Development of Constrained Fuzzy Logic for Modeling Biological Regulatory Networks and Predicting Contextual Therapeutic Effects

by

Melody K. Morris

B.S., University of Kentucky (2007)

Submitted to the Department of Biological Engineering
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Biological Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2012

Author ....
Department of Biological Engineering
May 18, 2012

Certified by ....
Douglas A. Lauffenburger
Professor
Thesis Supervisor

Accepted by ....
Forest White
Chairman, Department Committee on Graduate Theses

This Doctoral Thesis has been examined by the following Thesis Committee:

Douglas A. Lauffenburger, Ph.D.
Professor of Biological Engineering
Massachusetts Institute of Technology

Ernest Fraenkel, Ph.D.
Thesis Committee Chair
Associate Professor of Biological Engineering
Massachusetts Institute of Technology

Peter K. Sorger, Ph.D.
Professor of Systems Biology
Havard Medical School and Massachusetts Institute of Technology

Anand Asthagiri, Ph.D.
Associate Professor of Chemical Engineering
Northeastern University

Development of Constrained Fuzzy Logic for Modeling Biological Regulatory Networks and Predicting Contextual Therapeutic Effects

by
Melody K. Morris

Submitted to the Department of Biological Engineering
on May 18, 2012, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Biological Engineering

Abstract

Upon exposure to environmental cues, protein modifications form a complex signaling network that dictates cellular response. In this thesis, we develop methods for using continuous logic-based models to aide our understanding of these signaling networks and facilitate data interpretation. We present a novel modeling framework called constrained fuzzy logic (cFL) that maintains a simple logic-based description of interactions with AND, OR, and NOT gates, but allows for intermediate species activities with mathematical functions relating input and output values (transfer functions).

We first train a prior knowledge network (PKN) to data with cFL, which reveals what aspects of the dataset agree or disagree with prior knowledge. The cFL models are trained to a dataset describing signaling proteins in a hepatocellular carcinoma cell line after exposure to ligand cues in the presence or absence of small molecule inhibitors. We find that multiple models with differing topology and parameters explain the data equally well, and it is crucial to consider this non-identifiability during model training and subsequence analysis. Our trained models generate new biological understanding of network crosstalk as well as quantitative predictions of signaling protein activation.

In our next applications of cFL, we explore the ability of models either constructed based solely on prior knowledge or trained to dedicated biochemical data to make predictions that answer the following questions: 1) What perturbations to species in the system are effective at accomplishing a clinical goal? and 2) In what environmental conditions are these perturbations effective? We find that we are able to make accurate predictions in both cases. Thus, we offer cFL as a flexible modeling methodology to assist data interpretation and hypothesis generation for choice of therapeutic targets.

Thesis Supervisor: Douglas A. Lauffenburger
Title: Professor

Acknowledgments

The completion of this work would have not been possible without the help and support of many collaborators and friends.

My thesis committee has been instrumental in the development of this work and have been a pleasure to work with. Specifically, I thank Peter Sorger for his guidance in the writing of my initial manuscripts. I thank Ernest Fraenkel for stimulating discussions regarding the details of this work as well as his support in extending the work to additional levels of biological regulation. Anand Asthagiri joined the committee in the past year, and I have been very grateful for his fresh perspective on this project. Finally, I thank Doug Lauffenburger for his scientific guidance, support for my scientific and professional goals, and generosity in sharing his time and perspective. Doug has proven time and again that not only is he a skilled scientist, but also an admirable human being.

During my time at MIT, Julio Saez-Rodriquez has been an invaluable colleague, both for technical guidance as well as demonstrating what it means to be a colleague. To this day, I strive to follow his example in how to best interact with others. David Clarke has also been an amazing collaborator, frequently championing the approach presented in this work and always being willing to do the best, although not always easiest, experiment. From the Fraenkel lab, I thank Sara Gosline, Nurcan Tuncbag, Chris Ng, and Anthony Soltis for help in learning their analysis protocols. Doug Jones, Ken Lau, Sarah Schrier, Abby Hill, Dan Kiruoac and MingSheng Zhang have all taken an active interest in utilizing the approaches developed in this thesis, and I thank them for their interest and insights on how to improve the approach. I thank the menbers of Julio's lab at EBI, specifically Thomas Cokelaer and Aidan MacNamara, as well as Leonidas Alexopoulos, Ioannis Melas, and Alexander Mitsos for their willingness to integrate the cFL methodology into their workflows.

For stimulating discussions throughout my years at MIT, I thank Tom Schneider, Birgit Schoeberl, Kristen Naegle, Joel Wagner, Arthur Goldsipe, Emily Miraldi, Francisco Delgado, Brian Belmont, Steve Goldfless, Jeff Wagner, Edgar Sanchez, Ta-chun Hang, Sarah Kolitz, Bree Aldridge, Shannon Alford, and Nate Tedford. In particular, I would like to thank Brian Joughin for constantly demonstrating how to think rigorously and for always reminding me when it's time for lunch.

Prior to my arrival at MIT, the faculty of chemical engineering at the University of Kentucky were instrumental in preparing me for my graduate studies. In particular, I would like to thank Dr. Dibakar Bhattacharrya from that department and Athula Ekanayake from my internship at Procter and Gamble for their introduction to scientific research and advice throughout the years.

For keeping my life balanced outside of work, I thank Sarah Bashadi, Maggie Dolan, Emily Florine, Michelle Jackson, Katie Maloney, and Ranjani Paradise. I thank my parents, David and Karen Morris, and my brother, Chuck Morris. Finally, I thank my husband Jon Greiner for his unwavering love and support, willingness to listen to my worries and frustrations, and cooking of hundreds of wonderful dinners. Without Jon, I would have starved.

Contents

List of Figures

List of Tables

Chapter 1

Introduction:Logic-based models for the analysis of cell signaling networks [103]

1.1 Summary

Computational models are increasingly used to analyze the operation of complex biochemical networks, including those involved in cell signaling networks. Here we review recent advances in applying logic-based modeling to mammalian cell biology. Logic-based models represent biomolecular networks in a simple and intuitive manner without describing the detailed biochemistry of each interaction. A brief description of several logic-based modeling methods is followed by six case studies that demonstrate biological questions recently addressed using logic-based models and point to potential advances in model formalisms and training procedures that promise to enhance the utility of logic-based methods for studying the relationship between environmental inputs and phenotypic or signaling state outputs of complex signaling networks. We then relate the work presented in the remainder of this thesis to that described in this introduction.

1.2 Background

With accelerating pace, molecular biology and biochemistry are identifying complex patterns of interactions among intracellular and extracellular biomolecules. With respect to cell signaling in eukaryotes, the focus of this chapter, complex multicomponent networks involving many shared components govern how a cell will respond to diverse environmental cues. Powerful experimental approaches now exist for identifying components of these networks and for determining their biochemical activities, but understanding the networks as an integrated whole is difficult using intuition alone. Thus, mathematical and computational modeling is increasingly playing a role in data interpretation and attempts to extract general biological understanding [69, 46]. Depending on the network studied, the data available and the questions posed, a diverse spectrum of modeling approaches exists, ranging from the highly abstract to the highly specified [59, 57]. The goal of this chapter is to discuss logic-based modeling, an approach lying midway between the complexity and precision of differential equations on one hand and data-driven regression approaches on the other.

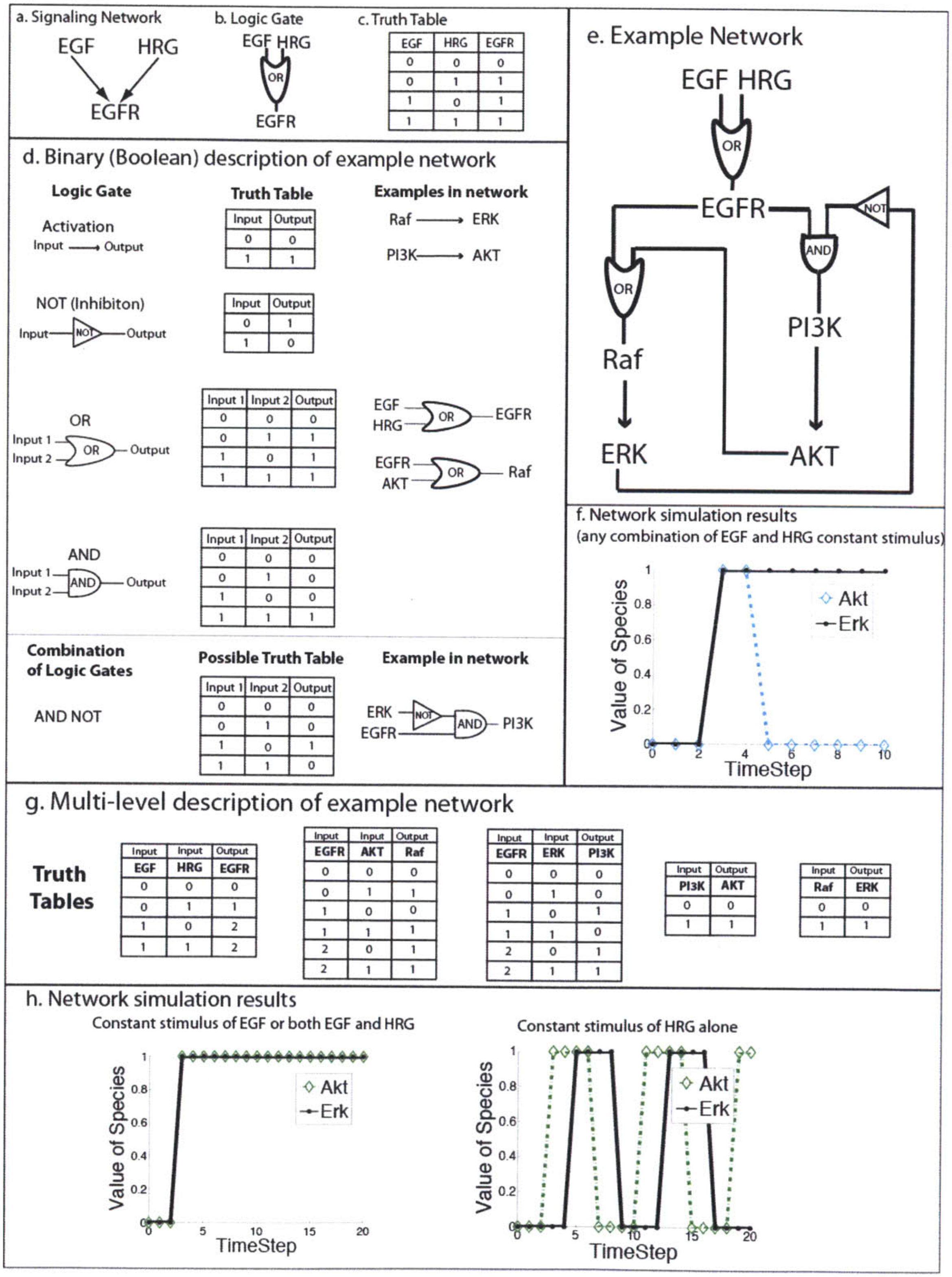
Within the spectrum of modeling methods currently being applied to cellular biochemistry, models involving differential equations bear the closest relationship to underlying biochemical rate laws. Sets of coupled ordinary differential equations (ODEs) can effectively represent chemical reactions when the number of molecules is large and mass action approximations are appropriate. Partial differential equations (PDEs) add the ability to represent spatial gradients [4] and stochastic methods make it possible to analyze systems in which the number of molecules is small[118]. Networks of differential equations can model the temporal and spatial dynamics of biochemical processes in considerable detail, making it possible to study chemical mechanism and predict network dynamics under various conditions. However, the topology of ODE and PDE-based models (that is, patterns of interaction among the species) must be specified in advance and model output is strongly dependent on the values of free parameters (typically initial protein concentrations and rate constants). Estimating these parameters is a computationally intensive task requiring substantial data. As networks get larger, ODE modeling becomes more and more challenging, and models that attempt to capture real biological data are currently limited to a few dozen components.

At the other extreme, a very active field has emerged to compute graphical representations of biological networks through literature analysis or identification of correlations in high throughput data. In these graphs, termed protein interaction networks (PINs or interactomes) or protein signaling networks (PSNs), genes and proteins are represented by nodes and potential interactions by edges (links). The edges can be directional or not and signed (inhibitory/activating) or not and typically represent a wide range of interaction modes from direct physical binding to correlated gene expression [8] or integrated database entries [121]. Graphs are an attractive way to summarize diverse relationships among large number of biomolecules across multiple organisms but they are not executable per se and cannot be used to compute input-output relationships. Moreover, network graphs rarely take into account dynamic changes in signaling activities, cell type-specific biochemistry or context-dependent variations [64].

Here, we review logic-based models, which represent a compromise between highly specified differential equation models and protein interaction graphs. Using logic-based methods, it is possible to model interactions among large numbers of protein species and perform model training, model validation and model-based prediction. The first application of logic-based modeling to biological pathways is credited to Kauffman, who used discrete logic to model the biological process of gene regulation [66]. Subsequent work focused on delineating theoretical properties of logic-based models of gene regulation [25, 147]. Huang and Ingber were among the first to apply logic-based modeling to cell signaling networks, demonstrating that specific cell phenotypes might correspond to dynamic steady states of a logic-based model of intracellular signaling species [53]. This example of linking environmental inputs to phenotypic outputs via a logic-based model of a biochemical signaling network has sparked considerable interest in the possibility of harnessing logic-based models to understand the relationship between biochemical signaling network and cell state, reflected in a large number of studies over the past few years [53, 10, 3, 34, 41, 47, 55, 67, 82, 87, 96, 116, 128, 124, 129, 131, 159, 165, 168, 161, 134, 98].

This chapter is divided into two sections. In the first, we describe the fundamentals of logic-based modeling; in the second, we discuss six applications of logic-based modeling to eukaryotic biology. We focus on logic-based models of biochemical signaling networks and refer the reader to the literature for a more in-depth explanation of theoretical considerations[148], applications of logic-based models to gene regulatory networks [25] and models of intercellular communication [68, 145].

Figure 1-1 *(facing page)*: Example logic-based network a. Protein signaling network. Biochemical species are represented as nodes. The interactions between these nodes interact is indicated with arrows. b. Logic Gate. Precisely how the nodes interact is specified with a simple Boolean logic gate. c. Truth table specifying the output node given possible combinations of its inputs nodes' values. d. Boolean logic gates an their truth tables. If the gates are used in the example network, the interaction is shown on the right. We also describe the AND NOT gate, which is used in the example network. We note that, in many applications of logic-based modeling, OR and AND gates are not explicitly indicated with their gate symbols. e. Example logic-based network structure. The model was simulated with synchronous updating using custom MatLab (Mathworks, Inc.) code (available as supplemental information). f. Network behavior with binary rules. Under initial conditions with different ligand stimulations, the network response was identical because the logic rules did not distinguish between EGF and HRG stimulation. g. Multi-state rule specification. The truth tables are given for each modeled species. These rules specify multiple states. The greater sensitivity of EGFR for EGF than HRG is encoded in the higher level it reaches upon stimulation by EGF. Rules that are different from the binary rules are highlighted. h. Network behavior with multi-state rules given in d. The rules specified that EGFR is more sensitive to EGF than HRG. Thus, the behavior differed depending on the stimulation condition. Under EGF or EGF and HRG stimulation, the states of ERK and AKT stabilized whereas they oscillated under HRG stimulation alone. This is because the rules specified that, with the highest activation level of EGFR (activation state two), the negative feedback by ERK was did not effectively inhibit PI3K, whereas with medium activation of EGFR (activation state one accessed with only HRG was present), the negative feedback was effective.



a. Signaling Network

b. Logic Gate

c. Truth Table

EGF	HRG	EGFR
0	0	0
0	1	1
1	0	1
1	1	1

d. Binary (Boolean) description of example network

Logic Gate | **Truth Table** | **Examples in network**

Activation

Input	Output
0	0
1	1

NOT (Inhibiton)

Input	Output
0	1
1	0

OR

Input 1	Input 2	Output
0	0	0
0	1	1
1	0	1
1	1	1

AND

Input 1	Input 2	Output
0	0	0
0	1	0
1	0	0
1	1	1

Combination of Logic Gates | **Possible Truth Table** | **Example in network**

AND NOT

Input 1	Input 2	Output
0	0	0
0	1	0
1	0	1
1	1	0

e. Example Network

f. Network simulation results
(any combination of EGF and HRG constant stimulus)

g. Multi-level description of example network

Truth Tables

Input	Input	Output
EGF	HRG	EGFR
0	0	0
0	1	1
1	0	2
1	1	2

Input	Input	Output
EGFR	AKT	Raf
0	0	0
0	1	1
1	0	0
1	1	1
2	0	1
2	1	1

Input	Input	Output
EGFR	ERK	PI3K
0	0	0
0	1	0
1	0	1
1	1	0
2	0	1
2	1	1

Input	Output
PI3K	AKT
0	0
1	1

Input	Output
Raf	ERK
0	0
1	1

h. Network simulation results

Constant stimulus of EGF or both EGF and HRG

Constant stimulus of HRG alone

1.3 Representing biochemical networks with logic-based models

1.3.1 What is a logic-based model?

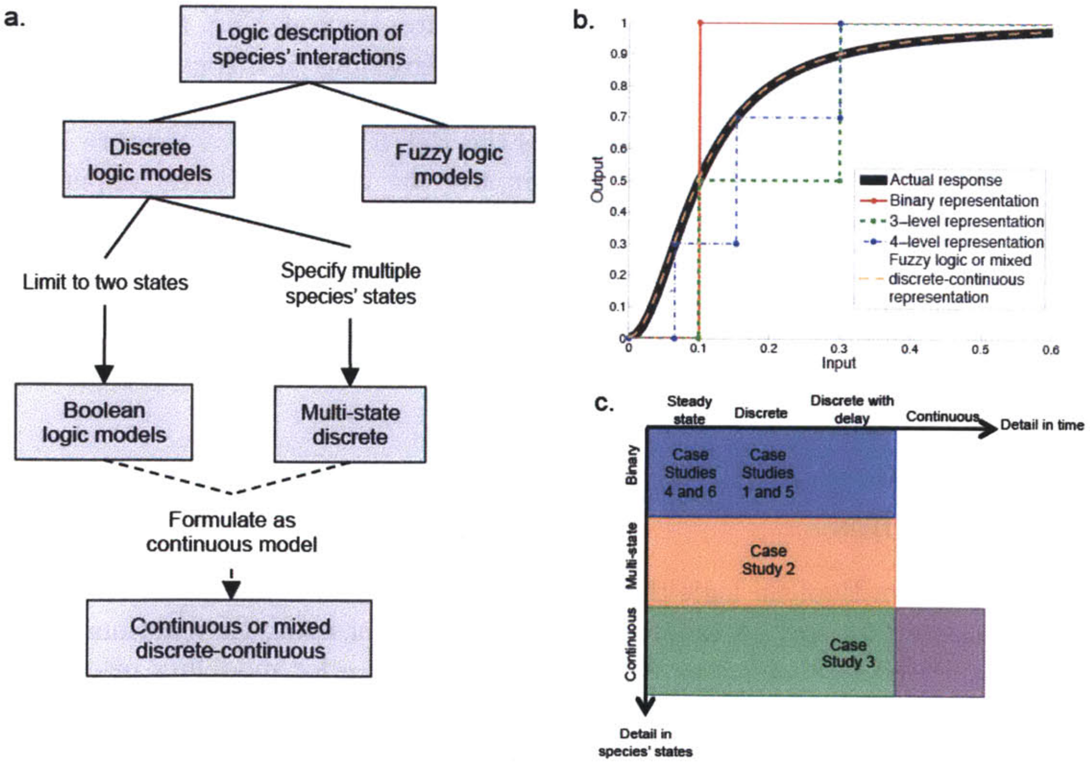
Consider the graphical representation of a signaling network common to protein interaction networks (Figure 1-1a): the nodes in the graph represent proteins, and the edges represent interactions. Such a graph depicts nodes that interact physically or have correlated expression or genetic profiles (depending on the underlying data source) but do not allow us to explicitly compute the state of activity of individual nodes given different inputs or initial network states. Performing such a calculation requires information on how each node reacts to the activities of its input nodes. In logic-based models, these dependencies are specified by 'gates' (Figure 1-1b) which, in Boolean logic, are specified by 'truth tables' that list output states for all possible combinations of input states (Figure 1-1c). Figure 1-1d shows the truth tables of the OR, AND, and NOT Boolean logic gates as well as a small network in which gates are assembled to create the AND-NOT logic gate.

To illustrate how logic-based modeling can be applied to a biological network, consider a hypothetical representation of epidermal growth factor receptor (EGFR) and several downstream proteins (Figure 1-1e). This toy network is too simple to be realistic but demonstrates several issues of importance when building a logic-based model. Either epidermal growth factor (EGF) or heregulin (HRG) can bind to and activate EGFR (Figure 1-1d,e). EGFR then stimulates the Raf/ERK and PI3K/AKT pathways (the multitude of known biochemical interactions in this case are modeled as a single 'activating' edge). ERK activity inhibits the EGFR-dependent PI3K activation whereas AKT positively regulates the Raf/ERK pathway (Figure 1-1d,e). With this information it is possible to compute the response of the unperturbed network to a given input as well as responses resulting from inhibition of a node (by a drug for example). However, under all simulated conditions (EGF or HRG alone or in combination, Figure 1-1f), the network response is the same. This is to be expected because binary logic cannot encode the differential sensitivities of EGFR to EGF and HRG, a point we return to below.

1.3.2 Modeling non-discrete processes using logic-based approaches

The assumption in Boolean logic that all species are either on or off (states 1 or 0) is clearly an unrealistic way to represent binding curves or catalytic reactions. Fortunately, logic-based modeling provides several approaches for modeling intermediate states of activity (Figure 1-2a). Multi-state discrete models specify additional levels between 0 and 1, whereas fuzzy logic allows for continuous node states. In fuzzy logic, which has found wide utility in industrial control systems, a set of user-defined functions transforms discrete logic statements into relationships between continuous inputs and output levels. Other methods of describing discrete or Boolean logic models

Figure 1-2: Description of logic-based formalisms. a. Description of various forms of logic-based models: All logic-based models describe species' interactions in terms of logical statements (or rules). Discrete logic can specify two or more levels for each modeled species whereas Boolean logic specifies only two levels of each species. In addition to these logic-based formalisms, various methods of describing discrete or Boolean logic models with piece-wise continuous equations [39] or logic-based ODEs [159] have been successfully implemented to represent biochemical signaling networks. b. Approximation of input-output relationship between hypothetical biological species (black solid line) with binary (red solid), ternary (green dashed), and quaternary (blue dash-dot) discrete logic gates. Various thresholds could be chosen for each discrete gate; chosen thresholds are purely hypothetical. c. Plane of granularity in species' states and treatment of time: Regions containing various logic-based modeling variants are denoted by shaded boxes. Boolean networks (blue region) and some discrete logic-based networks (orange region) are binary but their treatment of time ranges from logic steady state to discrete with delays. Discrete models with multiple species state cover a similar range in possible treatments of time. Fuzzy logic models (green region) describe a continuous range of species' states with the same range of time granularity. Conversion of Boolean or discrete models into logic-based ODEs, piecewise linear, and standardized qualitative dynamical system (purple region) result in models that are continuous in both species' states and time. Each case study is placed on the landscape according to how it represents the biological system of interest with a logic-based network.



as continuous or mixed discrete-continuous have also been implemented successfully (Figure 1-2a, dashed lines) [159, 39, 96].

How is a prototypical biological interaction approximated using discrete and non-discrete logic formalisms? In Figure 1-2b, a sigmoidal relationship between input and output level (e.g. a protein kinase acting on a substrate, black solid line) is approximated by a binary (red solid), ternary (green dashed), and quaternary (blue dash-dot) discrete logic functions. Fuzzy logic and mixed discrete-continuous logic can closely approximate the real response (orange dashed). It is important to note however, that the increased realism of multi-state or Fuzzy logic modeling comes at the cost of increased complexity, typically in the form of a threshold or transfer function having free parameters that must be estimated.

1.3.3 Treatment of time in a logic-based model

Figure 1-1g provides an example of how multi-state discrete logic can be used to represent the differing states of activation of EGFR when exposed to EGF and HRG stimulation, where an additional activation level of 'two' indicates that EGFR is more sensitive to EGF than HRG. In the model, addition of HRG alone causes AKT and ERK activity levels to oscillate (Figure 1-1h, right panel). These oscillations are caused by the negative feedback between ERK and PI3K. However, when either EGF alone or both EGF and HRG are present (Figure 1-1h, left panel), EGFR is in activation state two and the negative feedback inhibiting PI3K is absent. Thus, oscillations are not observed.

The presence of oscillations in this and other logic-based networks complicates analysis, and the actual form that the oscillations take depends on the treatment of time during the simulation. Logic-based models represent time with varying degrees of detail. We present this concept graphically in Figure 1-2c, where each modeling formalism is classified according to the detail in its representation in species' state and time. Table 1.1 presents a comparison of the approaches in tabular form. The activity of each species in discrete logic-based network simulations is determined by its input node states at some previous time step. The order in which node states are updated results in an implicit treatment of time scales.

Two primary node-updating schemes exist: synchronous and asynchronous [147, 146, 36]. Synchronous updating updates every node at each time step according to the states of its input nodes at the previous time step whereas asynchronous updating updates node states in random order. In practical terms, asynchronous updating involves updating an output node based on some of its input nodes at the current and others at a previous time step. Variants of both synchronous and asynchronous updating exist. Time delays can be specified with synchronous updating, allowing for a more refined description of dynamics. A variant of asynchronous updating, mixed asynchronous updating, allows some nodes be updated before others, making it possible to represent separation of time scales for fast (e.g., binding, phosphorylation) and slow (e.g., degradation, transcription) reactions, similarly to time delays[71]. Regardless of the updating scheme, it is frequently observed that logic-based models will settle into an 'attractor state' in which states no longer change (logic steady

Table 1.1: Description of logic modeling variants. Discrete time steps could use synchronous or asynchronous updating with or without delay or be examined at steady state. *Biochemical signaling network +Genetic network

Logic Modeling Variant	**Time Treatment**	**Detail of species' states**	**Use in biological modeling**
Boolean	Discrete time steps	Binary	*[53, 34, 41, 47, 82, 124, 128, 98] +[82, 15, 31, 81, 24]
Discrete logic	Discrete time steps	Multi-state; user-defined	*[87, 161, 134, 96] +[161, 93, 94]
Fuzzy Logic	Discrete time steps or time can be treated as a variable	Multi-state; user-defined and implicit in calculation of output state	*[10, 3, 55, 168]
Piece-wise linear	Continuous	Multi-state; user-defined and implicit in equations	+[15, 11]
Logic-based ODEs	Continuous	Multi-state; implicit in ODE equations	*[159]
Standardized qualitative dynamical systems	Continuous	Multi-state; implicit in formalism	*[116]

state) or states cycle in a pattern of activity (the oscillations in the example network are an example of a cyclic attractor state; Figure 1-1h). The continuous or mixed discrete-continuous methods mentioned previously formulate discrete logic as ordinary differential equations or piecewise-linear equations, respectively. This treatment allows one to model both species' state and time as continuous (Figure 1-2c) but at the cost of increased model complexity. Research into the influence of updating scheme on the segment polarity network of Drosophila melanogaster [15] and mammalian cell cycle [31] network have demonstrated that the different treatments of time can lead to unique biological interpretations. Generally, the most appropriate updating scheme is dependent on the type of model built as well as the questions that the model is meant to address.

Another extension of logic-based modeling aims to incorporate probabilistic interactions [137, 37]. This method allows one to account for uncertainty in knowledge of signaling networks as well as stochasticity in biological systems. Also noteworthy are a number of efforts to apply related formalisms such as Petri nets, cellular automata etc. to biological networks [32]. In some cases, these formalisms can be reduced to logic-based formalisms, and they provide an additional level of abstraction that makes it possible to perform formal network analysis[1]. Because these probabilistic and computational techniques involve slightly different considerations than previously discussed, we do not describe them further and instead point the interested reader to the references listed above.

This chapter focuses on a qualitative description of various logic-based formalisms. For readers interested in the actual computational procedures involved in carrying out these methods, an outline is provided Supplementary Figure A-1. Additionally, several dedicated software packages have been developed for logic-based modeling of biochemical signaling networks with varying degrees of detail and differing updating schemes; some of these are listed in Table 1.2. We refer the interested reader to the references in this table for descriptions of each simulation procedure, in particular the quantitative approaches not described here.

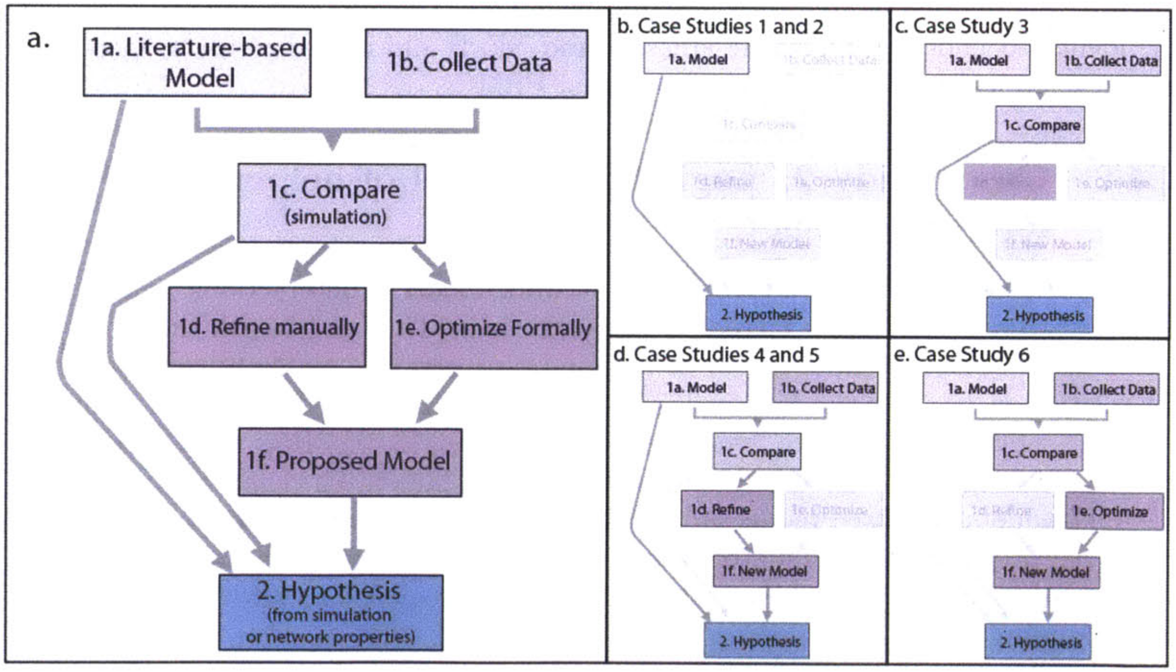
1.4 Applications of logic-based models to biochemical networks: Case Studies

Below we discuss six logic-based models of signal transduction as a means to highlight different methods, biological questions and opportunities for future development; we necessarily omit many details. Figure 1-3a shows a general workflow for applying logic-based modeling to signaling networks and serves as a means to summarize the key features of each case study: (i) Case studies 1 and 2 involve models built solely from literature-based prior knowledge (Figure 1-3b); (ii) Case 3 involves a comparison of models to data (Figure 1-3c); (iii) Cases 4 and 5 use manual refinement to fit experimental data to a fuzzy (case 4) or Boolean (case 5) logic-based model (Figure 1-3d); and, (iv) Case 6 presents a formal method for model optimization based on refining a literature-based Boolean model against high-throughput data (Figure 1-3e).

Table 1.2: Tools available for the logic modeling of biochemical signaling networks

Tool	**Type of logic**	**Functionality**	**Treatment of time**	**Ref**
BooleanNet	Boolean	Simulation and visualization	Synchronous, asynchronous, or piecewise-continuous	(56)
GinSim	Discrete (multi-state)	Model building, simulation and analysis	Synchronous, asynchronous, or mixed asynchronous	[14, 40]
CellNet Analyzer	Boolean (multi-state)	Network properties analysis	Logic-steady state	[71, 70]
CellNet Optimizer	Boolean	Model refinement	Logic-steady state	[124]
Odefy	Boolean and Logic-based ODEs	Simulation and visualization	Synchronous, asynchronous, or continuous	[159]
Genetic Network Analyzer	Piecewise-linear differential equations	Model building and simulation	Continuous	[26]
ChemChains	Boolean	Model simulation, visualization, and analysis	Synchronous (with delays) or asynchronous	[48]
MetaReg	Discrete (3 states)	Simulation and visualization; model refinement	Logic steady state	[151]
SQUAD	Standardized qualitative dynamical systems	Model simulation and analysis	Continuous	[95, 27]

Figure 1-3: Workflow of application of logic-based models to answer biological questions. a. General workflow. The workflow is divided into two phases: an initial model-building phase (purple boxes) and model prediction phase (blue box). Hypotheses are made from models built either from literature (box 1a) or from a comparison of a literature-based model with data (boxes 1a-c). In some cases, the models are refined manually (box 1e) or optimized formally (box 1f) with data and then used to make hypotheses (boxes 1a-f). b. Workflow of case studies 1 and 2. These case studies analyze network properties of logic-based models built from the literature and use them to make experimentally testable predictions. c. Workflow of case study 3. This case study compares the results of experiments to simulation results of a logic-based network to make predictions. They also analyze the network properties of their logic-based network. d. Workflow of case studies 4 and 5. Both case studies manually refine their models based on experimental data and, prior to refining, case study 5 first uses a model built from the literature to predict experimental outcome. e. Workflow of case study 6. This case study compares logic-based models to experimental data and presents a formal method of training a Boolean network model to data.



1.4.1 Case Study 1: Boolean logic model of leukemic T cell large granular lymphocytes [165]

Zhang et al. use a Boolean network model constructed from the literature to ask which proteins in leukemic T cell large granular lymphocytes (T-LGL) should be inhibited to induce apoptosis. Simulation of a 58-node logic model of the T-LGL survival-signaling network is used to address the following questions (i) What are minimal stimulation conditions that recapitulate observed deregulation of the T-LGL network and (ii) What perturbations might reverse deregulation and promote apoptosis?

A literature survey and experimental observations were combined to assemble a Boolean logic network describing signaling in T-LGL that affected cytoskeleton signaling, apoptosis, and proliferation. Simulations were compared when all nodes were free to vary and when some nodes were fixed (i.e. set to active or inactive and not allowed to change during the asynchronous updates). When the appropriate nodes were fixed, the model correctly recapitulated the situation in which leukemic T-LGL failed to undergo activation-induced cell death. Model analysis predicted a minimum set of stimuli that would result in the deregulated survival signaling previously observed in leukemic T-LGL. Experimental inhibition of this network state was shown to induce apoptosis in leukemic but not normal peripheral blood mononuclear cells (PBMCs). Intriguingly, the authors identified nodes whose activation or inactivation caused the apoptosis node to be activated. These nodes are potential therapeutic targets for induction of apoptosis in leukemic T-LGL. Chemical knockdown of two of the identified nodes, Sphingosine kinase 1 and NFκB, did indeed result in increased apoptosis in leukemic T-LGL but not normal PBMCs.

1.4.2 Case Study 2: Logic-based model of helper T cell differentiation [96]

Mendoza used a literature-derived logic network model of interactions among five cytokines and transcription factors in helper T Cells (Th Cells) to ask the following questions. (i) Do the final states of a logic-based network correctly represent the differentiation fates of the helper T cell (Th cell)? (ii) How do feedback loops in cytokine signaling interact to generate specific cell fates? (iii) How does perturbing nodes of the logic network change the differentiation fate of Th cells?

A 17-node logic-based model of the Th regulatory network was constructed from published literature and simulated under all combinations of initial node states until logic steady states were achieved. This analysis revealed four steady states: one corresponding to Th0 cells, one corresponding to Th2 cells, and two corresponding to Th1 cells. The Th1 cell attractors differed in their level of secretion of IFNγ but their level of IFNγ receptor was the same, a result supported by the literature. The feedback circuits that caused the network to reach these steady states were shown to correspond to experimental conditions known to induce Th0 cells to differentiate into Th1 or Th2 cells. Moreover, literature data validated several predictions based on single node perturbations that corresponded to deletion or over-expression.

This paper illustrates the utility of logic-based modeling when analyzing a network involving many positive and negative interactions whose net effect is not intuitively obvious. This type of model could be used to answer a number of interesting biological questions. For example: After a cell has entered one steady state, what cytokines or inhibitors must be present to switch it to another state? How might systemic cytokine administration affect the Th cell population? Can manipulation of normal nodes compensate for defects in nodes mutated in disease?

1.4.3 Case Study 3: Boolean logic model of ErbB receptor phospho-protein signaling data [131]

In the examples cited above, no direct link exists between the construction of the logic-based model and experimental data (Figure 1-3b). In contrast, Samaga et al. directly compared the outputs of a Boolean logic model constructed from the literature to data collected from cells (Figure 1-3c). The authors first developed a strategy for converting a biochemical network into Boolean logic. They then used this method to construct a complex Boolean logic model from a canonical graph of ErbB signaling that has been assembled by Kitano and colleagues [108]. Finally, they asked: Is the constructed Boolean model consistent with data from cells stimulated with ErbB ligands?

Model construction and simulation in Samaga et al. [131] was performed using the toolboxes ProMoT [127] and CellNetAnalyzer (CNA) [70] and data were obtained by exposing HepG2 liver cancer cells and primary human hepatocyte to various ErbB ligands in the presence and absence of specific small-molecule kinase inhibitors. Inconsistencies between model prediction and experimental observation generated a set of eleven hypotheses regarding ErbB signaling in HepG2 and primary cells. Five of the eleven were supported by literature (although not in the cell types used in this study); five pointed to the need to remove or add interactions in the network; and one suggested that a small molecule inhibitor did not have the expected specificity. Significantly, this work successfully converted a biochemical map into an executable logic-based model and then used experiments to explore model topology.

1.4.4 Case Study 4: Fuzzy logic model of protein signaling data [3]

As a means to analyze a set of continuous data, Aldridge et al. [3] built a fuzzy logic model of multiple growth factor and cytokine pathways based on prior literature knowledge and then refined the model manually based on measurements of signaling protein phosphorylation in cells treated with TNFα, EGF and insulin. During the model-building process, the authors asked: What interactions between TNFα and growth factors best explain experimental data?

This data consisted of total or phospho-protein levels for eleven signaling proteins following exposure of cells to TNFα, EGF and Insulin individually or in combination at thirteen time points from zero to 24 hours. Because Boolean logic was unable to capture important intermediate states of protein modification in the data, fuzzy logic modeling was used. Fully-implemented fuzzy logic is much more flexible than Boolean logic. Thus, the authors first selected a limited number of ways to represent interactions. Manual data fitting was used to optimize the interactions in the model and the shapes of the functions relating input and output species in the fuzzy logic gates. Time was included as a variable ("early" or "late") and time delays were included in the logical rules for several gates. Acceptable values for these delays were determined manually. During the model-building process, the authors uncovered unexpected interactions between ERK and IκK activities. This work demonstrates that fuzzy logic can be used to model and gain insight into signaling data that was not obvious from either inspection or partial least squares regression modeling. The authors also note that because manual fitting of large datasets to a fuzzy logic model is an arduous process, methods are required to automate the fitting process.

1.4.5 Case Study 5: Integration of logic-based modeling with experimental study of Trastuzumab-resistant breast cancer [129]

Sahin et al. first used a literature-derived Boolean logic model of a chemotherapeutic resistant cell line to ask the question: Knockdown of which molecular species will result in increased drug sensitivity? Because the model was unable to accurately predict experimental results, they attempted to deduce the network from experimental data but concluded that the most reliable network was one that they had manually refined (Figure 1-3c).

Trastuzumab is a monoclonal antibody against ErbB2 that has successfully treated a subset of ErbB2 positive breast cancers. However, two thirds of patients are Trastuzumab-resistant from the beginning of treatment. The authors hypothesize that this resistance is conferred by an escape from G1 cell cycle arrest. A Boolean logic network model of ErbB receptor regulation of the G1/S cell cycle transition was constructed based on published literature. Only the ErbB receptor dimerization events that were possible in the cell line model of Trastuzumab resistance were included in the model and initial node states were set based on the biological activity of the proteins in their experimental system, making the model specific to the experimental system of interest, a clear benefit for modeling a context-sensitive phenomenon such as Trastuzumab resistance, which is context sensitive.

The retinoblastoma protein (Rb) is phosphorylated under conditions of constant EGF stimulus and was postulated to allow cells to escape G1 cell cycle arrest. The model was queried to identify those nodes whose inactivation under conditions of constant EGF would result in pRb dephosphorylation and consequent G1 cell cycle arrest (resulting in restoration of Trastuzumab sensitivity). RNAi knockdown of all but two species in the network (including those not predicted a priori to confer Trastuzumab sensitivity) was then used to test model-based predictions, several of which were found to be correct. Manually refining a single logical rule substantially improved accuracy, correctly predicting all but one RNAi knockdown result. The authors attempted to reverse engineer the network using protein array data but were unable to explain this final inconsistency. Overall, this work nicely illustrates the power of integrating experimental and logic-based modeling to gain a more complete understanding of the system of interest. As with case study 4, it also points to a need for more reliable methods of training of logic-based networks.

1.4.6 Case Study 6: Training a Boolean logic model of HepG2 signaling [124]

The primary advance in Saez-Rodriguez et al. [124] is the development of a formal method for optimizing logic-based models against experimental data, implemented in the CellNetOptimizer software. The data in this case was fairly extensive, comprising 1000 phospho-protein measurements of sixteen signaling proteins in tumor cells stimulated with one of six growth factors or inflammatory cytokines (TGFα, IGF1, TNFα, IL1α, LPS, and IL6) in the presence or absence of one of seven small molecule kinase inhibitors. The starting point for model construction was a signed directed graph comprising 82 nodes and 116 interactions derived from pathways in the Ingenuity IPA software. The authors then asked: (i) Can a formal training process be developed to increase the predictive capacity of the nave model? (ii) Is the number of interactions in an optimized network similar to or smaller than the number in the nave model? (iii) Can interactions absent from the initial graph be indentified that increase predictive power? It was observed that data-optimized models contain many fewer interactions than the original network graph, suggesting the presence of many false-positive interactions at least for the HepG2 cells under study. Moreover, addition of a small number of links deduced directly from data improved predictive capacity while increasing model size only modestly. Support for these links was subsequently found in the literature. This work represents a first step in using logic-based models to generate executable models of network graphs and then refining the models to increase their reliability in specific cellular contexts. Direct extension of the methods should make it possible to compare different cell types directly and perhaps even identify drugs that affect diseased but not normal cells.

1.4.7 Outline of Thesis Work

In this chapter, we described how logic-based models can be used to represent biochemical signaling networks and illustrate some of the questions that logic-based modeling can address. The ability of discrete and fuzzy logic models to determine the effects of protein over-expression or inhibition on phenotype, elucidate network properties, and identify the network that best describes high-throughput experimental data has been illustrated with case studies. However, as illustrated by case studies 4 and 5, the ability to gain significant biological insights with the models was frequently hampered by the lack of a method to rigorously train the logic model to data. While this was accomplished in case study 6 for Boolean logic models, the inability to accurately model intermediate values is a significant limitation of the approach.

In the following chapters of this thesis, we present a novel logic modeling formalism called 'constrained fuzzy logic' (cFL) that retains the ability of traditional fuzzy logic to fit intermediate values but constrains the modeling flexibility such that it is able to efficiently model biological systems. We first use cFL to train logic models to data by altering the approach presented in case study 6 (Chapter 2). We demonstrate the the ability of cFL to fit intermediate values was crucial for accurately determining model topology and gaining important biological insights from our data. We perform a preliminary investigation of the ability of cFL models to predict species states in Chapter 2, and in Chapters 3 and 4 we significantly extend this capability by developing a complementary software tool called 'Querying Quantitative Logic Models' (Q2LM) that uses cFL model simulation results to determine therapeutic perturbations predicted to be effective at accomplishing a clinical goal in different contexts. We use Q2LM to answer these questions with cFL models constructed based solely on prior knowledge (Chapter 3) or trained to data (Chapter 4). Finally, we conclude with an examination of further opportunities for development and application of this modeling technique (Chapter 5).

Chapter 2

Training a constrained fuzzy logic model to data [102]

2.1 Summary

Predictive understanding of cell signaling network operation based on general prior knowledge but consistent with empirical data in a specific environmental context is a current challenge in computational biology. Recent work has demonstrated that Boolean logic can be used to create context-specific networks models by training proteomic pathway maps to dedicated biochemical data; however, that formalism is restricted to characterizing protein species as either fully active or inactive. To advance beyond this limitation, we propose a novel form of fuzzy logic sufficiently flexible to model quantitative data but also sufficiently simple to efficiently construct the models by training pathway maps on dedicated experimental measurements. Our new approach, termed constrained fuzzy logic (cFL), converts a prior knowledge network (obtained from literature or interactome databases) into a computable model that describes graded values of protein activation across multiple pathways. We train a cFL-converted network to experimental data describing hepatocytic protein activation by inflammatory cytokines and demonstrate the application of the resultant trained models for three important purposes: (a) generating experimentally testable biological hypotheses concerning pathway crosstalk, (b) establishing capability for quantitative prediction of protein activity, and (c) prediction and understanding of the cytokine release phenotypic response. Our methodology systematically and quantitatively trains a protein pathway map summarizing curated literature to context-specific biochemical data. This process generates a computable model yielding successful prediction of new test data and offering biological insight into complex datasets that are difficult to fully analyze by intuition alone.

2.2 Background

Signaling networks regulate cell phenotypic responses to stimuli present in the extracellular environment [63]. High throughput 'interactome' data provide critical information on the composition of these networks [156, 112, 28], but understanding their operation as signal processing systems is strongly advanced by direct interface with dedicated experimental data representing measured responses of biochemical species in the network (proteins, mRNA, miRNA, etc.) to stimulation by environmental cues in the presence or absence of perturbation [38, 58, 97, 84]. Immediate early responses are dominated by protein post-translational modifications (we focus here on phosphorylation), assembly of multi-protein complexes, and changes in protein stability and localization. Such responses are typically highly context dependent, varying with cell type and biological environment. A critical question for the field is how large scale measurements of these responses can be combined with a signed, directed protein signaling network (PSN) to better understand the operation of complex biochemical systems [85].

