calzone, L. Tournier, S. Fourquet, D. Thieffry, B. Zhivotovsky, E. Barillot, and A. Zinovyev, "Mathematical modelling of cell-fate decision in response to death receptor engagement.," *PLoS Computational Biology*, vol. 6, p. e1000702, Mar. 2010. 65

[135] A. Mitsos, I. Melas, P. Siminelakis, A. Chairakaki, J. Saez-Rodriguez, and L. G. Alexopoulos, "Identifying Drug Effects via Pathway Alterations using an Integer Linear Programming Optimization Formulation on Phosphoproteomic Data," *PLoS Comp. Biol.*, vol. 5, p. e1000591, Sept. 2009. 66, 76, 79, 102

[136] R. Sharan and R. M. Karp, "Reconstructing Boolean Models of Signaling," in *Research in Computational Molecular Biology*, pp. 261–271, Springer, 2012. 66, 76, 79

[137] S. Videla, C. Guziolowski, F. Eduati, S. Thiele, N. Grabe, J. Saez-Rodriguez, and A. Siegel, "Revisiting the Training of Logic Models of Protein Signaling Networks with ASP," in *10th Int. Conf. on Computational Methods in Systems Biology*, LNCS, pp. 342–361, Springer, 2012. 66

[138] C. D. Terfve, T. Cokelaer, D. Henriques, A. Macnamara, E. Goncalves, M. K. Morris, M. van Iersel, D. A. Lauffenburger, and J. Saez-Rodriguez, "CellNOptR: a flexible toolkit to train protein signaling networks to data using multiple logic formalisms," *BMC Syst Biol*, vol. 6, p. 133, Oct 2012. 66, 69, 70, 76

[139] T. Schaub and S. Thiele, "Metabolic network expansion with answer set programming," in *Logic Programming* (P. M. Hill and D. S. Warren, eds.), (Berlin, Heidelberg), pp. 312–326, Springer Berlin Heidelberg, 2009. 68

[140] T. Fayruzov, M. De Cock, C. Cornelis, and D. Vermeir, "Modeling Protein Interaction Networks with Answer Set Programming," in *Int. Conf. on Bioinformatics and Biomedicine, 2009.*, pp. 99–104, Nov. 2009. 68

[141] M. Durzinsky, W. Marwan, M. Ostrowski, T. Schaub, and A. Wagler, "Automatic network reconstruction using ASP," *Theory and Practice of Logic Programming*, vol. 11, pp. 749–766, 2011. 68

[142] I. Papatheodorou, M. Ziehm, D. Wieser, N. Alic, L. Partridge, and J. M. Thornton, "Using Answer Set Programming to Integrate RNA Expression with Signalling Pathway Information to Infer How Mutations Affect Ageing," *PLoS ONE*, vol. 7, p. e50881, Dec. 2012. 68

[143] O. Ray and T. Soh, "Analyzing Pathways Using ASP-Based Approaches," *Algebraic and Numeric Biology*, 2012. 68

[144] B. Néron, H. Ménager, C. Maufrais, N. Joly, J. Maupetit, S. Letort, S. Carrere, P. Tuffery, and C. Letondal, "Mobyle: a new full web bioinformatics framework.," *Bioinformatics*, vol. 25, pp. 3005–3011, Nov. 2009. 69

[145] L. G. Alexopoulos, J. Saez-Rodriguez, B. D. Cosgrove, D. A. Lauffenburger, and P. K. Sorger, "Networks inferred from biochemical data reveal profound differences in toll-like receptor and inflammatory signaling between normal and transformed hepatocytes.," *Molecular & Cellular Proteomics*, vol. 9, pp. 1849–1865, Sept. 2010. 70

[146] J. H. Morris, L. Apeltsin, A. M. Newman, J. Baumbach, T. Wittkop, G. Su, G. D. Bader, and T. E. Ferrin, "clustermaker: a multi-algorithm clustering plugin for cytoscape," *BMC bioinformatics*, vol. 12, 2011. 70

[147] G. Liu, T. Janhunen, and I. Niemelä, "Answer Set Programming via Mixed Integer Programming," in *13th Int. Conf. on Principles of Knowledge Representation and Reasoning*, AAAI Press, 2012. 76

[148] M. Ostrowski and T. Schaub, "ASP modulo CSP: The clingcon system," *Theory and Practice of Logic Programming*, vol. 12, pp. 485–503, July 2012. 76