PSNs are typically deduced by manual or automated annotation of the literature (e.g. [65]) or directly from high-throughput experimental data (e.g. [143, 121, 139]) using a variety of computational techniques. PSNs are represented as node-edge graphs [117], and although they provide high-level insight into the composition and topology of regulatory networks [83, 30, 135, 13, 16, 111], as currently constituted PSNs are not readily 'computable' in that they cannot be used to calculate activation states of the key proteins in a pathway given a set of input cues, nor can quantitative relationships between pathways be determined. This restricts the utility of PSNs for explicit prediction of responses and makes it difficult to compare network representations to functional experimental data. A chief motivation of our current work is to determine how information encoded in a PSN can be made computable and compared to experimental data from a specific cell type, resulting in a context-specific network model.

Logic-based models (e.g. [128, 96, 165, 129, 131, 12]; reviewed in [103, 157]) offer one means for converting interaction maps into computable models. We have previously used Boolean logic (BL) to convert a literature-derived signed, directed PSN (comprising for this purpose a 'prior knowledge network' [PKN]) into a computable model that could be compared to experimental data consisting largely of the phospho-states of signal transduction proteins in the presence of different ligands and drugs [124]. This approach allowed us to determine which links in the PKN were supported by the data, and generated models that were useful in making predictions about network topology [124, 125] and drug targets [98]. However, Boolean logic has a significant limitation, since real biochemical interactions rarely have simple on-off characteristics assumed by Boolean logic. Thus, we require a means to encode graded responses and typical sigmoidal biological relationships in a logical framework.

One way to accomplish this is to apply traditional fuzzy logic [FL], as demonstrated previously in modeling continuous input-output relationships to encode a complex signaling network [10, 3, 55]. In the realm of control theory, FL modeling is an established technique for predicting the outputs of complex industrial processes when the influences of inputs cannot be characterized precisely [164, 149, 153]. A central feature of FL is that it accounts for graded values of process states using a virtually unlimited repertoire of relationships between model species or components. However, for past application to biochemical signaling networks, the flexibility of conventional FL modeling necessitated that the network topology be fixed prior to either manual [10, 3] or computational [55] parameter fitting, rendering a formal training of network topology to experimental data infeasible.

In this chapter we develop and employ a new approach to fuzzy logic modeling of biological networks that we term 'constrained fuzzy logic' [cFL] for descriptive purposes. A key feature of cFL modeling is that it limits the repertoire of relationships between model species, enabling the formal training of a PKN to experimental data and resulting in a quantitative network model. To maximize broad dissemination across the computational biology community, we implement cFL in an exisiting software tool CellNetOptimizer v2.0 (CellNOpt), significantly extended to accommodate the further requirements of cFL while maintaining the BL analytic approach (freely available at http://www.ebi.ac.uk/saezrodriguez/software.html). We demonstrate the value of the CellNOpt-cFL method by elucidating new information from a recently published experimental dataset describing phospho-protein signaling in HepG2 cells exposed to a set of inflammatory cytokines [5]. We show that a cFL model can be trained against a training dataset and then validated by successful a priori prediction of test data absent from the training data. We also establish the benefits of cFL relative to BL in three key areas: (a) generation of new biological understanding; (b) quantitative prediction of signaling nodes; and (c) modeling quantitative relationships between signaling and cytokine release nodes. Particular examples of validated biological predictions include: (i) TGFα-induced partial activation of the JNK pathway and (ii) IL6-induced partial activation of multiple unexpected downstream species via the MEK pathway. Our work demonstrates the technical feasibility of cFL in modeling real biological data and generating new biological insights concerning the operation of canonical signaling networks in specific cellular contexts.

2.3 Results

2.3.1 Constraining fuzzy logic

Fuzzy logic is a highly flexible methodology to transform linguistic observations into quantitative specification of how the output of a gate depends on the values of the inputs [164, 42, 107, 43]. For example, in the simplest, 'Sugeno' form of fuzzy logic, one specifies the following quantities: 'membership functions' designating a variable number of discrete categories ('low, medium, high' etc.) as well as what quantitative value of a particular input belongs either wholly or partially to these categories; 'rules' designating the logical relationships between the gate inputs and outputs; AND and OR 'methods' designating the mathematical execution of each logical relationship; 'weights' designating the credence given any rule; and 'defuzzification' scheme designating how a final output value is determined from the evaluation of multiple rules [140]. This flexibility is important in industrial process control [23], which aims to use uncertain and subjective linguistic terms to predict how a controller should modulate a process variable to achieve desired process outputs.

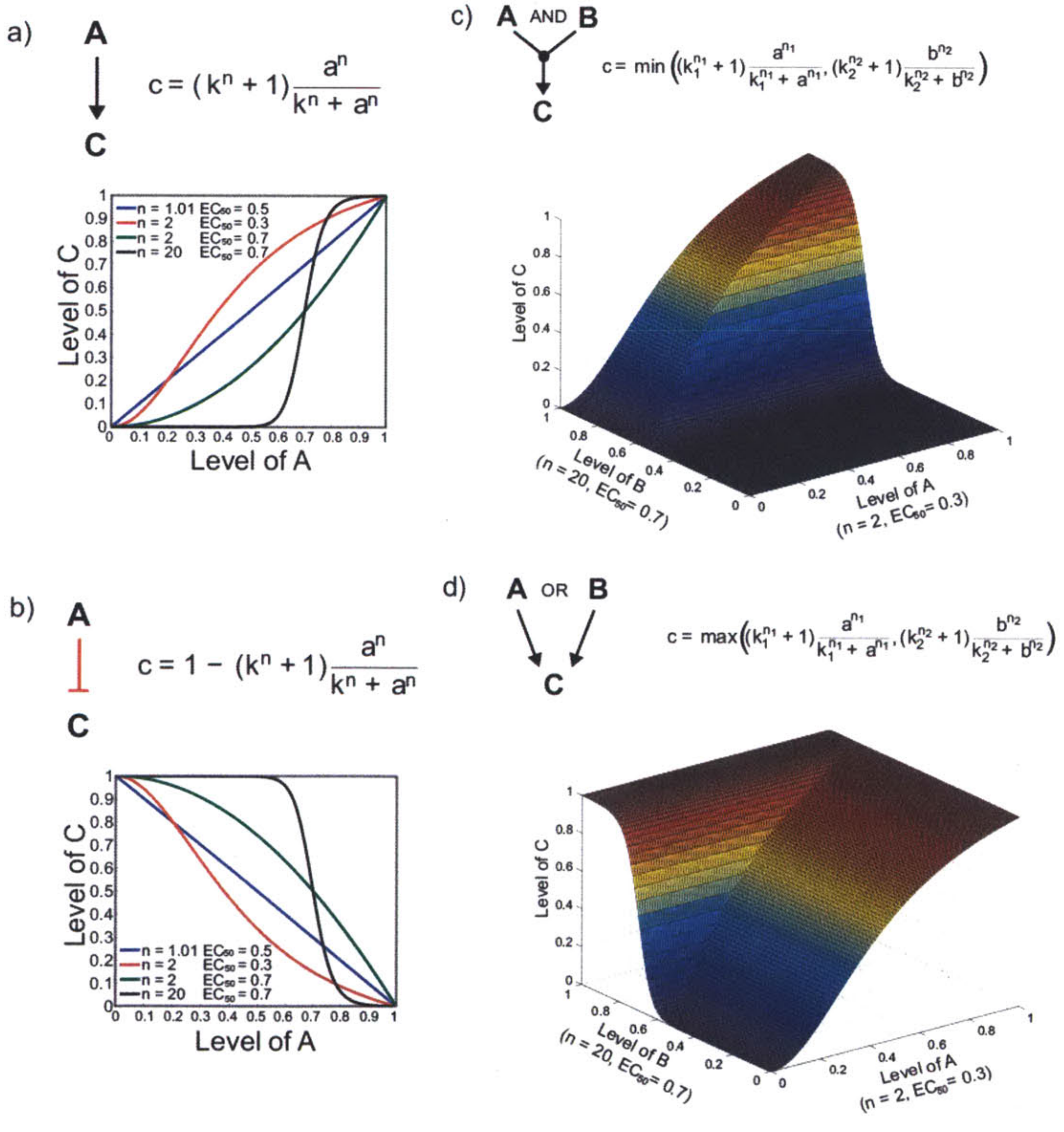
However, our goal is to train models on quantitative biological data that are inevitably incomplete in the sense that (i) measurements are not obtained under all possible conditions and (ii) available data are not sufficient to constrain both the topology and quantitative parameters of the underlying networks. Accordingly, we sought to develop a fuzzy logic system that minimizes the number of parameters to avoid over-fitting and simplifies the logic structure to facilitate model interpretability. Because we aim to represent relationships among proteins in enzymatic cascades, mathematical relationships should be need biologically relevant. We therefore use a simple Sugeno fuzzy logic gate with a defined form (see Supplementary Text B.1) based on transfer functions (mathematical functions describing the relationship between input and output node values) that approximate the Hill functions of classical enzymology.

Our 'constrained' fuzzy logic (cFL) framework uses a simplified fuzzy logic gate that is best described by the mathematical representation in Figure 2-1. The value of an output node of a one-input positive interaction is evaluated using a transfer function. In this work, 'input-output' refers to the nodes of a specific cFL logic gate, where 'node' are molecular species. We use 'model inputs' and 'model outputs' to refer to the overall relationship between model inputs such as ligand stimulation of cells and the collective output of the network (protein modifications or phenotypic states in our application). The transfer function underlying cFL gates is a normalized Hill function with two parameters: (1) the Hill coefficient, n, which determines the sharpness of the sigmoidal transition between high and low output node values and (2) the sensitivity parameter, k, which determines the midpoint of the function (corresponding to the EC_{50} value in a dose-response curve, Figure 2-1a). A negative interaction is represented similarly, except that the transfer function is subtracted from one, effectively inverting it (Figure 2-1b). Varying these parameters allows us to create a range of input-output transfer functions including linear, sigmoidal and step-like (Figure 2-1a). Moreover, this transfer function is biologically relevant: protein-protein interactions and enzymatic reactions can be described by Hill function formulations to a good approximation [51, 155, 130].

In some cases, use of a normalized function is too restrictive for practical application. For example, if model inputs are purely binary (values of either zero or one), the output of a normalized function would also be zero or one, making it impossible for a cFL gate to achieve intermediate states of activation. Accordingly, our cFL method allows for alternative transfer functions. For example, although the method is not limited to binary model inputs, the ligand inputs of our current work are binary (either present or not). If we used normalized transfer functions to relate these model inputs to downstream outputs, all model species would also be either zero or one. Thus, for these transfer functions, we used a constant multiplied by the binary ligand input value (see Materials and Methods 2.5).

If more than one input node influences an output node, this relationship is categorized as either an 'AND' or 'OR' interaction. An AND gate is used when both input nodes must be active to activate the output node, whereas an OR gate is used when either input node must be active. Mathematically, we represent AND behavior by evaluating each input-output transfer function and selecting the minimal possible output node value (i.e., applying the 'min' operator, Figure 2-1c) whereas we select

Figure 2-1: Construction of gates with constrained fuzzy logic (cFL). When node C depends only on node A, a normalized Hill function is used to calculate value of node C, 'c' given value of node A, 'a' where n is Hill coefficient and k is the sensitivity parameter specifying the EC_{50} for each gate. Several representative normalized Hill functions are shown for activating (a) and inhibiting (b) cFL gates. When C has more than one input (A and B, in this case), either an AND (c) or OR (d) gate must be used to model the interaction. In the case of the AND gate, the minimum possible value of C calculated from the transfer functions is used as the output node value. One possible response surface for levels of C given different levels of A and B with two transfer functions is demonstrated (c). For evaluation of an OR gate, the maximum value of C is used as the output node value, with the corresponding response surface (d).



the maximal value ('max' operator; Figure 2-1d) to evaluate an OR gate. Finally, if both AND and OR gates are used to relate input nodes to an output node, our formalism evaluates all AND gates prior to OR gates. This order of operations corresponds to the disjunctive normal or sum of products form [71].

Use of cFL to understand experimental data in the context of a prior knowledge network: CellNOpt-cFL The processing of training a cFL network (CellNOpt-cFL) has two starting requirements. The first is a prior knowledge network (PKN; Figure 2-2, box A). A PKN depicts interactions among the nodes as a signed, directed graph (such as a PSN) and can be obtained directly from the literature. Alternatively, a large number of commercial (e.g., Ingenuity Systems: www.ingenuity.com; GeneGo: www.genego.com) or academic (e.g., Pathway Commons: www.pathwaycommons.org, reviewed in [8]) pathway databases as well as integrative tools (e.g. [78, 77]) can be utilized to construct a PKN. The second requirement is a dataset describing experimental measurements characterizing node activities following stimulation of and/or perturbations in upstream nodes (ligand and inhibitor treatment in our example; Figure 2-2, box B). CellNOpt-cFL is then used to systematically and quantitatively compare the hypothesized PKN to the experimental dataset.

In practice, available experimental data is usually insufficient to fully constrain both the parameters and topology of the cFL models, and CellNOpt-cFL recovers many models that describe the data equally well. Due to this typical absence of firm structural and parametric identifiability [124, 75, 113], we examine families of models that fit the data equally well rather than attempting to identify a single global best fit. Specifically, we examined interactions in the PKN that were either retained or consistently removed by training. We also used individual models to predict input-output characteristics. This treatment allowed us to calculate both an average prediction as well as a standard deviation, which we show below was useful for discrediting inaccurate predictions.

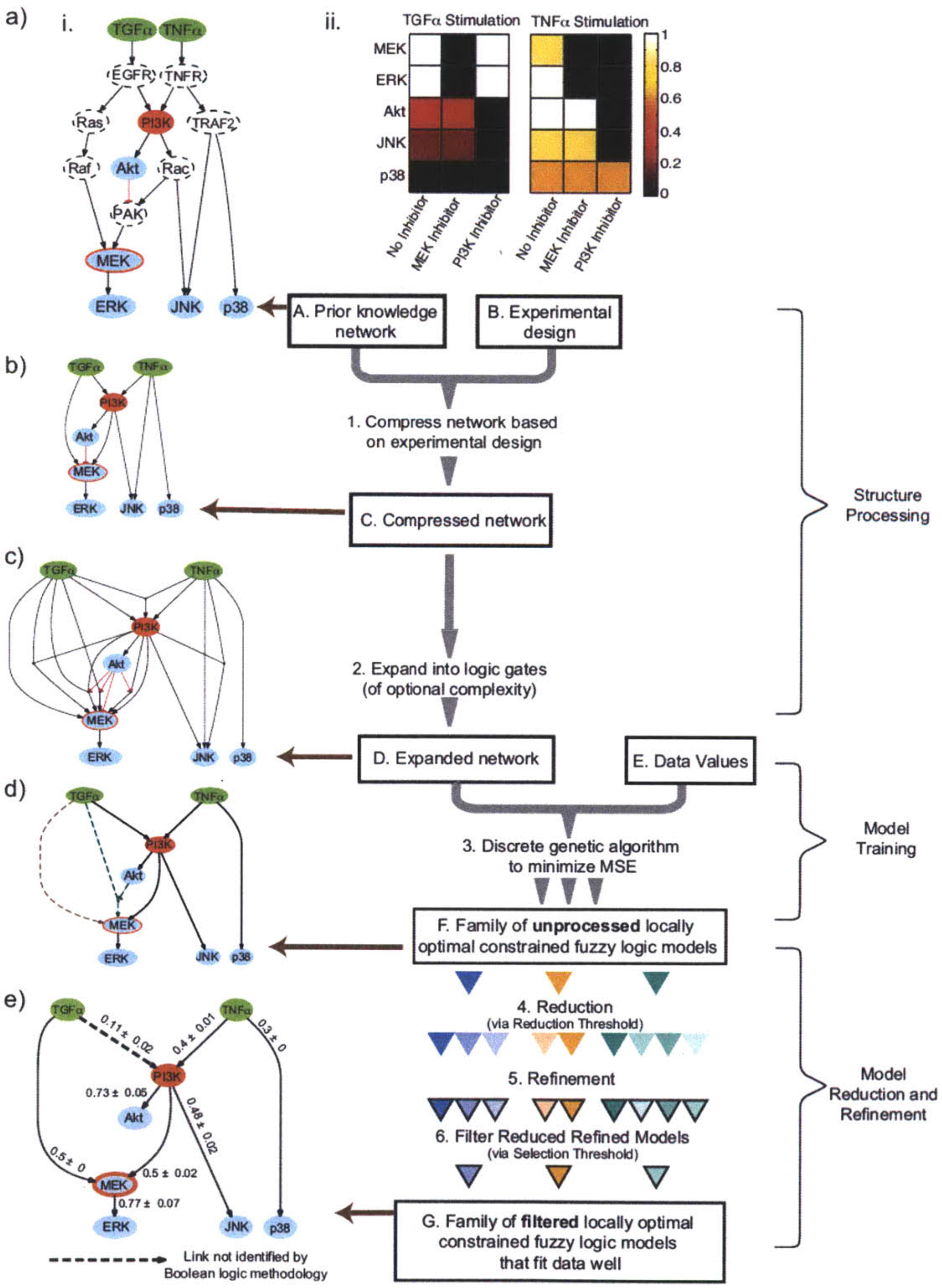
Our method comprises three main stages (Figure 2-2): (1) structure processing converts a PKN into a cFL model; (2) model training trains the model to experimental data; and (3) model reduction and refinement simplifies trained models. To illustrate CellNOpt-cFL, we examine a simple toy problem of training a PKN of the phosphoprotein signaling network response to TGFα and TNFα (Figure 2-2a.i) to *in silico* data of activation of several downstream kinases in response to these ligands in the presence or absence of PI3K or MEK inhibition (Figure 2-2a.ii).

2.3.2 PKN Processing

In the first step, we streamline the network to contain only measured and perturbed nodes as well as any other nodes necessary to preserve logical consistency between those that were measured or perturbed ([124]; Figure 2-2, Step 1), resulting in a compressed PKN (Figure 2-2 box C). In our example, many nodes that were in the original PKN were neither measured nor perturbed experimentally. Because these nodes can be removed without causing logical inconsistencies, they are not explicitly included in the compressed network (Figure 2-2b).

In the second step, we expand the network into the multiple logical relationships (combinations of AND and OR gates) that can relate output nodes to their input nodes (Figure 2-2, Step 2). For example, our toy PKN was expanded to include all possible two-input AND gates governing the response of nodes with more than one possible input node (Figure 2-2c).

Figure 2-2 *(facing page)*: Right side: Workflow (Boxes A through G and Steps 1-6) The methodology requires a dataset that describes some species in the prior knowledge network (PKN; Box A). Based on the data structure of the dataset (Box B), the map is compressed to contain only nodes measured (blue nodes), perturbed (green stimulated nodes and orange inhibited nodes), or necessary to maintain logical consistency between nodes (Step 1). The resultant compressed network (Box C) is then expanded to contain multiple possible logic descriptions of gates connecting more than one input node to a single output node (Step 2). The resultant expanded network (Box D) is compared to the data values (Box E) using several independent runs of a discrete genetic algorithm to minimize MSE (Step 3). Each independent run results in an unprocessed cFL models represented with a grey triangle. This results in a family of unprocessed cFL models (Box F). The result of each independent optimization run is now represented with a different colored triangle. Each individual unprocessed model is reduced with several reduction thresholds (Step 4), resulting in several reduced models (different triangles shadings). The parameters of each reduced model are then refined (Step 5), resulting in reduced-refined models (triangles outlined in black). Finally, one model is chosen to represent each original unprocessed model using a selection threshold (Step 6), resulting in a family of filtered models (Box G). Left side: Application to a toy model (panels a to e). A PKN was hypothesized from the Ingenuity Systems database (www.ingenuity.com) (a.i.) and compared to an *in silico* dataset generated by a simulation of a cFL model with known topology and parameters (a.ii.). The PKN contains 15 molecular species represented as nodes that are believed to positively (arrows) or negatively (blunt arrows) affect others species. These intermediate nodes summarize the possible paths between experimentally stimulated ligands (green) and measured (blue) or inhibited (orange) species. The model was compressed as described in [124] (b) and then expanded to contain all possible two-input AND gates (c). The expanded network was compared to the *in silico* dataset with twenty independent runs of the discrete genetic algorithm. The topologies of the resultant models were identical except in the case of the gate describing activation of MEK (d), with sixteen models modeling this interaction with an activating gate (brown, dashed gate) and four models using an AND-NOT gate (green, dashed gate). The TNFα JNK cFL gate was removed from all unprocessed models, reflecting that this interaction was inconsistent with the *in silico* data. The reduction process (Figure 2-3) showed that the AND-NOT cFL gate could be described more simply without significantly affecting the MSE, resulting in a family of filtered models (e). We have labeled each gate with the sensitivity of the gate (defined in Materials and Methods 2.5), where sensitivity is scaled between zero and one and a higher sensitivity indicates that the output node is more active at lower input node values. All maps and the graphs of cFL models were generated by a CellNOpt routine using the graphviz visualization engine (www.graphviz.org) followed by manual annotation in Adobe Illustrator.


a)
i.
TGFα
TNFα
EGFR
TNFR
Ras
PI3K
TRAF2
Raf
Akt
Rac
PAK
MEK
ERK
JNK
p38
ii.
TGFα Stimulation
TNFα Stimulation
MEK
ERK
Akt
JNK
p38
1
0.8
0.6
0.4
0.2
0
No Inhibitor
MEK Inhibitor
PI3K Inhibitor
No Inhibitor
MEK Inhibitor
PI3K Inhibitor
A. Prior knowledge network
B. Experimental design
1. Compress network based on experimental design
Structure Processing
b)
C. Compressed network
c)
2. Expand into logic gates (of optional complexity)
D. Expanded network
E. Data Values
d)
3. Discrete genetic algorithm to minimize MSE
Model Training
F. Family of **unprocessed** locally optimal constrained fuzzy logic models
e)
4. Reduction
(via Reduction Threshold)
5. Refinement
6. Filter Reduced Refined Models
(via Selection Threshold)
Model Reduction and Refinement
G. Family of **filtered** locally optimal constrained fuzzy logic models that fit data well
0.11 ± 0.02
0.4 ± 0.01
0.3 ± 0
0.73 ± 0.05
0.48 ± 0.02
0.5 ± 0
0.5 ± 0.02
0.77 ± 0.07
Link not identified by Boolean logic methodology

2.3.3 Model Training

In the third step, we train the cFL models to the data (Figure 2-2, Step 3). We start by limiting the possible parameter combinations to a subset of discrete parameter values that specify seven allowed transfer functions as well as the possibility that the input does not affect the output node (i.e. the cFL gate is not present). A discrete genetic algorithm determines transfer functions and a network topology that fit the data well by minimizing the mean squared error (MSE, defined in Materials and Methods 2.5) with respect to the experimental data.

Due to the stochastic nature of genetic algorithms, multiple optimization runs return models with slightly different topologies and transfer function parameters that result in a range of MSEs. Models with an MSE significantly higher than the best models are simply eliminated from further consideration. Models with similar MSEs but different topology and parameters result from the insufficiency of the data to constrain the model such that each model fits the data well albeit with slightly different features. We consider each individual in this family as a viable model, and all are included for subsequent analysis. Thus, after multiple independent optimization runs using the discrete genetic algorithm to train the expanded PKN against the data, a family of models with transfer functions chosen from a discrete number of possibilities is obtained.

For each of these models, we generate unprocessed models (Figure 2-2, box F) by removing all cFL gates that are logically redundant with other cFL gates (e.g., in the gate '(B AND C) OR B activate D', the AND gate is logically redundant with the 'B activates D' gate). These gates are removed because they increase model complexity by using multiple logic gates to encode a logic relationship that can be encoded in a simpler gate.

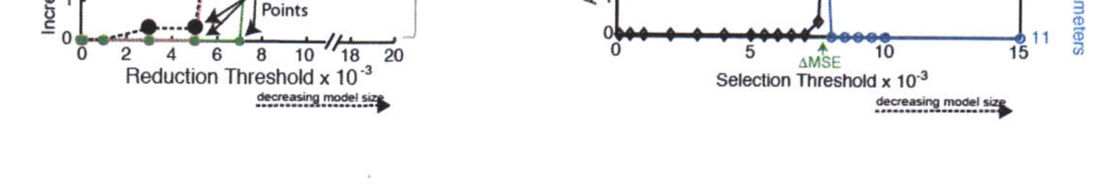
In our toy example, a family of twenty unprocessed models was obtained by training the expanded map (Figure 2-2c) to *in silico* data (Figure 2-2a.ii.) using the discrete genetic algorithm. The unprocessed models from different optimization runs had similar topologies with the exception of the gate describing the relationship of MEK to its input nodes: TGFα and Akt (Figure 2-2d, brown and green dashed gates). Sixteen of the unprocessed models described the activation of MEK as depending only on TGFα (brown, dashed gate) whereas four described activation using the AND NOT gate (green, dashed gate).

2.3.4 Model Reduction and Refinement

In the model reduction and refinement stage (Steps 4-6), we determine which gates can be removed altogether as well as AND gates that can be replaced with one-input cFL gates without significantly affecting MSE. We implemented the non-exhaustive heuristic search procedure described below on each unprocessed model and illustrate its application to our toy example (Figure 2-3).

In the fourth step, we remove or replace all gates for which the alteration does not increase the MSE of the unprocessed model over some threshold, which we term the 'reduction threshold'. We use a range of reduction thresholds such that each

Figure 2-3: Reduction of Trained cFL models. The unprocessed models resulting from twenty independent runs of the discrete genetic algorithm to compare the expanded network to an *in silico* dataset were reduced using several reductions thresholds and subsequently refined. The behavior of three representative models is shown (a). To develop a criterion for our model selection, we note that each individual model exhibits a drastic increase in refined MSE when reduced at some reduction threshold. For our toy model, the MSEs of some reduced-refined models increase significantly (ΔMSE of 7.7×10^{-3}) at a reduction threshold of greater than 5×10^{-3} (a., magenta line), whereas the MSEs of others only increase at a reduction threshold greater than 7×10^{-3} (a., green line). In our toy example, this increase in MSE of 7.7×10^{-3} is deemed significant because it corresponds to the models no longer fitting the *in silico* data of Akt and JNK under TGFα stimulation (the remaining data are still well fit). For each unprocessed model, we refer to the reduction threshold above which a significant increase in MSE is observed as the 'filter point' of the model. Each individual model has a filter point that is determined based on the amount that the reduced-refined models' MSE is allowed to increase. We term this allowable increase in MSE the 'selection threshold'. For example, one model of our toy example (black line) could be described as having a filter point of 1×10^{-3} or 5×10^{-3}, depending on the amount of increase in MSE allowed by the selection threshold. To choose a selection threshold, we compare the average increase in final MSE to the average decrease in the number of parameters in the resultant filtered family of models (b) and note that, at a selection threshold of 7.7×10^{-3}, the average MSE increases while at a selection threshold of 5×10^{-4}, average number of parameters decreases. Thus, a selection threshold of 5×10^{-4} to 7.6×10^{-3} results in the models at the 'filter points' noted in (a).



unprocessed model results in several models, one for each reduction threshold used. Following this step, the resultant models are considered reduced models.

In the fifth step, we fix the model topology to that obtained during Step 4 and treat the transfer function parameters in each reduced model (Figure 2-2, Step 5) as continuous parameters rather than the discrete set of transfer function parameters required for use of the discrete genetic algorithm. We use a Sequential Quadratic Programming method (Supplementary Text B.1) to refine the model parameters and further improve the fit of the models to the experimental data. The resulting models are termed reduced-refined models, which have a range of MSEs depending on the reduction threshold used (Figure 2-3a).

In the sixth and final step, we specify a reduced-refined model to represent each unprocessed model (Figure 2-2, Step 6). For each unprocessed model, we choose the reduced-refined model that has the fewest number of fitted transfer function parameters without increasing the MSE above a defined 'selection threshold'. The selection threshold is chosen by comparing the average number of parameters in the family of models to the average MSE of the models (Figure 2-3b). The net result is a set of reduced-refined-filtered models (hereafter referred to as filtered models, Figure 2-2, Box G).

In our toy example, the filtered models have identical topology and in no case does Akt inhibit MEK activation (Figure 2-2e). This topology is, in fact, the topology from which the *in silico* data was derived. The ability of cFL to fit intermediate values made it possible to recover the correct model topology, whereas BL did not identify the correct model, and a gate linking TGFα to PI3K was consistently missing (Figure 2-2e, dashed arrow). Specifically, BL was unable to return the correct topology because nodes downstream of PI3K (Akt and JNK) were partially activated (0.32 and 0.19, respectively) under conditions of TGFα stimulation, and a BL model that included the TGFα to PI3K gate had a higher error (MSE = 0.56) than a model that omitted the interaction (MSE = 0.07). In contrast, the improved ability of cFL to model graded activities made it possible to recover the true network topology.

2.3.5 Adjusting the complexity of CellNOpt-cFL model training

While the expansion step (Figure 2-2, step 2) captures the many possible combinations of AND and OR logic relationships between nodes, it also increases the complexity of the network, resulting in an increase in the size of the optimization problem. Depending on the biological network of interest, some or most of these AND gates might not be biologically relevant. For example, it is unlikely that six receptors must be active in order to activate another species, as would be the case for a six-input AND gate (instead, it is more likely to be a OR gate). A profusion of AND gates also makes the resultant networks difficult to interpret because most AND gates are in only a few models whereas the majority of models contain single-input and OR gates. Thus, the AND gates can effectively appear as system 'noise', interfering with visual assessment as well as computational analysis of the model topologies. Because of these potential complications, the expansion step can be limited to include only AND gates with a few inputs, depending on the complexity one would like to capture with the trained network models.

In the current chapter, we have limited the search in the discrete genetic algorithm to a set of seven transfer functions in the discrete genetic algorithm. Use of more or fewer transfer functions is possible, but we found that seven transfer functions allowed us to represent a variety of input-output relationships without unduly increasing problem complexity to the point that the discrete genetic algorithm no longer consistently returned models that fit the data well (see Materials and Methods 2.5).

2.3.6 Applying CellNOpt-cFL to protein signaling data from HepG2 cells

To test the ability of cFL modeling to analyze real biological data, we modeled a set of measurements describing the response of HepG2 hepatocellular carcinoma cell line to various pro-survival, pro-death, or inflammatory cytokines in the presence or absence of specific small molecule kinase inhibitors. This dataset was used to construct a recent BL model [124]. Here we ran an independent analysis using the cFL approach and compare the results to the BL previously reported. The dataset comprises measurement of phosphorylation states as a marker of activation of 15 intracellular proteins before and 30 minutes after stimulation by one of six cytokines in the presence or absence of seven specific small molecule kinase inhibitors (Figure 2-4a, Supplementary Figure B-1). The measurements were normalized to continuous values between zero and one using a routine implemented in the MatLab toolbox DataRail [126], as previously described ([124], see Supplementary Text B.1).

The HepG2 dataset was trained to several related PKNs which are enumerated in Table 2.1 and Supplementary Figure B-2 . These PKNs were derived, with various extensions, from the Ingenuity Systems database (www.ingenuity.com) with manual addition of literature data about IRS1 that was obviously missing from the Ingenuity database [124]. The first PKN, termed PKN0 was identical the one used previously for BL modeling [124]. In the course of our analysis, we found it necessary to search the literature for interactions missing in PKN0 but supported by the data, resulting in several PKNs (Table 2.1). Furthermore, we limited the manner in which the PKNs were expanded in two ways: (1) expansion into all possible two-input AND gates or (2) expansion into a two-input AND gate only when one input was inhibitory. In the second case, the expansion of inhibitory gates was necessary because, in logic terms, an inhibitory gate indicates that the output node is active when the input node is not present. In biological networks, this is true if the output node is constitutively active, which was not observed in the normalized HepG2 data. Thus, in order to accurately model the inhibitory effect, it had to occur in conjunction with activation by some other input node, which is captured with an AND gate. If a PKN was processed with both types of expansion, we include a superscript to differentiate between the two cases (i.e., PKN1^a for the expansion of all gates and PKN1^i for the expansion of only the inhibitory case).

2.3.7 CellNOpt-cFL Training of PKN0

PKN0 was expanded to include all possible two-input AND gates and trained to the HepG2 dataset with CellNOpt-cFL (Supplementary Figure B-2). The 90 unprocessed cFL models obtained after training showed that PKN0 exhibited a poor fit to IL1α-induced protein phosphorylation (Supplementary Figure B-3), a result we had also observed with BL analysis [124], confirming that the poor fit of BL was due to errors in the topology of PKN0 and not the inability of Boolean logic to fit intermediate values.

An inspection of systematic model/data disparity (Supplementary Figure B-3) immediately indicated that the models did not fit IL1α-induced phosphorylation of IRS1, MEK and several species known to be modulated by the MEK pathway. In PKN0, no paths between IL1α and MEK or IRS1 were present. Based on careful reading of the literature, we added two links to PKN0: a TRAF6 MEK link [119], and an ERK IRS1 link [162]. These links had been inferred by the BL framework [124] and were supported by further literature evidence. To add a link that provided a path between IL1α and MEK in the absence of BL inference results, for simplicity one

Figure 2-4 *(facing page)*: Training a family of cFL models to the HepG2 dataset. (a) Experimental design of a dataset describing the measured signaling response of the HepG2 cell line to six ligand stimulations in the presence or absence of inhibition of seven species. This dataset was used to train the PKNs (Supplementary Figure B-2) with CellNOpt-cFL. (b) The fraction of edges indicated were randomly removed from (solid line) or added to (dashed line) PKN1^i to result in at least 90 altered PKNs, which were subsequently trained to the HepG2 data. The average MSEs of the altered PKNs indicates that removal of edges reduced the ability of the trained models to fit the data (solid line). Because CellNOpt-cFL does not add links to the model, this result is as expected. The addition of edges to the PKN did not reduce the ability of the trained models to fit the data (dashed line) since edges that were inconsistent with the data could be removed during the training process (Supplementary Figure B-6). (c) Results of ten-fold cross-validation in which the data was randomly divided into ten subsets and the optimization procedure performed to obtain a family of at least 57 models from training data comprising nine of the ten subsets; the remaining subset was considered a test set. We thus obtained ten families of trained models, one family from the use of each subset as a test set. The fit of these families of models to their respective training and test sets was then plotted as a function of the selection threshold. As expected, on average the ability of the trained models to fit the test sets was slightly worse than, but comparable to, the ability to fit the training sets, suggesting that the models were predictive. The difference between MSEs of the test versus training sets did not change as a function of the selection threshold, suggesting that the models were not overfit, even at very low selection thresholds. (d) A comparison of the average final MSE with the average final number of parameters was used to determine a range of selection thresholds (1×10^{-3} through 7.5×10^{-3}) where the family of models has a slightly lower average number of parameters without greatly increasing the MSE.

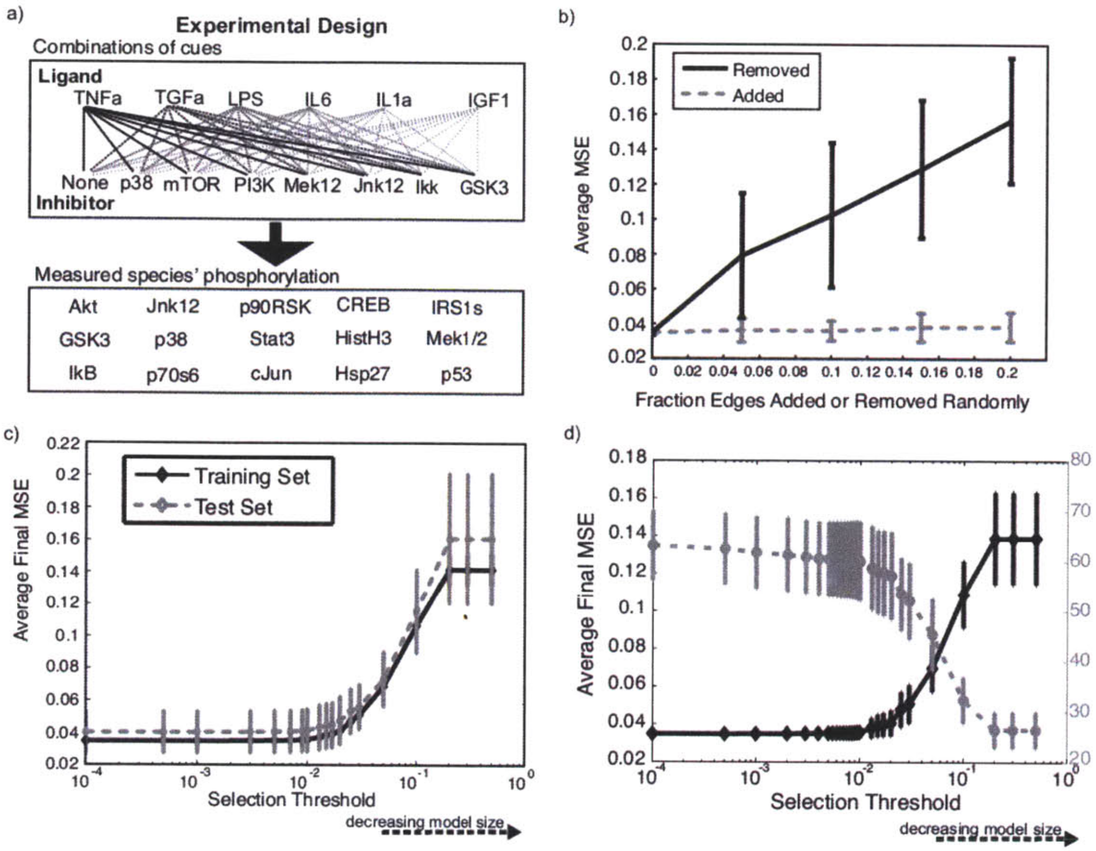


Table 2.1: Prior knowledge networks trained to HepG2 dataset. PKN0: Initial PKN shown to be insufficient for fitting HepG2 data. PKN1: Extended PKN used to compare two expansion limitations; PKN1^i was used for the majority of subsequent analysis. PKN2: PKNs used to determine mechanism of IL6-induced protein phosphorylation. PKN3: PKN further extended to model cytokine release.

Model ID	PKN0	PKN1^a	PKN1^i	A	B	C	D	PKN3
ERK to IRS1 [162]		X	X	X	X	X	X	X
TRAF6 to MEK [119]		X	X	X	X	X	X	X
Assay to PI3K				X	X	X	X	X
IL6R to PI3K [44]		X	X		X		X	
IL6R to Ras [44]						X	X	X
Protein Signals to Cytokine Release								X
Gates expanded into all possible 2-input AND gates (Step 2)	All	All	Only Inhib	Only Inhib	Only Inhib	Only Inhib	Only Inhib	Only Inhib

should first consider links from species that IL1α is already known to activate. In this case, TRAF6 is the most upstream species which experimental evidence suggests can activate MEK [119]. In the case of IRS1 signal activation, the specific phosphorylation site measured should be considered. Our data included measurements of phospho-S636/639, and S636 is a known phosphorylation site of ERK2 [162].

A novel finding from CellNOpt-cFL analysis of the HepG2 data was that IL6 treatment led to phosphorylation of several downstream proteins. Similarly to the links just considered, PKN0 included no paths between IL6 stimulation and these downstream proteins, resulting in an inability to fit this pattern of phosphorylation. Importantly, however, BL analysis would not have recognized this partial activation due to its inability to fit intermediate activation values (as illustrated in our earlier toy example). Because IL6 was observed to partially activate Akt in the data and known mechanisms exist for this activation [44], we added a prospective IL6R $\rightarrow$ PI3K link to the PKN, thus providing an extended PKN (PKN1) that we use below for subsequent CellNOpt-cFL analysis.

2.3.8 CellNOpt-cFL Training of PKN1

PKN1 was expanded to include all possible two-input AND gates (PKN1^a) for a total of 170 discrete parameters corresponding to 105 logic gates. The resultant network was trained to the HepG2 data. Reduction of the PKN1^a-derived models indicated that almost all AND gates could be removed or replaced by single-input gates. Since the AND gates appeared to add unnecessary complexity to the cFL models, we also expanded PKN1 to only include AND gates if an input node was inhibitory (PKN1^i; Table 2.1), resulting in only 60 discrete parameters corresponding to 56 logic gates. We then compared the PKN1^a- and PKN1^i-derived cFL models.

The comparison of these two PKN-derived model families revealed a clear tradeoff between model fit and complexity. The more complex PKN1^a-derived models were able to fit the data slightly better than the PKN1^i-derived models (average unprocessed model MSE of 0.032 ± 0.002 compared to 0.035 ± 0.002, $p < 0.001$). However, the more complex PKN1^a-derived models contained many more parameters than the PKN1^i-derived models both before and after optimization (170 compared to 60 discrete parameters before optimization and an average of 72.8 ± 4.9 compared to 66.6 ± 3.9 continuous parameters after optimization ($p < 0.001$); Supplementary Figure B-4). The simpler PKN1^i-derived models used fewer initial and final parameters to arrive at a fit to the data only 9% worse than PKN1^a-derived models. Since the 9% deviation is in the range of error in the normalized data (error estimated to be 10% by comparing similar stimulation conditions), we focused subsequent analysis on the simpler PKN1^i-derived models. For completeness, we include the results of PKN1^a-derived models as supplemental information (Supplementary Figure B-5).

2.3.9 Statistical significance of cFL models trained to PKN1^i

To determine the statistical significance of our results, we compared the family of 243 unprocessed models with unprocessed models obtained from either training PKN1^i to randomized data or training a randomized PKN1^i to the data (Supplementary Table B.1). Data was randomized by pairwise exchange of all data values while network topologies were randomized either by generation of an entirely random topology or by random pairwise exchange of gate inputs, gate outputs, or nodes' inputs [124]. When compared to the results of all types of randomization, models trained to the real data and PKN1 were highly significant ($p < 0.001$, Supplementary Table B.1), indicating that the family of trained cFL models fit the data better than expected by random chance.

To probe the dependence of the CellNOpt-cFL training process on the quality of the PKN used, we randomly added links to or removed links from the PKN and trained the resultant PKN to the data. As expected, the models derived from PKNs with links randomly removed had a poorer fit to data than those derived from the complete PKN1^i (Figure 2-4b, solid line). Conversely, when links were randomly added to the PKN, cFL-CellNOpt effectively removed the links (Supplementary Figure B-6), resulting in models with similar goodness of fit as models derived from PKN1^i (Figure 2-4b, dashed line). We thus conclude that an incomplete PKN degrades the ability of CellNOpt-cFL to fit the data whereas models derived from a PKN with extraneous links retain the ability to fit the data.

As an initial investigation of model predictive capacity to check for over-fitting, we performed a ten-fold cross-validation by randomly dividing the HepG2 data into ten subsets and, for each subset, reserving one as a test set while training with the remaining nine data subsets. The similar fits of the training and test data provided evidence that the family of models obtained from this procedure were predictive, and the difference in test and training MSEs did not depend on selection threshold, a measure of model size, suggesting that the models were not over-fit (Figure 2-4c).

Analysis of this cross-validation result combined with a plot of average filtered model size and fit (MSE) as a function of selection threshold (Figure 2-4d) suggested that a selection threshold in the range $1 \times 10^{-3} - 1 \times 10^{-2}$ would result in a family of models that contain slightly fewer number of parameters than lower thresholds (Figure 2-4d, dashed line) while retaining the ability to fit the data well (Figure 2-4d, solid line). We used a threshold of 5.0×10^{-3} for the remainder of our analysis unless otherwise noted.

Finally, we obtain a family of 243 filtered models for further analysis (Figure 2-5). By taking note of which cFL gates are removed during the CellNOpt-cFL training and reduction processes, one can generate hypotheses regarding these gates. Table 2.2 summarizes a set of biological hypotheses readily suggested by our cFL model topologies.

2.3.10 Validated Biological Hypothesis 1: Crosstalk from TGFα to the JNK pathway

Analysis of error between the family of cFL models and experimental data (Supplementary Figure B-7) highlighted consistent error in TGFα-induced partial activation of c-Jun. Both PKN0 and PKN1 allowed for TGFα-induced activation of c-Jun by

Figure 2-5: Results of training PKN1^i to HepG2 dataset. Topologies of the family of filtered cFL models trained to the HepG2 dataset. Unprocessed cFL models can be found in Supplementary Figure B-6 and fit of the filtered models to the data in Supplementary Figure B-7 . Nodes represent proteins that were either ligand stimulations (green), inhibited (orange), measured by a phospho-specific bead-based antibody assay (blue), or could not be removed without introducing potential logical inconsistency (white). The grey/black intensity scale of the gates corresponds to the proportion of individual models within the family that include that gate. Thus, links colored black were present in all models whereas links colored grey were present in a fraction of the models. Where visually feasible, cFL gates are labeled with a numerical value that corresponds to a quantitative sensitivity of the input-output relationship. Sensitivity is calculated as described in the Materials and Methods 2.5. The larger this value, the lower the level of the input nodes' activity required for generating significant output node activity (i.e. a gate with a high sensitivity indicates that the output node is sensitive to a low value of its input node). The uncertainties in these values arise from the various best-fit EC_{50} for each family member in the refinement step. The graph of the cFL models was generated by a CellNOpt routine using the graphviz visualization engine (www.graphviz.org) followed by manual annotation in Adobe Illustrator.

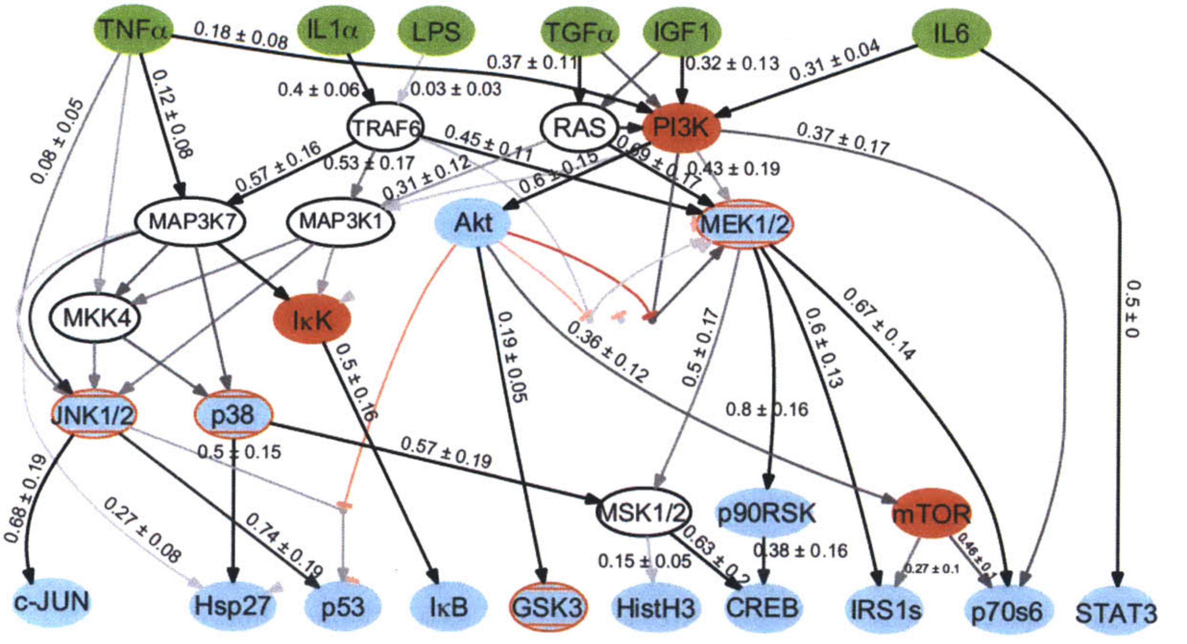


Table 2.2: Biological hypotheses about signaling network operation. Hypotheses suggested by gates removed during CellNOpt-cFL analysis

Hypothesis	**Evidence in cFL Models**	**Evidence in data**
Akt → Iκk crosstalk is inconsistent with the data.	Akt → Iκk gate is not present in unprocessed models (Supplementary Figure B-6)	Phosphorylation of Akt and Iκb are not positively correlated (correlation coefficient of -0.24).
Crosstalk from the growth and survival pathways (MEK/ERK and PI3K/Akt) to the inflammatory pathways (Nfκb, JNK, and p38) is not necessary to fit the data well.	Akt → Iκk gate is not present in unprocessed models and frequencies of other relevant crosstalk gates (Ras → MAP3K1 and PI3K → MAP3K1) are low in unprocessed models and decrease in filtered models.	
Crosstalk from the MEK/ERK pathway is not necessary to describe Hsp27 phosphorylation.	MEK → Hsp27 gate is not present in unprocessed models.	Phosphorylation of MEK and Hsp27 is not strongly correlated (correlation coefficient of 0.43) but phosphorylation of JNK and Hsp27 is strongly correlated (correlation coefficient of 0.91)
HistH3 data is not well described by PKN1.	Frequency of MSK1/2 → HistH3 gate is low in unprocessed models and decreases in filtered models and models do not fit HistH3 data well (Supplementary Figure B-7)	Phosphorylation of HistH3 and neither MEK nor p38 are strongly correlated (correlation coefficients of 0.55 and 0.47, respectively)
LPS does not activate the measured signaling nodes.	Frequency of LPS → TRAF6 gate is low in unprocessed models and decreases in filtered models.	The only protein that is consistently phosphorylated under LPS stimulation is Akt

the JNK pathway via crosstalk from Ras or PI3K to MAP3K1. In the BL methodology, this crosstalk was removed due to the inability to fit partial activation, and no BL model allowed for activation of c-Jun after TGFα stimulation. However, we found that a subset of cFL models accounted for this c-Jun partial activation by including crosstalk between Ras or PI3K and MAP3K1. These models also partially activated JNK after TGFα stimulation, a feature that was inconsistent with the training data (Supplementary Figure B-8). Thus, these models predict that JNK was actually phosphorylated under conditions of TGFα stimulation, but our measurements did not detect it.

To test this prediction directly, we undertook *de novo* measurement of JNK and c-Jun phosphorylation following stimulation with different doses of TGFα (Figure 2-6a). These new data show that JNK does indeed become phosphorylated upon stimulation of HepG2 cells with TGFα. Thus, the cFL models containing crosstalk from Ras or PI3K to MAP3K1 were the correct models. Combined with Table 2.2, this analysis highlighted the partial activation of the JNK pathway after TGFα stimulation as a singular instance of crosstalk from a pro-growth ligand to an inflammatory pathway. In support of the significance of our finding here, we note that TGFα-induced JNK activation has been shown to be important for hepatic regeneration [158] and stimulation of DNA synthesis [7] in primary rat hepatocytes.

2.3.11 Validated Biological Hypothesis 2: Mechanism of IL6-Induced Protein Phosphorylation

As previously mentioned, PKN0 was unable to fit IL6-induced protein phosphorylation (a feature of the data unappreciated by the BL methodology). Because Akt was observed to be partially phosphorylated under these conditions and we found literature evidence for a prospective IL6R → PI3K link, we added the link to PKN1. However, the media-only condition also induced partial phosphorylation of Akt. Discovery of the partial activation of Akt in the media-only control led us to consider that perhaps the IL6-induced phosphorylation of Akt was simply an assay artifact. Thus, we inserted an Assay → PI3K link into the PKN. This 'Assay' node represents cell stress arising from changing environmental conditions during the assay (media change, etc.); it is postulated to activate PI3K because only Akt is consistently active in the untreated control. Having accounted for the potential that IL6-induced partial phosphorylation of Akt was an artifact, we undertook a series of computational experiments to determine the mechanism of IL6-induced phosphorylation of downstream proteins.

Upon exposure to IL6, SHP2 has been reported to bind to gp130, a subunit of the IL6 receptor complex. SHP2 is then phosphorylated in a JAK1-dependent manner. This phosphorylation can lead to PI3K/Akt pathway activation through interactions with Gab-1 or IRS1 or Ras/MEK/ERK pathway activation through Grb2 or Gabl [44]. Thus, our computational experiments were designed to infer which pathway (PI3K/Akt or Ras/MEK/ERK) was mediating the IL6-induced protein phosphorylation. Four families of 150 filtered models were examined, all of which were obtained

Figure 2-6: Validation of cFL crosstalk predictions. (a) Analysis of systematic error as well as the topologies of the family of trained cFL models (Figure 2-5) indicated that c-Jun was partially activated after TGFα stimulation. Models with crosstalk from Ras or PI3K to Map3K1 predicted that JNK was partially activated under these experimental conditions even though it was not partially activated in the dataset. We tested whether JNK was actually partially activated under these conditions by stimulating HepG2 cells with TGFα and measuring levels of phosphorylated JNK and c-Jun by a bead-based antibody assay after 30 minutes. Fold increase in measured phosphorylation over un-stimulated control for c-Jun (black) and JNK (red) is shown. Where available, biological replicates are indicated with filled circles. Solid lines indicate the averages of the replicates. This experiment indicates that JNK was partially phosphorylated under TGFα stimulation and the cFL models with crosstalk from Ras or PI3K to MAP3K1 were correct. (b) CFL analysis of the topologies and fit of the HepG2 training dataset to several PKNs suggested that IL6 activated downstream nodes through the Ras/MEK pathway (Table 3). To test this prediction, a validation dataset was examined [5]. This validation dataset showed that the activation of nodes other than STAT3 that responded robustly to IL6 stimulation was ablated by pretreatment with a small molecule MEK inhibitor but not other inhibitors, demonstrating that the Ras/Raf/MEK pathway mediates this crosstalk.

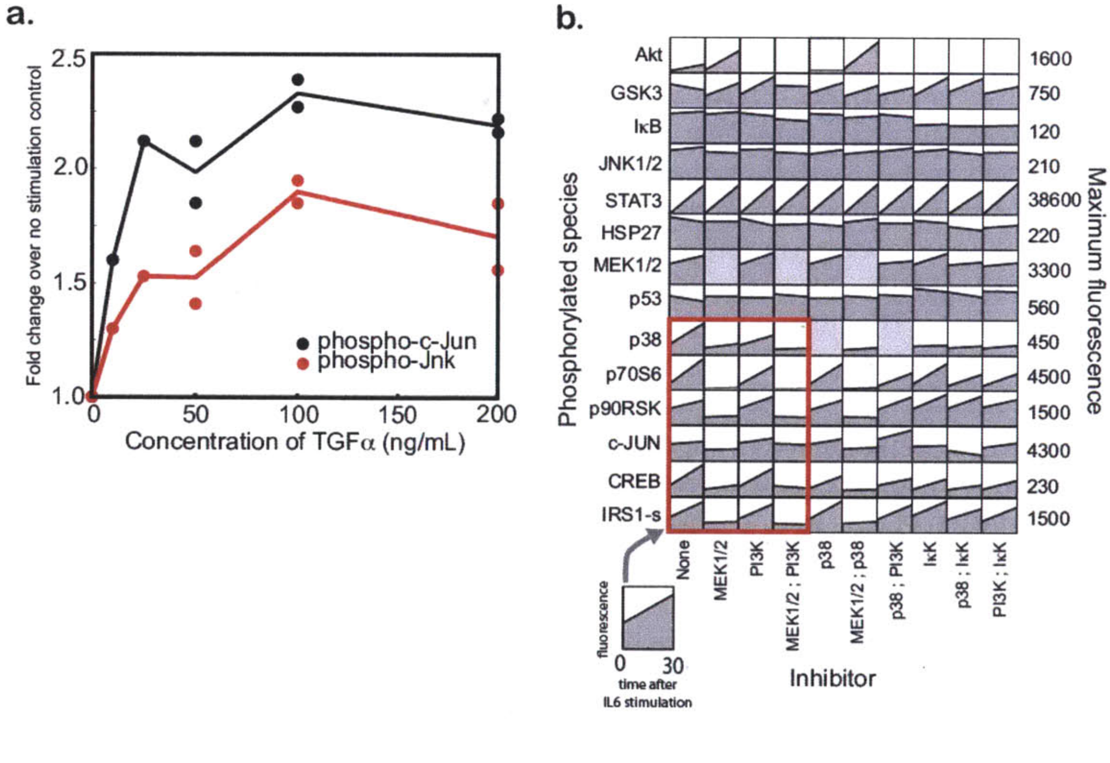


Table 2.3: Results of cFL training of various prior knowledge networks for the investigation of IL6 crosstalk

PKN	Assay to PI3K?	IL6R to PI3K?	IL6R to Ras?	MSE_{IL6}
PKN1^i	-	100%	-	0.040 ± 0.004
PKN2A	100%	-	-	0.052 ± 0.004
PKN2B	97%	56%	-	0.046 ± 0.008
PKN2C	99%	40%	95%	0.028 ± 0.004
PKN2D	99%	-	98%	0.028 ± 0.004

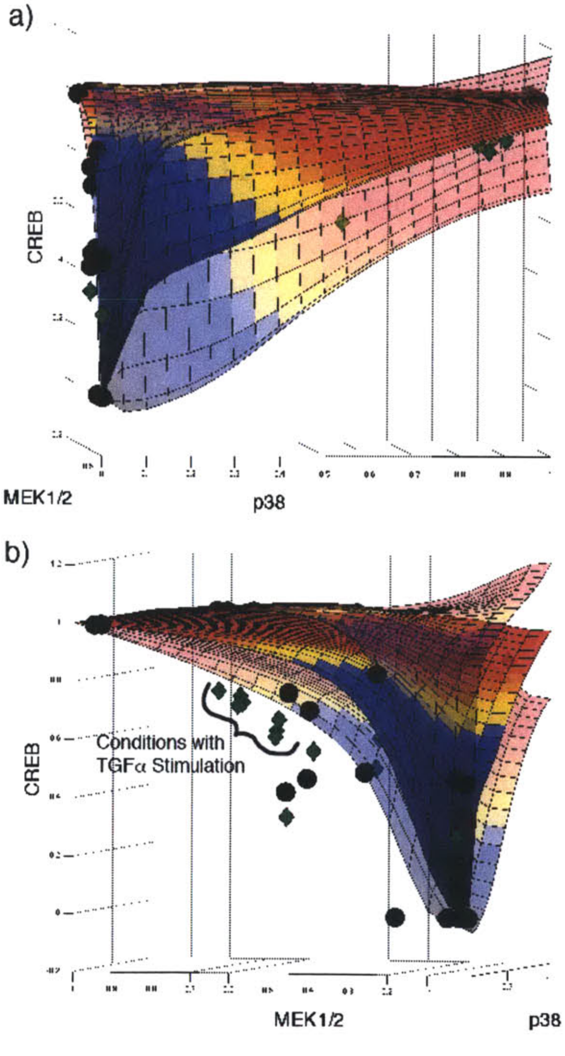
after training a new PKN to the normalized HepG2 dataset (Table 2.3 , PKN2A through PKN2D). The inability of PKN2A-derived cFL models with only the Assay → PI3K link to fit well the IL6-induced protein phosphorylation data suggested that some other link was necessary to fit this data. In our trained networks, the IL6R → PI3K link was present in only a fraction of the relevant trained models (PKN2B and PKN2C), but the IL6R → Ras link was present in more than 90% of relevant trained models (PKN2C and PKN2D). Additionally, models with IL6R → Ras links were better able to fit the IL6-induced protein phosphorylation. Consequently, our cFL results supported the hypothesis that IL6R activates downstream proteins through the Ras/Raf pathway. This hypothesis is supported by an independent dataset [124], where the IL6-induced protein phosphorylation response was more robust than in the training data (Supplementary Figures B-1 and B-9). Inhibition of MEK either alone or in combination with other inhibitors resulted in ablation of downstream protein activation whereas inhibition of PI3K did not (Figure 2-6b). Thus, we infer that IL6-induced protein phosphorylation was not an assay artifact and was instead mediated by the Ras/Raf pathway.

2.3.12 Predicting node-to-node transfer functions

CFL relates nodes in a network with transfer functions that describe quantitative input-output relationships between protein species represented as network nodes. To investigate the ability of the cFL models to predict these transfer functions, we simulated the PKN1^i-derived, filtered cFL models to determine the activation state of a specified node under many theoretical combinations of its input nodes. We then plotted the model predictions of quantitative input-output relationships. As one instance, Figure 2-7 shows the predicted average and standard deviation of the quantitative values of CREB phosphorylation as a function of the activation of its input nodes, p38 and MEK1/2. The resulting plots indicated that we were able to predict the activation response of CREB to the entire range of p38 and MEK1/2 although training set measurements were limited to a few values of these nodes (Figure 2-7, black circles).

We tested this prediction using a set of data with combinations of ligands and inhibitors not present in the training data ([124], Supplementary Figure B-9). Roughly

Figure 2-7: Transfer functions predicted by trained cFL models. The output value of the CREB node was predicted by computationally simulating each individual model in the family of cFL models with 441 combinations of p38 and MEK1/2. Three-dimensional plots were generated in MatLab showing the average prediction (opaque surface) as well as the average prediction plus or minus the standard deviation of the predicted value (semi-transparent surfaces). The training data (black circles) and validation data (green diamonds) are also plotted. The 3-D plots have been rotated to highlight the influence of either (a) p38 or (b) MEK1/2. The predicted transfer functions agree with the validation data reasonably well except for the overestimation of CREB activation for conditions with TGFα stimulation as one of the ligands.



20% of the test conditions were also present in the training data set, allowing us to control for differences between both data sets. When we compared this dataset to the predicted transfer functions, we observed that most of the data fell within one standard deviation of the predicted value (Figure 2-7, green diamonds) with exception of overestimation under conditions of TGFα stimulation. This overestimation is expected, as a comparison of common conditions between the training and test dataset indicated that the normalized experimental values of CREB in the validation dataset were 38±4% lower than that in the training set.

This result demonstrates the ability of the trained cFL models to predict the quantitative relationship between nodes in the network. We also found that the family of cFL models was able to fit the phospho-protein signaling response in the validation dataset well, which we demonstrate as supplementary information (Supplementary Figure B-9).

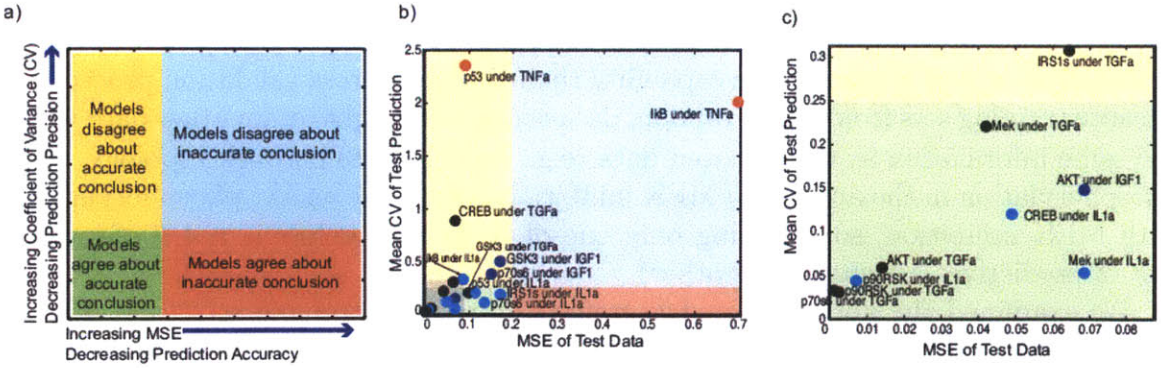
2.3.13 Predictive capability of a cFL model family

We performed a series of nineteen cross-validation experiments to further investigate the ability of our methodology to predict signaling response under conditions that were not represented in the training data. For each experiment, we used training data from which we had removed the phosphorylation data of a specific protein signal, s, under a single ligand stimulation condition and all inhibitor treatments. Nineteen signal/stimulation combinations were chosen to be test sets according to two criteria: (1) s is at least partially activated under the stimulation condition of interest and (2) s is at least partially activated under some other stimulation condition (Supplementary Table B.2). These criteria ensured that the remaining training data contained some information regarding the activation of s but it did not contain information regarding the activation of s under the stimulation condition of interest. This procedure is a more stringent test for predictive capability than a random cross-validation procedure because training sets from which random data is removed might retain other data with the same information as the removed data (e.g., based on the network topology, Akt phosphorylation in the absence of MEK inhibition is the same as Akt phosphorylation with MEK inhibition, so removing only one of these data points is not a stringent test of predictive capacity of the other).

We examined the ability of models trained on reduced training sets ($n > 45$ for each case) to predict phosphorylation of the test protein signals. Because we used each individual in the family of models to predict the test signal, we could determine if the models were constrained in their predictions by examining the coefficient of variance (CV; standard deviation divided by mean) of the prediction. If the CV was high, the models were not constrained to a specific prediction (i.e. the prediction was imprecise), and the average prediction should be discounted. Thus, for these cross-validation results, we compared the precision (CV) and accuracy (MSE) of the models' predictions, where precise and accurate predictions exhibited both a low CV and low MSE (Figure 2-8a).

We found that the families of models trained on these reduced training sets were able to precisely predict phosphorylation of the test protein signals in twelve of the

Figure 2-8: Accuracy vs. precision of cross-validation experiments. (a) Model predictions can be assessed based on both how well the family of models agree on a prediction (precision) as well as their accuracy. If a prediction is imprecise (i.e. the models do not agree), the models are not constrained to any single prediction. Thus, precision can be used to discredit predictions. Predictions can be both precise and accurate (green field), imprecise but accurate on average (yellow field), imprecise and inaccurate (blue field), or precise but inaccurate (orange field). Predictions that are precise and accurate (green field) are preferred. (b) The importance of considering the precision of a prediction amongst a family of models was demonstrated by a cross-validation study in which a signal under a single ligand stimulation condition in the presence or absence of any inhibitor was removed from the training data set. The mean coefficient of variance (CV) as a function of the error in the prediction (MSE) is plotted for all tests. One prediction was highly inaccurate. However, it was also imprecise (blue field), whereas no predictions were precise and inaccurate (orange field), demonstrating that taking the precision of a prediction into account can help to discredit inaccurate predictions. (c) The grey-boxed subset of (b) highlights the test sets that were precisely and accurately predicted by the family of cFL models.



nineteen cases (Figure 2-8b and c, green field). In six of the test sets, the models did not agree, although their average prediction was reasonably accurate (Figure 2-8b and c, yellow field). We observed no test sets for which the training sets agreed about an inaccurate prediction (Figure 2-8b, orange field). In one case (prediction of Iκb signaling under TNFα stimulation), the predicted phosphorylation state was highly inaccurate (MSE $>$ 0.20). However, this prediction was also very imprecise (CV $>$ 0.25), indicating that the average prediction was unreliable (Figure 2-8b, blue field). Thus, by taking the precision of the models' predictions into account, we were able to discredit an inaccurate prediction. This result underscores the importance of considering consensus among the family of models rather than examining the results of only one cFL model.

2.3.14 Using cFL models to relate phospho-protein signaling to cell phenotypic response

The ability to quantitatively model protein signal activation with cFL offers the prospect of predicting phenotypic response upon exposure to stimuli and inhibitors. To investigate the ability of cFL to model phenotypic data, we turned to data describing cytokine release three hours after stimulation under the same conditions as the phosphorylation data [5]. As a first approach, we linked the output of our family of cFL models to a partial least squares regression model [58] obtained by regressing normalized data of release of five cytokines (IL1B, IL4, G-CSF, IFNg, and SDF1a) to the normalized protein phosphorylation measurements (see Supplementary Text B.1).

The cFL models linked to a PLSR model were able to model phenotypic response with an accuracy of $R^2 = 0.79$, near that of the PLSR model ($R^2 = 0.81$; see Suppementary Figure B-10). However, we found that the correlation indicated by regression coefficients did not lead to easily interpretable insights about phenotype because proteins in the same pathway were also highly correlated with each other.

To obtain a more interpretable model, we utilized a second approach where we included nodes specifying cytokine release in the PKN and linked them to a few protein signaling nodes. These nodes were chosen based on principle component analysis: if protein signals in a pathway clustered together in principle component space, the signal most downstream in the pathway was linked to cytokine release. Based on this analysis, the following protein signaling nodes were linked to each cytokine release node: MEK1/2, CREB, GSK3, c-Jun, Hsp27, Iκb, and STAT3 (Table 2.1, PKN3). We then trained a family of cFL models to the normalized dataset comprised of cytokine release at three hours and protein signaling at thirty minutes.

The models resulting from cFL method were able to fit the cytokine release data reasonably well ($R^2 = 0.78$ for the average predicted by a subset of best-fitting models). Furthermore, the low frequency of several gates in the resultant family of cFL models (Supplementary Figure B-11 , Supplementary Table B.3) indicated that, although the promoters of several of the modeled cytokines contained binding sites of transcription factors are known to be modulated by the MEK1/2, GSK3, and CREB pathways (Supplementary Table B.4), activation of these nodes did not predict cytokine release. Thus, we altered our previous PKN by removing the links between these protein signaling and cytokine release nodes and trained it to the data. The resultant family of cFL models (Figure 2-9) indicated that STAT3 activation explained cytokine release after IL6 stimulation and other signals (Iκb, c-Jun, and Hsp27) explained cytokine release three hours after TNFα or IL1α stimulation.

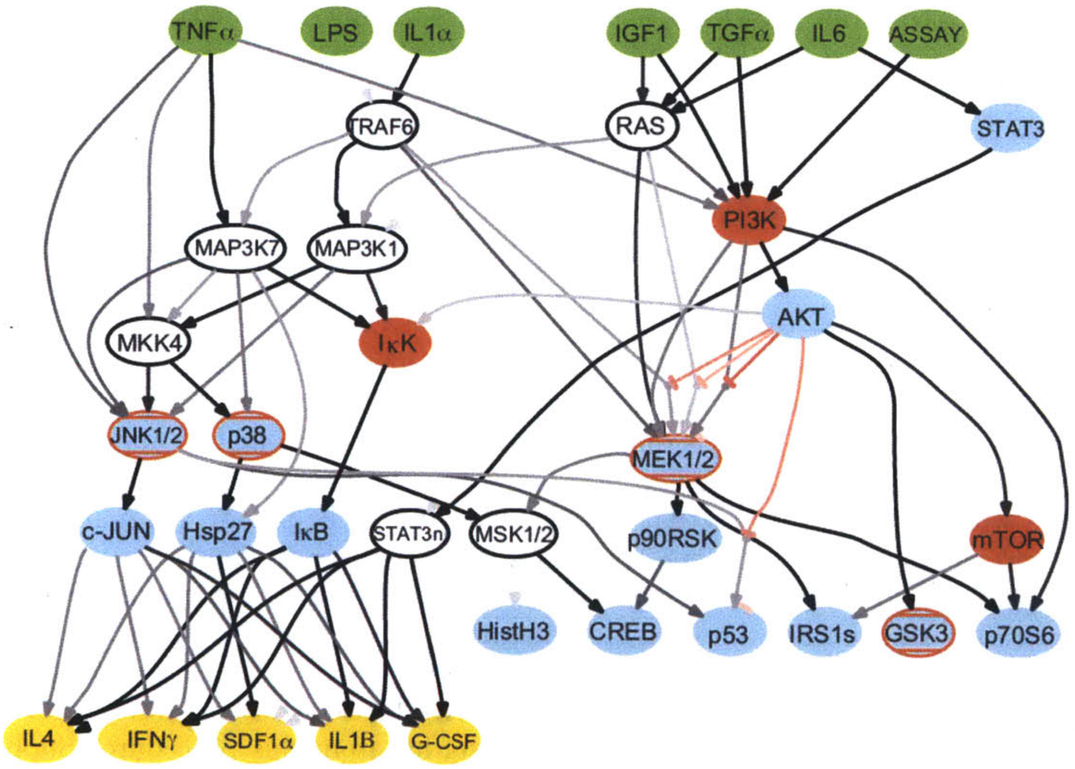
2.4 Discussion

In this chapter, we have described cFL for formal training of a prior knowledge network obtained from a protein interaction or signaling network map to experimental data and demonstrated that the ability of cFL to fit intermediate activities was crucial for understanding key features of a biological network. We validated two important biological insights concerning network operation in the HepG2 cells under inflammatory cytokine and growth factor treatment: (i) identification of c-Jun as a downstream locus of crosstalk between growth factor and inflammatory cytokine treatments and (ii) the Ras/Raf/MEK pathway as an avenue for activation of key downstream proteins following exposure of cells to IL6. Both of these insights were dependent on the ability of our cFL models to fit partial protein activation and were thus not appreciated by BL modeling.

We note that the ability of cFL to model intermediate activity data comes at the cost of increased model complexity. This complexity calls into question the identifiability of a cFL model (i.e. ability of the CellNOpt-cFL training process to train both parameters and topology given limited data). To address this concern, we considered families of models where each individual model predicted signaling states and the resulting predictions had an average and standard deviation. The standard deviation provided a metric for discrediting predictions for which the models were not constrained. With regard to topology, we considered how often a gate was present in the trained cFL models. This allowed us to determine hypothesized links (those present in the PKN) that were either inconsistent with the data (cFL gates removed from unprocessed models) or only marginally important for fitting the data (cFL gates removed from filtered models). Thus, the consideration of consensus and variation in an ensemble of models allowed us to account for the non-identifiability of any individual model.

We also illustrated the use of CellNOpt-cFL to (i) predict quantitative phenotypic response data with the same quality as a regression-based approach and (ii) increase the biological understanding of a phenotypic response by generating hypotheses regarding protein signaling pathways that lead to cytokine release. Transcriptional and/or non-transcriptional mechanisms could underlie the biological link between the signaling network activation and cytokine release profiles. We investigated predicted and known transcription factor binding sites in the promoters of relevant genes (Supplementary Table B.4), finding that several transcription factors hypothesized by CellNOpt-cFL to drive cytokine release (STAT3 and Nfκb) could, in concert with IRF1, potentially lead to the production and secretion of the observed cytokines.

Figure 2-9: Trained cFL models linking ligand cues, phospho-protein signals, and cytokine release phenotypic responses. A dataset describing release of five cytokines after three hours under conditions identical to those under which protein phosphorylation was measured after thirty minutes was combined with the phospho-protein dataset. PKN2D was further extended to include links from protein signals that occupied unique principle component space (Supplementary Text B.1) to nodes of cytokine release after three hours. Training this network to the data indicated that the growth and survival pathways were not needed to describe cytokine release. Thus, the PKN was revised to link only Stat3, Nfκb, c-Jun, and Hsp27 to the cytokine release nodes, and this PKN was trained to the experimental dataset of both cytokine release and protein phosphorylation. In contrast to the cFL models describing only signaling activation, we found that the family of 141 cFL models fit the cytokine response data with a wider distribution of MSE. The resultant sub-family of seven filtered cFL models that fit the data with a MSE less than the average plus one standard deviation of the family MSE is shown. Nodes represent proteins that were either ligand stimulations (green), inhibited (orange), phosphorylation states measured (blue), cytokine secretion measured (yellow) or could not be removed without introducing potential logical inconsistency (white). The grey/black intensity scale of the gates corresponds to the proportion of individual models within the family that include that gate. The graph of the cFL models was generated by a CellNOpt routine using the graphviz visualization engine (www.graphviz.org) followed by manual annotation in Adobe Illustrator.



Our subsequent test of this notion by qRT-PCR measurement, however, yielded a negative result; expression of the HepG2-secreted proteins were not significantly upregulated by IL6 stimulation (data not shown). Thus, it appears more likely that non-transcriptional mechanisms, such as exocytosis of secretory vesicles [122, 105] or proteolytic cleavage of pro-forms at the cell plasma membrane [9, 89], was responsible for the cytokine release observations.

We have shown that CellNOpt-cFL is useful for systematically and quantitatively comparing experimental datasets to a PKN that summarizes decades of dedicated biochemical studies. However, our aim in this work is not to argue for exclusive use of cFL modeling instead of BL or other modeling approaches, but rather to delineate key advantages of cFL modeling for addressing data with intermediate activity values. Training with CellNOpt-cFL is a more difficult optimization problem that is not efficiently solved for networks much larger than those in this work. The BL optimization problem scales as 2^w, where w is the number of gates in the processed PKN, whereas the CellNOpt-cFL optimization problem scales as $(1+a)^h$, where a is the number of transfer function in the set chosen by the genetic algorithm $((1+a) \geq 2; (1+a) = 8$ as formulated here) and h is the number of possible input-output transfer functions in the network ($h \geq w$). Additionally, as was the case with the reformulation of the BL optimization problem with Integer Linear Programming [98], we acknowledge that there may be more efficient, rigorous ways to solve the optimization problem presented by CellNOpt-cFL.

When training a prior knowledge network to data, we often encountered the need to add links to the prior knowledge network in order to fully describe the data. In this study, this was done manually simply by searching the literature. In the absence of such information, one should automate the process of testing many candidate links. A simple heuristic procedure such as the one we employed for the BL methodology based on mismatches between the best-fit models and data is one option [124]. Alternatively, more complex reverse engineering techniques could be used. The additional complexity of cFL modeling poses significant complications for the implementation of a simple heuristic or reverse engineering technique, but future efforts should investigate best practices for the automation of this process.

An additional prospective application of CellNOpt-cFL is to use a trained cFL model to inform the construction of a model with a different mathematical formalism. One intriguing possibility is that the CellNOpt-cFL methodology might be used to determine topologies to translate into a system of ordinary differential equations (ODEs) with methods such as that presented in [159]. The precise relationship between cFL and ODE parameters is unclear, but the ease of translating from one formalism to the other might be facilitated through the use of continuous AND and OR operators rather than the Min/Max operators utilized in this study. As a first step, we have retrained one of our main results (that presented in Figure 2-5) using the product of possible outputs to evaluate AND gates and the sum of possible outputs to evaluate OR gates. The models resulting from this procedure (Supplementary Figure B-13) were similar to those obtained previously (Figures 2-4c, 2-5), demonstrating the flexibility of this approach to accommodate different AND and OR operators as well as transfer function forms. Such flexibility should aid future attempts to translate CellNOpt-cFL results into other mathematical formalisms.

Finally, the dataset used here was gathered for training a BL model. This dataset was explicitly designed to maximally stimulate or inhibit pathways through the application of saturating doses of ligand and drugs. However, cells *in vivo* face a much more subtle and interesting situation in which ligands are present in combination, often at very different levels. Because cFL can model the graded activation of cell signaling pathways, we suspect that CellNOpt-cFL should prove particularly useful with signaling data collected under more physiological conditions. Our laboratories are currently pursuing experimental studies in this direction.

2.5 Materials and Methods

2.5.1 Optimization Procedure

Model compression and expansion was performed with CellNOpt as previously described [124]. The discrete genetic algorithm in the CellNOpt-BL variant was adapted so that discrete variables specified a transfer function rather than the gate type. Because our datasets (toy example and HepG2) only contained saturating concentrations of ligand stimulation, the normalized values of ligand model inputs were one or zero. In this instance, using normalized Hill functions to model interactions downstream of these zero or one inputs would result in all downstream nodes also reaching levels of zero or one (a Boolean simulation). To circumvent this issue, we represented interactions linking a ligand input to a downstream component with linear transfer functions with a y-intercept of zero and possible values of slope of 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, and 0.8 as well as the absence of the interaction. All other interactions were modeled with the normalized Hill function described in Figure 2-1 where the following transfer functions were possible: gate not active, approximately linear transfer function ($n = 1.01$, $k = 68.5098$ chosen for computation efficiency and numerical stability), or sigmoidal transfer function ($n = 3$) with an EC_{50} of 0.2, 0.3, 0.4, 0.5, 0.6, or 0.7 (Supplementary Figure B-14). These transfer functions were chosen because the models resulting from the training represented many different topologies while still fitting the data well. We found that including a subset of three to five of the aforementioned transfer functions would have also succeeded in this case, but a family of models obtained when ten transfer functions were used contained some models that did not fit the data well. This necessitated the addition of a step to choose a subset of well-fitted models from the family of trained models; this subset of well-fitted models did not significantly differ from the family of models obtained with fewer possible transfer functions. Given that more transfer functions allow to more accurately represent parameter space, this implied that the genetic algorithm was converging to poorly-fit local minima because the search space was too large. We therefore concluded that usage of eight transfer functions (seven transfer functions and the possibility of no interaction) balanced coverage of search space and ability to identify well-fitting models.

2.5.2 Sensitivity of a cFL gate

Sensitivity is calculated as $(1 - EC_{50})$ for cFL gates modeled with normalized Hill functions and $(0.5 \times slope)$ for cFL gates modeled with linear transfer functions.

2.5.3 Calculation of MSE

Mean squared error was calculated with the following formula

$$MSE = \frac{1}{N} \sum_{i=1}^{N_{sig}} \sum_{j=1}^{N_{stim}} \sum_{k=1}^{N_{inhib}} (x_{i,j,k}^{pred} - x_{i,j,k}^{obs})^2 \tag{2.1}$$

where N is the total number of data points, $Nsig$ is the number of protein signals measured, $Nstim$ is the number of cytokine or growth factor stimulations, $Ninhib$ is the number of inhibition conditions used, and $x_{i,j,k}^{pred}$ and $x_{i,j,k}^{obs}$ are the predicted and observed protein level of the ith protein signal under the jth stimulation and kth inhibition condition, respectively. In some cases, only the MSE of a subset of the data points is calculated for more specific error analysis. In these instances, the previous formula holds, but signal and/or stimulation conditions are constant and indicated with subscripts (e.g. MSE_{IL6} is the MSE of all signal measurements under all inhibition conditions and IL6 stimulation).

2.5.4 Measurement of protein phosphorylation and cytokine release

Protein phosphorylation and cytokine release were measured as described in [5]. Briefly, cells were incubated with small molecule inhibitor before exposure to ligand. Luminex bead-based bioassays were used to determine protein phosphorylation in cell lysate collected immediately before and 30 minutes after ligand exposure. Three hours after ligand exposure, supernatant was collected and Luminex bead-based bioassay used to measure the amount of cytokine that had been secreted.

Chapter 3

Q2LM analysis of cFL models based solely on prior knowledge [104]

3.1 Summary

Using intuition alone to predict the response of a complex biological system to perturbation is difficult, even when individual processes comprising the system are well understood. Mathematical models have substantially improved our ability to predict these responses, but their use is typically limited by difficulty in specifying model topology and parameter values. Additionally, incorporating entities across different scales ranging from molecular to organismal in the same model is not trivial. Here, we present an open source MATLAB framework that we call 'querying quantitative logic models' (Q2LM) for building and asking questions of constrained fuzzy logic (cFL) models. CFL is a recently developed modeling formalism that uses logic gates to describe influences among entities, with transfer functions to describe quantitative dependencies. Q2LM does not rely on dedicated data to train the parameters of the transfer functions, and it permits straight-forward incorporation of entities at multiple biological scales. The Q2LM framework can be employed to ask application-oriented questions about the system, such as: Which potential therapeutic perturbations accomplish a designated goal, and under what environmental conditions will these perturbations be effective? We demonstrate the utility of this framework for generating testable hypotheses in two diverse examples: (a) a model for intracellular signaling network regulation of transcription factor activity; and (b) a model for physiological pharmacokinetics and pharmacodynamics of cell/cytokine interactions; in the latter, we validate hypotheses concerning molecular design of granulocyte colony stimulating factor with the goal of enhancing neutrophil production from hematopoietic stem cells.

3.2 Background

Based on current understanding of a biological system, bioengineers predict how the system will respond to designed perturbations. One important manifestation of this process is predicting whether exposing a patient to a drug with a pre-defined target will result in a favorable clinical outcome. This approach works well when few relevant components of the system are considered. However, it is more difficult to propagate possible effects through a complex system using intuition alone, which hinders the capability for reliable prediction.

To aid intuition, a broad spectrum of mathematical and computational models have been developed [57]. For example, 'theory-driven' differential equations (DEs) based on physico-chemical mechanisms have been used to model and make predictions in biological systems ranging from virus population dynamics in a host organism [114] to receptor trafficking through cellular compartments [49] to enzymatic phosphorylation cascades [45]; at the other end of the continuum, 'data-driven' algebraic and statistical algorithms have been used to understand the integrated influence of multiple signaling pathways on cell phenotypic outcomes [58, 160]. While these various approaches have proven useful in biological and pharmaceutical contexts, their ability to make reliable predictions depends heavily on a large amount of appropriate experimental data for determining relationships, topologies, and parameter values. This critical dependence creates a high barrier-to-entry for using mathematical models to guide scientific decisions on a day-to-day basis. Furthermore, using these methods to describe relationships between different biological scales, such as the exchange of a molecule from tissues to individual cells and subsequent molecular interactions within the cell, is a significant challenge and an active area of research [56, 154, 20].

Logic-based models are an attractive intermediate alternative on the continuum of mathematical/computational approaches because they are readily derivable from either theory-driven or data-driven foundation [103], and they have been successfully used to predict the response of a biological system to perturbation (e.g. [165, 124]). A deficiency of discrete (e.g. Boolean) logic models, in which all species are found categorically in one of a few levels of activity, is that they are often too simple to adequately describe biological systems. Furthermore, increasing the number of possible species levels beyond two or three generally causes the process of specifying the gates in a logic model to become unwieldy. Recently, some have proposed transforming discrete logic models into either ordinary or piecewise linear differential equations [39, 96, 159]. While some software tools for building and simulating models of these types exist (reviewed in [103]), changes to parameters of such models affect the differential equations governing each species, and it is not immediately evident how such changes affect the quantitative relationships among the species in the system. Moreover, use of these tools to determine the effect of perturbation to species or parameters requires familiarity with the particular software and is not straightforward.

To alleviate these difficulties, we present here a new analysis framework for asking questions of logic-based models, which we term 'querying quantitative logic models' (Q2LM). We use the constrained fuzzy logic (cFL) formalism recently developed for training a logic model to data [102], but here demonstrate the Q2LM approach on models based solely on prior knowledge of the biological system. This logic formalism allows species in a biological system to be modeled with a continuous range between zero and one using mathematical functions that directly relate input and output species (transfer functions). The transfer functions contain parameters with distinct interpretations, allowing for the direct exploration of the effect of these mathematical relationships on model predictions. Importantly, the Q2LM approach facilitates querying of these models for more efficiently making predictions about the behavior of biological systems in response to perturbation. Q2LM is freely available at http://sites.google. com/site/saezrodriguez/software.

Because we use a simple logic-based framework, Q2LM is flexible enough to concomitantly incorporate multiple scales of biology-from molecular species to whole organisms. We illustrate the use of Q2LM to build and query a logic model with a simple example intracellular signaling model. Subsequently, we investigate a logic model of multiscale pharmacokinetics and pharmacodynamics of granulocyte colony stimulating factor (GCSF) with the objective of predicting the molecular-level alterations to the system that would best stimulate maturation of precursor neutrophils.

3.3 Methods

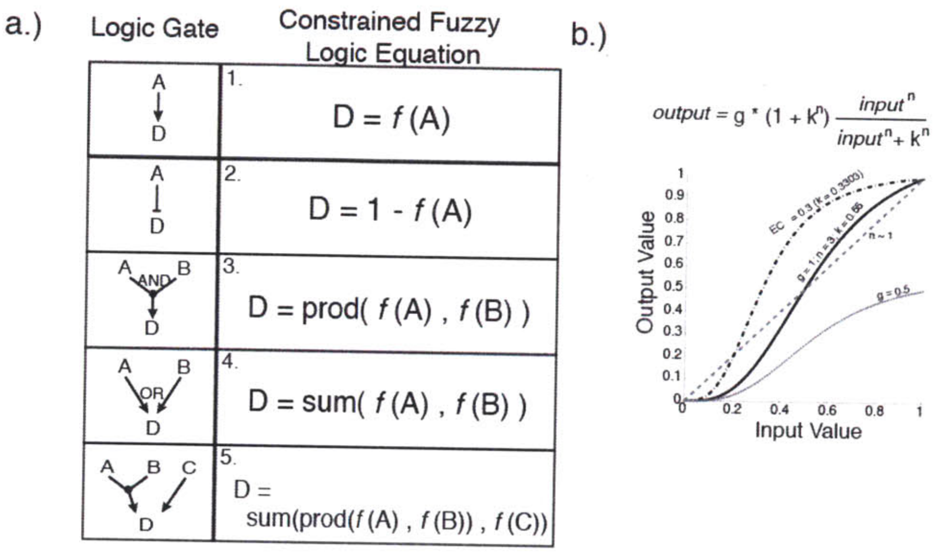
3.3.1 What is a constrained fuzzy logic model?

In a constrained fuzzy logic (cFL) model, the relationship between species is described by logic gates with transfer functions, from 'upstream' parent node(s) to 'downstream child node. In the simplest logic gate, one input parent species activates an output species, designated by an arrow between the two (Figure 3-1a). In cFL, this activating relationship is represented with a transfer function, which is simply a mathematical function used to evaluate the value of the output species given the value of the input species (Figure 3-1b).

In the current implementation, each transfer function is a normalized Hill function with a gain, where the gain, g, is a constant between zero and one, n is the Hill Coefficient, and k is the parameter that determines the EC_{50} of the function. If the input species inhibits the output species (a NOT gate in traditional logic modeling, Figure 3-1a), the transfer function is subtracted from one, effectively inverting it. We have found this transfer function form to be useful because it is simple yet flexible enough to accommodate a variety of biologically relevant functional relationships including linear, sigmoidal, and digital. Furthermore, each parameter of the transfer function determines a specific aspect of the function shape: g determines the maximum value that the output species reaches given maximal input species value; k determines the value of input species necessary for the output to reach activation at half of its maximum (EC_{50}), and n determines whether the shape is linear or sigmoidal. Thus, changing any of these parameters changes the transfer function shape in a predictable manner (Figure 3-1b).

Transfer functions are specified for every relationship between species and provide the basis for all quantitative relationships between species in a cFL model. If

Figure 3-1: Description of constrained fuzzy logic. (a) Constrained fuzzy logic describes interactions between biological species with logic gates. The logic gates are evaluated based on the output of the transfer function (f) that quantitatively relates the input and output species. In this example, AND gates are evaluated with the PROD operator and OR gates are evaluated with the SUM operator. Evaluation of the AND and OR gates with the MIN and MAX operators, respectively, is also possible, and Q2LM supports both types of operators. Note that the SUM operator is not identical to arithmetic sum, but rather, the logical sum of two possible values is equal to the first plus the second minus the product of the two (i.e. V1 + V2 - V1V2, where V1 is the value of one possible output and V2 is the value of the other). (b) The quantitative relationship between any two species is specified with a transfer function. In this work, we use a normalized hill function multiplied by a gain as the transfer function, although other functional forms can easily be imagined.



an output species has more than one input species, multiple transfer functions are evaluated for each input-output relationship, resulting in multiple possible values for the output species. The final value for the output species is then determined based on these possible values as well as the logic of the interactions. For example, if an output species has two inputs species, both could be necessary to affect the output species (an AND gate) or they could affect the output species independently of one another (an OR gate). If both AND and OR gates are used to relate inputs species to an output species, AND gates are evaluated before the OR gates (i.e. the sum-of-products formalism, Figure 3-1a).

3.3.2 Building a cFL model

To build a logic-based model, one must first identify the species in the biological system of interest to be included in the model. These species might be intra- or extra-cellular molecules, specific cell types, or the 'state' of a molecule or cell; thus, within the model a single entity can be represented by several species (e.g. ligand-bound and unbound cell receptors, differentiated or undifferentiated hematopoietic cells), where the name of the species is used to distinguish among various states of a single entity. Denoting various species names to identify any sort of entities enables a logic model to concomitantly incorporate processes at multiple biological scales.

The next step for building a logic model is to specify the interactions between species both in terms with what species interact as well as whether the interaction is activating or inhibitory. Knowledge of these interactions can come from a variety of sources. An expert may have accumulated enough knowledge to build such a model using intuition alone. Additionally, a wealth of databases exists that contain such interactions [8]. It is important to document sources used during the model building process so that, if discrepancies arise between the model simulations and what is known about the system, the knowledge basis of the model can easily be revisited.

The most challenging aspect of building a logic model is specifying AND or OR logic gates for species with more than one input parent species. While in previous work we used the CellNOpt software to train logic gates to dedicated experimental data [124, 102], in this work we rely on prior knowledge to determine the logic of the relationships. An AND gate should be used if the input species 'work together' to affect the output. Alternatively, one can identify an AND gate by asking 'Should the output be affected with only that input, or are other species necessary?' If other species are necessary, an AND gate should be used. For example, a molecular binding event is represented with an AND gate because both binding partners are necessary to form the bound species.

The final step is to write the model in a form readable by the software. For Q2LM, this involves making a spreadsheet that specifies the interactions and parameters of the transfer functions used to evaluate the effect (Figure 3-2c).

3.3.3 Simulating a cFL model

Q2LM simulates a cFL model with synchronous updating by calculating species values at each simulation step based on the values of their input species at the previous step. The simulation terminates when either the values of all species stabilize (the so-called 'logic steady-state') or once a pre-defined maximum number of steps has been reached. Species that have been designated as 'stimuli' are maintained at the stimulated value or, if its input species specify it to be a larger value during simulation, it is assigned the maximum of the stimulated and calculated values. The value of an inhibited species is multiplied by the percent inhibition at each simulation step. The initial values of all non-stimuli species is designated as Not-a-Number (NaN) and are ignored until their values have been specified by an upstream interacting species.

3.3.4 Querying a cFL model

We demonstrate Q2LM here by posing the following example questions: 1) What perturbations to species in the system result in a desired outcome? and 2) In what environmental conditions are these perturbations effective? To answer these questions, one must provide environmental conditions (the 'environment'), the perturbations ('experiments') and the desired outcome (the 'criteria'). Environmental conditions are considered invariant while experimental perturbations are varied and the effect of the experimental perturbation on each environmental condition determined. This effect is then compared to the criteria to reveal if the experimental perturbation 'met' the criteria. Strictly speaking, only an environment is required to simulate the model while experiments and criteria are used to answer a specific query. We will consider two vignettes motivated by previous studies by the Lauffenburger laboratory with various collaborators: one on the relationship of inflammatory cytokine and growth factor signaling to transcription factor regulation, and the other on systemic pharmacokinetic and pharmacodynamic behavior of neutrophilic cells to hematopoietic factors.