[149] S. E. Kolitz and D. A. Lauffenburger, "Measurement and Modeling of Signaling at a Single-Cell Level," *Biochemistry*, p. 120906164635004, Sept. 2012. 77

[150] M. Razzaq, L. Paulevé, A. Siegel, J. Saez-Rodriguez, J. Bourdon, and C. Guziolowski, "Computational discovery of dynamic cell line specific Boolean networks from multiplex time-course data," *PLoS Comput Biol*, vol. 14, p. e1006538, 10 2018. 5, 6, 78, 99, 100, 104

[151] M. Ostrowski, L. Paulevé, T. Schaub, A. Siegel, and C. Guziolowski, "Boolean network identification from multiplex time series data," in *Computational Methods in Systems Biology* (O. Roux and J. Bourdon, eds.), (Cham), pp. 170–181, Springer International Publishing, 2015. 5, 78, 79, 84, 104

[152] M. Ostrowski, L. Paulevé, T. Schaub, A. Siegel, and C. Guziolowski, "Boolean network identification from perturbation time series data combining dynamics abstraction and logic programming," *Biosystems*, vol. 149, pp. 139–153, 2016. 5, 78, 79, 81, 82, 84, 85

[153] S. Watterson, S. Marshall, and P. Ghazal, "Logic models of pathway biology," *Drug discovery today*, vol. 13, no. 9, pp. 447–456, 2008. 79

[154] R. Samaga and S. Klamt, "Modeling approaches for qualitative and semi-quantitative analysis of cellular signaling networks," *Cell communication and signaling*, vol. 11, no. 1, p. 43, 2013. 79

[155] A. MacNamara, C. Terfve, D. Henriques, B. P. Bernabé, and J. Saez-Rodriguez, "State–time spectrum of signal transduction logic models," *Physical biology*, vol. 9, no. 4, p. 045003, 2012. 79

[156] S. A. Kauffman, *The origins of order: Self-organization and selection in evolution*. Oxford University Press, USA, 1993. 79

[157] R. Thomas, "Laws for the dynamics of regulatory networks.," *International Journal of Developmental Biology*, vol. 42, no. 3, pp. 479–485, 2002. 79

[158] M. L. Wynn, N. Consul, S. D. Merajver, and S. Schnell, "Logic-based models in systems biology: a predictive and parameter-free network analysis method," *Integrative biology*, vol. 4, no. 11, pp. 1323–1337, 2012. 79

[159] A. Almudevar, M. N. McCall, H. McMurray, and H. Land, "Fitting boolean networks from steady state perturbation data," *Statistical applications in genetics and molecular biology*, vol. 10, no. 1, p. 47, 2011. 79

[160] P. Zhu, H. M. Aliabadi, H. Uludağ, and J. Han, "Identification of potential drug targets in cancer signaling pathways using stochastic logical models," *Scientific reports*, vol. 6, 2016. 79

[161] S. M. Hill, L. M. Heiser, T. Cokelaer, M. Unger, N. K. Nesser, D. E. Carlin, Y. Zhang, A. Sokolov, E. O. Paull, C. K. Wong, *et al.*, "Inferring causal molecular networks: empirical assessment through a community-based effort," *Nature methods*, vol. 13, no. 4, pp. 310–318, 2016. 81, 82

[162] S. M. Hill, N. K. Nesser, K. Johnson-Camacho, M. Jeffress, A. Johnson, C. Boniface, S. E. Spencer, Y. Lu, L. M. Heiser, Y. Lawrence, *et al.*, "Context specificity in causal signaling networks revealed by phosphoprotein profiling," *Cell systems*, vol. 4, no. 1, pp. 73–83, 2017. 81, 82, 95

[163] J. Dorier, I. Crespo, A. Niknejad, R. Liechti, M. Ebeling, and I. Xenarios, "Boolean regulatory network reconstruction using literature based knowledge with a genetic algorithm optimization method," *BMC bioinformatics*, vol. 17, no. 1, p. 410, 2016. 82

[164] S. A. Kauffman, "Metabolic stability and epigenesis in randomly constructed genetic nets," *Journal of theoretical biology*, vol. 22, no. 3, pp. 437–467, 1969. 84

[165] K. Inoue, "Logic programming for boolean networks," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, vol. 22 of *IJCAI'11*, pp. 924–930, AAAI Press, 2011. 84

[166] A. Cheng, J. Esparza, and J. Palsberg, "Complexity results for 1-safe nets," *Theoretical Computer Science*, vol. 147, no. 1, pp. 117 – 136, 1995. 84

[167] A. Cimatti, E. Clarke, E. Giunchiglia, F. Giunchiglia, M. Pistore, M. Roveri, R. Sebastiani, and A. Tacchella, "Nusmv 2: An opensource tool for symbolic model checking," in *International Conference on Computer Aided Verification*, pp. 359–364, Springer, 2002. 86

[168] G. Wu, E. Dawson, A. Duong, R. Haw, and L. Stein, "Reactomefiviz: a cytoscape app for pathway and network-based data analysis," *F1000Research*, vol. 3, 2014. 87

[169] G. Wu, E. Dawson, A. Duong, R. Haw, and L. Stein, "Reactomefiviz: a cytoscape app for pathway and network-based data analysis," *F1000Research*, vol. 3, 2014. 87

[170] D. E. Carlin, *Computational evaluation and derivation of biological networks in cancer and stem cells*. University of California, Santa Cruz, 2014. 96

[171] J. Romero, T. Schaub, and P. Wanko, "Computing diverse optimal stable models," in *ICLP (Technical Communications)*, vol. 52 of *OASICS*, pp. 3:1–3:14, Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016. 97

[172] M. Razzaq, R. Kaminski, J. Romero, T. Schaub, J. Bourdon, and C. Guziolowski, "Computing diverse boolean networks from phosphoproteomic time series data," in *Computational Methods in Systems Biology - 16th International Conference, CMSB 2018, Brno, Czech Republic, September 12-14, 2018, Proceedings* (M. Ceska and D. Safránek, eds.), vol. 11095 of *Lecture Notes in Computer Science*, pp. 59–74, Springer, 2018. 97

[173] S. J. Dunn, G. Martello, B. Yordanov, S. Emmott, and A. G. Smith, "Defining an essential transcription factor program for naïve pluripotency," *Science*, vol. 344, pp. 1156–1160, Jun 2014. 101

[174] D. Meistermann, A. Bruneau, S. Loubersac, A. Reignier, J. Firmin, V. François-Campion, S. Kilens, Y. Lelièvre, J. Lammers, M. Feyeux, P. Hulin, S. Nedellec, B. Bretin, G. Castel, N. Allègre, S. Covin, A. Bihouée, M. Soumillon, T. Mikkelsen, P. Barrière, C. Chazaud, J. Chappell, V. Pasque, J. Bourdon, T. Fréour, and L. David, "Integrated pseudotime analysis of human pre-implantation embryo single-cell transcriptomes reveals the dynamics of lineage specification," *Cell Stem Cell*, vol. 28, pp. 1625–1640, 09 2021. 101

[175] V. G. Tusher, R. Tibshirani, and G. Chu, "Significance analysis of microarrays applied to the ionizing radiation response," *Proc Natl Acad Sci U S A*, vol. 98, pp. 5116–5121, Apr 2001. 103

[176] P. Geeleher, N. J. Cox, and R. S. Huang, "Clinical drug response can be predicted using baseline gene expression levels and in vitro drug sensitivity in cell lines," *Genome Biol*, vol. 15, p. R47, Mar 2014. 104

[177] F. E. Faisal and T. Milenkovic, "Dynamic networks reveal key players in aging," *Bioinformatics*, vol. 30, 2014. 104

[178] P. Creixell, A. Palmeri, C. J. Miller, H. J. Lou, C. C. Santini, M. Nielsen, B. E. Turk, and R. Linding, "Unmasking determinants of specificity in the human kinome," *Cell*, vol. 163, pp. 187–201, Sep 2015. 104

[179] G. Collet, D. Eveillard, M. Gebser, S. Prigent, T. Schaub, A. Siegel, and S. Thiele, "Extending the metabolic network of ectocarpus siliculosus using answer set programming," in *Logic Programming and Nonmonotonic Reasoning* (P. Cabalar and T. C. Son, eds.), (Berlin, Heidelberg), pp. 245–256, Springer Berlin Heidelberg, 2013. 105

[180] C. Frioux, T. Schaub, S. Schellhorn, A. Siegel, and P. Wanko, "Hybrid metabolic network completion," in *Logic Programming and Nonmonotonic Reasoning* (M. Balduccini and T. Janhunen, eds.), (Cham), pp. 308–321, Springer International Publishing, 2017. 105