3.4 Results

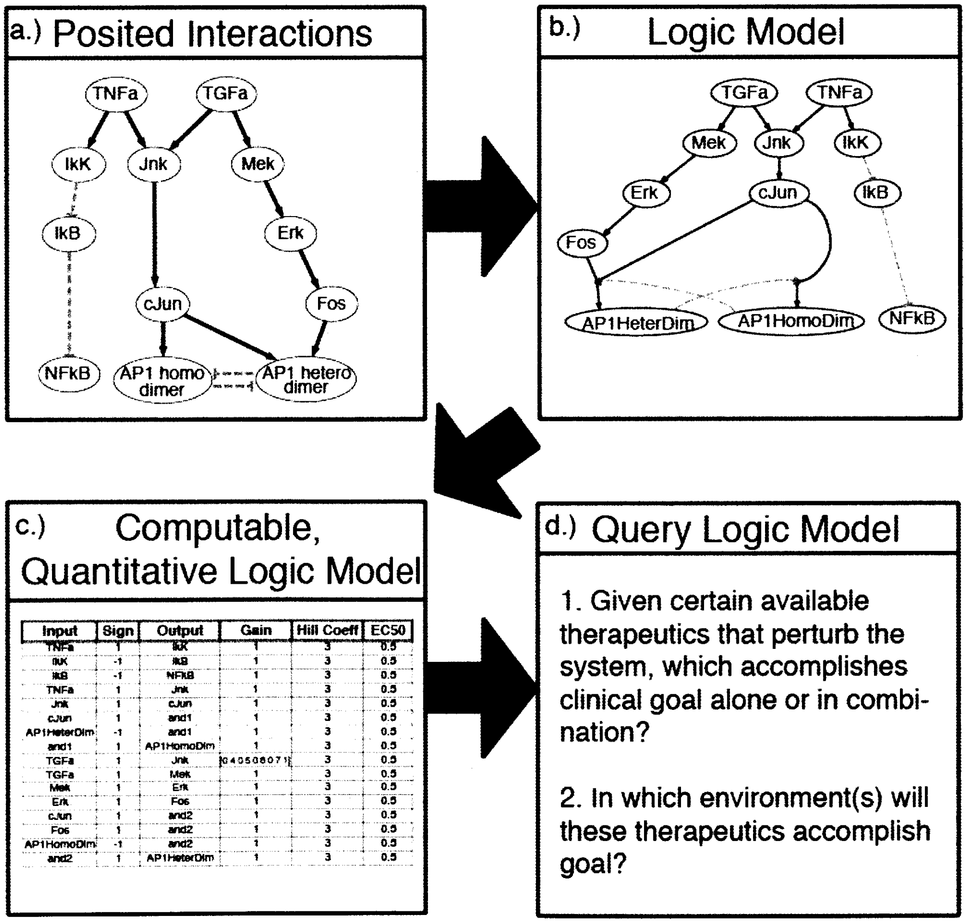
3.4.1 Logic-based model of intracellular signaling network

We first investigate potential crosstalk between TNFα- and TGFα-induced signaling pathways in activating downstream transcription factor activation with a highly simplified network. In a cFL model trained to intracellular signaling data from HepG2 cells, we previously noticed that both TNFα and TGFα stimulation of HepG2 cells activated the JNK/c-Jun pathway while only TGFα stimulation activated the MEK/ERK pathway [102]. These two pathways activate a variety of transcriptional programs; here, we focused on AP1 transcription factor activation, which involves the oligomerization of c-Jun and Fos, and IκK-mediated activation of the NFκB transcription factor. We postulated from literature evidence that ERK phosphorylates Fos, which facilitates its dimerization with c-Jun, thus forming AP1 heterodimers. Alternatively, c-Jun can be phosphorylated via the JNK pathway and dimerize to form AP1 homodimers [19, 152, 50]. For our Q2LM analysis, we focus on questioning whether inhibiting the activation of MEK, ERK, or JNK would increase the amount of AP1 homodimers.

From our understanding of this simple biological system, we specified the interactions between species in the network (Figure 3-2a). In most cases, increasing the value of the input species increased the value, or activity, of the output species. However, there were a few cases of inhibitory interactions: IκB sequesters and inhibits the activity of NFκB, and increased activity of IκK decreases the ability of IκB to sequester NFκB. For this example, we also assumed that there was limited c-Jun available in the system which resulted in stoichiometry-driven inhibitory relationships between AP1 hetero- and homo-dimers because the presence of one dimer form indicated that there was less c-Jun available to form the other.

To convert these interactions into a logic model (Figure 3-2b), we considered species with more than one parent inpu species as possible AND gates. JNK has more than one parent input (TGFα and TNFα), but TGFα and TNFα activate JNK independently of one another. Thus this gate is an OR gate and not an AND

Figure 3-2: Converting posited interactions of intracellular signaling into a logic model. (a) The relationship between species in an intracellular signaling network is depicted. These relationships are based on a model trained to biochemical data of HepG2 signaling protein activation after exposure to extracellular ligands [102] with additional links to AP1 homo- and hetero-dimerization based on [19, 152, 50]. (b) To convert the posited interactions in (a) into a logic model, we consider if the logic describing the relationship between input and output species should include an AND gate for species with more than one input, and find that two AND gates are necessary (see text for details). (c) The logic model is recorded as a spreadsheet to be loaded into the Querying Quantitative Logic Model (Q2LM) software. The first three columns specify which species interact as well as the logic of these relationships. An AND gate is specified by 'linking together' input and output species with a 'dummy' species indicated with 'and' followed by a number identifier. The last three columns specify the parameters of the transfer functions of the interaction indicated by that row. (d) The Q2LM software has been specifically designed to ask academically and industrially relevant questions.

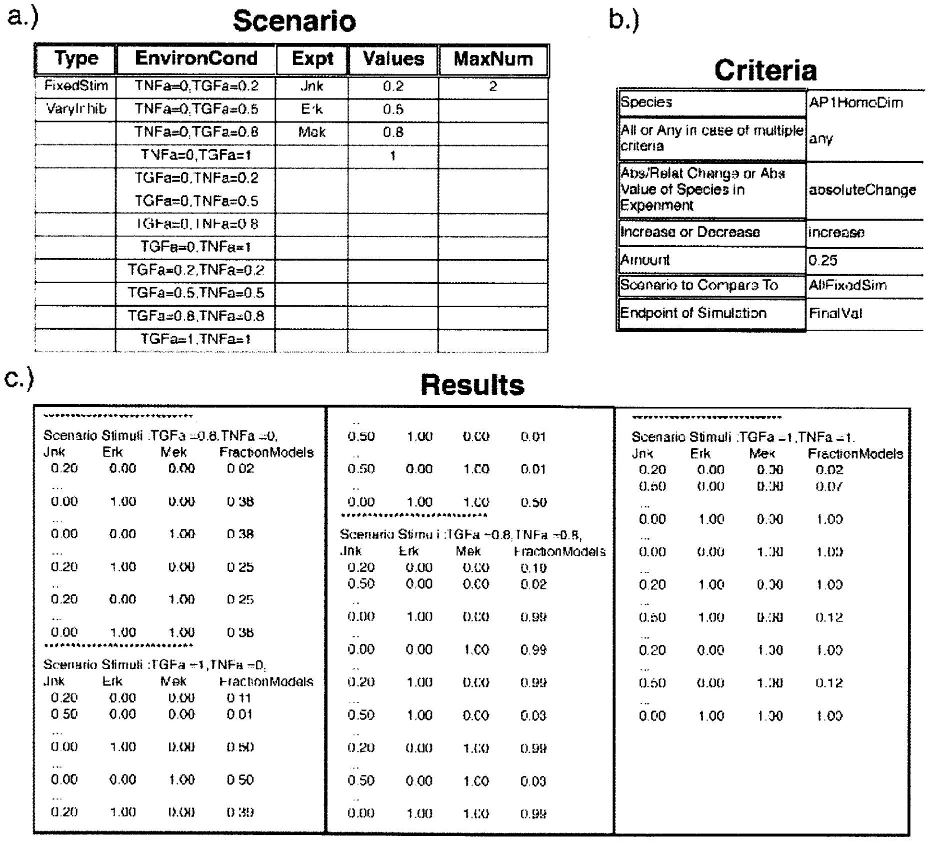


Input	Sign	Output	Gain	Hill Coeff	EC50
TNFa	1	IkK	1	3	0.5
IkK	-1	IkB	1	3	0.5
IkB	-1	NFkB	1	3	0.5
TNFa	1	Jnk	1	3	0.5
Jnk	1	cJun	1	3	0.5
cJun	1	and1	1	3	0.5
AP1HeterDim	-1	and1	1	3	0.5
and1	1	AP1HomoDim	1	3	0.5
TGFa	1	Jnk	[0 4 0 5 0 6 0 7 1]	3	0.5
TGFa	1	Mek	1	3	0.5
Mek	1	Erk	1	3	0.5
Erk	1	Fos	1	3	0.5
cJun	1	and2	1	3	0.5
Fos	1	and2	1	3	0.5
AP1HomoDim	-1	and2	1	3	0.5
and2	1	AP1HeterDim	1	3	0.5

1. Given certain available therapeutics that perturb the system, which accomplishes clinical goal alone or in combination?

2. In which environment(s) will these therapeutics accomplish goal?

Figure 3-3: Files to use Q2LM to examine intracellular signaling logic model. (a) Example of a scenario file that Q2LM imports to simulate experimental perturbations in a variety of environmental conditions. A detailed description of all file types is provided in the software's manual. In this case, environments with partial or full stimulation of TNFα and TGFα alone or in combination will be simulated with inhibition of the 'Expt' species JNK, ERK, and MEK at levels listed in the 'Values' column alone or in combination, where the maximum number of species to inhibit at any one time is listed in the 'MaxNum' column. (b) Example of a criteria file that Q2LM imports. Simulation results from environments with perturbation are compared to environments without perturbation and Q2LM calculates if the criteria have been met. In this case, the criteria is that the AP1homoDim species increase in value by at least 0.1 with perturbation compared to without. (c) Example of a Results file Q2LM outputs to indicate, for each environment, the values of perturbation that met the criteria in (b) and in what fraction of models they were effective.



a.)

Scenario

Type	EnvironCond	Expt	Values	MaxNum
FixedStim	TNFa=0,TGFa=0.2	Jnk	0.2	2
VaryInhib	TNFa=0,TGFa=0.5	Erk	0.5	
	TNFa=0,TGFa=0.8	Mek	0.8	
	TNFa=0,TGFa=1		1	

	TGFa=0,TNFa=0.2
	TGFa=0,TNFa=0.5			

	TGFa=0,TNFa=0.8			
	TGFa=0,TNFa=1			
	TGFa=0.2,TNFa=0.2			
	TGFa=0.5,TNFa=0.5			
	TGFa=0.8,TNFa=0.8			
	TGFa=1,TNFa=1			

b.)

Criteria

Species	AP1HomoDim
All or Any in case of multiple criteria	any
Abs/Relat Change or Abs Value of Species in Experiment	absoluteChange
Increase or Decrease	increase
Amount	0.25
Scenario to Compare To	AllFixedSim
Endpoint of Simulation	FinalVal

c.)

Results

```
******************************
Scenario Stimuli :TGFa =0.8,TNFa =0,
Jnk      Erk      Mek      FractionModels
 0.20     0.00     0.00     0.02
 ...
 0.00     1.00     0.00     0.38
 ...
 0.00     0.00     1.00     0.38
 ...
 0.20     1.00     0.00     0.25
 ...
 0.20     0.00     1.00     0.25
 ...
 0.00     1.00     1.00     0.38
******************************
Scenario Stimuli :TGFa =1,TNFa =0,
Jnk      Erk      Mek      FractionModels
 0.20     0.00     0.00     0.11
 0.50     0.00     0.00     0.01
 ...
 0.00     1.00     0.00     0.50
 ...
 0.00     0.00     1.00     0.50
 ...
 0.20     1.00     0.00     0.39
```

```
 ..
 0.50     1.00     0.00     0.01
 ..
 0.50     0.00     1.00     0.01
 ..
 0.00     1.00     1.00     0.50
******************************
Scenario Stimuli :TGFa =0.8,TNFa =0.8,
Jnk      Erk      Mek      FractionModels
 0.20     0.00     0.00     0.10
 0.50     0.00     0.00     0.02
 ..
 0.00     1.00     0.00     0.99
 ..
 0.00     0.00     1.00     0.99
 ..
 0.20     1.00     0.00     0.99
 ...
 0.50     1.00     0.00     0.03
 ..
 0.20     0.00     1.00     0.99
 ...
 0.50     0.00     1.00     0.03
 ..
 0.00     1.00     1.00     0.99
```

```
******************************
Scenario Stimuli :TGFa =1,TNFa =1,
Jnk      Erk      Mek      FractionModels
 0.20     0.00     0.00     0.02
 0.50     0.00     0.00     0.07
 ...
 0.00     1.00     0.00     1.00
 ...
 0.00     0.00     1.00     1.00
 ...
 0.20     1.00     0.00     1.00
 ...
 0.50     1.00     0.00     0.12
 ...
 0.20     0.00     1.00     1.00
 ...
 0.50     0.00     1.00     0.12
 ...
 0.00     1.00     1.00     1.00
```

gate. The AP1 heterdimers species also has more than one parent input species (c-Jun, Fos, and NOT AP1 homodimers). Because a heterodimer consists of both c-Jun and Fos, both are necessary to increase the amount of heterodimer, and an AND gate was used to model their logic. The presence of AP1 homodimer limits the amount of AP1 heterodimer, but only when c-Jun and Fos are present to make a heterodimer. Thus, it is also a parent input for the AND gate.

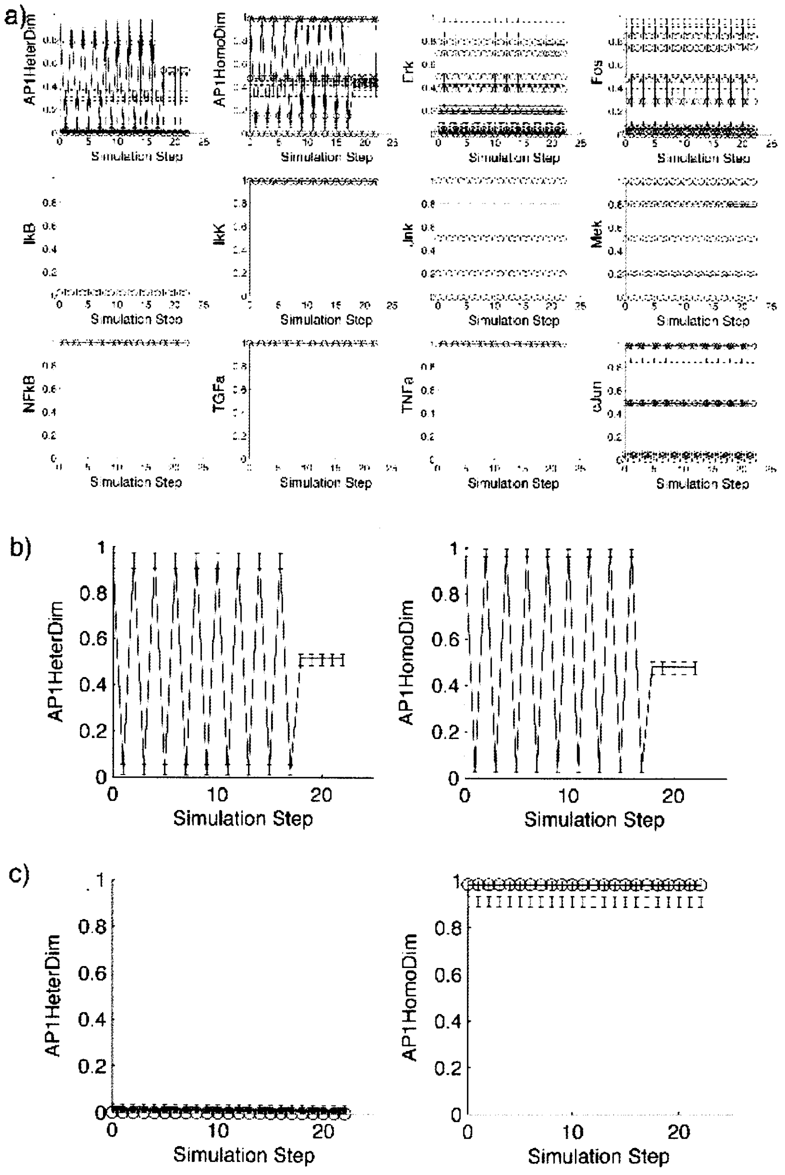
Finally, we wrote our logic model in a spreadsheet compatible with Q2LM (Figure 3-2c). TGFα did not activate the JNK pathway as strongly as TNFα in our initial dataset [102], but since we were not certain of the relative activating potentials we made several models, each with a different gain parameter for this interaction. This was indicated in the spreadsheet by including an array of gain parameters in the corresponding entry (Figure 3-2c). Additionally, when we loaded the model, we added normally distributed noise to each parameter to simulate biological noise.

We queried our intracellular signaling model to determine if inhibiting MEK, ERK, and JNK alone or in combination would increase AP1 homodimers in specific environments composed of varying levels of TNFα and TGFα alone or in combination. We simulated these environments with partial or complete inhibition of MEK, ERK, and JNK and then compared the resulting levels of AP1 homodimers with the levels that resulted without inhibition. This information was encoded in two input files: (1) the Scenario file included the environments and species to perturb with inhibition (Figure 3-3a) and (2) the Criteria file specified that the software should return experimental conditions that increase AP1 homodimers (Figure 3-3b).

Q2LM results revealed perturbations that increased the values of AP1 homodimers (Figure 3-3c), which corresponded to our criteria. These perturbations were stored in a separate file. We found that partially or completely inhibitting ERK and/or MEK increased AP1 homodimers in environments featuring high values of TGFα stimulation, but had minimal effect in those featuring low values of TGFα stimulation. Furthermore, this result implied that inhibiting JNK was not an effective strategy for increasing AP1 homodimer levels. However, inhibiting ERK and MEK either alone or in combination increased homodimers only if the network was fully stimulated by TGFα. Because this example served only to illustrate the use of Q2LM, a test of this hypothesis was out of the scope of this work. However, we note that because the software asked questions of the model in a manner analogous to experimental queries, experimental tests are easy to specify. For this example, a follow-up experiment to test this hypothesis would be to stimulate cells with low and high concentrations of TGFα in the presence or absence of ERK or MEK inhibition and to measure the resulting AP1 homodimer levels.

We next investigated how the system evolved during model simulation (Figure 3-4). It was apparent that the values of the AP1 homo- and hetero-dimers oscillated in several inhibition conditions. This is a common occurrence in models with feedback that have been simulated with discrete updating [39]. Q2LM offers two alternative treatments for environment/perturbation combinations that exhibit oscillations: 1) they can be ignored when delineating conditions that meet the designated criteria, or 2) the average value calculated over some pre-defined number of simulation steps can be used as the representative value for that species. In this case, we used the av-

Figure 3-4: Species values as a function of simulation step during simulation of intracellular signaling model. a) For each indicated species, the median value for all models at the final 19 simulation steps is shown (Q2LM does not save all simulation steps when memory is a limitation). Upper and lower error bars indicate the third and first quartile, respectively. Simulation conditions: TGFα = 1; TNFα = 1; Perturbation with different combinations of JNK, MEK, and ERK inhibition is indicated by different line color. Different line styles represent different models. b) Median value for AP1 homo- or hetero-dimers with no inhibitor perturbations. c) Median value for AP1 homo- or hetero-dimers for inhibitor combinations that met criteria of increasing the value of AP1HomDimer by at least 0.1 in at least 2% of the models.



erage to analyze oscillations. Because the interpretation of a condition that produces oscillations might differ from that of a condition that does not, it is important to plot the simulation evolution. In this case, the oscillations did not hinder our interpretation. In the absence of perturbation, AP1 homo- and hetero-dimers oscillated due to the negative feedback between them (Figure 3-4b). Thus, these two species were calculated to have values of 0.5 based on the average of their values over multiple stimulation steps. In inhibitor combinations that met the designated criteria, no oscillations were observed (Figure 3-4c). Instead, the values of the homo- and hetero-dimer species approached unity and zero, respectively. Thus, these conditions increased homodimers because they were no longer limited by negative feedback from heterodimers. By examining the system evolution, we confirmed that the conditions met our criteria by directly considering oscillations.

3.4.2 Logic-based modeling of pharmacokinetics of GCSF

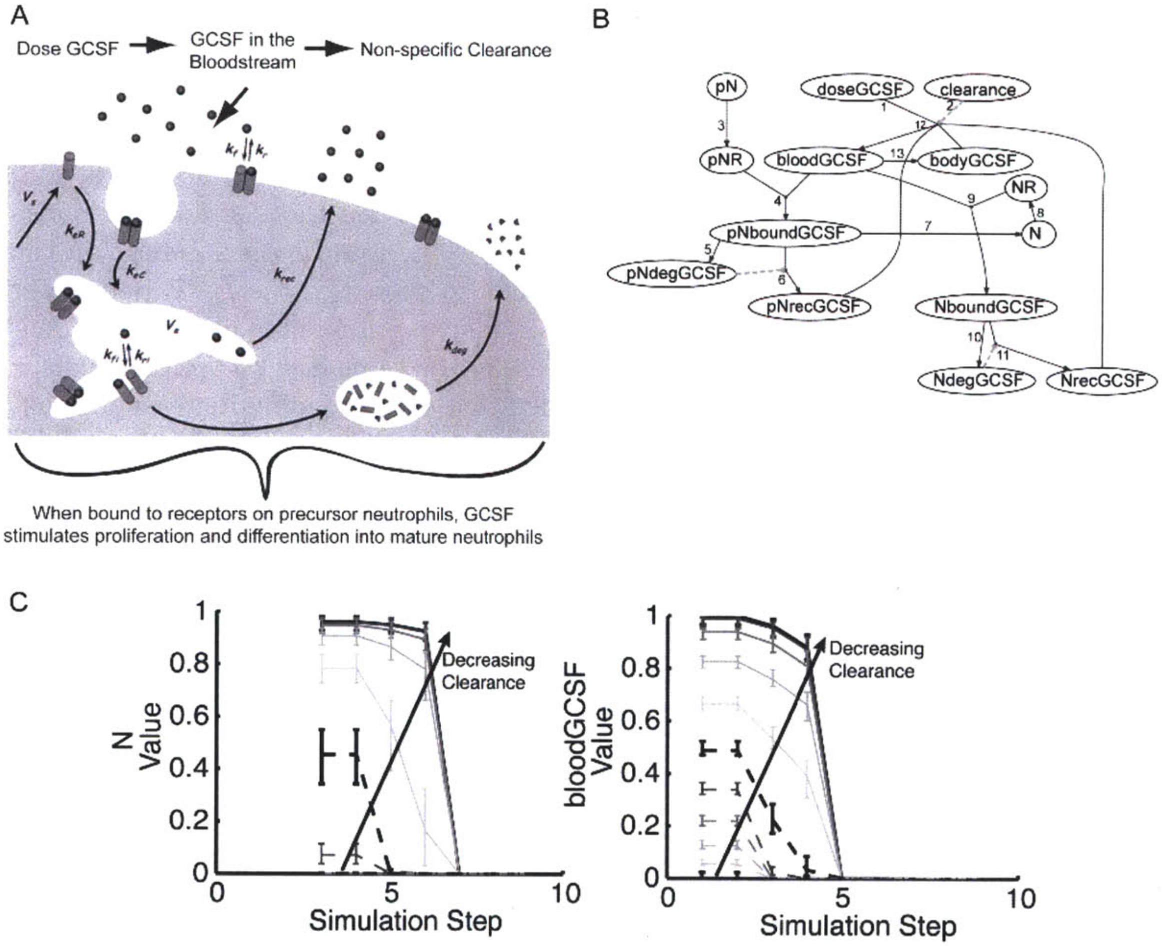
For our second example, we investigated whether Q2LM could be useful for multiscale models of physiological significance by using it to address the pharmacokinetics and pharmacodynamics of GCSF (Figure 3-5a). GCSF is used clinically to restore neutrophil levels to normal in situations generating neutropenia, such as cancer chemotherapy treatment. It is administered intravenously to stimulate the maturation of precursor neutrophils. After binding its receptor, it is internalized and either degraded in endosomes or recycled back into the bloodstream. Additionally, GCSF is cleared from the blood through non-specific clearance mechanisms, primarily renal clearance. Sarkar et al. used a DE model for GCSF PK/PD to ascertain a finding that when non-specific mechanisms are not the dominant mechanism of clearance, decreasing the rate of endosomal degradation of GCSF is more effective in stimulating neutrophil maturation than increasing the binding affinity of GCSF to its receptor [132]. This insight was consistent with the effects of engineerd GCSF variants *in vitro* [133] but had not been verified *in vivo*. Here, we examined whether a simpler cFL model would allow us to reach comparable conclusions without the requirement of estimating model parameter value for a complicated mechanistic DE model.

We built a logic model of the GCSF system by converting the linguistic description above (Figure 3-5a) into our cFL framework (Figure 3-5b). Although no dedicated experimental data were used to train this model in a traditional sense, it was derived from literature knowledge describing PK/PD of GCSF [115, 76]. Rather than using kinetic parameters to describe intracellular trafficking and non-specific clearance mechanisms, we use an AND gate to model these processes as limiting the amount of GCSF available in the bloodstream. The logic description therefore allowed us to easily relate tissue level phenomena to cellular- and molecular-level phenomena.

To explore system behavior predicted by the cFL model, we simulated model behavior under several conditions and plotted the species' values at each simulation step. We found that with decreasing clearance, the maximum value of both mature neutrophil (N) and GCSF in the blood (*bloodGCSF*) species values increased (Figure 3-5c). Although these species eventually reached a value of zero due to GCSF being degraded via receptor-mediated endocytic uptake, in some cases these decreases occurred at later simulation steps. This result agrees with how we understand the system to behave: a decrease in rate of clearance leads to an increase in total amount of GCSF that reaches precursor neutrophils due to increased half-life, but GCSF is nevertheless eventually cleared from the system. From this analysis, we identified two criteria to consider for assessing the impact of a perturbation on the *N* and *bloodGCSF* species: 1) maximum value attained; and 2) the number of simulation steps during which the nodes were at a value greater than zero.

Having established that the model was recapitulating known behavior, we used it to explore the effects of altering GCSF properties on physiological effectiveness, as measured by *N* and *bloodGCSF* levels. In particular, we calculated the above criteria under two conditions: 1) diminished degradation modeled by multiplying the *pNdegGCSF* and *NdegGCSF* species by a percent inhibition; or 2) enhanced binding modeled by increasing the minimal value of the *boundGCSF* species. We then compared the values of criteria under these conditions to those from simulations with no such perturbation (Figure 3-6a and b). Our results indicated that when the degradation nodes (*pNdegGCSF* and *NdegGCSF*) were inhibited by more than 50% at low values of clearance, there was a substantial increase in the number of simulation steps for the *bloodGCSF* species to reach zero. However, there was no effect on maximal value of *N* or *bloodGCSF* (Figure 3-6a). On the other hand,

Figure 3-5 *(facing page)*: GCSF administration as a logic model. (a) Depiction of GCSF pharmacokinetics at the tissue, cellular, and molecular level. Altered from [132]. (b) Logic model based on (a). All transfer functions have the parameters g = 1; n = 3; and EC_{50} = 0.5. Normally distributed noise with a standard deviation of five percent was added to each parameter 100 times to generate 100 models. Further analysis indicated adding noise with a standard deviation of up to 25 percent led to identical conclusions. Arrow labels indicate the following steps of the pharmacokinetics of the molecule: (1) When GCSF is administered intravenously (doseGCSF), it enters the bloodstream where it is subject to (2) nonspecific clearance (mainly renal clearance; clearance). (3) Precursor neutrophils (*pN*) possess receptors (*pNR*), which (4) bind GCSF in the blood (*pNboundGCSF*). (5) Bound GCSF can be degraded (*pNdegGCSF*), and (6) what is not degraded is recycled back into the bloodstream (pNrecGCSF). (7) Bound GCSF also stimulates proliferation and differentiation into mature neutrophils (*N*). (8) Mature neutrophils possess receptors (*NR*) that can (9) bind GCSF (*NboundGCSF*). Bound GCSF is then (10) degraded (*NdegGCSF*) or (11) recycled (*NrecGCSF*). (12) Value of GCSF in the blood (*bloodGCSF*) is limited by the dose, clearance, and amount recycled. (13) An additional species bodyGCSF represents the exchange of GCSF from the blood to the body cavity and is necessary in the logic model to ensure that the *bloodGCSF* node is also limited by its own value. (c) The GCSF logic model was simulated under non-limiting precursor neutrophils and dose conditions (*pN* = 1 and *doseGCSF* = 1) with multiple levels of clearance (0, 0.1, 0.2, etc.). Median value of the neutrophil and GCSF levels in the blood nodes (*N* and *bloodGCSF*) were plotted as a function of simulation step, with error bars indicating the first and third quartile of predictions of 100 models with noise added to the parameters. As levels of clearance decreased, maximal values of *N* and *bloodGCSF* increased as well as the number of simulation steps until the species values decreased to zero.


A
Dose GCSF
GCSF in the Bloodstream
Non-specific Clearance
k_f
k_r
V_s
k_{eR}
k_{eC}
k_{rec}
V_e
k_{fi}
k_{ri}
k_{deg}
When bound to receptors on precursor neutrophils, GCSF stimulates proliferation and differentiation into mature neutrophils
B
pN
doseGCSF
clearance
1
2
3
12
pNR
bloodGCSF
13
bodyGCSF
4
9
NR
8
pNboundGCSF
7
N
5
6
pNdegGCSF
pNrecGCSF
NboundGCSF
10
11
NdegGCSF
NrecGCSF
C
N Value
0
0.2
0.4
0.6
0.8
1
0
5
10
Simulation Step
Decreasing Clearance
bloodGCSF Value
0
0.2
0.4
0.6
0.8
1
0
5
10
Simulation Step
Decreasing Clearance

increasing binding by setting the minimum of the *pNboundGCSF* species to a value greater than zero resulted in no decay of the *N* node (i.e., a logic steady state value greater than zero, Figure 3-6b). This result was expected because the *pNboundGCSF* species directly activated the *N* species, so fixing the minimum value of one should directly affect the value of the other. This effect was also reflected in an increase in the maximum value that the *N* species attained. However, the maximal value of the *bloodGCSF* species did not increase, and in fact the number of simulation steps for the *bloodGCSF* species to reach zero decreased in many conditions (Figure 3-6b). These results provide a first indication that inhibiting degradation is the better strategy for increasing numbers of mature neutrophils.

As a complementary approach for exploring whether increasing binding affinity or decreasing endosomal degradation would be more effective, we examined the effect of varying the parameters controlling the processes of binding and degradation (Figure 3-6c and d). We varied the gain parameter of the *boundGCSF*-to-*degGCSF* transfer function to represent varying the fraction of *boundGCSF* that was degraded, and found that these results recapitulated those obtained when the degradation nodes were inhibited: steps to decay of *bloodGCSF* and *N* increased with no effect on the maximal level of these species (Figure 3-6c). We also decreased the EC_{50} parameter of *bloodGCSF* to *pNboundGCSF* to represent an increase in binding affinity. By definition, decreasing the EC_{50} results in an increase in the value of *pNboundGCSF* for a given value of *bloodGCSF*. This perturbation led to a corresponding increase in maximum value of *N* while the value of *bloodGCSF* remained constant for intermediate values of clearance (Figure 3-6d). At high or low values of clearance, this effect was not observed, pointing to another interesting aspect of our system: at high values of clearance, *bloodGCSF* never reached a value large enough to activate the *pNboundGCSF* and *N* nodes while at low values of clearance, the *N* species reached a large value at the default EC_{50} (Figure 3-5c), so only minimal effects were observed when affinity was further increased. Changing these parameter values had no substantial

Figure 3-6 *(facing page)*: In all parts, perturbations to species (a,b) or model parameters (c,d) were made when the GCSF logic model was simulated under non-limiting precursor neutrophils and dose conditions (i.e. pN = 1 and doseGCSF = 1) with multiple levels of clearance (0, 0.1, 0.2, etc.), with each color and line style corresponding to a different fixed value of the clearance species as shown in the legend in the rightmost panel for each part. a) The median effect of increasing inhibition of the *pNdegGCSF* and *NdegGCSF* nodes on each criteria is plotted, with error bars indicating the first and third quartile of predictions of 100 models. b) The median effect of varying the minimal possible value of the *pNboundGCSF* node, with error bars indicating the first and third quartile of predictions of 100 models. Because the *N* species was not observed to decay in these simulation, the first panel is the increase in logic steady state value of N, not steps until decay. c) The median effect of changing the gain of the transfer function relating *pNboundGCSF* to *pNdegGCSF* and *NboundGCSF* to *NdegGCSF* on each criteria is plotted, with error bars indicating the first and third quartile of predictions of 100 models. d) The effect of changing the EC_{50} of the *bloodGCSF* to *pNboundGCSF* interaction, with error bars indicating the first and third quartile of predictions of 100 models.

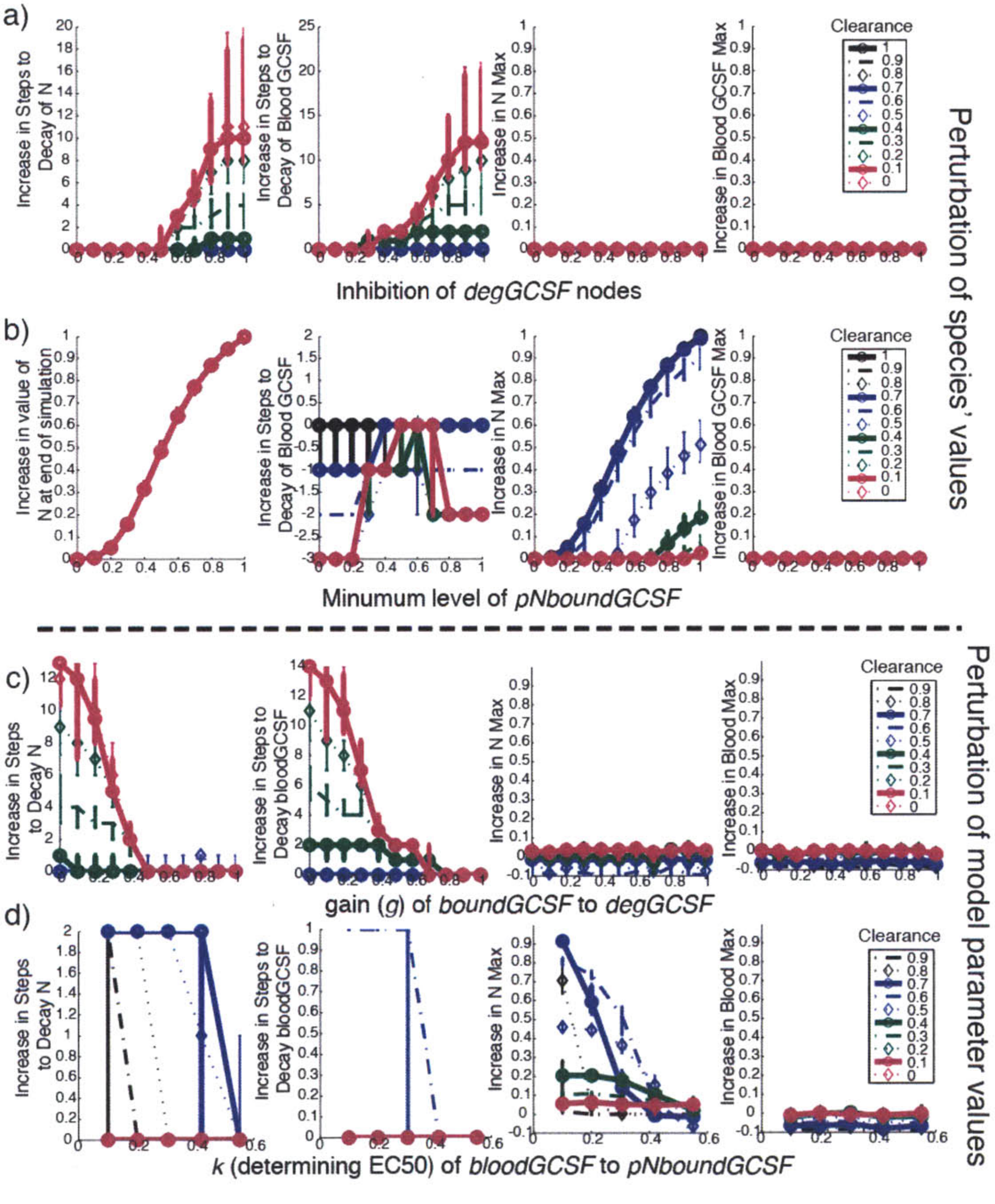
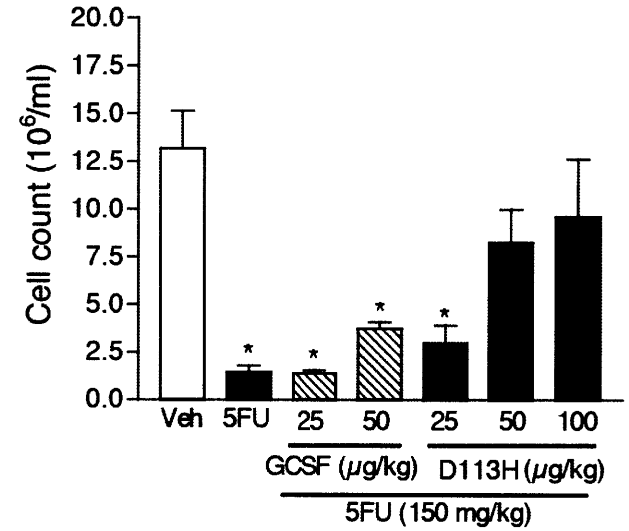

a)
Increase in Steps to Decay of N
Increase in Steps to Decay of Blood GCSF
Increase in N Max
Increase in Blood GCSF Max
Clearance
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Inhibition of degGCSF nodes
b)
Increase in value of N at end of simulation
Increase in Steps to Decay of Blood GCSF
Increase in N Max
Increase in Blood GCSF Max
Minumum level of pNboundGCSF
Perturbation of species' values
c)
Increase in Steps to Decay N
Increase in Steps to Decay bloodGCSF
Increase in N Max
Increase in Blood Max
gain (g) of boundGCSF to degGCSF
d)
Increase in Steps to Decay N
Increase in Steps to Decay bloodGCSF
Increase in N Max
Increase in Blood Max
k (determining EC50) of bloodGCSF to pNboundGCSF
Perturbation of model parameter values

Figure 3-7: *in vivo* increase in WBC associated with decreased GCSF degradation. Veh denotes the WBC count in animals sham treated with PBS. Animals (n = 5) were treated with 5FU (a drug that acts on the bone marrow and inhibits haematopoiesis, administered at 150 mg/kg). Treatment with the colony stimulating factor was started 24 h after the administration of 5FU, and continued for 9 days. Animals were sacrificed and blood collected by cardiac puncture. WBCs were concentrated after hemolysing the RBCs using a RBC lysis solution. Cell count was performed using a Coulter counter. *$p < 0.001$ versus vehicle-treated controls.



effects on the number of simulation steps until decay.

In summary, these results indicated that while increasing the binding affinity of GCSF to its receptor might result in an increase in N for a given level of *bloodGCSF* (Figure 3-7b), this effect occured only in a limited range of clearance values, and an increase in bound receptor also had the deleterious effect of decreasing the number of simulation steps required for decay of *bloodGCSF* (Figure 3-6b). In contrast, decreasing the amount of degradation consistently increased the number of simulation steps required for decay of *bloodGCSF* (Figure 3-6a and c). We therefore concluded that decreasing degradation of GCSF is the superior strategy for stimulating neutrophil maturation.

Thus far, we have used *in silico* logic model simulations to generate hypotheses about optimization of GCSF potency in living systems. This work suggests decreasing degradation of receptor bound GCSF is an effective strategy for improving potency *in vivo.* In previous work, a mutant GCSF with weaker receptor binding affinity at the endosomal pH exhibited decreased degradation *in vitro* through increased recycling of internalized receptor, resulting in increased potency of the molecule *in vitro* [133]. To examine whether decreased degradation had any effect in an *in vivo* setting, we determined white blood cell (WBC) counts in mice following treatment with wild-type GCSF or mutant GCSF engineered for increase dissociation at an endosomal pH (mutant D113H). Mice were first treated with 5-Fluorouracil (5FU) for 24 hours to inhibit haematopoiesis followed by administration of either wild-type or mutant GCSF. The mutant was more effective in increasing WBC count than wild-type GCSF (Figure 3-7) . This result illustrates that cFL models can faithfully represent complex multi-scale systems and that the hypotheses generated from the Q2LM analysis presented here are relevant in both *in vitro* and *in vivo* settings.

3.5 Discussion

In this work we presented Q2LM as a means for generating insights from a cFL model of a biological system based on literature knowledge. We queried the model to address two questions highly relevant to translational research: 1) which therapeutic perturbation of a system will result in a pre-defined clinical goal and 2) in which environments will this perturbation be effective? We used this software framework to explore two biological systems of different scales. With the first, an intracellular signaling model, we illustrated how the software can be used to make testable hypotheses. With the second, a multi-scale model of GCSF administration, we generated and tested hypotheses to show that a logic model was able to recapitulate the experimentally validated results of a mechanistic ordinary differential equation without the prerequisite of estimating a multitude of kinetic parameters.

Building a logic model requires converting a linguistic description into logic gates, which requires a significant amount of abstraction of the system. Logic models are a natural framework for modeling intracellular signaling networks because relationships between proteins are commonly described in terms of their influence (e.g. 'Phosphorylation by JNK activates c-Jun' and ' TGFα stimulation activates the MEK/ERK pathway'). However, building a logic model of a biological system describing interactions between species at the tissue, cellular, and molecular level is arguably less intuitive, in part because the relationships between these types of interactions and logic gats are less obvious (e.g. it is initially unclear how 'binding a receptor' and 'intracellular degradation' can be described with logic gates; these considerations are further explored in Supplementary Text C). Nevertheless, with our logic model of GCSF we demonstrated that transforming such linguistic descriptions into a logic model can provide valuable insights into the operation of a system.

Along with abstracting the relationships between species by describing them as logic gates, the concepts of time and amount are also abstracted in a logic model. The plots presented in Figure 3-4 and 3-5 are similar in appearance to time courses. However, the values of species were plotted as a function of simulation step, not time. Thus, these plots allow one to directly 'follow the logic' of environmental conditions and perturbations, which is not equivalent to examining the value of a species as a function of time. The exact relationship between simulation steps and time cannot be ascertained without additional information regarding the dynamic behavior of the system. Similarly, the meaning of the values of species in relation to a physical descriptor such as concentration is unclear without additional information. Nevertheless, the relative values of species in simulations of the same model carry interpretable information regarding the qualitative effect of perturbations (e.g., the value of N is nonzero for more simulation steps when degradation is inhibited than when it is not) from which we can form a testable hypotheses (e.g., inhibiting degradation will increase the process of neutrophil maturation).

Because the quantities resulting from cFL models are abstract, it raises the question of whether modeling with ostensibly simpler Boolean or discrete logic would be sufficient for the analysis we present here. Indeed, cFL models use traditional AND, OR, and NOT gates to specify the topology of a network, such that tools developed for either analysis are readily interchangeable. However, the use of cFL is justified for several reasons. First, discrete models lack transfer functions such that analyses similar to that shown in Figure 3-7 could not easily be performed with a discrete model. Furthermore, analysis with cFL is no more difficult than one with discrete logic because of the simplicity of the cFL formalism and ease of specifying a model and its transfer functions in Q2LM. Moreover, cFL modeling allows one to explorethe effects of a number of additional parameters, such as the amount of perturbation, different implementations of perturbations, and the effect ofnoise in the transfer function parameters. Such explorations allow one to ascertain whether the predictions are robust to variations of the model, which if confirmed, increases the confidence in their reliability.

One of the main results of this work is a novel 'seamless' approach to multi-scale modeling, exemplified by our logic model of GCSF administration that integrates ligand/receptor binding and endocytic trafficking at the molecular level, the transition between differentiation states at the cellular level, and systemic pharmacokinetics at the tissue level. The insights from this model were validated both *in vitro* and *in vivo.* Thus, the relevance of this model to the therapeutic administration of other receptor agonists should be considered. Because intracellular trafficking is important for cellular responses to other stimulatory ligands such as EGF and IL2 [80], it is likely that the insights from this model will be applicable to the administration of these molecules. More broadly, these results may be applicable to therapeutics for which endosomal degradation is an important mechanism for clearance, underscoring the importance of understanding intracellular trafficking when administering receptor agonists as therapeutics [61, 163].

From this work overall, we submit that our Q2LM framework holds promise for effective use toward generating testable hypotheses of interest in academic as well as industrial settings. Additionally, the further development of cFL will enable the prediction of perturbation effects on a complex system without requiring a large amount of experimental data, thereby facilitating the use of mathematical models for guiding scientific decisions.

Chapter 4

Investigation of the ability of trained cFL models to make precise predictions with Q2LM

4.1 Background

In previous chapters, we have developed a formalism for modeling quantitative relationships between species called constrained fuzzy logic (cFL) and demonstrated its use for enhancing insights gleaned when training logic models of signaling networks [102]. We have also used cFL to make predictions regarding context-specific therapeutic effects with logic models built entirely based on prior knowledge (i.e. not trained to data) using the software framework Querying Quantitative Logic Models (Q2LM) [104]. However, the ability of cFL models trained to data to make these predictions has not been fully investigated.

In this work, we adapt Q2LM to make predictions with models trained to data. We divide our results into three main sections: (1) Model training and Q2LM Prediction; (2) Examination of conditions in our initial data set useful for training models that will make precise predictions; and (3) Experimental design for suggesting more informative conditions. We first train logic models to a new dataset describing the signaling response of HepG2 cells to various environmental stimuli. We train multiple models (a 'family' of models) because the data is not sufficient to constrain both the topology and parameters of the models, a non-identifiability problem common in training biological network models [75]. We then use the trained models and Q2LM to ask: (1) What therapeutic or combination of therapeutics will result in a desired outcome? and (2) In what environmental contexts will the therapeutics be effective? We experimentally test several of these predictions and find that these predictions are highly accurate in a statistical sense. However, in the course of making predictions, we find that some were ambivalent, in that half of the models predict a therapeutic will be effective whereas the other half do not. Such ambivalent predictions indicate that the data did not sufficiently constrain the models to be able to make all predictions of interest. Thus, we undertake a series of computational analyses to determine if we can suggest experiments that would better constrain our models. We find that inclusion of ligand doses is particularly important for training models that will have few ambivalent predictions. Additionally, we are able to suggest conditions predicted to be informative, but they are not helpful in producing more precise predictions. Thus, in the absence of a principled method of ensuring constrained models and precise predictions, we emphasize the importance of considering model ambivalence during model training and analysis.