[181] M. Razzaq, R. Kaminski, J. Romero, T. Schaub, J. Bourdon, and C. Guziolowski, "Computing diverse boolean networks from phosphoproteomic time series data," in *Computational Methods in Systems Biology* (M. Češka and D. Šafránek, eds.), (Cham), pp. 59–74, Springer International Publishing, 2018. 105

[182] E. G. Cerami, B. E. Gross, E. Demir, I. Rodchenkov, O. Babur, N. Anwar, N. Schultz, G. D. Bader, and C. Sander, "Pathway Commons, a web resource for biological pathway data," *Nucleic Acids Res*, vol. 39, pp. D685–690, Jan 2011. 105

[183] J. Moreaux, B. Klein, R. Bataille, G. Descamps, S. Maïga, D. Hose, H. Goldschmidt, A. Jauch, T. Rème, M. Jourdan, M. Amiot, and C. Pellat-Deceunynck, "A high-risk signature for patients with multiple myeloma established from the molecular classification of human myeloma cell lines," *Haematologica*, vol. 96, pp. 574–582, Apr 2011. 106

[184] D. Chiron, S. Maïga, S. Surget, G. Descamps, P. Gomez-Bougie, S. Traore, N. Robillard, P. Moreau, S. Le Gouill, R. Bataille, M. Amiot, and C. Pellat-Deceunynck, "Autocrine insulin-like growth factor 1 and stem cell factor but not interleukin 6 support self-renewal of human myeloma cells," *Blood Cancer J*, vol. 3, p. e120, Jun 2013. 106

[185] L. Bodet, P. Gomez-Bougie, C. Touzeau, C. Dousset, G. Descamps, S. Maïga, H. Avet-Loiseau, R. Bataille, P. Moreau, S. Le Gouill, C. Pellat-Deceunynck, and M. Amiot, "ABT-737 is highly effective against molecular subgroups of multiple myeloma," *Blood*, vol. 118, pp. 3901–3910, Oct 2011. 106

[186] S. Surget, D. Chiron, P. Gomez-Bougie, G. Descamps, E. Ménoret, R. Bataille, P. Moreau, S. Le Gouill, M. Amiot, and C. Pellat-Deceunynck, "Cell death via DR5, but not DR4, is regulated by p53 in myeloma cells," *Cancer Res*, vol. 72, pp. 4562–4573, Sep 2012. 106

DOCTORAT BRETAGNE LOIRE / MATHSTIC

Nantes Université

**Titre :** Modélisation de réseaux biologiques à l'aide des programmes logiques

**Mot clés :** Programmation logique, réseaux de régulation, modélisation de systèmes biologiques

**Résumé :** Dans ce manuscrit nous explorons deux représentations d'un système biologique en utilisant des modélisations informatiques. Les résultats de ces deux représentations ont été diffusés et valorisés au travers de publications scientifiques méthodologiques et applicatives ; et pour certains d'entre eux, au travers de projets de recherche en étroite collaboration avec des biologistes.

Notre première contribution a été dans la modélisation par la *consistance des signes*. Ici, un réseau de régulation (graphe orienté et signé) est confronté à des données expérimentales contenant un ensemble d'observations des gènes du système dans deux conditions différentes. Cette confrontation a été implémentée dans un programme logique écrit dans le paradigme de la Programmation par Ensemble Réponses. Ce programme détecte une mesure de consistance entre le graphe et le jeu de données expérimentales, et propose des corrections automatiques minimales dans le cas d'inconsistance. Une fois que la consistance est établie pour le système, une liste de déductions peut être énumérée, ces déductions logiques seront appelées les prédictions du système. Notre outil se nomme *Iggy* et utilise le solveur `clasp`. Nous avons appliqué cette modélisation à plusieurs systèmes biologiques, notamment chez l'humain. Nous détaillons, dans ce manuscrit, son application dans la modélisation du Myélome Multiple, qui nous a permis de mettre en lumière des espèces clés dans le réseau de régulation. Les prédictions de ces espèces clés nous ont permit de discriminer des patients ayant une meilleur survie. Dans une autre étude, nous avons étendu Iggy pour développer un nouveau système *MajS* dont la finalité est de faire le lien entre modélisation d'un réseau des gènes et métabolisme.