4.2 Results

4.2.1 Model training and Q2LM Prediction

We first gathered a dataset of phosphorylation of seventeen intracellular proteins in HepG2 cells exposed to one of several small molecule inhibitors followed by stimulation with a growth factor or inflammatory cytokine, all at various doses (Figure 4-2a). Briefly, cells were incubated with small molecule inhibitor for 30 minutes prior to exposure to ligand. For the initial dataset (Figure 4-2), BioPlex bead-based bioassays were used to determine protein phosphorylation in cell lysate collected prior to any perturbation (inhibition or ligand exposure) as well as a pooled lysate collected 10 and 30 minutes after ligand exposure. For the validation data sets (Supplementary Figures D-7 and D-7), protein phosphorylation was measured with BioPlex in cell lysate collected before perturbation as well as 30 minutes after ligand exposure.

For the initial dataset, each data point was normalized using the 'Booleanizer' method described previously [124]. Briefly, this method calculated the relative fold change of each signal upon stimulation and transformed it using a hill function to smooth the output. A noise penalty was then applied by multiplying the resultant relative fold change by the Langmuir transformation of the value of the signal relative to the maximum value observed in the experiment. Importantly, in this work when calculating the relative fold change, we used the basal value of the signal adjusted for vehicle effects. We defined 'basal value' as the value of the signal before either stimulation or inhibition had occurred. Thus, if a signal decreased due to inhibition of an upstream node, and the stimulation condition did not activate the signal, its normalized value would be negative. This property had important implications to model simulation.

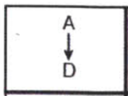
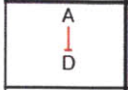
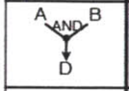
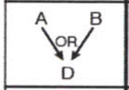
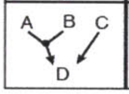
In order to correctly model our normalized data, an alternative simulation procedure was used. In this simulation procedure, the constrained fuzzy logic formalism using min/max operators for AND and OR gates described in Figure 4-1a and b was used. This formalism is equivalent to that in Chapter 2, Figure 2-1 with the exceptions that negative inputs are supported and inhibitory effects were modeled using the formula $output = -f(input)$. To simulate the behavior of the network, effects of inhibitors on the basal level of downstream proteins were evaluated (Figure 4-1c through e). Next, effect of stimuli on the signal was evaluated by propagating the activating effect of the ligand to downstream nodes without considering the effect of the inhibitor on basal level. Finally, the effect of the inhibitor on the ability of the stimuli to activate downstream species was evaluated by using the values from the previous step as the initial points for this step. The inhibition amount was subtracted from inhibited nodes' values and this effect was propagated to downstream nodes. To obtain the simulated species' values, the effect of the inhibitor on basal levels was added to values from the last step (Figure 4-1e). This simulation procedure allowed the inhibition effects on basal values to be modeled separately from effects on the ability of a stimulation condition to activate its downstream species.

A prior knowledge network (PKN) was constructed using that from [102] as a basis. However, this PKN modeled species as one single 'activity' (e.g. "Mek activates Erk") rather than incorporating knowledge of the influence of phosphorylation of specific sites on catalytic activity of the kinase (e.g. "serine phosphorylation of Mek results in increased phosphorylation of the activation loop of Erk). Because we were measuring the phosphorylation of these specific sites and not activity as a whole, we extended the PKN to include information describing phosphorylation of different protein domains. Different domains were included as additional nodes in the network, and their phosphorylation by upstream kinases and influence on activity of the protein encoded as interactions deduced from a variety of sources [166, 167, 60, 62, 6, 34, 2, 52, 74, 109, 92, 138, 141]. This PKN (Supplementary Figure D-1) was then trained to the data.

Model training followed the procedure described in Chapter 2 with the addition of a PKN processing step that added nodes and interactions to allow for the training to capture affects of inhibition on basal signal value. Additionally, in the first training step in which transfer functions are chosen from a predefined suite of possible transfer functions using a genetic algorithm, the suite of transfer functions was different than that used in Chapter 2. In the current application, the same suite was used for all interactions and consisted of nine transfer functions with fixed hill coefficient ($n = 3$) and all combinations of $EC_{50} = 0.3, 0.5, 0.7$ and $g = 0.3, 0.7, 1$ for a total of nine possible transfer functions.

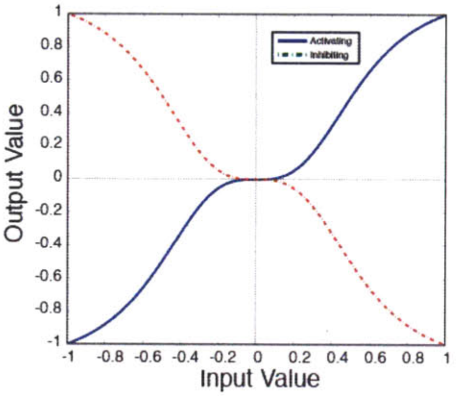
The CellNOpt training process was repeated 993 times, resulting in 993 constrained fuzzy logic (cFL) models. Of these, we used 120 models that fit the data

Figure 4-1 *(facing page)*: Alteration to cFL methodology and simulation to allow for normalizing to basal value. (a) The modeling formalism for evaluating logic gates. (b) Transfer function used to evaluate negative effects. For example plot shown, $g = 1$, $n = 3$, and $EC_{50} = 0.5$. (c) Network model used to demonstrate simulation procedure. Note that inhibited species are represented by two nodes, one that participates in the signaling network as activated by upstream stimuli, and one marked with a 'B' that only affects basal levels upon inhibition. (d) Data that the model in (c) produces upon various stimulation and perturbation conditions. (e) Simulation procedure for each condition in (d). Effect of inhibition on basal values is determined (Step 1) by propogating the effects on inhibition on the inhibitted species' nodes marked with a 'B'. In step 2, the effects of stimulation are determined without considering basal level effects. In step 3, the effects of inhibition of the nodes participating in the network are determined based on the results of step 2. To obtain the final species' values, the basal affect is added to the effect of the stimuli and inhibitor a.) Logic Gate / Constrained Fuzzy Logic Equation

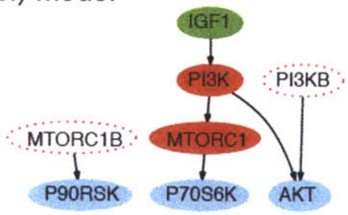
Logic Gate	Constrained Fuzzy Logic Equation
	1. $D = f(A)$
	2. $D = -f(A)$
	3. $D = \min(f(A), f(B))$
	4. $D = \max(f(A), f(B))$
	5. $D = \max(\min(f(A), f(B)), f(C))$

b.)

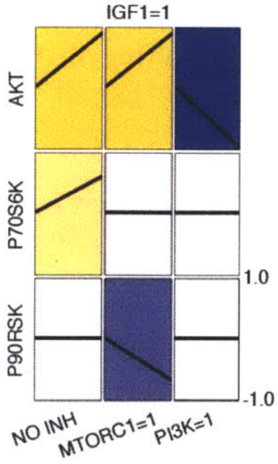
$$output = \begin{cases} g * (1 + K^n) \dfrac{input^n}{input^n + k^n} & input \geq 0 \\ -1 * g * (1 + k)^n \dfrac{|input|^n}{|input|^n + k^n} & input < 0 \end{cases}$$



c.) Model



d.) Data



e.) Simulation Procedure

Condition: IGF1 = 1

	Akt	p70s6k	p90RSK
1. Inhib Basal Effect	0	0	0
2. Stim Effect	0.9	0.6	0
3. Inhib & Stim Effect	0.9	0.6	0
Final Value (Step 1 + 3)	0.9	0.6	0

Condition: IGF1 = 1, PI3K inhibition

	Akt	p70s6k	p90RSK
1. Inhib Basal Effect	-0.7	0	0
2. Stim Effect	0.9	0.6	0
3. Inhib & Stim Effect	-0.3	0	0
Final Value (Step 1 + 3)	-1	0	0

Condition: IGF1 = 1, MTORC1 inhibition

	Akt	p70s6k	p90RSK
1. Inhib Basal Effect	0	0	-0.7
2. Stim Effect	0.9	0.6	0
3. Inhib & Stim Effect	0.9	0	0
Final Value (Step 1 + 3)	0.9	0	-0.7

within 6.6 percent of the best fit for subsequent analysis (see Supplementary Figure D-5 for dependence of kept models' features on solution pool size). The fit of these models to the data and their topology is summarized in Figure 4-2c and d. The results of 10-fold cross validation (Supplementary Figure D-6) provided a first indication that cFL modeling of this data resulted models that were predictive and not over-fit.

While the topology and fit of the trained models yielded interesting hypotheses regarding the network structure (Supplemental Table D.1), we focused our analysis here on the ability of these models to predict species state. We first extended the software package Querying Quantitative Logic Models (Q2LM) to allow for simulation of cFL models trained to data. We then used Q2LM to predict the values of species in our models in three scenarios.

Q2LM predictions in this chapter are made using three components: (1) cFL models trained to data; (2) "Scenario" file specifying environmental conditions and perturbations; and (3) an optional "Criteria" file indicating the species to be compared across conditions. The scenario file was used to generate indicated combinations of stimuli and inhibitors. Each 'simulation project' was then simulated with the cFL models. If a criteria was provided, the simulation results in the perturbed conditions were compared to those without perturbation and the relevant criteria evaluated (Figure 4-3).

To specify a scenario, we specified two components: environments and perturbations. Environments were considered independent aspects of the cellular milieu whereas perturbations were the inhibitors or stimuli we proposed to expose to the cells in order to result in a desired outcome. This semantic distinction was used to set up the problem efficiently recognizing that, in some instances, a species might be an environmental species whereas in others, it might be a perturbation.

The scenarios used in this work contained different 'environmental' stimulation conditions and the same perturbation conditions: varying inhibition of Map3k7, p38, IκK, PI3K, or Mek at various doses alone or in combination. The environmental stimulation conditions were chosen to approximate growth, inflammatory, and mixed environments, as described in Table 4.1.

To evaluate the simulation predictions, we assessed how constrained they were by evaluating the inner quartile range (IQR) of each predicted species value across all models. For example, if a prediction had a small IQR across all models, it was constrained in that the models were able to precisely predict its value (Figure 4-4a). Conversely, if it a large IQR across models, it was unconstrained in that the models did not agree to a precise prediction of its value (Figure 4-4a). A plot of the distribution of IQR for all predictions indicated that between 65 and 80 percent of the predictions for each scenario were predicted with an IQR of less than 0.2 and were thus fairly well constrained (Figure 4-4b and c).

We next simulated our models to address our main question of interest: Which inhibition conditions (perturbations) met a criteria and in which stimulation conditions (environments) were they effective? We focused on inhibition of various downstream 'effecter' proteins with eight criteria described in Table 4.2. To determine if an inhibition condition met our criteria, we evaluated whether the relative fold change of

Figure 4-2: Training cFL models to dataset of HepG2 signaling response. The prior knowledge network in (a) was processed and trained to the dataset in (b) as described in Chapter 2. This analysis yielded the family of models shown in (c) where the grey/black intensity scale of the gates corresponds to the proportion of individual models within the family that include that gate. Thus, links colored black were present in all models whereas links colored grey were present in a fraction of the models. The fit of the models to the experimental data is shown in (d) where the average simulation result is shown with a dashed blue line and the absolute difference in measured and average simulated signal level is indicated with a background color ranging from green (good fit) to red (bad fit). Larger views of all figures can be found in Supplementary Figures D-1 through D-4

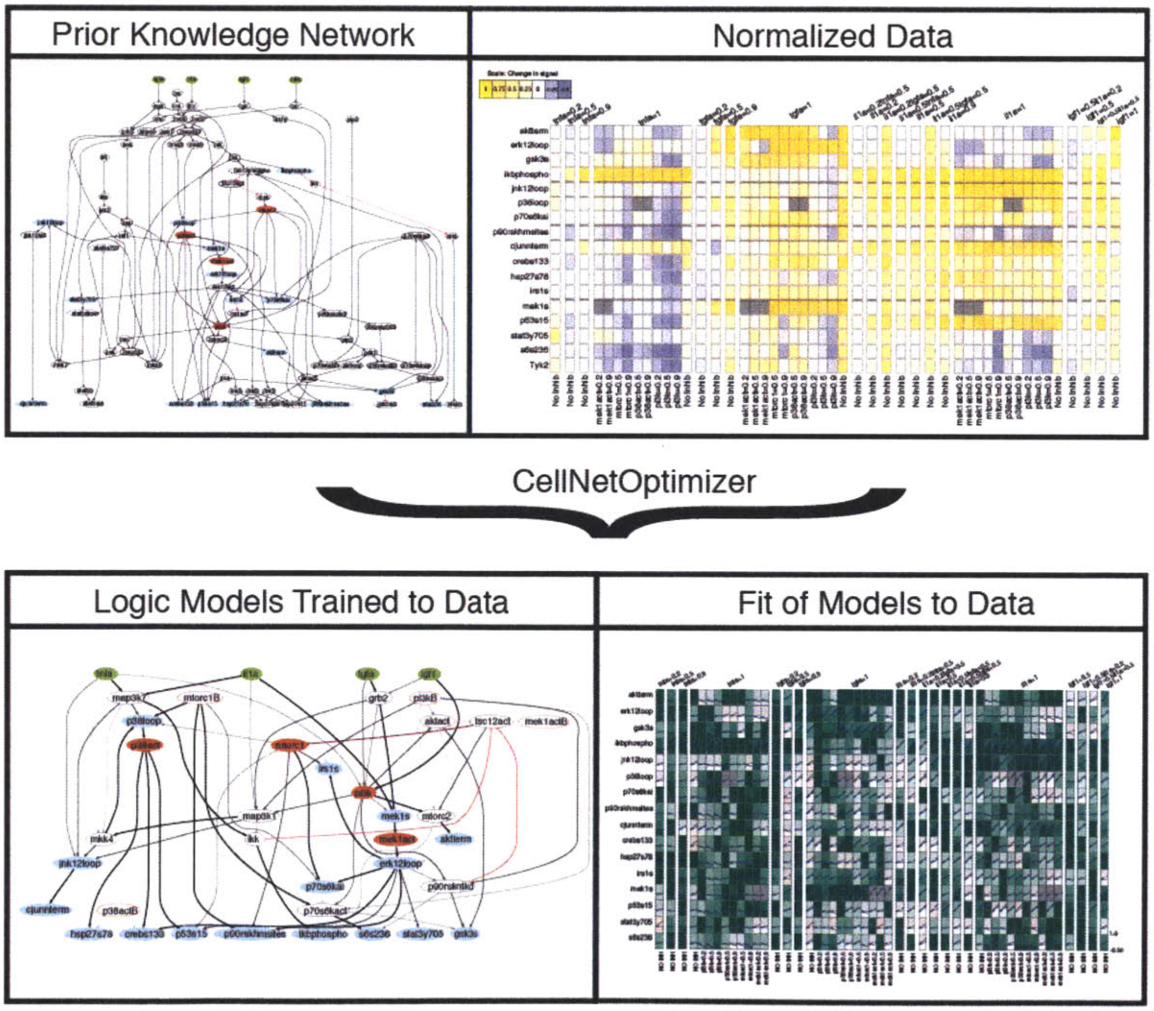


Figure 4-3: Workflow for use of Q2LM for species value prediction and criteria evaluation. Steps on the left side (colored black with sold outlines) were necessary for species' predictions. Steps on the right side (colored grey with dashed outlines) were only necessary if a criteria was to be evaluated.

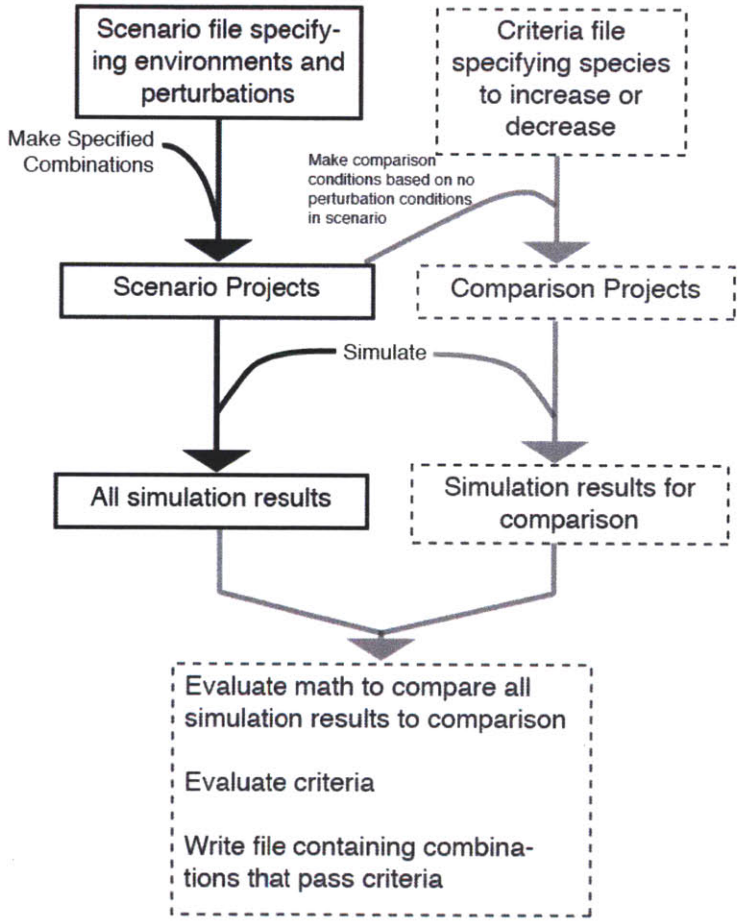

Scenario file specifying environments and perturbations
Criteria file specifying species to increase or decrease
Make Specified Combinations
Make comparison conditions based on no perturbation conditions in scenario
Scenario Projects
Comparison Projects
Simulate
All simulation results
Simulation results for comparison
Evaluate math to compare all simulation results to comparison
Evaluate criteria
Write file containing combinations that pass criteria

Figure 4-4: The inner quartile range as an indication of the precision of predictions. (a) Hypothetical distributions of three species' value predictions for one condition across the 102 models. A lower inner quartile range (IQR) indicates the predictions were more precise (Hsp27 and Erk plots) while a higher IQR indicates the models were not constrained in their prediction of the species' value (Map3k1 plot). (b) Cumulative distribution of IQR of predictions across models. The distribution for each prediction scenario (Table 4.1) is indicated with a dotted line whereas a black line indicates the distribution for the conditions that the models were fit to. For the prediction scenarios, 65 - 80% of the predictions had an IQR across models of ≤ 0.2, indicating that they were precisely predictions. (c) To simplify visualization of these types of distributions, we will present bar graphs of the percent of predictions that were reasonably well constrained (i.e. IQR ≤ 0.2). As expected, a higher fraction of predicted species' values for fitted conditions had an IQR of ≤ 0.2.

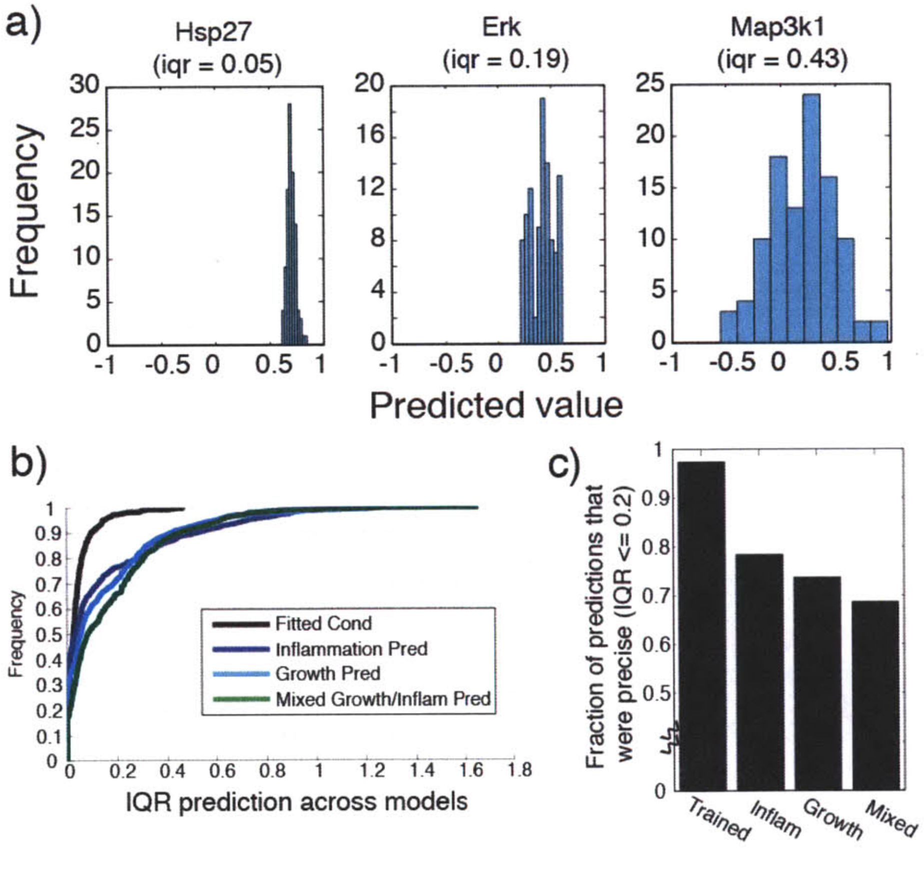


Table 4.1: Scenarios simulated by Q2LM for predicting therapeutic effects

	"Environmental" Stimuli (partial and full activation)	Inhibitor "Perturbations" (partial or full single inhibitors and all pairs)
Inflammatory Scenario	TNFα and IL1α alone or in combination	Map3k7, p38, IκK, PI3K, MEK
Growth Scenario	TGFα and IGF1 alone or in combination	Map3k7, p38, IκK, PI3K, MEK
Mixed Growth/Inflammatory Scenario	TGFα in combination with TNFα or IL1α	Map3k7, p38, IκK, PI3K, MEK

the species of interest decreased by at least 50 percent with perturbation compared to without (i.e. stimulation and inhibition compared to stimulation alone) for each model simulated. We then determined the fraction of models that predicted that the condition met our criteria. Perturbation conditions that met the criteria in a high fraction of models were considered good therapeutic candidates for their specific environments whereas those that met the criteria in a low fraction of models were not. Inhibition conditions that met the criteria in an intermediate fraction of models could not be categorized because the models were ambivalent as to whether or not they would be effective. The results of this analysis is summarized for all predictions in Table 4.3. Overall, we found that 16% of predictions were ambivalent.

Table 4.2: Therapeutic effect criteria evaluated by Q2LM

Criteria	Species that should decrease by a relative change of 50%
1	cJun
2	Hsp27
3	Creb
4	s6
5	cJun, Hsp7
6	Creb, s6
7	cJun, Hsp27, IκB
8	Creb, s6, p70s6k

Table 4.3: Summary of Q2LM Evaluation of Criteria across Conditions. The first four columns contain the number of perturbation conditions that the indicated fractions of models predicted to be effective. The last column contains the fraction of 'ambivalent' predictions (i.e. the faction of perturbations predicted to be effective in between 20 and 80% of models).

Criteria Species	Scenario	Zero Percent	Between Zero and 20%	Between 20 and 80%	Greater than 80%	Fraction Ambivalent
cJun	Inflam	1628	47	996	44	0.37
	Growth	1695	136	272	612	0.10
	Mixed	2084	349	764	51	0.23
Hsp27	Inflam	196	144	684	1620	0.25
	Growth	555	360	0	1800	0.00
	Mixed	179	428	832	1819	0.26
Creb	Inflam	175	308	434	1798	0.16
	Growth	591	216	612	1296	0.23
	Mixed	866	313	966	1113	0.30
s6	Inflam	299	1052	344	1020	0.13
	Growth	915	780	0	1020	0.00
	Mixed	828	1206	0	1224	0.00
cJun, Hsp27	Inflam	1628	47	996	44	0.37
	Growth	1695	136	272	612	0.10
	Mixed	2086	393	728	51	0.22
Creb, s6	Inflam	549	1082	64	1020	0.02
	Growth	919	776	234	786	0.09
	Mixed	1061	973	624	600	0.19
cJun, Hsp27, IκB	Inflam	1731	182	758	44	0.28
	Growth	1695	604	128	288	0.05
	Mixed	2240	428	566	24	0.17
Creb, s6, p70s6k	Inflam	613	1082	52	968	0.02
	Growth	919	776	273	747	0.10
	Mixed	1290	744	792	432	0.24
Sum		26456	12624	11391	19033	0.16

We experimentally tested a few conditions of interest that the models suggested would be effective for decreasing phosphorylation of either cJun, Hsp27, and IκB or Creb, s6, and p70s6k (experimental data shown in Supplementary Figure D-7). Although we were initially interested in Map3k7 (TAK1) inhibition alone, off target effects of the inhibitor used (5Z-7-Oxozeaenol) were evident in a subsequent dataset (Figure 4-12) and cited as possible from the literature [106], so we considered application of this inhibitor to inhibit both Map3k7 and Mek.

Relative fold decrease for the experimental data and model predictions is depicted in Figure 4-5c and d). We found that, for most protein signals, the models correctly predicted whether or not the signal would decrease (Figure 4-5a and b). The computed accuracy of the predictions was dependent on the exact threshold for considering a decrease significant, but for a reasonable choice of thresholds (average model value decreasing by 50% and average experimental value decreasing by 40%), the true positive rate was 86% and false positive rate was 44%. The contingency table associated with these thresholds for decreases was statistically signifiant ($p = 0.0028$) determined with a fisher exact test.

This analysis also yielded the observation that, for some conditions and signals, model predictions were ambivalent (i.e. some predicted that a signal would substantially decrease whereas others did not; marked with an asterisk in Figure 4-5d), leading to a bimodal predicted response of cJun in conditions with IL1α stimulation and Map3k7 inhibition (Figure 4-5e). To determine differences in models that were driving these predictions, we classified each model as either predicting that the signal would decrease or not and plotted the fraction of models in each set that contained each interaction in the processed PKN (Figure 4-5f). We found that all models that predicted cJun would not decrease in the relevant conditions contained an interaction linking IL1α and Map3k1 whereas only a few models that predicted cJun would

Figure 4-5 *(facing page)*: Experimental validation of Q2LM predictions. The computed accuracy of the predictions was dependent on the exact threshold for considering a decrease significant. Here, we varied the threshold for considering an experimental or simulated value decrease significant and calculated the true positive rate (TPR) and false positive rate (FPR) for each threshold pair. The resultant Receiver Operator Characteristic Curve (ROC) indicated that model predictions behaved as expected and accurately predicted significant decreases in phosphorylation of (a) single and (b) pairs of signals for a reasonable choice of thresholds that determined if a decrease is deemed significant. The ROC for predicting if three signals would decrease was similar to that for pairs (not shown). For the thresholds indicated by the arrow in (a), the magnitude of the decrease in signal is depicted in (c) for the experimental data and (d) for average model prediction. The colors in (c) and (d) indicate if the change is classified as significant (blue for yes, white for no). In (d), the species marked with an asterisk were predicted to decrease in some models but remain the same in others, resulting in a bimodal distribution of predictions (demonstrated in (e) for species marked with a yellow asterisk). (f) For the case in (e), we divided the models into two sets: those that did and did not predict cJun phosphorylation would decrease and plotted the fraction of models containing each interaction in the processed PKN fore each set.

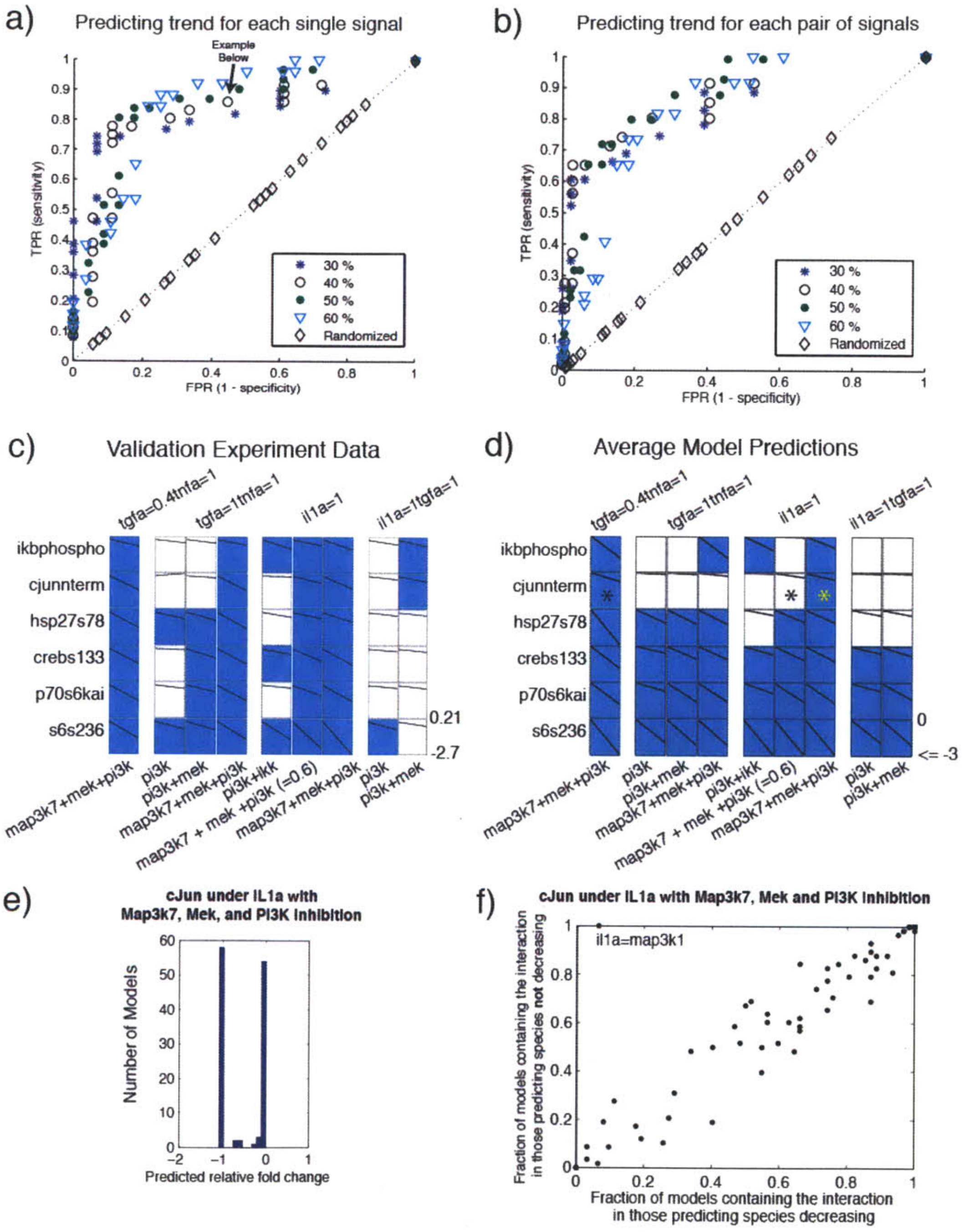

a)
Predicting trend for each single signal
Example Below
TPR (sensitivity)
FPR (1 - specificity)
30 %
40 %
50 %
60 %
Randomized
b)
Predicting trend for each pair of signals
TPR (sensitivity)
FPR (1 - specificity)
30 %
40 %
50 %
60 %
Randomized
c)
Validation Experiment Data
tgfa=0.4tnfa=1
tgfa=1tnfa=1
il1a=1
il1a=1tgfa=1
ikbphospho
cjunnterm
hsp27s78
crebs133
p70s6kai
s6s236
0.21
-2.7
map3k7+mek+pi3k
pi3k
pi3k+mek
map3k7+mek+pi3k
pi3k+ikk
map3k7 + mek +pi3k (=0.6)
map3k7+mek+pi3k
pi3k
pi3k+mek
d)
Average Model Predictions
tgfa=0.4tnfa=1
tgfa=1tnfa=1
il1a=1
il1a=1tgfa=1
ikbphospho
cjunnterm
hsp27s78
crebs133
p70s6kai
s6s236
0
<= -3
map3k7+mek+pi3k
pi3k
pi3k+mek
map3k7+mek+pi3k
pi3k+ikk
map3k7 + mek +pi3k (=0.6)
map3k7+mek+pi3k
pi3k
pi3k+mek
e)
cJun under IL1a with Map3k7, Mek, and PI3K Inhibition
Number of Models
Predicted relative fold change
f)
cJun under IL1a with Map3k7, Mek and PI3K Inhibition
il1a=map3k1
Fraction of models containing the interaction in those predicting species not decreasing
Fraction of models containing the interaction in those predicting species decreasing

Table 4.4: Structural elements of trained models with a correlation coefficient of at least 0.6 in filtered models

	Positively Correlating Interactions
1.	erk12loop=p90rskhmsites with map3k1=mkk4
2.	mtorc1B=p90rskhmsites with map3k1=mkk4
3.	p90rskntkd=gsk3s with pi3k=p90rskntkd
4.	aktact=gsk3s with pi3k=aktact
5.	mtorc1B=p70s6kact with p70s6kact=s6s236
6.	pi3kB=aktact with aktact=tsc12act
7.	mtorc1B=p90rskhmsites with erk12loop=p90rskhmsites
8.	pi3kB=p90rskntkd with p90rskntkd=tsc12act
	Negatively Correlating Interactions
9.	tgfa=pi3k with grb2=pi3k
10.	igf1=grb2 with pi3k=mek1s
11.	p90rskntkd=s6s236 with pi3k=p53s15
12.	erk12loop=p53s15 with mtorc2=p53s15
13.	pi3kB=mtorc2 with mtorc1=p90rskhmsitest

decrease contained this interaction. The fact that cJun did decrease in the experimental test of this prediction (Figure 4-5c) indicated that models that did not contain this interaction were correct. While additional experimental controls are necessary to verify this aspect of the network, it would be a particularly interesting feature because it implicates Map3k1 as a key node in the network allowing growth factors to activate some inflammatory pathways while decoupling that activation from the action of inflammatory stimuli, which seem to signal primarily through Map3k7.

The finding that in some cases there were topological distinctions between models making specific predictions(Figure 4-5) led us to determine if a more general analysis of structural features would yield such insights. Thus, we clustered the trained models based on whether or not they contained each interaction and calculated the correlation between structural features of the network. We found that, while no distinct patterns emerged from the clustering results (Supplementary Figure D-8), the correlation matrix pointed to interesting relationships between interactions (Table 4.4 and Supplementary Figure D-9). While some positively correlated pathways were coincidental (Numbers 1 and 2), those between successive interactions in pathways (Numbers 3 - 9) indicated that if the one interaction was included to fit the data during training, the other was also necessary. The pathways that were negatively correlated demonstrated pathways that were redundant (Numbers 9 - 13). All in all, this analysis provided initial insights into the general structural features of the trained models, but it did not immediately point out differences that would result in the bimodal distributions of predictions described above.

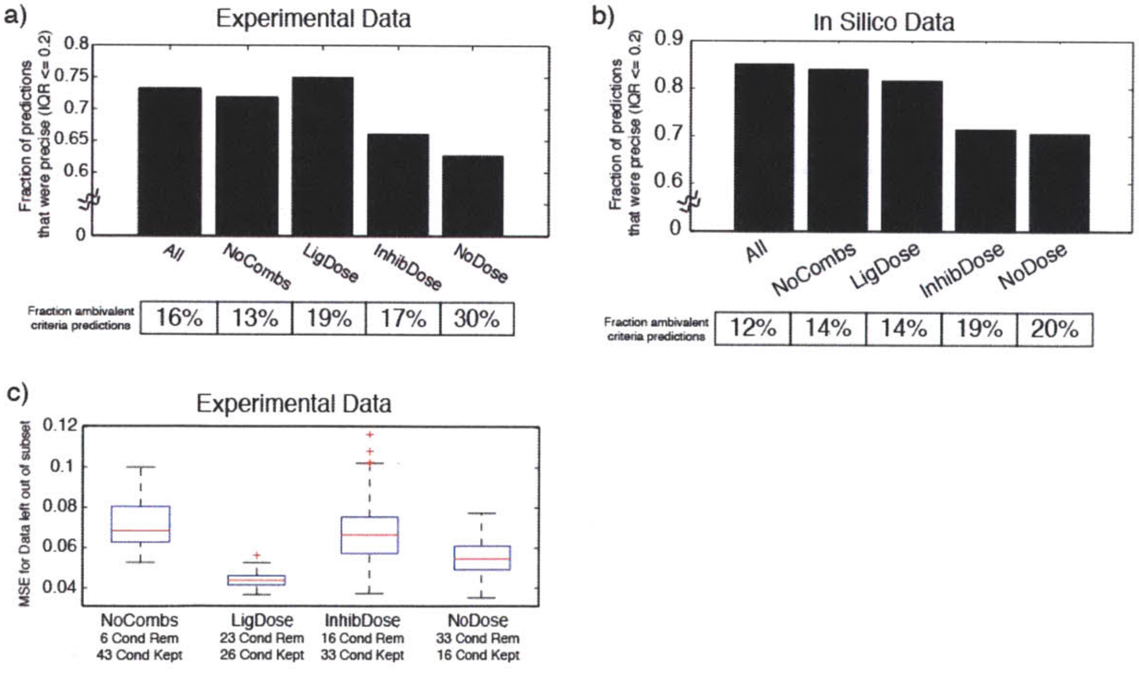
Table 4.5: Subsets of original dataset used to query what aspect of initial training set was useful in constraining the models' topology and predictions.

Subset Name	Data Removed	Data Remaining
All	None	All
NoCombs	Combinations (Combs)	Single Lig Doses + Inhib Doses
LigDose	Combs + Inhib Doses	Single Lig Doses + Full Inhib
InhibDose	Combs + Lig Doses	Full Single Lig + Inhib Doses
NoDose	Combs + Lig Doses + Inhib Doses	Full Single Lig + Fill Inhib

4.2.2 Evaluating original conditions for usefulness in constraining predictions

Given that we had initially conducted and experiment of 49 conditions and found that the resultant models still led to some ambivalent predictions, we wondered how many of the original 49 conditions were useful in training the models. Thus, we created four subsets of our original dataset, and trained the models to the reduced datasets (Table 4.5). To our surprise, we found only negligible differences between the distribution of topology and parameters of the family of models resulting from training to each subset (Supplementary Figure D-10). Furthermore, an analysis of the precision of model predictions indicated that while inhibitor doses were also useful in producing models that made fewer ambivalent criteria predictions, they did not constrain the species' value predictions overall as much as ligand doses. Thus, the main determinant in making constrained predictions was whether or not the dataset contained conditions with ligand doses (Figure 4-6a).

Figure 4-6: Original Training Conditions Necessary to Constrain Models. (a) Experimental data. Cumulative distribution functions for IQR of predictions as an indication of prediction precision. Training to subsets 'NoCombs' and 'LigDose' resulted in predictions that were as precise as those from training to the full training set ('All'). However, training to 'InhibDose' and 'NoDose' resulted in less precise predictions in general. These two subsets did not contain ligand doses, indicating that the ligand dose conditions were useful in training models that would result in precise predictions. (b) The analysis in part *a.* was repeated with *in silico* data that did not contain biological or technical noise. Thus, a comparison of *a.* and *b.* revealed the influence of noise in part *a.* This result indicated that noise was the likely cause of more data being detrimental for making constrained predictions as evidenced by the smaller fraction of precise predicted for the 'All' and 'NoCombs' subsets than the 'LigDose' subset. However, this result did not reveal why ligand doses were especially helpful in making precise predictions. (c) The ability of the models trained to each subset to fit the data left out of that subset indicates that there was overlapping information in the data remaining in the subsets and the data left out.



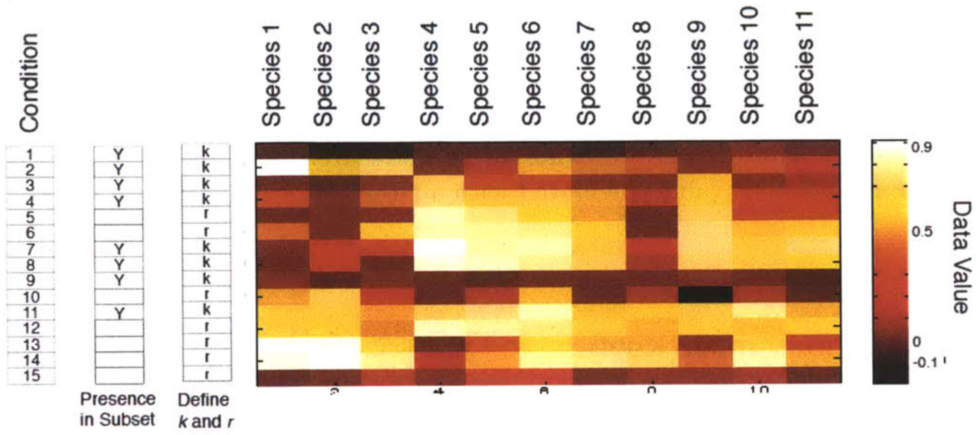
From Figure 4-6a, we noticed two important features: (1) more data was sometimes detrimental for making precise predictions as evidenced by the smaller fraction of precise predictions for the 'All' and 'NoCombs' subsets than the 'LigDose' subset and (2) the inclusion of ligand dose data was most important for making precise predictions. For the former, we believed that the subsets with more data were less successful at constraining the models due to inherent noise in the data. We tested this hypothesis by training models to data sets of the same composition using 'perfect' *in silico* data obtained by simulating one of our trained models (Figure 4-6b). The removal of noise indicated that more data was not detrimental in constraining the model predictions, and we concluded that the cause of more data being detrimental in the model training was due to noise in the experimental data. However, for the latter, we obtained the same general result: the only major difference was in the utility of ligand doses for making precise predictions (Supplementary Figure D-11).

Because noise in the data did not explain the additional utility of the ligand dose conditions, we hypothesized that some conditions were not useful because of redundancy of information contained in the new conditions. We first tested for this effect by determining if the models trained to the subsets could predict the data removed. If so, the models already contained the information regarding the value of signals under the removed conditions. Thus, we would not expect that adding the data to the training set would be helpful in constraining the models. Figure 4-6c indicated models trained to each subset were reasonably successful at predicting the value of those removed from the subset, suggesting that the reason additional conditions were not always informative was indeed that the information contained in the removed conditions was redundant with those kept.

To further investigate if conditions removed from each subset contained information that was redundant with those kept, we first quantified information lost when creating data subsets by computing the minimum distance between signal values of conditions removed and kept from each data subset using the correlation distance metric (Figure 4-7). We found that, when compared to randomly chosen conditions of the same size, conditions removed from the subset 'InhibDose' contained significantly higher distances between kept conditions (Figure 4-8) indicating that conditions removed from the subset (i.e. the ligand dose and combination conditions) contained more unique information than expected by random chance. Unfortunately, this metric was a relative metric that was only meaningful when compared to random subsets of the same size. Thus, we sought an absolute quantity indicating how much information was lost upon removal of conditions from the subsets.

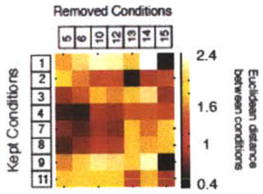
Figure 4-7 *(facing page)*: Use of relative distance metric for determining information lost upon removal of conditions. The data was first split into subsets either randomly or by choice. A standard distance metric (in this case correlation) was used to compute the distance between all kept and removed conditions. The minimum distance between each removed condition and kept conditions represented how 'unique' the removed condition was compared the kept conditions. The more unique (higher minimum distance), the more information that was removed. To interpret the distances, they were compared to those obtained by randomly creating subsets of the same size. If the minimum distances were larger for the chosen than random, more information was removed than expected by random chance.

Split data into subsets: *r* is the set of conditions removed *and k* is the set kept in *m*

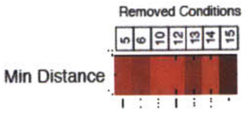


Condition	Presence in Subset	Define *k* and *r*
1	Y	k
2	Y	k
3	Y	k
4	Y	k
5		r
6		r
7	Y	k
8	Y	k
9	Y	k
10		r
11	Y	k
12		r
13		r
14		r
15		r

Calculate correlation distance metric of species values between all kept and removed conditions

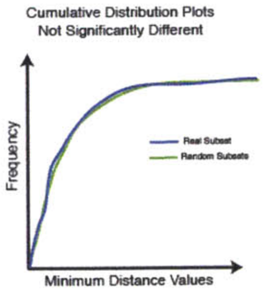


Calculate minimum across kept conditions, resulting in the minimum distance between each removed condition and kept conditions. The higher this number is, the more 'unique' the removed condition was from the kept conditions



Repeat for each subset and randomly chosen subsets.

Compare minimum distance values for each real subset to randomly chosen subsets of the same size. If the minimum distance values are higher for real subsets than those of the randomly chosen subset, the data removed from the real subsets was more unique than we would expect by random chance.