Une deuxième contribution a été dans l'apprentissage des familles des réseau booléens (RBn). Ce formalisme consiste à apprendre de familles de RBs à partir d'une connaissance préliminaire de régulation (ou *prior knowledge network*, PKN) en la confrontant avec des données d'expression (de gènes ou protéines) issues de multiples perturbations expérimentales. Les familles de RBs seront optimales car elles auront une taille minimale et qu'elles expliqueront de façon optimale les observations obtenues à travers les multiples perturbations. Le premier système conçu a été *caspo*, également implémenté avec des programmes logiques. Une extension de *caspo* a été conduite pour proposer des expériences de perturbations pour réduire la taille de la famille des RBs apprise. Plus tard, nous avons proposé *caspo-ts* qui peut assimiler des séries temporelles des données et qui propose des RBs dynamiques. *caspo-ts* a été appliqué sur les données d'une défi internationale, nommé le *HPN-DREAM challenge*, appliqué à des lignées cellulaires du cancer du sein. L'objectif était de déterminer des de mécanismes de régulations ou des fonctions booléennes différentes qui s'expriment dans des lignées cellulaires. En parallèle de ces travaux, nous avons proposé une méthode (basé sur la programmation logique), pour générer de données de *pseudo-perturbations* à partir de données expérimentales pour des systèmes où il n'est pas possible de réaliser des perturbations pour des raisons éthiques. Cette méthode a été utilisée pour analyser des données issues de patients ayant développé une Leucémie Myeloïde Aiguë, et pour discriminer les patients ayant une meilleure réponse au traitement employé. Nous sommes actuellement en train d'adapter cette méthode pour l'appliquer à des données de *single cell*, dans une étude du développement embryonnaire chez l'humain.

**Title:** Modeling Biological Networks as Logic Programs

**Keywords:** Logic programming, regulatory networks, biological systems modeling

**Abstract:** In this manuscript it is proposed to explore two representations of a biological system using computational modeling. These representations both gave birth to several methodological publications, and in some cases research projects in close collaboration with biologists.

One is done through the *sign-consistency* modeling. In this approach a regulatory network (signed directed graph) is combined with a dataset of gene expression observations, using a logic program. This logic program, written in Answer Set Programming, expresses a rule that has to be valid for each species in the network, which relates the *sign* of a network species with its direct predecessors *influences and signs*. This rule is tested in a global way, through all the network species by using an efficient solver, `clasp`. The sign-consistency modeling framework we proposed is named *Iggy*. Iggy performs as well automatic and optimal correction of sign inconsistencies. The sign-consistency modeling framework has been applied to different biological case studies. For example, the signaling pathway of Hepatocyte Growth Factor, where some of the computational predictions of our model were validated experimentally. A case-study well described in this manuscript is the modeling of Multiple Myeloma patients gene expression data. Our main results on this system was to propose Multiple Myeloma markers, that is, species in the network, coupled with our computational predictions, that allow to identify patients having a better survival. *Iggy* has inspired *MajS*, our last sign-consistency modeling framework contribution. In this on-going research project we plan to integrate gene regulatory and metabolic network modeling.

A second modeling approach is *learning Boolean network families*. In this framework, given a regulatory network (also called Prior Knowledge Network, PKN) and a set of network species observations upon multiple perturbations over the system, our framework learns a family of Boolean Networks (BNs), compatible with the PKN topology, that fits the perturbation data with minimal error. The first system we conceived is named *caspo*. It is also implemented using Answer Set Programming. An extension of *caspo* was implemented, so that new experimental designs (*i.e.* new experimental perturbations) can be proposed to decrease the number of learned BNs. Later, we proposed a system named *caspo-ts*, which deals with perturbation time-series data, and the output of this system is a family of dynamic BNs. *caspo-ts* has been applied to the data of HPN-DREAM challenge, concerning Breast cancer cell lines. Our objective was to identify the different BNs underlying the four Breast Cancer cell lines considered. In parallel, since multiple perturbation data, essential for *caspo* or *caspo-ts*, is sometimes hard to obtain in Human systems because of ethical reasons; we have begun a research subject towards the extraction of multiple pseudo-perturbations from non perturbed datasets, such as proteomics or RNA-Seq datasets. This method has been applied to discriminate Acute Myeloid Leukemia patients having different treatment prognosis. Currently, we are exploring to extract multiple pseudo-perturbations from single cell data, in the study of Human embryo development.