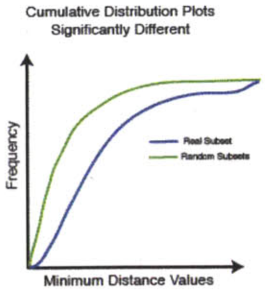
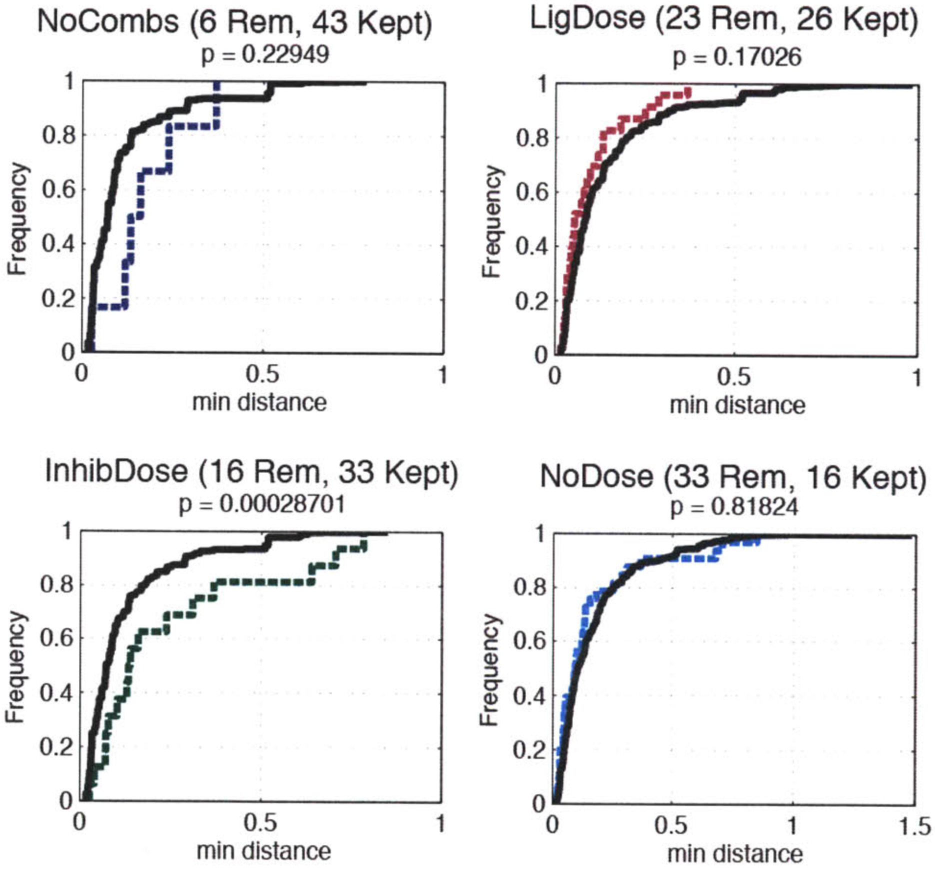


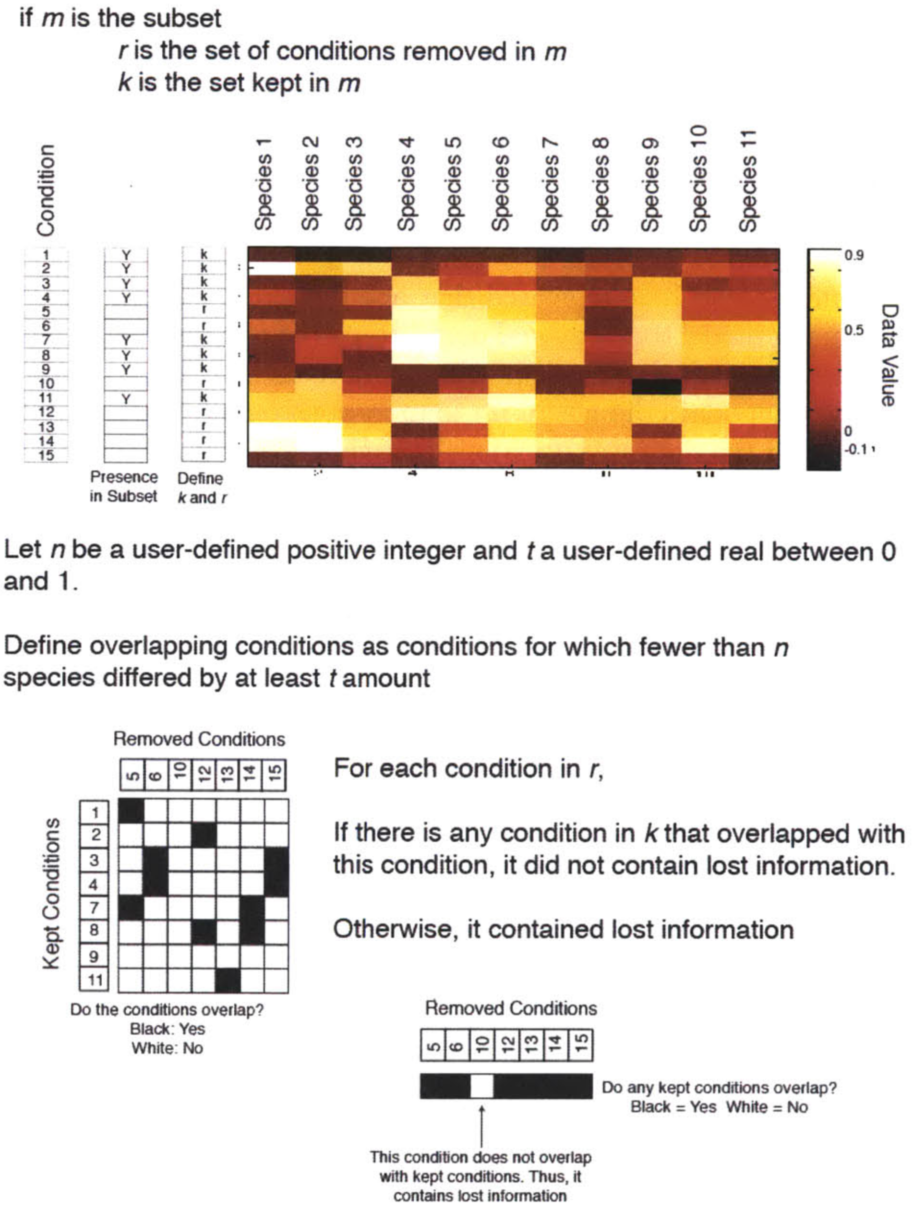
Figure 4-8: Relative distance metric for determining information in data subsets. Correlation distance metric as an indication whether or not more information was lost than expected by random chance. p-value calculated with a t-test



We first attempted to quantify this information with a quantity from information theory, joint information entropy. However, we found that this quantity was not flexible enough for our purposes for two reasons: (1) the requirement that the data first be discretized led to a strong dependency the discretization protocol employed and (2) most conditions differed by at least one discretized signal level. Thus, they were unique and joint entropy was maximized for all sets. We defined a more general metric for lost information by answering the following questions: Did several signal measurements in removed conditions significantly differ from kept conditions? If so, the removed condition contained information that was 'lost' from the remaining data (Figure 4-9). In this metric, there were two thresholds: (1) how much should the signals differ quantitatively to be considered different and (2) how many species should be different to consider the condition to have contained additional information? We calculated the number of conditions containing lost information for several threshold combinations and (Figure 4-10) and found that indeed more conditions contained lost information in the subsets that did not contain ligand dose conditions than those that did.

As a whole, our analysis here indicated that additional data was not particularly helpful in constraining model topology and parameters, but the addition of ligand dose conditions resulted in models that made precise predictions. Noise in the data was not the sole reason that more data did not result in more constrained models. Rather, the information content of the data was a major determinant of how helpful a particular set of conditions would be. We found that some conditions contained redundant information, such that their removal did not adversely affect model training.

Figure 4-9: Metric for quantifying information lost upon removal of conditions. To quantify the absolute number of conditions removed from a subset that contained lost information, we defined a metric by determining if several signal measurements in removed conditions significantly differ from kept conditions based on two thresholds (t and n).



if *m* is the subset
r is the set of conditions removed in *m*
k is the set kept in *m*

Condition	Presence in Subset	Define *k* and *r*
1	Y	k
2	Y	k
3	Y	k
4	Y	k
5		r
6		r
7	Y	k
8	Y	k
9	Y	k
10		r
11	Y	k
12		r
13		r
14		r
15		r

Let *n* be a user-defined positive integer and *t* a user-defined real between 0 and 1.

Define overlapping conditions as conditions for which fewer than *n* species differed by at least *t* amount

Removed Conditions

Kept Conditions	5	6	10	12	13	14	15
1	■	□	□	□	□	□	□
2	□	□	□	■	□	□	□
3	□	■	□	□	□	□	■
4	□	■	□	□	□	□	■
7	■	□	□	□	□	■	□
8	□	□	□	■	□	■	□
9	□	□	□	□	□	□	□
11	□	□	□	□	■	□	□

Do the conditions overlap?
Black: Yes
White: No

For each condition in *r*,

If there is any condition in *k* that overlapped with this condition, it did not contain lost information.

Otherwise, it contained lost information

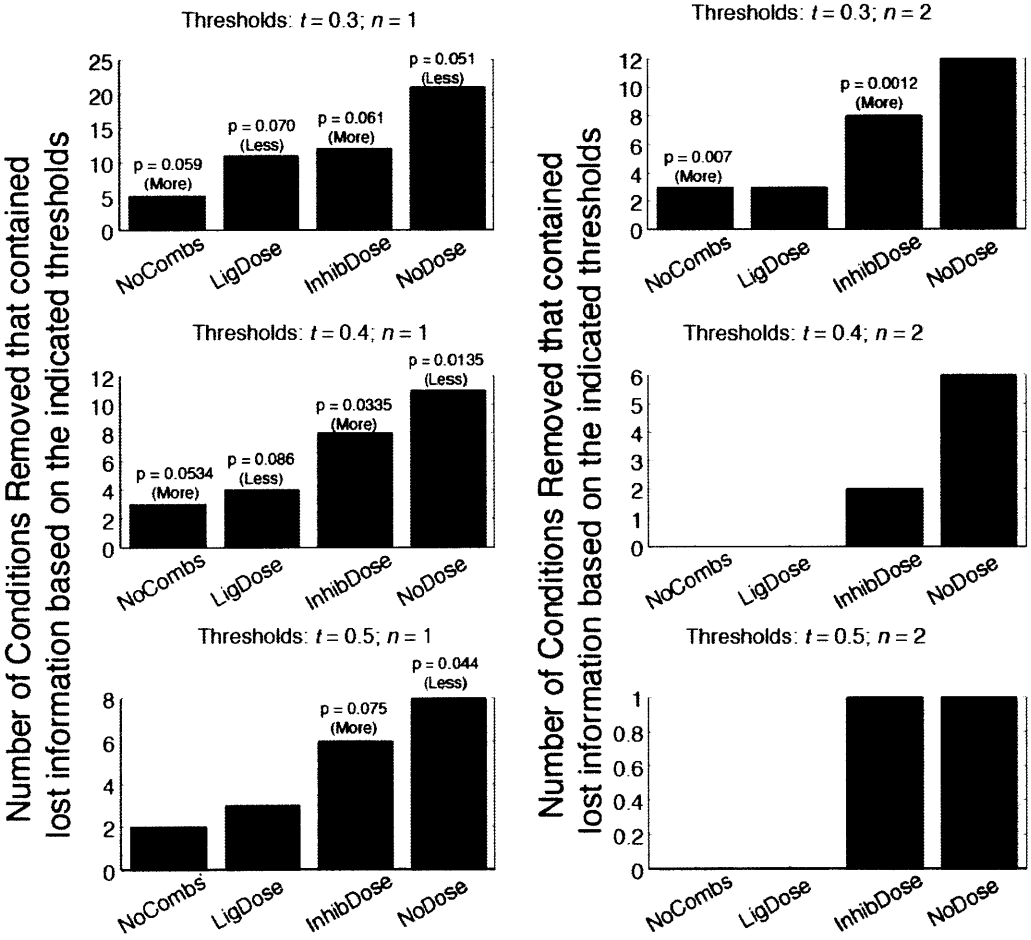
Removed Conditions

5	6	10	12	13	14	15
■	■	□	■	■	■	■

Do any kept conditions overlap?
Black = Yes White = No

This condition does not overlap with kept conditions. Thus, it contains lost information

Figure 4-10: Quantification of data lost from subsets. Titles indicate parameters used in discretization protocol. The first number in title is threshold for difference in signal value for measurement to be considered different, and the second number is how many species had to be different for the condition to have contained lost information. If more or less removed conditions contained lost information than expected by random chance ($p \leq 0.1$), it is indicated above the bars.



NoCombs: 6 Cond Removed; 43 Kept
LigDose: 23 Cond Removed; 26 Kept
InhibDose: 16 Cond Removed; 33 Kept
NoDose: 33 Cond Removed; 16 Kept

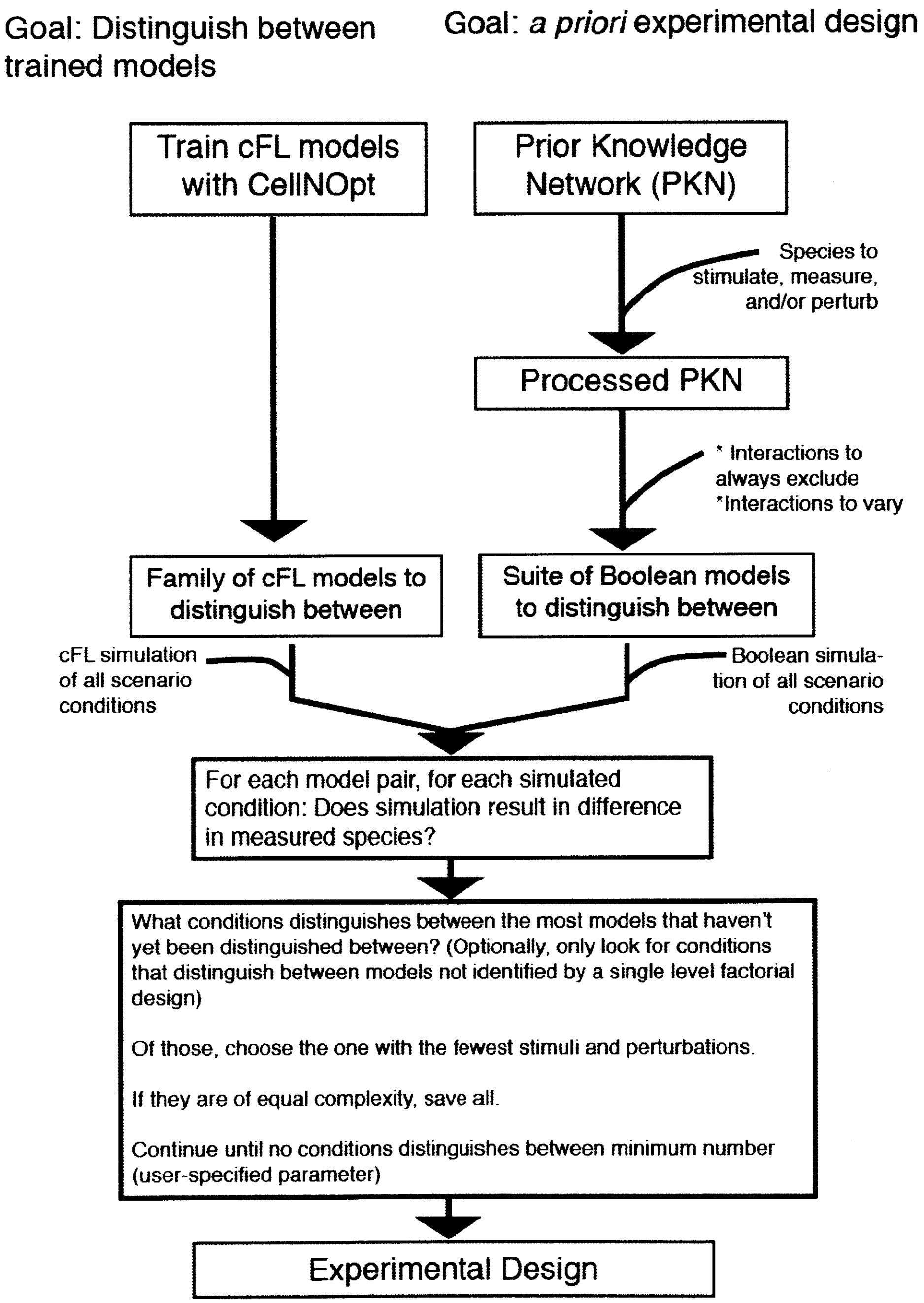
4.2.3 Experimental Design for more precise predictions

Determining conditions to distinguish between trained models

The finding that many of our initially chosen conditions were not useful in training constrained models for making precise predictions led us to question if further experiments could better constrain the models. To determine these experiments, we first simulated a set of models with a multitude of conditions. Conditions for which the models predicted very different values for any measured species were considered to be good candidates for further experiments because they would allow us to distinguish between the models. We further extended Q2LM to suggest combinations of experiments that would allow us to discriminate between many models based on the simulation predictions (Figure 4-11, left side).

Based on this analysis, we conducted an experiment of 9 additional conditions (Supplementary Table D.2) that were predicted to differentiate between models for 5116 of 5151 pairs (99.3%) of 102 models initially trained to all conditions (in later analysis, we discovered that the genetic algorithm had not converged when we initially trained our models. We had based this analysis on these models, but all other analysis in this chapter is based on models trained subsequently to convergence.) Our experiment also contained several control conditions to allow for comparison between experiments. Unfortunately, the results of this experiment indicated that day-to-day variability was high and could not be adjusted for with our experimental controls, such that models that fit the discriminating conditions well were mainly fitting the discrepancy between the quantitative values of the two experiments rather than the specific protein signals whose response to these specific conditions was intended to discriminate between models. This result underscores the importance of controlling

Figure 4-11 *(facing page)*: Workflow for use of Q2LM for experimental design (dQLM). Left Side: In the case of dQLM for conditions to distinguish between trained models, predictions for conditions contained in scenario files were evaluated to determine if they were able to distinguish between models. DQLM evaluated the difference between all model predictions of all measured species for all conditions. If the condition resulted in predictions of any measured species that differed by more than a predefined threshold (0.5 in this case), it was said to be able to distinguish between the two models. DQLM then used a greedy search to determine the simplest sets of experiments that distinguished between as many models as possible. Right Side: In the case of dQLM for *a priori* experimental design, the prior knowledge network was first processed as described in Chapter 2 based on proposed stimuli, inhibitors, and measured species. User-specified interactions were excluded from the general scaffold, and other specified interactions were varied systematically. The resultant Boolean models were then simulated with the scenario conditions, and conditions that resulted in different predictions for measured species defined as able to distinguish between two models. In this case, we assumed that the basis for any design would be a factorial design in which cells were exposed to all stimuli and inhibitors alone and in combination. DQLM then used a greedy search to determine the simplest sets experiments containing combinations of stimuli or inhibitors (or both) that distinguished between as many additional models as possible


Goal: Distinguish between trained models
Goal: *a priori* experimental design
Train cFL models with CellNOpt
Prior Knowledge Network (PKN)
Species to stimulate, measure, and/or perturb
Processed PKN
* Interactions to always exclude
*Interactions to vary
Family of cFL models to distinguish between
Suite of Boolean models to distinguish between
cFL simulation of all scenario conditions
Boolean simulation of all scenario conditions
Experimental Design

For each model pair, for each simulated condition: Does simulation result in difference in measured species?

What conditions distinguishes between the most models that haven't yet been distinguished between? (Optionally, only look for conditions that distinguish between models not identified by a single level factorial design)

Of those, choose the one with the fewest stimuli and perturbations.

If they are of equal complexity, save all.

Continue until no conditions distinguishes between minimum number (user-specified parameter) for day-to-day variability when integrating new conditions.

Experimental design *a priori*

Because so few of the conditions in our initial experiment were useful in constraining the trained models and incorporation of new experiments was complicated by the need for many experimental controls, we wondered if we could use Q2LM *a priori* to determine conditions that would allow us to distinguish between models within our prior knowledge network. To make this challenge tractable, we used Boolean simulation to focus on full stimulation and inhibition conditions that distinguished between different topologies in the prior knowledge network. Additionally, we reasoned that the basis for any design would be a standard one-level factorial design (single stimuli in the presence or absence of single inhibitors). Thus, we only considered conditions with combinations of either stimuli or inhibitors that would allow us to distinguish between more models than a standard one-level factorial design.

We first specified which aspects of the topology we would like to distinguish between and the species that we could stimulate, inhibit, or measure. Regarding the specification of aspects of topology to distinguish between, we could either specify that we would like to determine if a specific edge was present or what edges linking to a given model species were present (i.e. what logic gate controlled the species' activation). We chose the latter for this analysis. The software then simulated different conditions with all models and suggested combinations of experiments that would allow us to distinguish between many models based on the simulation predictions (Figure 4-11). In total, this analysis involved the simulation of 176 conditions in 377,408 models. The results of this analysis (Table 4.6) pointed to a set of ten conditions that would allow us to distinguish between at least ten specified model pairs in at least one of the PKN-derived model topology combinations specified. It was possible to suggest more conditions by adding conditions that distinguished between fewer than ten model pairs, but we chose a 'model pair' threshold of ten based on the ability to distinguish between many models without drastically adding experimental complexity (Supplementary Figure D-12).

Table 4.6: Combination conditions in addition to a single-level factorial design predicted to differentiate between additional models in PKN

Stimuli	Inhibitors
TNFα	Map3k7 + PI3K
IGF1 + IL1α	
IGF1 + IL1α	Map3k7
IGF1 + IL1α	p38
IGF1 + TGFα	
IGF1 + TGFα	p38
IL1α + TGFα	
IL1α + TNFα	
IL1α + TNFα	Map3k7
IL1α + TNFα	p38

We performed an *in silico* experiment with these conditions by simulating one of our trained models with the conditions chosen by the experimental design process as well as randomly chosen conditions for comparison. We also considered conditions with ligand doses since they had previously been demonstrated to be useful (see Figure 4-12 for experimental design). As before, we found that models trained to the various subsets did not differ significantly in how constrained the model topologies were. A few subsets did differ in terms of how constrained the parameters were and featured more constrained EC_{50} parameters but *less* constrained Hill coefficients (Supplementary Figure D-14). These subsets also different in terms of how precise the predicted species values were (Figure 4-13a), and, as before, the sole determinant of how constrained the predictions were was whether or not the models contained ligand doses. Because no beneficial effect was observed for subsets additionally containing the designed conditions, this result demonstrated that the experimental design process was not useful in constraining the models.

Nevertheless, we performed a wet lab experiment with these conditions (Figure 4-12 and Supplementary Figure D-13). To our surprise, we found that the results differed from the *in silico* result, and the designed conditions appeared to result in models that yielded more precise predictions (Figure 4-13b). However, the evaluation of criteria using models trained to the designed conditions were actually more ambivalent than those without these conditions (Figure 4-13b). When compared to randomly chosen subsets of the same size as the designed conditions, we found that predictions with models trained to the designed conditions were slightly more constrained than those trained to randomly chosen conditions, but the effect was negligible. Additionally, models trained to randomly chosen conditions resulted in fewer ambivalent predictions when evaluating criteria (Figure 4-13c).

We suspected that the disparity between the success of the experimental design *in silico* versus experimental data was due to the noise inherent in biological data as well as the inability of the PKN to account for all aspects of the experimental data. This hypothesis was supported by the fact that models trained to all of the data were less precise for the conditions and species to which they were actually fit (Figure 4-13d). To further investigate the data and model features underlying imprecise model fit, we visualized the species that were fit imprecisely and the conditions for which the models' fits were imprecise (Figure 4-14). We found that p70s6K and IRS1 phosphorylation were imprecisely fit for many conditions. Specifically, IRS1 was imprecisely fit in conditions with IL1α stimulation and either MEK or MTORC1 inhibition. To examine structural features underlying this result, we divided the models into two sets: those that predicted the first quartile of values (bottom 25%) and those that predicted the fourth quartile of values (top 25%). We then plotted the fraction of interactions in the processed PKN (Figure 4-14). We found that structural features underpinning the imprecise fits weren't obvious for the conditions with MTORC inhibition, but for cases with MEK inhibition, the pathway contained in the models that was used to activate the TSC complex resulted in differing IRS1 predictions.

From this analysis, we concluded that ligand doses were of utmost importance in training models that would make constrained predictions. While we were able to

Figure 4-12: Experimental Test of Effectiveness of Experimental Design. Conditions used in test of designed experiments. Because of off-target effects evident in the data and cited as possible from the literature [106], the Map3k7 inhibitor actually inhibited both Map3k7 and Mek . Thus, Map3k7 inhibition was changed to Map3k7 + Mek inhibition in any condition it was included in both the real and *in silico* data sets, and the *in silico* result was verified to be the same when this substitution was not made.

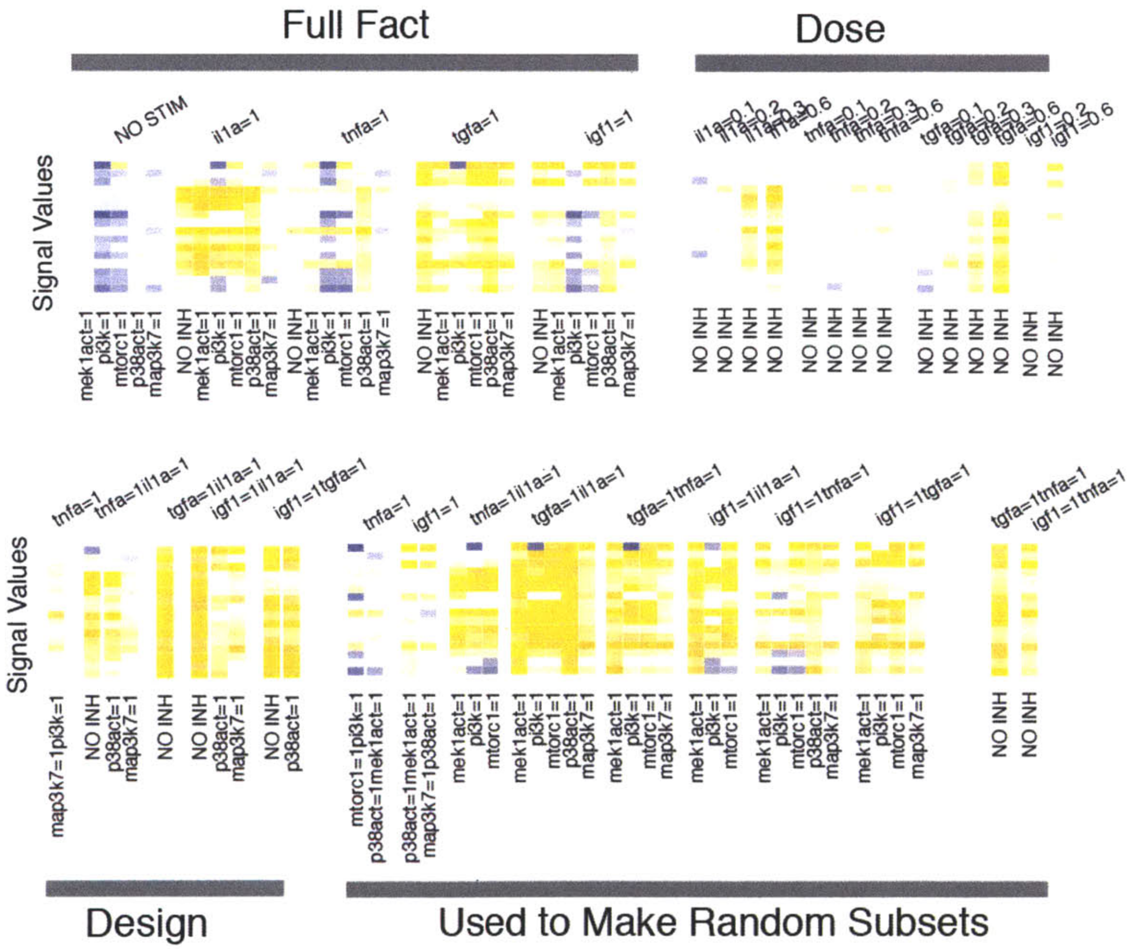
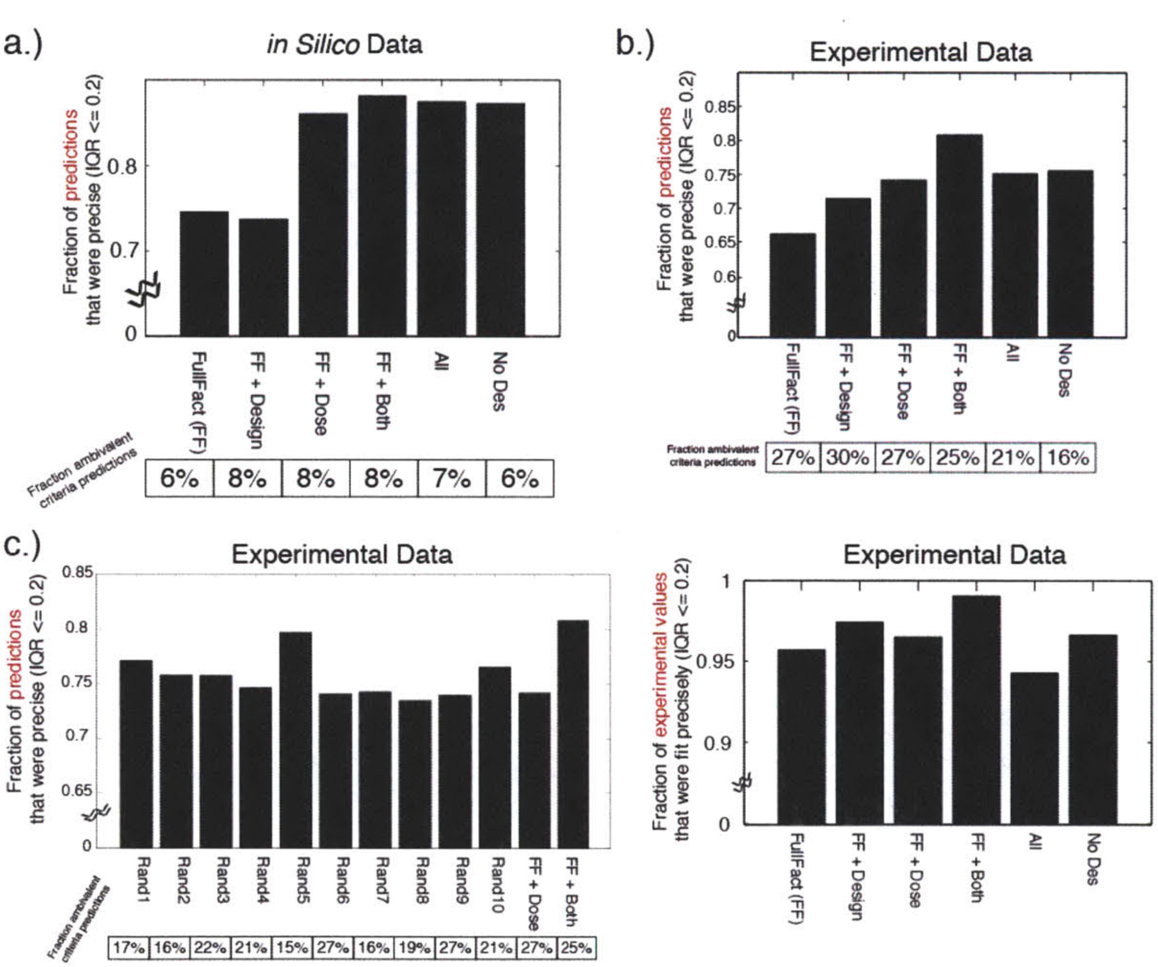
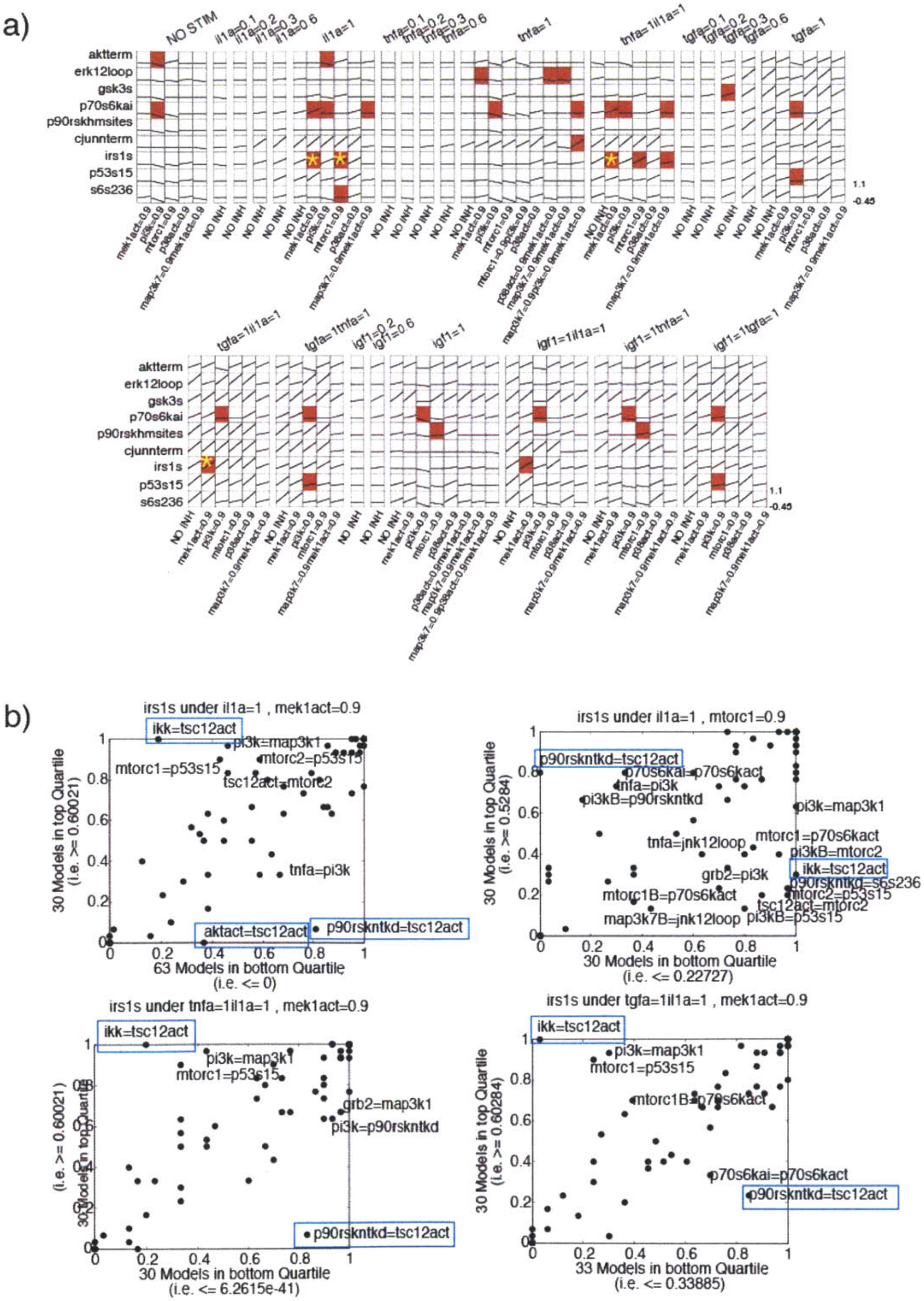


Figure 4-13: Effectiveness of Experimental Design. No significant differences between topologies and parameters were observed for models trained to different subsets. However, differences in the ability to make constrained predictions were observed.



predict a reasonably small set of experiments predicted to better constrain the model predictions, they were not helpful when tested with *in silico* data. Experimental data indicated that some subsets of conditions might have been helpful in constraining the model predictions, but subsequent analysis suggested that this effect was mainly due to inclusion of data that the model training was unable to reconcile either due to noise in the data or lack of appropriate edges in the PKN.

Figure 4-14 *(facing page)*: Analysis of model variability. (a) Data values that the trained models fit imprecisely (i.e. with an IQR of more than 0.25) are colored red. (b) By dividing the models into two sets: those that predicted the first quartile of values (bottom 25%) and those that predicted the fourth quartile of values (top 25%), and plotting the fraction of interactions in the processed PKN, we were able to observe structural features underpinning the imprecise fits in some cases. We show a few examples here. For IRS1 phosphorylation under conditions with MEK inhibition, were were able to see the disparity in fits was due to differences in which pathways were used to activate MTORC1 (which then activated IRS1). For IRS1 phosphorylation under conditions with MTORC1 inhibition, no obvious trends emerged, although similar pathways seemed to be involved.


a)
NO STIM
il1a=0.1
il1a=0.2
il1a=0.3
il1a=0.6
il1a=1
tnfa=0.1
tnfa=0.2
tnfa=0.3
tnfa=0.6
tnfa=1
tnfa=1il1a=1
tgfa=0.1
tgfa=0.2
tgfa=0.3
tgfa=0.6
tgfa=1
aktterm
erk12loop
gsk3s
p70s6kai
p90rskhmsites
cjunnterm
irs1s
p53s15
s6s236
1.1
-0.45
NO INH
mek1act=0.9
pi3k=0.9
mtorc1=0.9
p38act=0.9
map3k7=0.9mek1act=0.9
mtorc1=0.9pi3k=0.9
p38act=0.9mek1act=0.9
map3k7=0.9pi3k=0.9mek1act=0.9
tgfa=1il1a=1
tgfa=1tnfa=1
igf1=0.2
igf1=0.6
igf1=1
igf1=1il1a=1
igf1=1tnfa=1
igf1=1tgfa=1
b)
irs1s under il1a=1 , mek1act=0.9
ikk=tsc12act
pi3k=map3k1
mtorc2=p53s15
mtorc1=p53s15
tsc12act=mtorc2
tnfa=pi3k
aktact=tsc12act
p90rskntkd=tsc12act
30 Models in top Quartile (i.e. >= 0.60021)
63 Models in bottom Quartile (i.e. <= 0)
irs1s under il1a=1 , mtorc1=0.9
p90rskntkd=tsc12act
p70s6kai=p70s6kact
tnfa=pi3k
pi3kB=p90rskntkd
pi3k=map3k1
mtorc1=p70s6kact
tnfa=jnk12loop
pi3kB=mtorc2
ikk=tsc12act
grb2=pi3k
p90rskntkd=s6s236
mtorc1B=p70s6kact
mtorc2=p53s15
tsc12act=mtorc2
map3k7B=jnk12loop
pi3kB=p53s15
30 Models in top Quartile (i.e. >= 0.5284)
30 Models in bottom Quartile (i.e. <= 0.22727)
irs1s under tnfa=1il1a=1 , mek1act=0.9
ikk=tsc12act
pi3k=map3k1
mtorc1=p53s15
grb2=map3k1
pi3k=p90rskntkd
p90rskntkd=tsc12act
30 Models in top Quartile (i.e. >= 0.60021)
30 Models in bottom Quartile (i.e. <= 6.2615e-41)
irs1s under tgfa=1il1a=1 , mek1act=0.9
ikk=tsc12act
pi3k=map3k1
mtorc1=p53s15
mtorc1B=p70s6kact
p70s6kai=p70s6kact
p90rskntkd=tsc12act
30 Models in top Quartile (i.e. >= 0.60284)
33 Models in bottom Quartile (i.e. <= 0.33885)
0
0.2
0.4
0.6
0.8
1

4.3 Conclusions

In this work we trained a family of constrained fuzzy logic models to a dataset describing signaling response. In order to train our models to data normalized to pre-inhibited basal level, we introduced a new simulation engine that modeled the effects of inhibitors to basal value of signals separately from the effect on ability of species to be activated upon stimulation. Additionally, we explicitly modeled the biochemistry of the sites actually measured by phospho-specific antibodies through incorporation of nodes describing different protein domains into the prior knowledge network. We subsequently investigated the ability of the models to make predictions that efficiently screened for what therapeutic or combination of therapeutics would result in a desired outcome and in what environmental contexts they would be effective. We assessed the predictions in terms of both accuracy and precision and discovered interesting facets of model topology.

The evaluation of a trained logic model both in terms of accuracy and precision of predictions is a concept introduced in [124] and explored for constrained fuzzy logic in Chapter 2 [102]. In this work, we further demonstrated the utility of this concept by determining the accuracy of several predictions (Figure 4-5a and b) and further examining differences in models that predicted different responses for the signals (Figure 4-5e and f). We found that examining the topological features underpinning different predictions yielded further insight into network topology that was not discernible from a more general clustering and correlation analysis (Table 4.4 and Supplementary Figures D-8 and D-9). Thus, we concluded that differences in model topology should be considered and examined throughout use of the models to make predictions because a general approach, while useful, did not present a complete picture of the implications of topological disparities.

As described in [102], a family of models was trained to the data because available data did not completely constrain both model topology and transfer function parameters. While considering this lack of identifiability was useful when making predictions, it would be preferable to obtain data that better constrained the model predictions. We addressed this issue in two ways: (1) investigating what subset of the data we originally obtained was useful in constraining the models and (2) developing methods to determine experimental conditions that would allow us to distinguish between models.

In our investigation of what portion of the data was useful in constraining the models, we found that inclusion of ligand dose data was important for training models that would make precise predictions (Figure 4-6a). It remains to be seen if this will be true for other data sets, but it stands to reason that inhibitor doses will generally add less information than ligand doses. In the case of partial inhibition of downstream nodes, other conditions could contain information regarding the effect of partial activity of that node due to differential activation under other experimental conditions. However, information regarding effects of partial activity of the most upstream nodes (the ligand stimuli) could not be contained in any condition other than one featuring partial stimulation of that species. Thus, the ligand dose condition would be necessary to determine this effect.

Our efforts to quantify the information content of the data subsets led us to propose a new metric for data information that wasn't reliant on a comparison with randomly chosen conditions. We developed a new metric to determine if removal of a condition resulted in 'lost' information and found that it allowed us to better determine how many conditions removed in each subset contained lost information. However, future work should aim to use this metric in a variety of data sets of various size and composition to ensure its general usefulness and applicability.

Toward the goal of developing a method to determine informative experimental conditions, we performed the design analysis both to distinguish between our trained models (Table D.2) and to suggest experiments *a priori* that distinguished between models derived from the prior knowledge network (Figure 4-11). We found that we were able to propose a set of experiments a fraction of the size of that tested (less than 10% of conditions tested were found to be informative). In our test of conditions predicted to distinguish between trained models, we found that day-to-day experimental variability prevented us from being able to use these experiments as planned. We had allowed the software to propose experiments that contained different doses of ligands. The use of these smaller doses exacerbated the effect of day-to-day variability, as small differences in seeding density could lead to large differences in the number of molecules of ligand stimulation per cell, which has been shown to greatly affect cellular response [21]. The variability was systematic in that the observed dose response for the ligands appeared shifted. Thus, we could not correct for the appearance of this experimental noise nor could we have accounted for it during our design calculations. If we had included dose response conditions of each ligand and inhibitor as part of the validation study, we could have corrected for the noise. Alternatively, we could limit this effect by only considering conditions that contain full stimulation and inhibition. An example design incorporating this limitation is shown in Supplementary Table D.3.

In our test of *a priori* design for conditions that distinguished between models derived from the prior knowledge network, we found that while we were able to suggest conditions that were predicted to be informative, a test of the ability of models trained to these additional conditions in making constrained predictions indicated that the designed conditions were not additionally informative for *in silico* data. For real data, some conditions appeared to be helpful in constraining the resultant models' predictions. However, on closer inspection, the success of the designed conditions seemed to be due to noise in other conditions and not the actual information content of the data. This noise (or, in some cases, differing responses in similar but not identical conditions) led to imprecise models because the training process could not rectify the data in terms of the prior knowledge network, which caused the models to choose one of several reasonable explanations for the data and led to ambivalent models. Because the success of the designed conditions was attributable to experimental noise rather than the information content of the data, we concluded that the experimental design procedure was not successful. In this procedure, we assumed that potential stimuli, inhibitors, and measured species were known. In order to obtain an effective design strategy in future work, it might be beneficial to also consider what species it would be most informative to additionally measure or perturb.

We concluded that, because we are currently unable to design experiments that will not result in a single model that fits the data rather than a family of models, we should consider ways to investigate model 'ambivalence' during entire modeling process: from training to topology analysis to making predictions. We demonstrated several strategies for such analysis of model ambivalence. Initially, we trained many models individually and used the resultant family of well-fit models for further analysis. By exploring what is kept and removed in many models, we determined aspects of our prior knowledge network that were inconsistent with or unnecessary to fit the data well. Additionally, we performed correlation analysis to determine interactions that are correlated, resulting in a deeper understanding of the ambivalence in the topology of the models. Importantly, we also considered model ambivalence when making predictions. We used each model to predict if a condition would meet our criteria, and then determined the fraction of models that predicted a conditions would be effective. If many models predicted that it would be effective while many others predicted it would not, the prediction was ambivalent. In such cases, we demonstrated that by considering differences between structural features in each group of models, we gained further insight into the regulatory network dictating cellular response.

Chapter 5

Conclusions and Perspectives

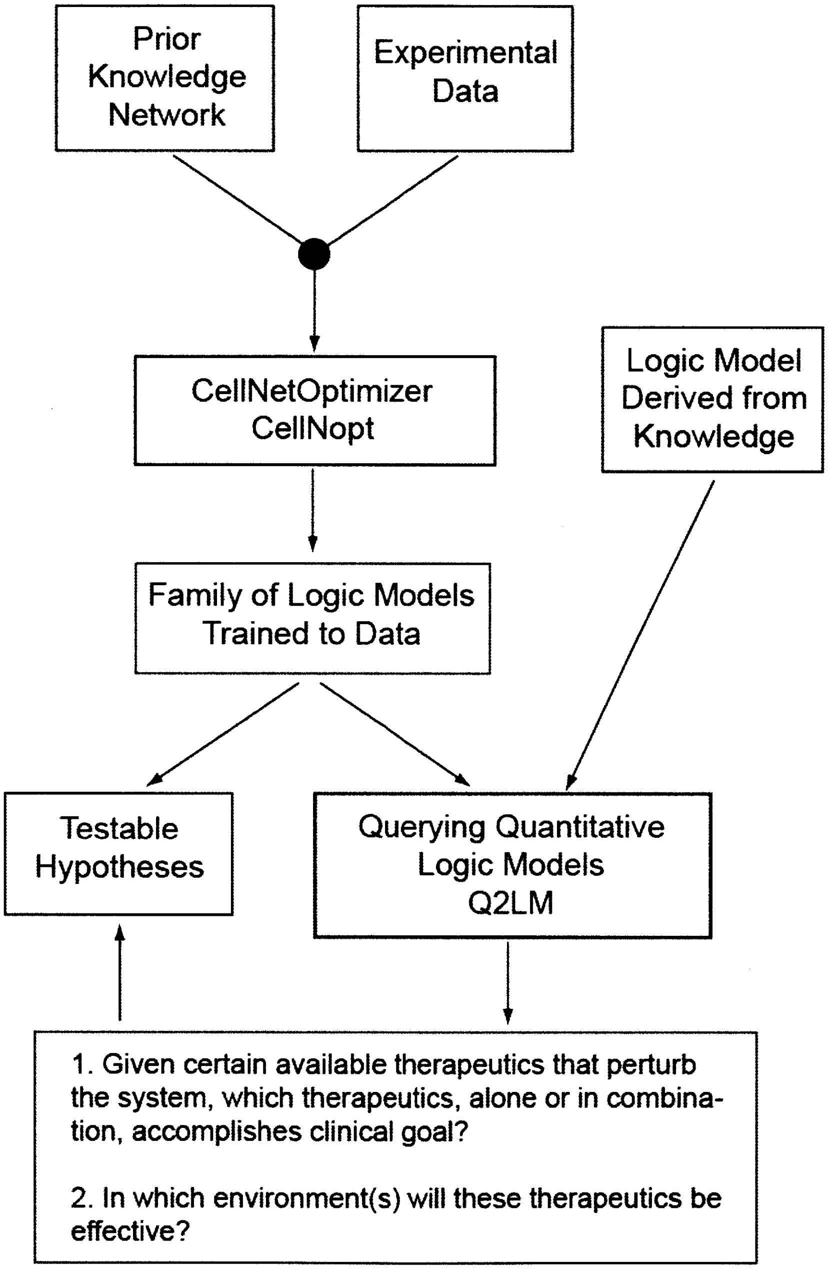
5.1 Summary of thesis contents

Signaling networks contain many protein species with many interactions between them, all of which could play a role in eliciting a cellular response. The complicated nature of these signaling networks leads to difficulty in predicting their response using intuition alone. Rather, mathematical models can aide in probing network properties and interpreting data describing signal activation under specified environmental conditions.

In this thesis, we developed methods for using continuous logic-based models to aide our understanding of biological systems and facilitate data interpretation (Figure 5-1). Chapter One describes various types of logic models as well as how they have been applied to biological networks in the past [103]. A logic-based network model encodes observations of how molecular species interact with logic rules comprised of AND, OR, and NOT gates. The simplest and most common form of logic is Boolean logic, where species are described as either active or inactive. However, in biological networks, proteins often access a range of intermediate activation states. Aldridge et al manually determined fuzzy logic relationships and tuned parameters to fit the model to the data. During the training process, the authors made interesting observations regarding interactions in the signaling network. However, the necessity to manually fit the both the topology (in the form of fuzzy logic rules) and parameters was quite cumbersome, pointing to the need to simplify the formalism in order to enable automation of the model training [3]. Thus, we have developed a logic framework called constrained fuzzy logic (cFL) that maintains the simple description of interactions with AND, OR, and NOT gates, but allows for intermediate species activities through the use of mathematical functions relating input and output values (transfer functions).

In Chapter Two, we used cFL to train a prior knowledge network (PKN) to data, which revealed what aspects of the dataset agreed or disagreed with prior knowledge [102]. The PKN represented how we expected the signals to interact and was constructed from the literature or curated databases. To train the PKN to data, we employed a strategy previously developed to train Boolean logic models by enumer-

Figure 5-1: Summary of Thesis Work


Prior Knowledge Network
Experimental Data
CellNetOptimizer CellNopt
Logic Model Derived from Knowledge
Family of Logic Models Trained to Data
Testable Hypotheses
Querying Quantitative Logic Models Q2LM
1. Given certain available therapeutics that perturb the system, which therapeutics, alone or in combination, accomplishes clinical goal?
2. In which environment(s) will these therapeutics be effective?

ating possible logic gates from the PKN and testing the ability of models containing a subset of those gates to fit the data [124, 125]. We extended this strategy to train cFL models by testing models containing a subset of logic gates as well as transfer functions chosen from a list of possible functions. We implemented our approach as a significant extension of the publicly available software CellNetOptimizer [101].

In our first application to data in Chapter Two, we trained cFL models to a dataset describing fifteen signaling proteins in the HepG2 liver cancer cell line after exposure to six extracellular ligand cues in the presence or absence of seven small molecule inhibitors. We demonstrated that the ability to capture intermediate information is essential for full understanding of the network structure. Furthermore, our trained models generated new biological understanding of network behavior as well as quantitative predictions of signaling protein activation. Specifically, we learned that the ligand cue TGFα induced partial activation of the JNK pathway and that IL6 induced partial activation of multiple unexpected downstream species via the MEK pathway.

In addition to training a cFL model, in Chapter Three, we developed a tool called Querying Quantitative Logic Models (Q2LM) to quickly and easily construct a cFL model from existing knowledge of a biological system [104]. No dedicated data was gathered in these cases. Rather, we described the logic of the system based on observations made from diverse studies of the system. The model was then queried to determine if a prediction made using intuition alone was consistent with the current understanding of the system. We also varied the parameters of the model to determine if a prediction was consistent with many possible models, or only models with certain parameter values. Furthermore, this tool was specifically designed to screen for the effectiveness of combinations of therapeutics across posited extracellular environments. These two features are of utmost importance because cellular response can vary greatly across different environments, and combinations of inhibitors are often needed to elicit the correct response since many paths can activate the same signal in these networks. We used this platform for the analysis of a multiscale model of GCSF pharmacokinetics and pharmacodynamics and discovered that the cFL model was able to predict behavior that was reflected of the system *in vivo*.

We also explored the ability of cFL to model multiple stages of a biological response - from environmental cues to protein activation to subsequent transcriptional response (Appendix E). We collected datasets of both early protein phosphorylation and later mRNA transcription of the HepG2 cell line response to various ligand stimulation conditions, and explored means to train a cFL model to the diverse information contained in these datasets. We concluded that cFL training of a prior knowledge network linking protein and transcriptional regulation was useful for systematically determining if condition specific hypotheses were consistent with a general picture, but further development is necessary to fully evaluate the ability of this methodology to provide additional insight to this type of analysis.

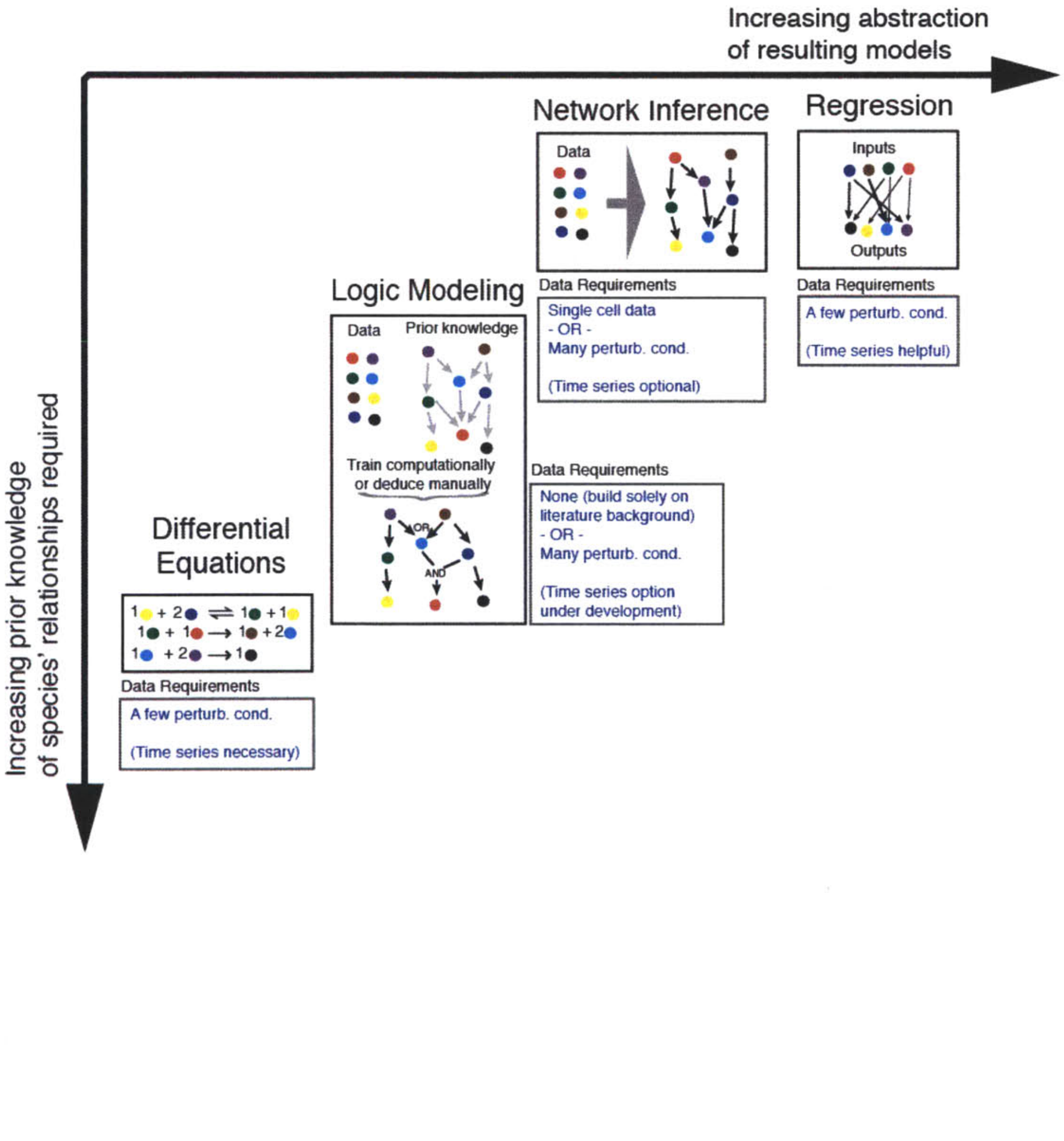
In our final application of trained cFL models, we investigated the ability of these models to predict therapeutics that would be effective at accomplishing a clinical goal in different cellular environments. We first trained a PKN to a new dataset describing activation of sixteen signaling proteins in the HepG2 cell line after exposure to several doses of four extracellular ligand cues and four small molecule inhibitors. As is often the case for training models to data, we found that several models could fit the data well because the data did not completely constrain both the transfer function parameters and topology of the models. Thus, we used all trained models to make predictions with Q2LM, which simulated models to determine what therapeutic perturbation or combination would be effective at accomplishing a clinical goal. In the course of this analysis, it became clear that some predictions were ambivalent in that some models predicted a perturbation would be effective whereas others did not. We investigated means of experimental design to reduce such ambivalence. However, we found that while we were able to suggest useful 'rules of thumb' for what conditions would allow models to produce constrain predictions, at least 10% of the predictions were ambivalent. Thus, we conclude that, since such ambivalence is currently unavoidable in modeling biological systems, it is crucial to explicitly address the ambivalence during model training and analysis. We demonstrate several means of exploring the general cause of model ambivalence as well as its effect on model prediction.

5.2 Relationship between cFL and other modeling formalisms

Models are typically used for two important purposes: (1) to understand biological relationships and (2) to predict outcomes in specified conditions. To accomplish these goals for a variety of applications involving different data and prior knowledge availability, several modeling approaches have proven useful (Figure 5-2). Correlation-based approaches such as principle components analysis and partial least squares regression require little prior knowledge and can be used to understand the correlative relationship between molecular species and phenotypes measured [58, 97, 5]. However, they do not typically incorporate causative network links that indicate which upstream species activate downstream species such that predictions of therapeutic effect do not take into account cellular context [57]. Network inference approaches (e.g. Bayesian networks and mutual information networks) propose networks of relationships between species with minimal or no prior knowledge but typically require large amounts of data to train and do not explicitly take into account cellular context [123, 160].

On the other end of the spectrum, differential equations (DEs) based on mechanistic representation of molecular interactions require detailed knowledge of biochemical interactions as well as large amounts of kinetic data to train. Thus, while additional insights regarding what insights interact are limited, these models are well suited to make detailed predictions regarding how a molecular alteration to an inhibitor or biologic will effect efficacy (e.g. [33]) as well as what aspects of the system are important for determining efficacy (e.g. [17, 72]). By describing networks with less mechanistic and time-resolved detail than DEs but incorporating more prior knowledge than a reverse-engineering approach, logic-based models typically require much less data to

Figure 5-2: Relationship between cFL and other modeling techniques. Each modeling approach answers different questions regarding biological data and has specific prior knowledge and data requirements.



train than either approach while still providing information regarding hypothesized network connections and allowing for context-specific prediction of species' states [124, 102, 103].

5.3 Limitations of cFL modeling

While complementary to other model approaches, cFL modeling has several limitations that should be overcome to further extend its use. As currently formulated, cFL models are trained only to 'static' data in that only one time point is considered. Thus, the resultant models represent the early activation of signals while assuming that later mechanisms of signal attenuation are on a timescale slower than that reflected by the data. It would be preferable to model these negative feedback mechanisms explicitly, but negative feedback frequently leads to oscillations in discrete simulation procedures. To handle these cases in the case of logic models constructed based on prior knowledge (Chapter Three), we offer the option of solving the system of nonlinear equations specifying the network for the root 'nearest' the last simulation step. While this procedure could be employed in the model training case, it would greatly decrease computational efficiency. Rather, an attractive alternative that allows for the annotation of negative feed back as either strong or weak has been developed for the Boolean logic case [86], and application to the cFL case would be highly informative for application of the method to study biological networks involving both intracellular and intercellular regulation in the future.

An additional limitation of the cFL model training approach presented in Chapter Two is that it cannot add interactions to those deemed feasible by the prior knowledge network (PKN). If the PKN does not contain an interaction between species that is necessary to fit the data, the trained models simply return with systematic error for that data. The scientist must then recognize the error and propose interactions by manual inspection of the data and literature. These interactions are added to the PKN, which is subsequently retrained. Methods to automate this process would be less subjective and potentially less time consuming. One option is to develop a means to detect systematic error caused by links missing from the PKN during model training and initiate an additional training process to propose connections from a predetermined list or based on correlation between measured species Alternatively, the Saez-Rodriguez lab at EMBL-EBI recently proposed the addition of links to the prior knowledge network based on correlation analysis prior to model training [29].

Finally, cFL models are deterministic in that, for a given set of stimuli and perturbations, simulation of a given model will always return identical species' values. Thus, cFL as currently implemented cannot be used to probabilistically model single cell data. One alternative for training a PKN to single cell data is the use of a different mathematical formalism to represent the networks: specifically, a Bayesian network. While applications of Bayesian network training have typically focussed on network inference, it is possible to imagine either altering network inference procedures to incorporate a strong prior that recapitulates the processed prior knowledge network or developing a new search procedure to evaluate Bayesian networks limited to those contained in the prior knowledge network. Work toward the latter is ongoing in the Sorger laboratory at Harvard Medical School.

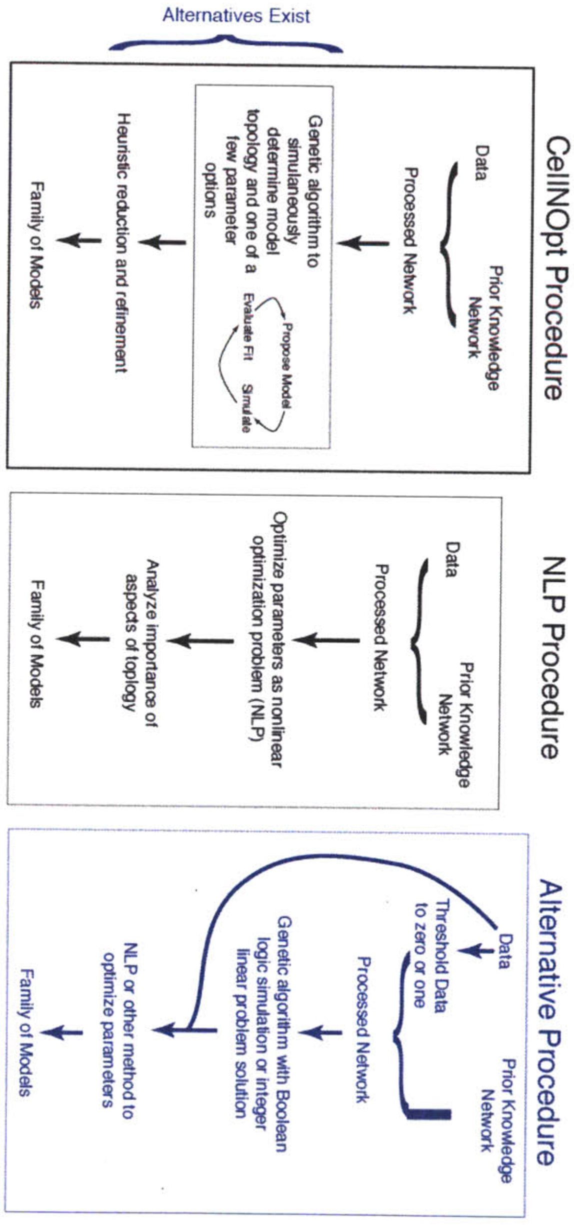
5.4 Overcoming computational limitations of cFL training procedure

CFL model training is currently computationally limited because of the platform used for implementation as well as the optimization procedure employed. CellNOpt was initially implemented in MATLAB. While every effort was made to increase computational efficiency, the training of a cFL model of the size presented in Chapters Two and Four can take between three and twenty-four hours. If the PKN contains many possible interactions such that 1000 models must be trained to ensure a sufficient number of well-fit models, a vast amount of computational time is necessary to obtain the solution pool. Furthermore, if the models are trained in parallel, a sufficient number of MATLAB licenses must be obtained, representing an additional limitation to model training.

To overcome the requirement of the proprietary MATLAB platform, CellNOpt has been implemented as an R packages for both Boolean logic and cFL model training [144]. Unfortunately, the implementation in R has not yet been optimized for speed. Although it is unclear if the R version will overcome the computational time requirement issues, it will be a viable alternative that alleviates the MATLAB requirement. If the R version cannot be made more computationally efficient, another programming platform could be sought. Whichever platform is used, great care should be taken to ensure that not only does the interface for training models to data remain user friendly, but also the post-training analysis. One of the great advantages to CellNOpt in the current MATLAB implementation is the availability of functions for analysis of trained models' structures and fit to the data as well as the interface to Q2LM for extended prediction capability. It is crucial to maintain ability to efficiently learn not only how to train models but also how to analyze them once they have been trained.

Toward improvement of the training optimization procedure (Figure 5-3), the cFL network has been recast as a regular nonlinear optimization problem (NLP) by describing the network with equations and trained to data [99]. This procedure first optimizes the network parameters and then examines what parts of the topology are necessary, so the model topology and parameters are not considered simultaneously as in CellNOpt. Additionally, in order to recast the network as an NLP amenable to training with available solvers, the mathematical form of the relationship between species must be well established. In the CellNOpt procedure, the genetic algorithm optimizes based on a comparison of simulated and experimental values. Thus, it is relatively straight forward to incorporate an alternative simulation procedure into the CellNOpt but not the NLP procedure. This flexibility is sometimes necessary to accurately recapitulate data (as exemplified in Chapter Four). If the current CellNOpt workflow is preferred, improvements in specific aspects should be considered. For example, a simulated annealing or other global optimizer could be substituted for the

Figure 5-3: Alternative logic model training procedures. The CellNOpt procedure was developed and used in this thesis work. Recently, an NLP procedure has been proposed [99]. A proposed additional alternative is depicted in purple.



Alternatives Exist
CellNOpt Procedure
Data
Prior Knowledge Network
Processed Network
Genetic algorithm to simulaneously determine model topology and one of a few parameter options
Propose Model
Simulate
Evaluate Fit
Heuristic reduction and refinement
Family of Models
NLP Procedure
Data
Prior Knowledge Network
Processed Network
Optimize parameters as nonlinear optimization problem (NLP)
Analyze importance of aspects of topology
Family of Models
Alternative Procedure
Data
Prior Knowledge Network
Threshold Data to zero or one
Processed Network
Genetic algorithm with Boolean logic simulation or integer linear problem solution
NLP or other method to optimize parameters
Family of Models

genetic algorithm. Alternatively, the entire genetic algorithm followed by heuristic refinement procedure could be accomplished using different optimizers and heuristics. Development toward establishing a more efficient and rigorous workflow for logic model training is an area of active study in the laboratory of Julio Saez-Rodriguez at EMBL-EBI as well as several collaborators.

One additional workflow that could be investigated is training the model topology with Boolean logic methods and subsequently training parameters with cFL methods (Figure 5-3). In early stages of this work, this procedure was attempted and found lacking because the optimal cFL parameter values were simply those that recapitulated a Boolean logic gate. However, if the data was thresholded properly to zero or one for Boolean logic topology training and then the un-thresholded continuous values used to train cFL parameters, this procedure could still be a viable alternative.

5.5 Concluding remarks

While further work remains toward development of the most rigorous and efficient optimization protocol, in this thesis we have developed the novel modeling approach, constrained fuzzy logic. We have demonstrated that cFL models can be trained to data or constructed directly from prior knowledge and used these models to increase understanding about signaling network responses and make predictions under a variety of environmental conditions. Ultimately, we hope that development of this flexible methodology will assist scientists in making self-consistent, informed decisions for further experiments or choice of therapeutic targets.

Appendix A

Supporting information for Chapter 1

A.1 Supplementary Figures

Figure A-1 *(facing page)*: (a.) Example of equation description of logic. Practically, discrete models are often described by equations. There are several ways to represent logic as equations. We have shown one possible representation. For the simplest binary case, aij is 1 if x_j is and activator or x_i, -1 if xj is an inhibitor of x_i and 0 if x_j does not influence x_i. In order to describe multiple activation states of the species, θ_i and $a_{i,j}$ can take on virtually any user-supplied value. (b.) Truth table description of logic. The truth tables indicate the values of the output species given specific values of each input values. Only binary (Boolean) truth tables for interactions with two inputs are described. The Boolean function notation is shown above each gate. All gates shown are monotonic with the exception of XOR, which has nonmonotonic behavior: increasing input levels may lead to an increase in output level, but upon further input level increase, a descrease in output level is observed. The disjunctive normal form (sum-of-products) description of this gate is shown above this truth table. The alternate conjunctive normal form (product of sums) is (c.) Continuous or mixed discrete-continous description of logic. Species states are determined using differential equations. The activation of any given species is determined by a function (G_i). The specific function used depends on the specific formalism (logic based ODEs, piecewise-linear, or standardized qualitative dynamical systems, etc.). Each formalism models decay as a first-order process. (d.) Fuzzy logic description of logic. Given one or several input species levels, possible output levels are calculated from user-defined functions. The final output value is calculated based from the possible output levels using other user-defined functions.

a. Example equation formalism

$$x_i(t+1) = \begin{cases} 1 & \sum_{j} a_{ij}x_j(t) - \theta_i > 0 \\ 0 & \sum_{j} a_{ij}x_j(t) - \theta_i < 0 \\ x_i(t) \text{ or } 0 & \sum_{j} a_{ij}x_j(t) - \theta_i = 0 \end{cases}$$

where
x = node state
t = refers to time
a_{ij} = how node x_j relates to node x_i
θ_i = threshold for activation for x_i

b. Example 2-Input Truth Tables

A ∧ B

A AND B → C		
Inputs		Output
A	**B**	**C**
0	0	0
0	1	0
1	0	0
1	1	1

A ∨ B

A OR B → C		
Inputs		Output
A	**B**	**C**
0	0	0
0	1	1
1	0	1
1	1	1

A ∧ ¬B

A AND not B → C		
Inputs		Output
A	**B**	**C**
0	0	0
0	1	0
1	0	1
1	1	0

(A ∧ ¬B) ∨ (¬A ∧ B)

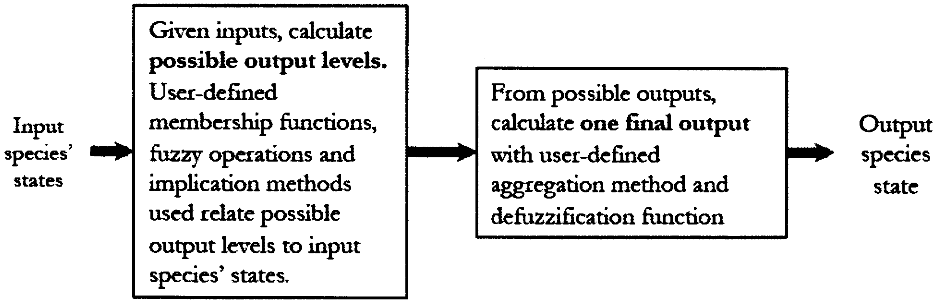
A XOR B → C		
Inputs		Output
A	**B**	**C**
0	0	0
0	1	1
1	0	1
1	1	0

c. Continuous or mixed discrete-continous formalisms

$$\frac{dx_i}{dt} = G_i(\vec{x},t) - \gamma_i x_i$$

where
x = node state
t = refers to time
γ_{ij} = decay rate of node x_i
G_i = Function describing activation of node x_i. It is dependent on values of other nodes (denoted by the x vector) and, optionally, time

d. Fuzzy logic formalism



Input species' states

Given inputs, calculate **possible output levels.** User-defined membership functions, fuzzy operations and implication methods used relate possible output levels to input species' states.

From possible outputs, calculate **one final output** with user-defined aggregation method and defuzzification function

Output species state

Appendix B

Supporting information for Chapter 2

B.1 Supplementary Methods

B.1.1 HepG2 training and follow-up data phospho-ERK measurement inconsistencies

In the training dataset, ERK was phosphorylated solely under one stimulatory condition (TGFα) and did not increase with increasing MEK phosphorylation. When compared the raw data between this and the follow-up dataset discussed below, ERK did indeed respond in many other conditions in the initial training dataset, but signal intensity was much lower in this experiment, causing many ERK phosphorylation measurements to be below the noise threshold and recorded as zero during the normalization process. Because of these discrepancies with the phospho-ERK measurements, we did not include it in our training dataset.

subsectionConstrained fuzzy logic

Prior to implementing constrained fuzzy logic for network training, we investigated the use of Mamdani [88] and Sugeno [140] fuzzy logic gates with varying number and functional forms of membership functions. The cFL framework we use in this work represents each biological interaction with Sugeno gates with normalized Hill input membership functions and constant output membership functions of zero and one. Each AND gate is a fuzzy logic rule with an AND operator of "min." In this formalism, OR gates are evaluated by the "max" defuzzification method that operates on the outputs of fuzzy logic rules.

The use of normalized Hill functions assumes that species reach the same level of saturation under activation by any of its possible inputs. Biologically, this assumption does not always hold. However, during our initial methods development, we deemed this assumption acceptable as the use of the normalized Hill function did not cause any noticeable issues during the model training and allowed each parameter to have a distinct meaning with the sensitivity parameter, k, specifying EC_{50} and the Hill coefficient, n, specifying sharpness of transition.

B.1.2 Simulation

A set of functions was implemented in MATLAB (Mathworks, Inc.) and integrated into CellNetOptimizer to convert BL models to cFL models and determine the logic steady state of node states of a given cFL network under given experimental conditions. To calculate the logic steady state, nodes of the network are updated until they reach a stable state. If the network contains negative feedback, a logic steady state cannot be computed, similar to the Boolean case [71]. Penalization of not-computed stated states leads then to the absence of negative feedback in resulting models [70]. To increase the efficiency of the training process, in the HepG2 prior knowledge network we determined the negative interactions that would result in negative feedback in CellNOpt using the MATLAB (Mathworks, Inc.) software CellNetAnalyzer [126], and we removed them prior to optimization.

B.1.3 Model Refinement

Model parameters were refined using the MATLAB active-set algorithm, a Sequential Quadratic Programming method for nonlinear constrained optimization. http://www.mathworks.com/access/helpdesk/help/toolbox/optim/ug/brnoxzl.html

B.1.4 PLSR model of cytokine release data

Luminex data describing release of 50 cytokines at time zero and three hours after stimulation was examined. Twenty cytokines were chosen to model based on the consistency and reliability of the data (e.g. if the data was grossly inconsistent under similar experimental treatment conditions, it was not considered). Data for these cytokines were normalized similarly to the phospho-signal dataset [124] except no data was considered below the lower level of detection because it had already been filtered for consistency.

A preliminary three-component PLSR model was constructed using DataRail [126] by regressing the normalized cytokine release data against the signaling data. Five cytokines (IL1β, IL4, G-CSF, IFNγ, and SDF1α) were chosen for further study based on the criteria that the R2 values of the PLSR model for those cytokines be greater than 0.70. Further analysis suggested that cytokine measurements with lower R2 values were not robust (i.e. varied in measured value even under similar stimulation and inhibition conditions).

A new PLSR model was then generated with DataRail by regressing the normalized cytokine release data against the signaling data. Three components were chosen to be optimal by seven-fold cross-validation.

B.1.5 Linking prior knowledge network to cytokine release nodes

The clustering of the protein signals in principle components space of both a principle component model of the signals as well as the principle components of the PLSR model was considered when choosing signaling nodes to link to the cytokine release nodes. If protein signals clustered together consistently, the signal most downstream in the prior knowledge networks was chosen. Based on this analysis, the following protein signaling nodes were linked to each cytokine release node: MEK1/2, CREB, GSK3, c-Jun, Hsp27, IκB, and STAT3.

B.2 Supplementary Figures

Figure B-1 *(facing page)*: Experimental dataset describing HepG2 signaling response. Each small rectangle represents phosphorylation of the protein indicated on the left at zero and thirty minutes as measured by Luminex bead-based Elisa. HepG2 cells were exposed to the inhibitor indicated below the column and stimulated with the ligand indicated above as described in [5]. Raw intensity (a) and normalized (b) values are shown. Data was normalized as previously described [124] using DataRail software [126]. Briefly, data values below the background or above the saturation signal of the Luminex instrument were not included in the training set (grey fill). The absolute difference between the signal at the time of stimulation and 30 minutes thereafter was divided by the signal at time zero and transformed using a nonlinear Hill transformation. The resulting value was multiplied by a Hill-transformed ratio of the value of the signal at 30 minutes to its maximum value across all conditions. The resulting value was the normalized value. Plots were generated by the open-source MATLAB toolbox DataRail [126].

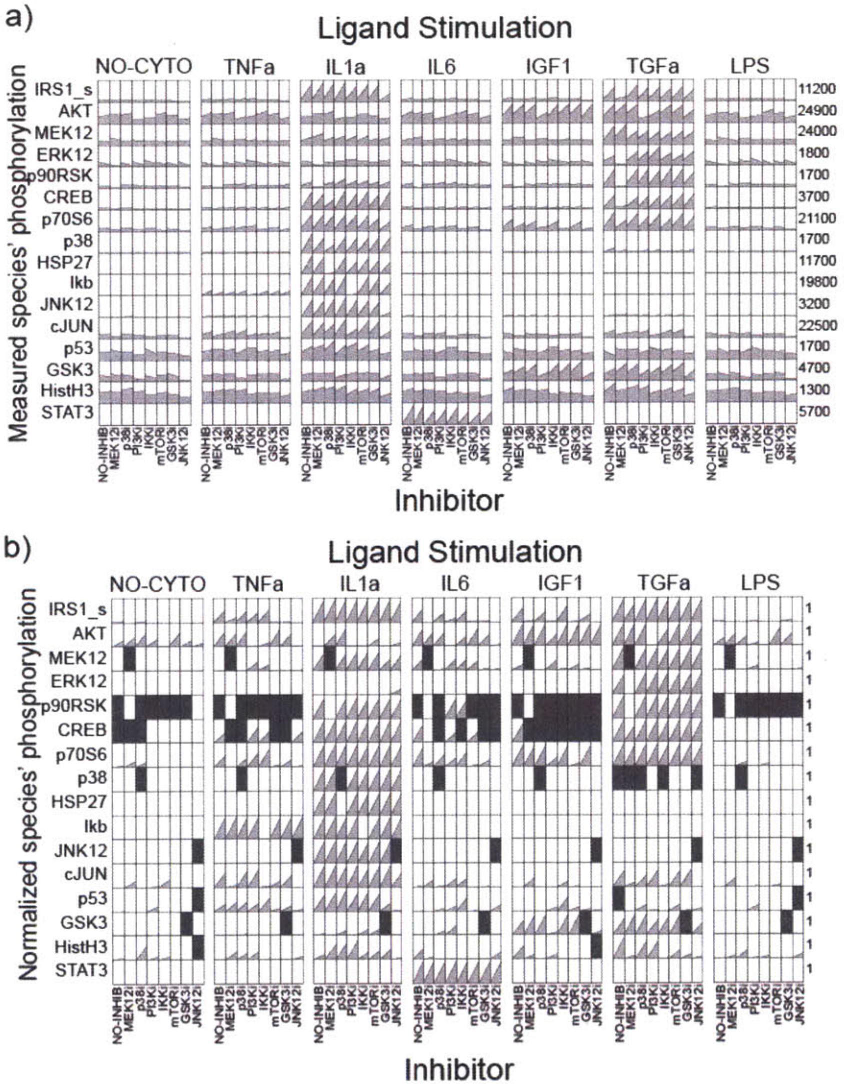

a)
Ligand Stimulation
NO-CYTO
TNFa
IL1a
IL6
IGF1
TGFa
LPS
Measured species' phosphorylation
IRS1_s
AKT
MEK12
ERK12
p90RSK
CREB
p70S6
p38
HSP27
Ikb
JNK12
cJUN
p53
GSK3
HistH3
STAT3
11200
24900
24000
1800
1700
3700
21100
1700
11700
19800
3200
22500
1700
4700
1300
5700
NO-INHIB
MEK12i
p38i
PI3Ki
IKKi
mTORi
GSK3i
JNK12i
Inhibitor
b)
Ligand Stimulation
NO-CYTO
TNFa
IL1a
IL6
IGF1
TGFa
LPS
Normalized species' phosphorylation
IRS1_s
AKT
MEK12
ERK12
p90RSK
CREB
p70S6
p38
HSP27
Ikb
JNK12
cJUN
p53
GSK3
HistH3
STAT3
1
NO-INHIB
MEK12i
p38i
PI3Ki
IKKi
mTORi
GSK3i
JNK12i
Inhibitor

Figure B-2 *(facing page)*: Prior knowledge networks (PKNs). PKN0 derived from Ingenuity and used in the BL methodology validation (a, map without purple dashed arrows) was first processed to include two-input AND gates (b) and then used with the CellNOpt-cFL methodology to determine the cFL networks representing this dataset. Results of this analysis led to extension of the PKN to PKN1 (a, purple dashed arrows) which was processed to include either two-input AND gates ($\mathrm{PKN1}^a$, c) or only include AND gates when an inhibitory interaction was being modeled ($\mathrm{PKN1}^i$, d). These processed PKNs were then compared to the HepG2 dataset with CellNOpt-cFL. All maps were generated with a CellNOpt routine using the graphviz visualization engine (www.graphviz.org) followed by manual annotation in Adobe Illustrator.

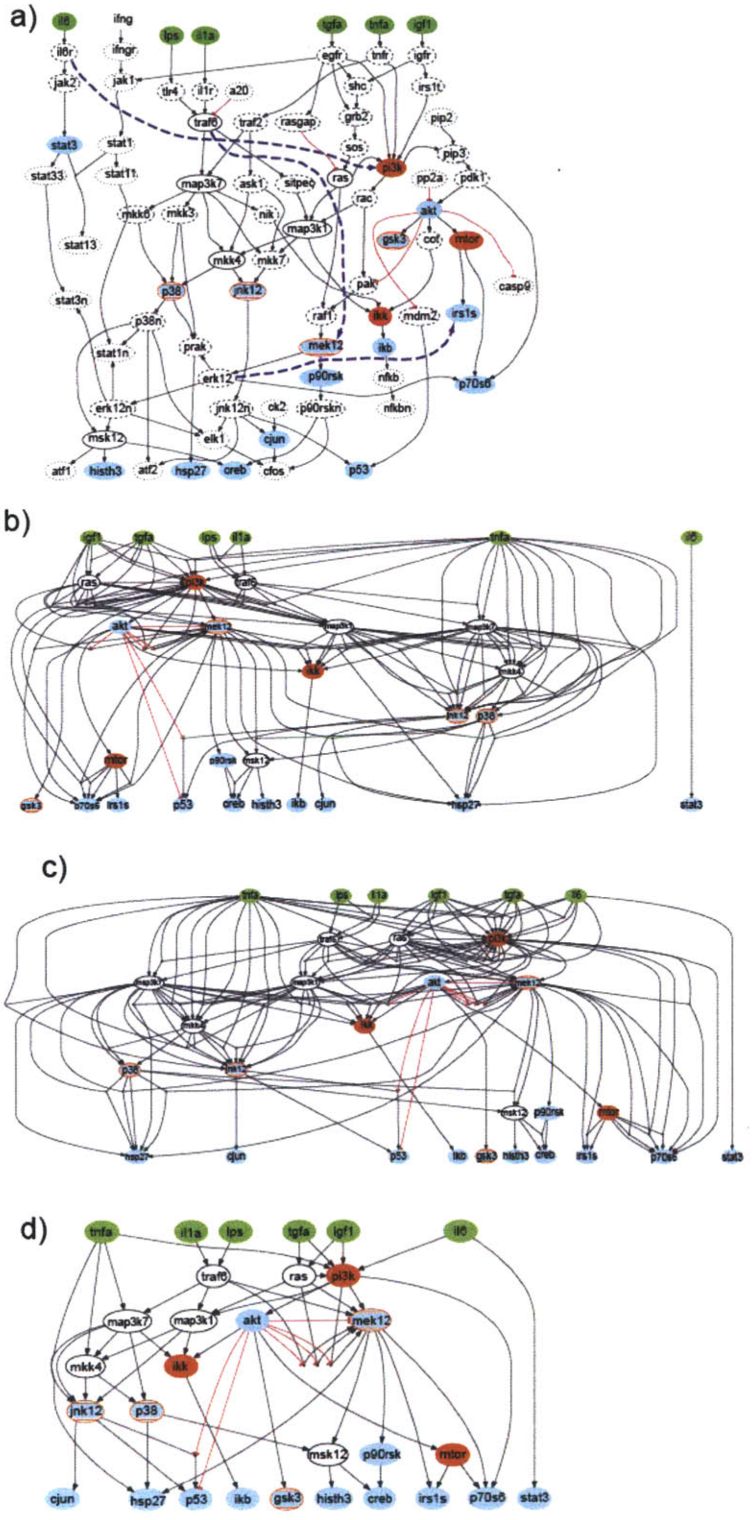

a)
il6
ifng
lps
il1a
tgfa
tnfa
igf1
il6r
ifngr
egfr
tnfr
igfr
jak2
jak1
tlr4
il1r
a20
shc
irs1t
traf6
traf2
rasgap
grb2
pip2
stat3
stat1
sos
pip3
stat33
stat11
map3k7
ask1
sitpec
ras
pi3k
pp2a
pdk1
mkk6
mkk3
nik
rac
akt
map3k1
stat13
mkk4
mkk7
gsk3
cot
mtor
pak
p38
jnk12
casp9
stat3n
raf1
ikk
mdm2
irs1s
p38n
stat1n
prak
mek12
ikb
erk12
p90rsk
nfkb
p70s6
erk12n
jnk12n
ck2
p90rskn
nfkbn
msk12
elk1
cjun
atf1
histh3
atf2
hsp27
creb
cfos
p53
b)
igf1
tgfa
lps
il1a
tnfa
il6
ras
pi3k
traf6
akt
mek12
map3k1
map3k7
ikk
mkk4
jnk12
p38
mtor
p90rsk
msk12
gsk3
p70s6
irs1s
p53
creb
histh3
ikb
cjun
hsp27
stat3
c)
tnfa
lps
il1a
igf1
tgfa
il6
traf6
ras
pi3k
map3k7
map3k1
akt
mek12
mkk4
ikk
p38
jnk12
msk12
p90rsk
mtor
hsp27
cjun
p53
ikb
gsk3
histh3
creb
irs1s
p70s6
stat3
d)
tnfa
il1a
lps
tgfa
igf1
il6
traf6
ras
pi3k
map3k7
map3k1
akt
mek12
mkk4
ikk
jnk12
p38
msk12
p90rsk
mtor
cjun
hsp27
p53
ikb
gsk3
histh3
creb
irs1s
p70s6
stat3

Figure B-3: Fit of unprocessed cFL networks trained with PKN0. PKN0 (Supplementary Figure B-2a) was processed to include all two-input AND gates (Supplementary Figure B-2b) and CellNOpt-cFL used to train 90 network models to the HepG2 dataset. The data is displayed as described in Supplementary Figure B-1, with the exception that the average simulation result is shown with a dashed blue line and the absolute difference in measured and average simulated signal level is indicated with a background color ranging from green (good fit) to red (bad fit). Note that, under the IL1α and IL6 stimulation conditions, many signals are not fit well (as indicated by the red and white coloring). Plots were generated by CellNOpt.

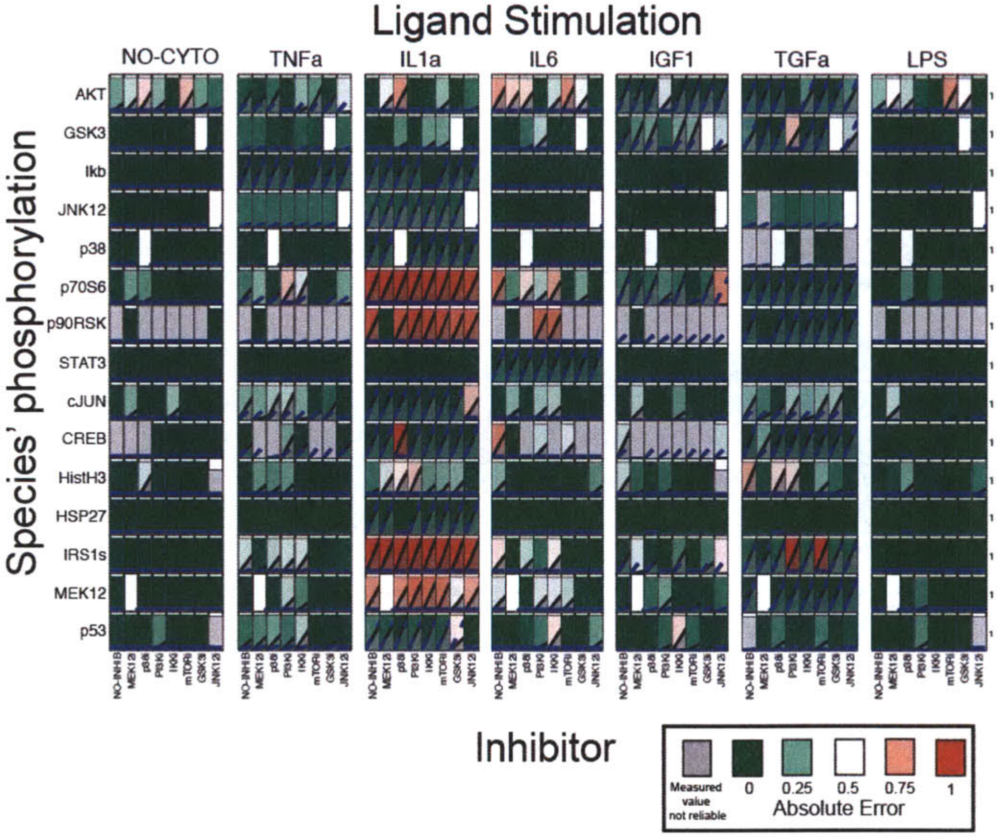


Figure B-4: Comparison of MSE and number of parameters of PKN1^a and PKN1^i. The Cumulative Distribution functions of the MSE and number of final parameters of unprocessed (a,b) and filtered (c,d) models with or without expansion into all plausible two-input AND gates are shown. For both the unprocessed and filtered models, the error of the models expanded with all plausible two-input AND gates is significantly less than those not fully expanded ($p = 4.3x10^{-32}$ from a Kolmogorov-Smirnov two-sided test of the filtered models). However, both unprocessed and filtered models expanded with all plausible two-input gates also contained more parameters than those not fully expanded ($p = 3.2x10^{-14}$ from a Kolmogorov-Smirnov two-sided test of the filtered models). The skewing of the filtered models (d) is due to the heuristic reduction procedure, which sometimes does not remove any parameters from the models.

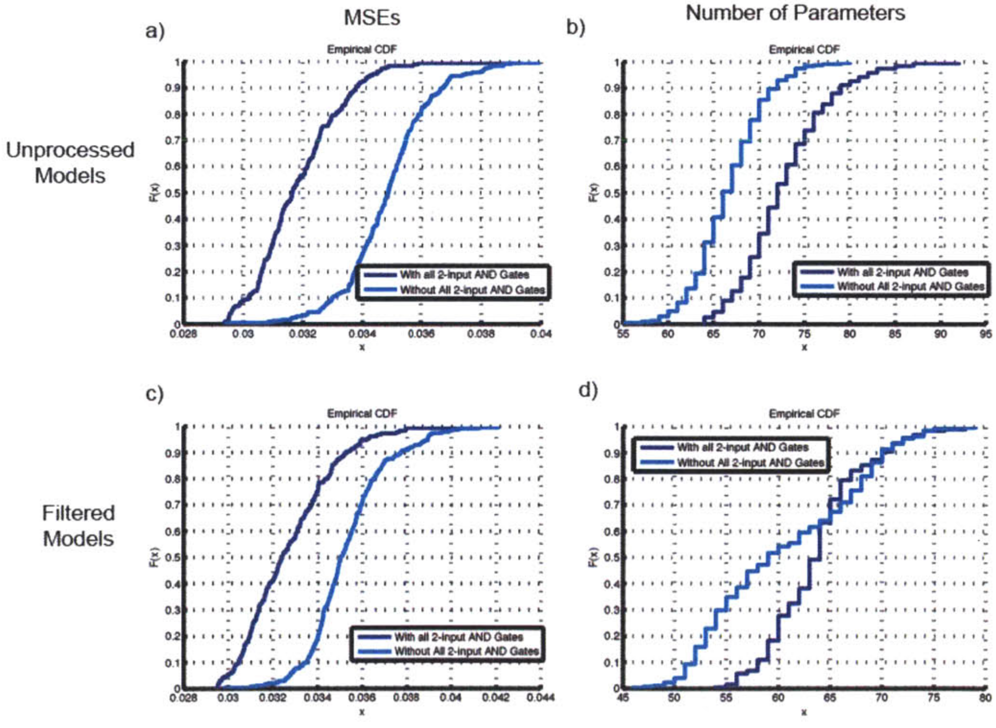


Figure B-5 *(facing page)*: Filtered cFL network models derived from training the HepG2 dataset to PKN1 processed to include two-input AND gates (PKN1^a). The PKN1 (Supplementary Figure B-2a) was processed to include all two-input AND gates (Supplementary Figure B-2c) and CellNOpt-cFL used to train 191 network models to the HepG2 dataset. Reduction of family of cFL models indicates that cFL AND gates can be removed without greatly affecting the resulting refined model score (a and b; b is a portion of the graph shown in a). The structures of the family of cFL network models trained to the HepG2 dataset are shown (c). Links colored black were present in all models whereas links colored grey were present in a fraction of the models (a darker grey indicates that the cFL gate was present in more models). Filtered cFL network models are shown. Fit to experimental data (d) is displayed as described in Supplementary Figures B-1 and B-3. Plots were generated by CellNOpt. Note that, when this PKN1^a is used to train the networks, most trained models include the Ras -to- Map3k1 cFL gate. The inclusion of this link is in contrast to the models obtained when two-input AND gates are only included for inhibitory interactions (Figure 2-5, Supplementary Figure B-6), where only a few models include this link. This difference is also reflected in the fact that cFL network models processed to include all two-input AND gates are better able to fit data describing c-Jun activation under TGFα stimulation (d compared to Supplementary Figures B-7 and B-8). Graphs of cFL network models were generated a CellNOpt routine using the graphviz visualization engine (www.graphviz.org) followed by manual annotation in Adobe Illustrator.

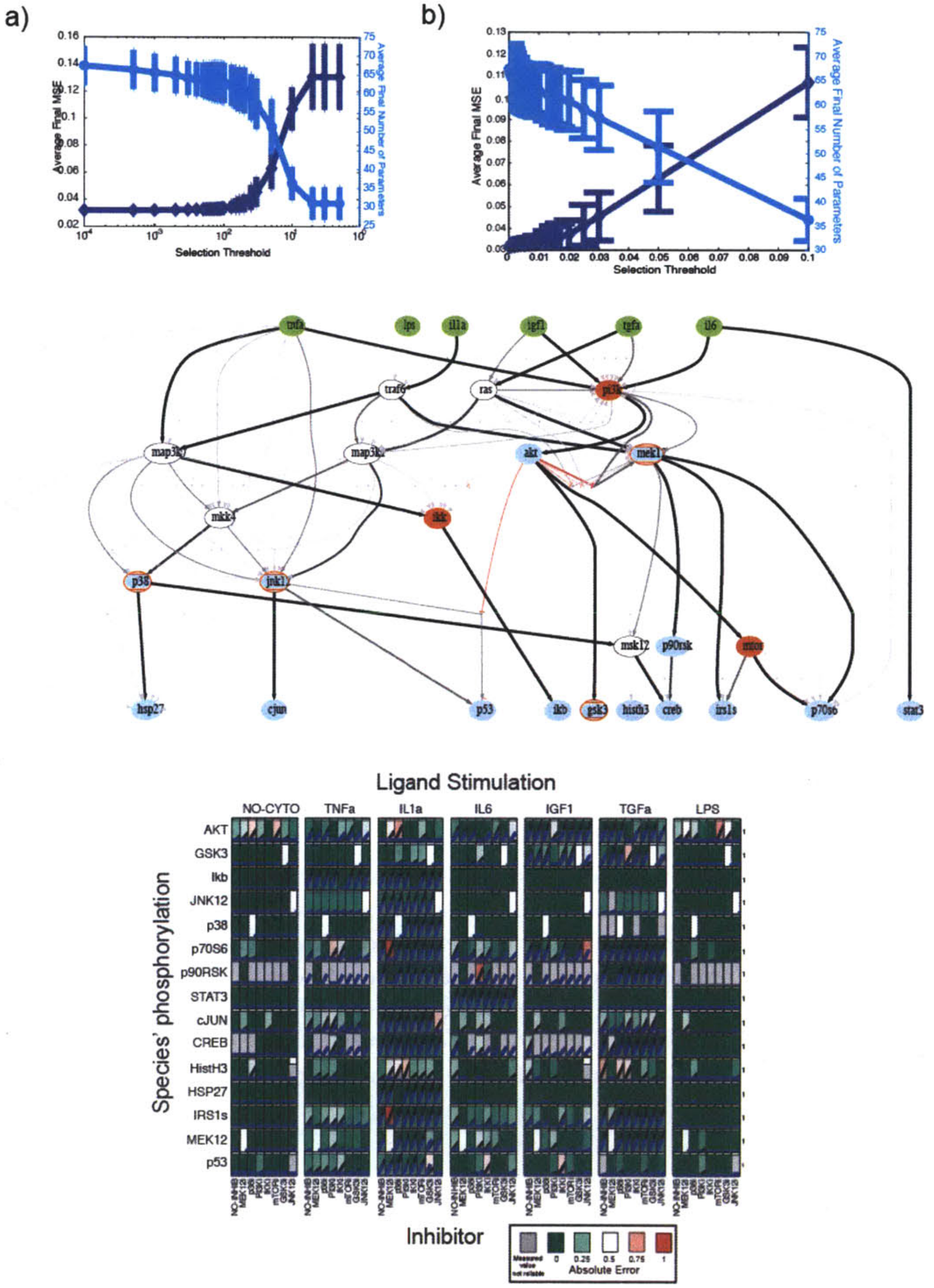

a)
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
Average Final MSE
10^{4}
10^{3}
10^{2}
10^{1}
10^{0}
Selection Threshold
75
70
65
60
55
50
45
40
35
30
25
Average Final Number of Parameters
b)
0.13
0.12
0.11
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
Average Final MSE
0
0.01
0.02
0.03
0.04
0.05
0.06
0.07
0.08
0.09
0.1
Selection Threshold
75
70
65
60
55
50
45
40
35
30
Average Final Number of Parameters
c)
tnfa
lps
il1a
igf1
tgfa
il6
traf6
ras
pi3k
map3k7
map3k1
akt
mek12
mkk4
ikk
p38
jnk12
msk12
p90rsk
mtor
hsp27
cjun
p53
ikb
gsk3
histh3
creb
irs1s
p70s6
stat3
d)
Ligand Stimulation
NO-CYTO
TNFa
IL1a
IL6
IGF1
TGFa
LPS
Species' phosphorylation
AKT
GSK3
Ikb
JNK12
p38
p70S6
p90RSK
STAT3
cJUN
CREB
HistH3
HSP27
IRS1s
MEK12
p53
NO-INHIB
MEK12
p38
PI3K
IKK
mTOR
GSK3
JNK12
Inhibitor
Measured value not reliable
0
0.25
0.5
0.75
1
Absolute Error

Figure B-6 *(facing page)*: Unprocessed cFL network models derived from training the HepG2 dataset to PKN1^{i}. a) Structures of the family of unprocessed cFL network models obtained by training the PKN1^{i} (Supplementary Figure B-2d) to the HepG2 dataset. Links colored black were present in all models whereas links colored grey were present in a fraction of the models (a darker grey indicates that the cFL gate was present in more models). Theses models were compared to the randomization controls, both for the determination of a p-value of the models (Supplementary Table B.1) as well as the investigation of the influence of the PKN on the model training process (b,c). The graph of the cFL network models was generated with a CellNOpt routine using the graphviz visualization engine (www.graphviz.org) followed by manual annotation in Adobe Illustrator. (b) We compared unprocessed models derived from a PKN with edges randomly added to those derived from the original PKN1^{i}. After structure processing (Figure 2-2 Steps 1-2), a model derived from a PKN with random edges added might have a different number of species as well as interactions than those derived from the original PKN. Thus, to compare these models, we further compressed the networks to include only interactions between the treated, measured, and inhibited species. This treatment allowed us to directly compare models with different intermediate species. When compared to the original PKN1^{i}, several edges were added which increased as a function of edges added to the pre-processed PKN, as expected (solid line). For the trained models, we compared edges present frequently in the family of models trained to the original PKN1^{i} (i.e. those present in $\geq 25\%$ of the models in a.) to those trained to each randomly extended PKN (dashed line). The fraction of different edges in the structures of the trained randomly extended models to those trained to the original PKN1^{i} increased slightly with increasing number of edges added randomly. (c) Comparing between the randomly extended PKNs and models derived from them, connections between treated, measured, and inhibited species that were in the randomly extended PKN but not the original PKN were often but not always removed during the training process. This is to be expected, as not all of the randomly added edges would not be inconsistent with the data, and some might allow the models to fit the data better than the original PKN.

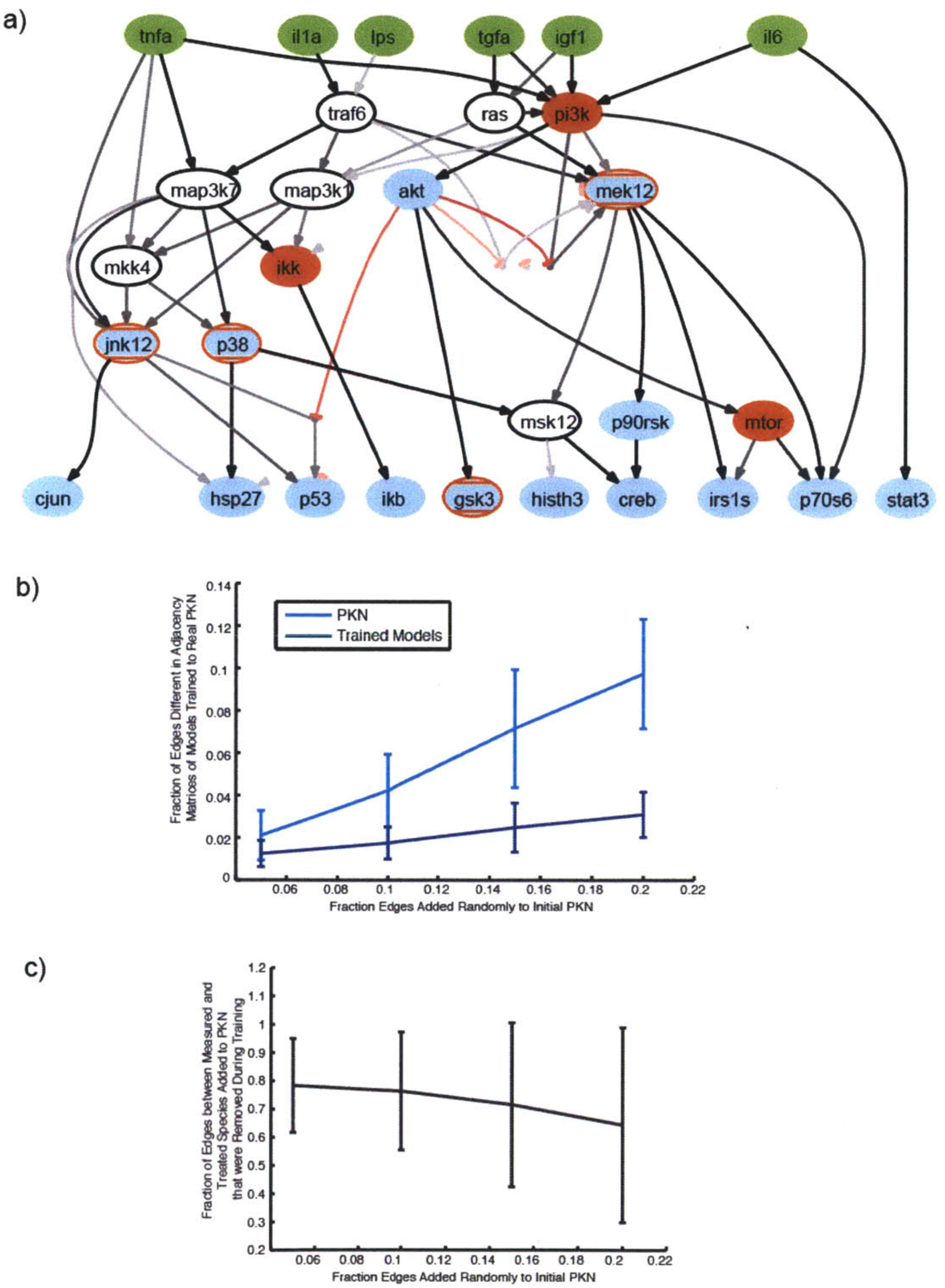

a)
tnfa
il1a
lps
tgfa
igf1
il6
traf6
ras
pi3k
map3k7
map3k1
akt
mek12
mkk4
ikk
jnk12
p38
msk12
p90rsk
mtor
cjun
hsp27
p53
ikb
gsk3
histh3
creb
irs1s
p70s6
stat3
b)
PKN
Trained Models
Fraction of Edges Different in Adjacency Matrices of Models Trained to Real PKN
0.14
0.12
0.1
0.08
0.06
0.04
0.02
0
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
0.22
Fraction Edges Added Randomly to Initial PKN
c)
Fraction of Edges between Measured and Treated Species Added to PKN that were Removed During Training
1.2
1.1
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.06
0.08
0.1
0.12
0.14
0.16
0.18
0.2
0.22
Fraction Edges Added Randomly to Initial PKN

Figure B-7: Fit of cFL networks trained using the extended PKN1^{i}. The extended prior knowledge network (Supplementary Figure B-2a) was processed to include all two-input AND gates only when an inhibitory interaction was modeled (Supplementary Figure B-2d) and CellNOpt-cFL used to train 243 network models to the HepG2 dataset. The data is displayed as described in Supplementary Figures B-1 and B-3. Plots were generated by CellNOpt.

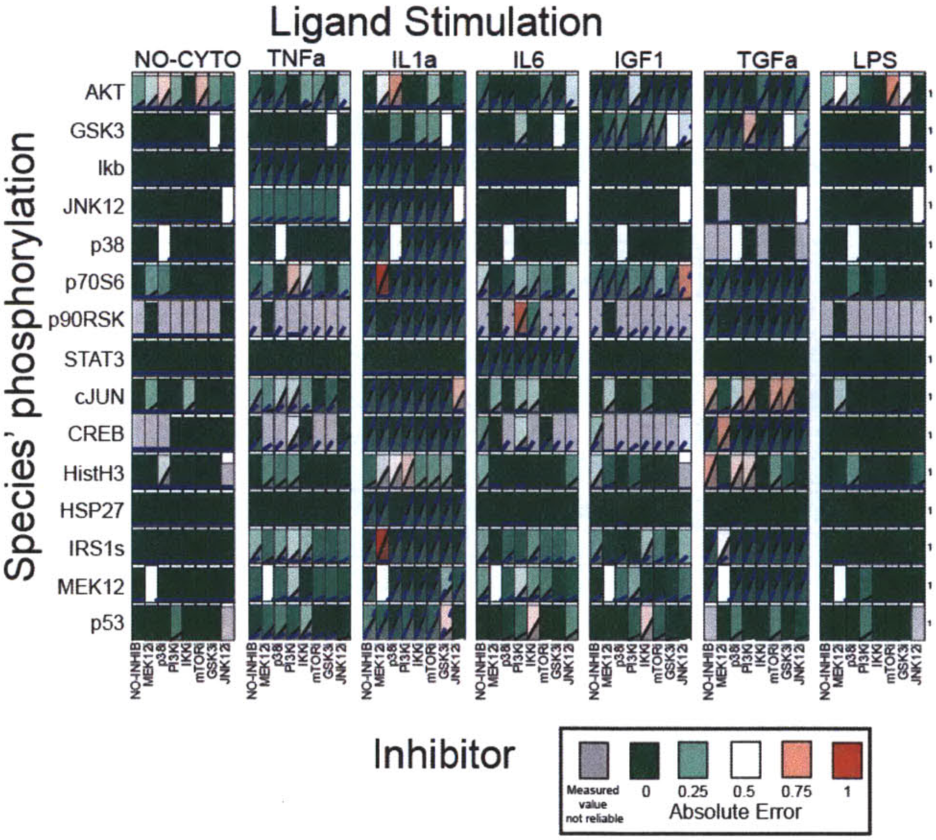


Figure B-8: Analysis of systematic error in c-Jun under TGFα stimulation. Both the training and follow-up datasets indicate that c-Jun but not JNK is phosphorylated upon TGFα stimulation. In PKN1, the only path for c-Jun activation is by JNK activation. The cFL networks account for this discrepancy in one of two ways: (1) Partial activation of the JNK node (increasing error) and amplification of this signal to further activate the c-Jun node (decreasing error). CFL networks that followed this treatment contained Ras -to- MAP3K1 or PI3K -to- MAP3K1 links (blue 'With Crosstalk' case). (2) No activation of c-Jun under TGFα stimulation, increasing error in only the c-Jun signaling node. CFL networks that followed this treatment contained neither Ras -to- MAP3K1 nor PI3K -to- MAP3K1 links (red 'Without Crosstalk' case). No significant differences in ability to fit the other signals are observed. Each of these treatments of c-Jun activation corresponds to a different biological explanation. The first treatment corresponds to the explanation that JNK was partially activated but our measurement did not reflect this while the second treatment corresponds to the explanation that an interaction we did not include in PKN1 was causing c-Jun to be activated.

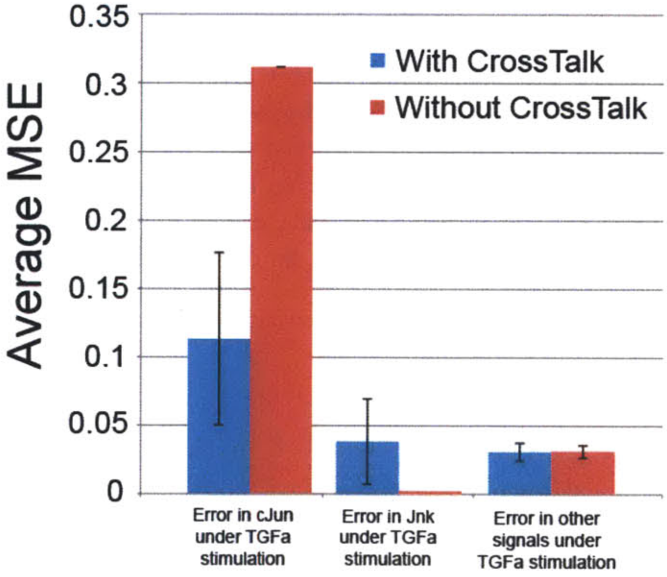


Figure B-9 *(facing page)*: Fit of cFL networks to follow up data. (a) Experimental design of follow-up dataset describing the HepG2 response to combinations of ligand and inhibition treatments. (b) Raw data was rescaled using common conditions as described in Prill et al., in preparation. (see http://wiki.c2b2.columbia.edu/dream/data/scripts/DREAM4/ for Challenge_3 data scaling scripts). Briefly, a linear correlation the log-normalized signals under common conditions of the training and validation data was fit. Parameters of this line were used to scale the log-normalized validation data, which was then transformed back into the linear range. The resulting rescaled values are shown. (c) CFL networks were trained to the HepG2 dataset using PKN1^{i}. The data is displayed as described in Supplementary Figures B-1 and B-3. The filtered models were able to fit the validation data with an MSE of 0.076 ± 0.005. Some of this error (13%) was expected, as these conditions were similar to the experimental conditions under which the main discrepancies between the training data and models were observed (phosphorylation of IRS1s and p70s6 under IL1α stimulation and MEK inhibition). An additional 25% of the error can be accounted for by variation in the normalized data of the common conditions of the two datasets. Plots were generated by CellNOpt (fit to data) and the open-source MATLAB toolbox DataRail [126] (raw data).

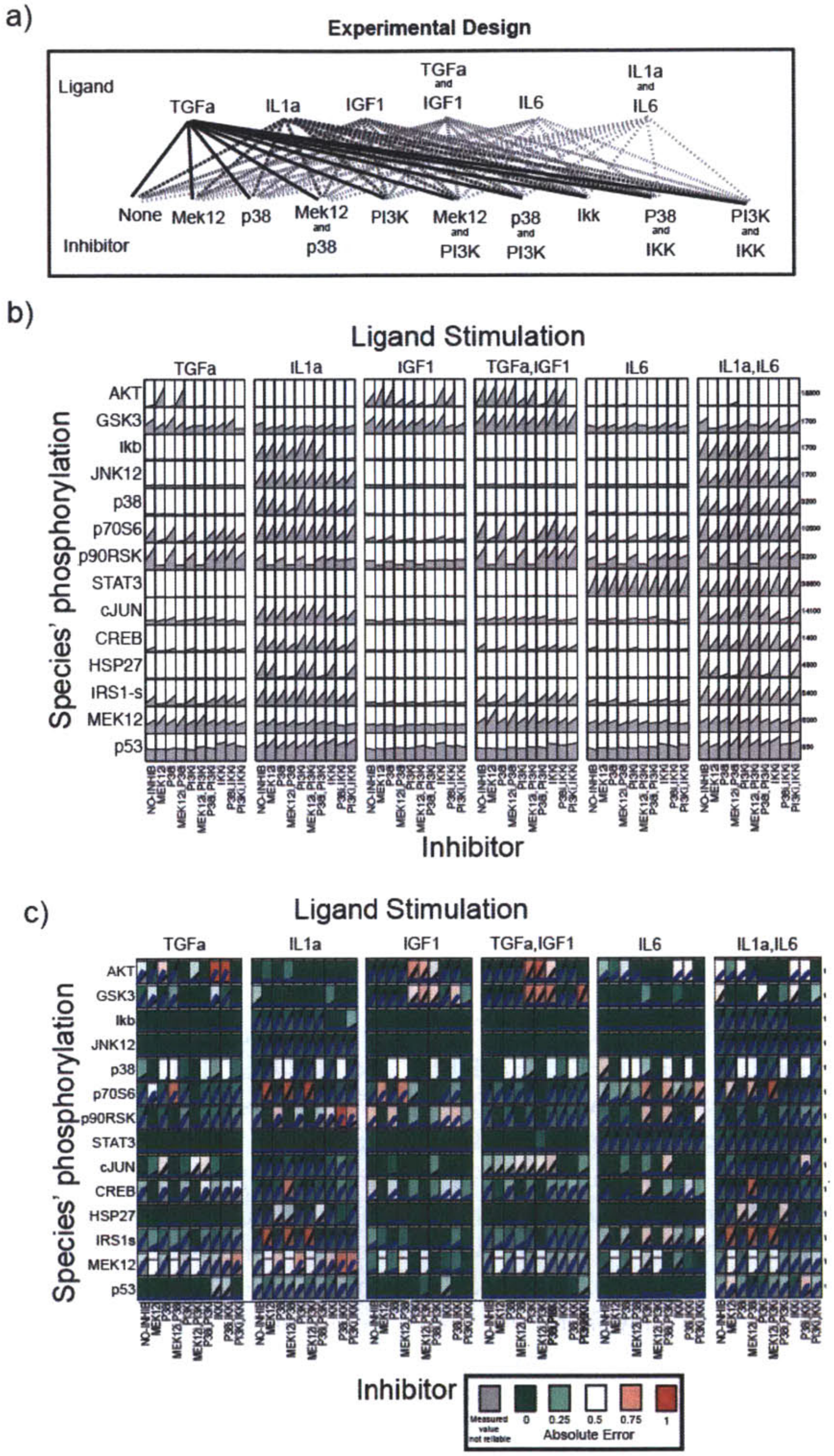

a)
Experimental Design
Ligand
TGFa
IL1a
IGF1
TGFa and IGF1
IL6
IL1a and IL6
None
Mek12
p38
Mek12 and p38
PI3K
Mek12 and PI3K
p38 and PI3K
Ikk
P38 and IKK
PI3K and IKK
Inhibitor
b)
Ligand Stimulation
TGFa
IL1a
IGF1
TGFa,IGF1
IL6
IL1a,IL6
Species' phosphorylation
AKT
GSK3
Ikb
JNK12
p38
p70S6
p90RSK
STAT3
cJUN
CREB
HSP27
IRS1-s
MEK12
p53
NO-INHIB
MEK12
P38
MEK12,P38
PI3K
MEK12,PI3K
P38,PI3K
IKK
P38,IKK
PI3K,IKK
Inhibitor
c)
Ligand Stimulation
TGFa
IL1a
IGF1
TGFa,IGF1
IL6
IL1a,IL6
Species' phosphorylation
AKT
GSK3
Ikb
JNK12
p38
p70S6
p90RSK
STAT3
cJUN
CREB
HSP27
IRS1s
MEK12
p53
NO-INHIB
MEK12
P38
MEK12,P38
PI3K
MEK12,PI3K
P38,PI3K
IKK
P38,IKK
PI3K,IKK
Inhibitor
Measured value not reliable
0
0.25
0.5
0.75
1
Absolute Error

Figure B-10: Fit of PLSR model of phenotypic cytokine release data. A three-component PLSR model fit normalized cytokine release data well in most cases except the condition of TNFα stimulation and Iκb inhibition. The data is displayed as described in Supplementary Figures B-1 and B-3. Plots were generated by CellNOpt.

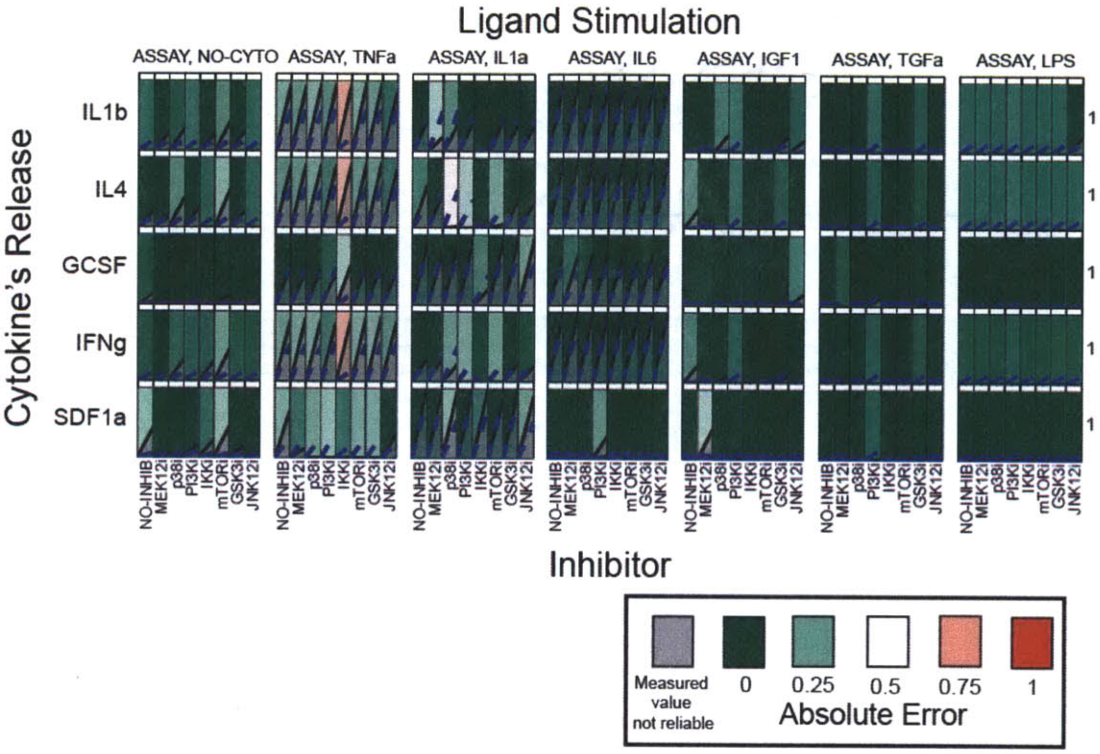


Figure B-11 *(facing page)*: Fit of cFL models linking protein signals to phenotypic cytokine release. Several signaling nodes (MEK1/2, CREB, GSK3, c-Jun, Hsp27, Iκb, and STAT3) were linked to cytokine release nodes (IL1B, IL4, GCSF, IFNg, and SDF1a) in an extended PKN (Table 3, PKN2D) and trained to the HepG2 dataset of both protein signaling and cytokine release data. The fit of the family of cFL models (a) was similar to other cFL models for the signaling data but slightly worse than the PLSR model fit to cytokine release data (Supplementary Figure B-10). A subset of these models had MSEs less than one standard deviation of the mean MSE of the family of models. Those models were deemed most reliable because they fit the data very well. The fit of the average prediction of these models is shown in (b). These average structure for this subset can be found in Supplementary Figure B-12. The data is displayed as described in Supplementary Figures B-1 and B-3. Plots were generated by CellNOpt. Because few cFL network models contained links between MEK1/2, CREB, and GSK3 to cytokine release (Supplementary Table B.3), these links were removed from the extended prior knowledge network and the resultant network trained to the data.

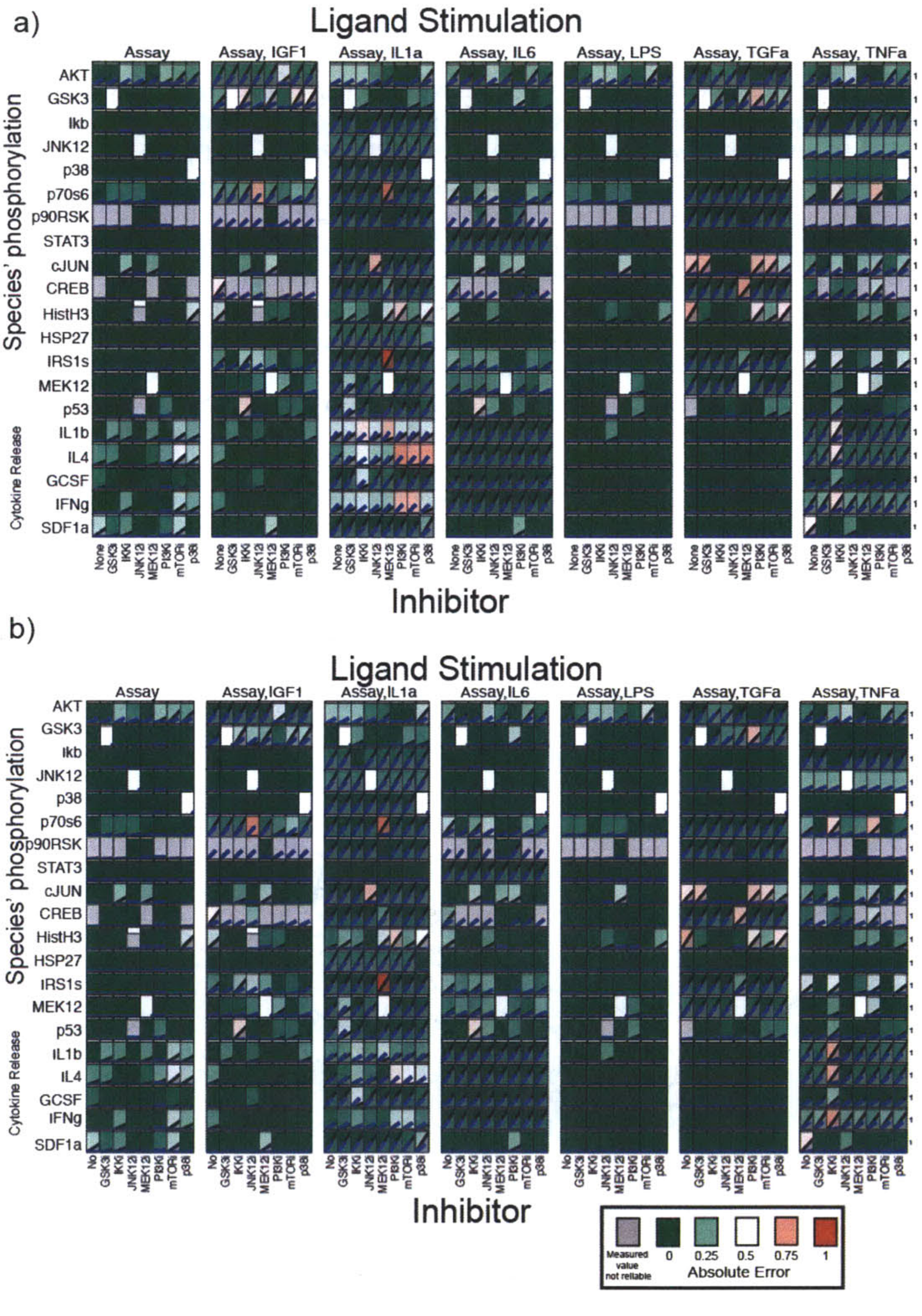

a)
Ligand Stimulation
Assay
Assay, IGF1
Assay, IL1a
Assay, IL6
Assay, LPS
Assay, TGFa
Assay, TNFa
Species' phosphorylation
AKT
GSK3
Ikb
JNK12
p38
p70s6
p90RSK
STAT3
cJUN
CREB
HistH3
HSP27
IRS1s
MEK12
p53
Cytokine Release
IL1b
IL4
GCSF
IFNg
SDF1a
None
GSK3i
IKKi
JNK12
MEK12
PI3Ki
mTORi
p38i
Inhibitor
b)
Ligand Stimulation
Assay
Assay,IGF1
Assay,IL1a
Assay,IL6
Assay,LPS
Assay,TGFa
Assay,TNFa
Species' phosphorylation
Cytokine Release
No
GSK3i
IKKi
JNK12
MEK12
PI3Ki
mTORi
p38i
Inhibitor
Measured value not reliable
0
0.25
0.5
0.75
1
Absolute Error

Figure B-12: Structure of filtered cFL models linking protein signals to phenotypic cytokine release. Several signaling nodes (MEK1/2, CREB, GSK3, c-Jun, Hsp27, Iκb, and STAT3) were linked to cytokine release nodes (IL1B, IL4, G-CSF, IFNg, and SDF1a) in an extended prior knowledge network (Table 3, PKN2D) and trained to the HepG2 dataset of both protein signaling and cytokine release data. Structures of the subset of 31 filtered cFL network models with MSE less than one standard deviation from the mean of the entire family is shown. Links colored black were present in all models whereas links colored grey were present in a fraction of the models (a darker grey indicates that the cFL gate was present in more models). Graph of cFL network models was generated a CellNOpt routine using the graphviz visualization engine (www.graphviz.org) followed by manual annotation in Adobe Illustrator. Because few cFL network models contained links between MEK1/2, CREB, and GSK3 to cytokine release (Supplementary Table B.3), these links were removed from the extended prior knowledge network and the resultant network trained to the data. The average prediction of these models fit similarly to those in Supplementary Figure B-11 and the models' structures can be found in Figure 2-9.

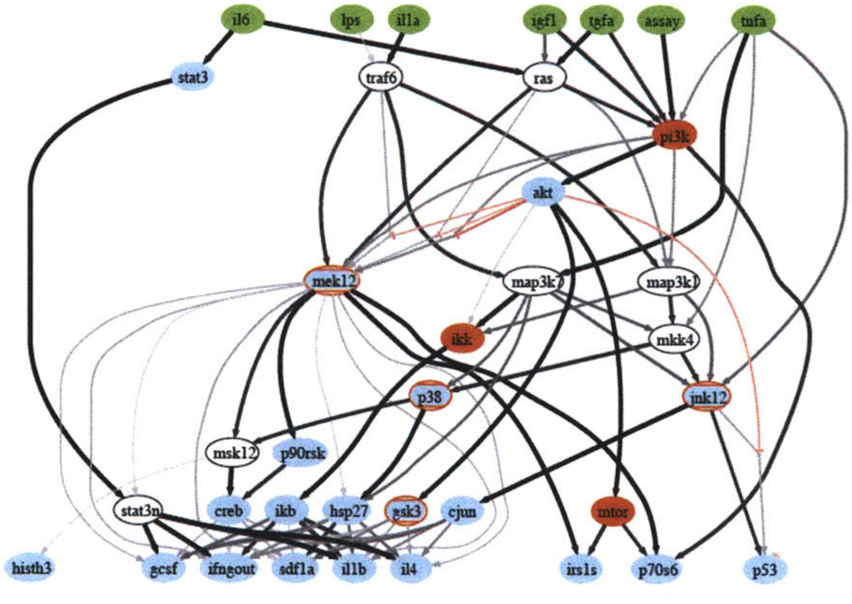


Figure B-13 *(facing page)*: Investigating the use of alternate mathematical operators to evaluate AND and OR gates. The extended prior knowledge network (Supplementary Figure B-2a) was processed to include all two-input AND gates only when an inhibitory interaction was modeled (Supplementary Figure B-2d, PKN1^i) and CellNOpt-cFL used to train 149 network models to the HepG2 dataset. However, the cFL formalism was altered slightly so that an AND gate was evaluated using the product operator and an OR operation evaluated with the sum operator, where the scaling was maintained to between zero and one by limiting the maximum value of any species to one. Note the similarity of these results to those obtained with Min/Max operators are used to evaluate AND and OR gates, respectively (compare Figure 2-4c to part a of this figure, Figure 2-5 to part b, and Figure 2-7 to part c). Reduction of the family of cFL models indicates that a Selection Threshold of 0.005 is also appropriate in this case. The structures of the family of cFL network models trained to the HepG2 dataset are shown (b). Links colored black were present in all models whereas links colored grey were present in a fraction of the models (a darker grey indicates that the cFL gate was present in more models). Filtered cFL network models are shown. Fit to experimental data (c) is displayed as described in Supplementary Figures B-1 and B-3. Plots were generated by CellNOpt.

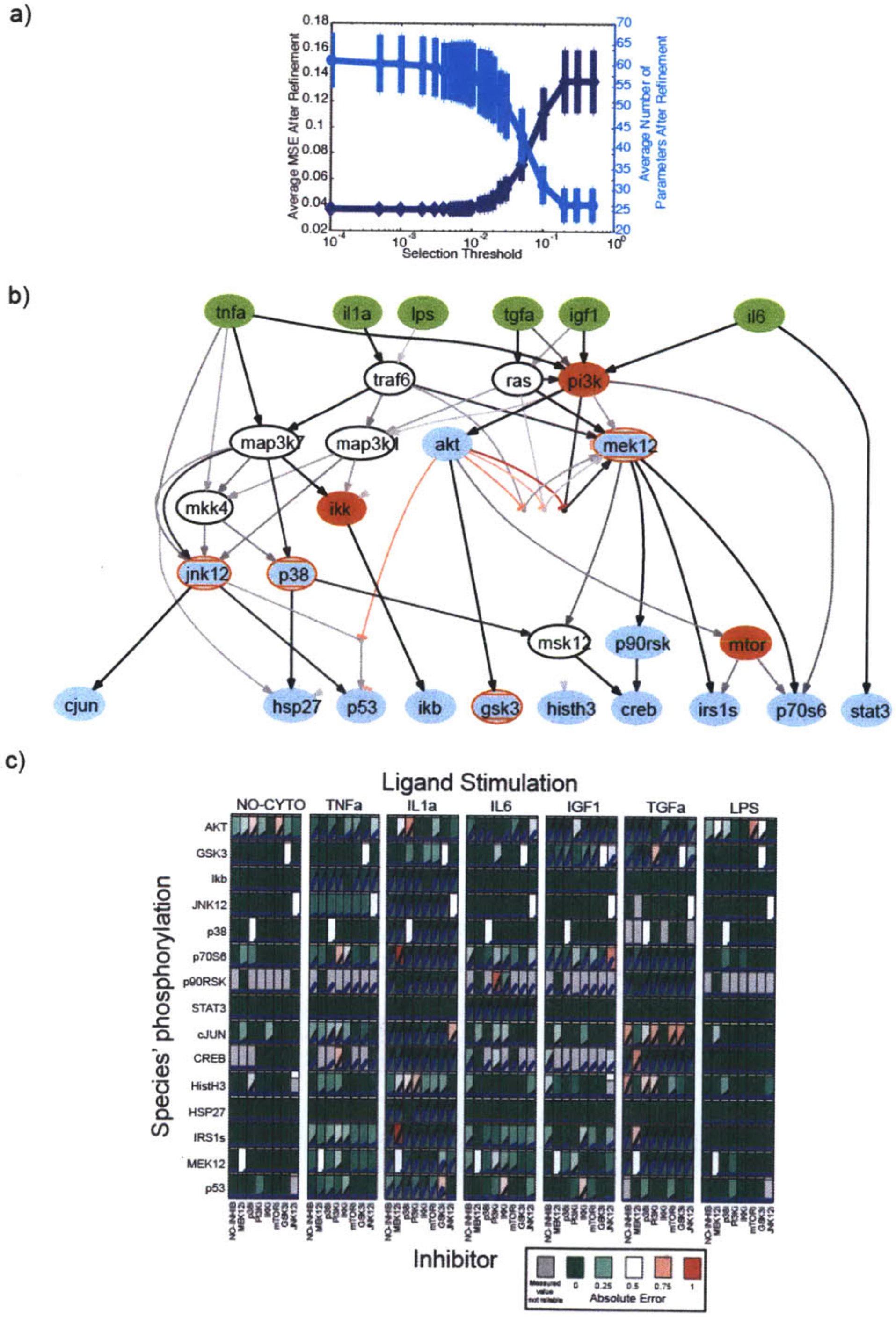
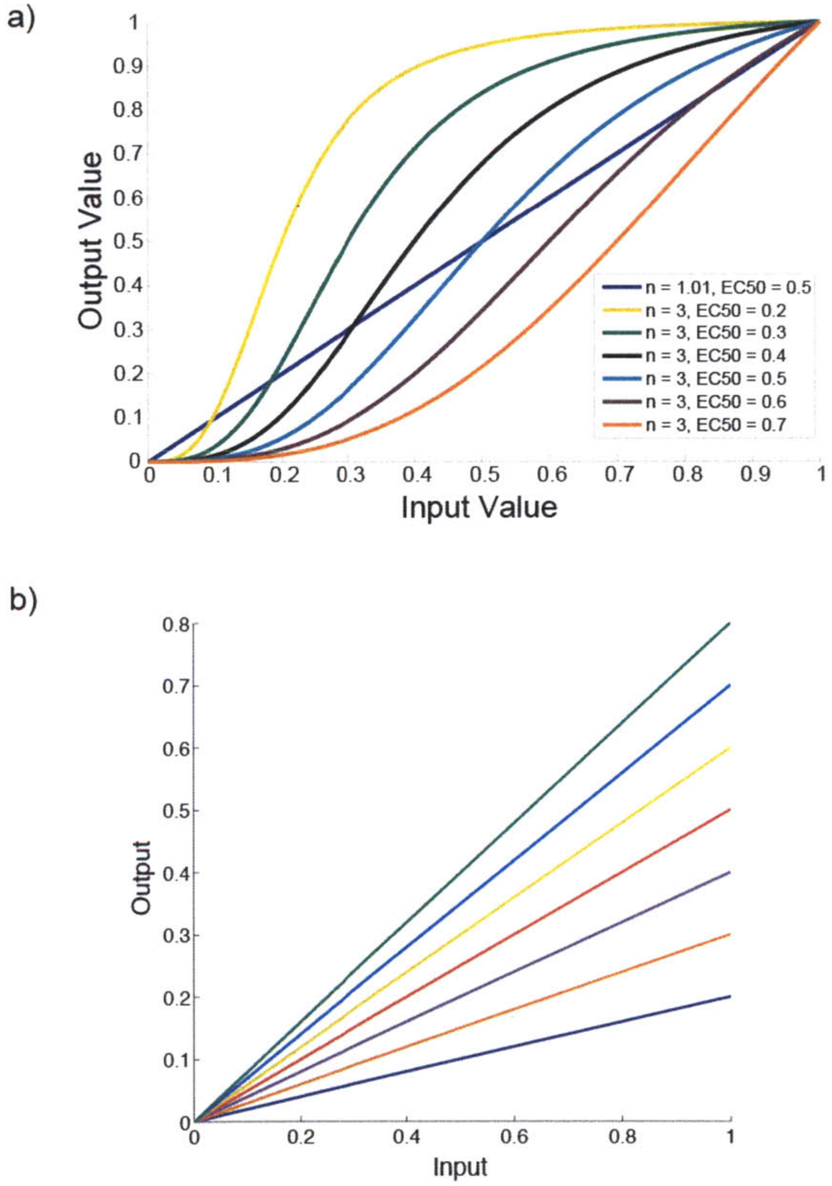

a)
Average MSE After Refinement
0.18
0.16
0.14
0.12
0.1
0.08
0.06
0.04
0.02
Average Number of Parameters After Refinement
70
65
60
55
50
45
40
35
30
25
20
10^{-4}
10^{-3}
10^{-2}
10^{-1}
10^{0}
Selection Threshold
b)
tnfa
il1a
lps
tgfa
igf1
il6
traf6
ras
pi3k
map3k7
map3k1
akt
mek12
mkk4
ikk
jnk12
p38
msk12
p90rsk
mtor
cjun
hsp27
p53
ikb
gsk3
histh3
creb
irs1s
p70s6
stat3
c)
Ligand Stimulation
NO-CYTO
TNFa
IL1a
IL6
IGF1
TGFa
LPS
Species' phosphorylation
AKT
GSK3
Ikb
JNK12
p38
p70S6
p90RSK
STAT3
cJUN
CREB
HistH3
HSP27
IRS1s
MEK12
p53
NO-INHIB
MEK12
p38
PI3K
IKK
mTOR
GSK3
JNK12
Inhibitor
Measured value not reliable
0
0.25
0.5
0.75
1
Absolute Error

Figure B-14: Transfer functions included in the discrete genetic algorithm optimization process. The discrete genetic algorithm chose one of the transfer functions with the indicated parameter sets during the optimization process to relate each input species' value to the output species' value. (a) Transfer functions used to relate species within the network. (b) Transfer functions used to relate ligand input values to the species immediately downstream of them.



B.3 Supplementary Tables

Table B.1: Assessing statistical significance of cFL models derived from PKN1^{i}. Data randomization was accomplished by pairwise exchange of data points. Several types of network randomization were performed. In "Swap Heads" randomization, the input of each interaction was randomly exchanged with the input of another interaction while in "Swap Tails" this process was executed for outputs of each interaction. "Swap Inputs" randomization involved swapping the inputs of all interactions with a randomly chosen output node with the inputs of all interactions with another randomly chosen output node. Finally, completely random networks were generated with the same number of nodes and edges as the extended prior-knowledge network, at least one edge per node, and no incoming but at least one outgoing edge for each network input [124]. For the random data case, P-Values were calculated for each model trained to the real dataset using the Z-score of the model MSE compared to the distribution of randomized data models' MSEs. For the random networks case, the distribution of MSEs was not normal as assessed by the Jarque-Bera test at $\alpha \geq 0.001$. In this case, P-value was calculated as the instance of random models with score less than that of the trained model, of which no instance was observed for any model.

Randomization Method	Average P-Value	Maximum P-Value
Randomize Data (n=312)	9.6×10^{-68}	1.8×10^{-65}
Swap Heads (n=1027)	$< 1.0 \times 10^{-3}$	$< 1.0 \times 10^{-3}$
Swap Tails (n=1059)	$< 1.0 \times 10^{-3}$	$< 1.0 \times 10^{-3}$
Swap Inputs (n=1016)	$< 1.0 \times 10^{-3}$	$< 1.0 \times 10^{-3}$
Completely Random Model (n=1104)	$< 1.0 \times 10^{-3}$	$< 1.0 \times 10^{-3}$

Table B.2: Test sets for cross validation experiment. In each test case, the measured signal under one stimulation condition with all inhibitor conditions was used as the test data. The remaining data was training data.

	Stimulation Condition	Measured signal left out
1	IGF1	GSK3
2	TGFα	GSK3
3	IGF1	Akt
4	TGFα	Akt
5	TNFα	p53
6	IL1α	p53
7	TNFα	IκB
8	IL1α	IκB
9	TGFα	CREB
10	IL1α	CREB
11	TGFα	p90RSK
12	IL1α	p90RSK
13	TGFα	Mek
14	IL1α	Mek
15	TGFα	IRS1s
16	IL1α	IRS1s
17	TGFα	p70s6K
18	IL1α	p70s6K
19	IGF1	p70s6K

Table B.3: Frequency of interactions linking protein signals to phenotypic cytokine release. Frequency in subset of 31 cFL models with MSEs lower than one standard deviation of the family of models

Unprocessed Models					
Input/Output	IL1β	IL4	G-CSF	IFNγ	SDFα
MEK1/2	0.26	0.29	0.29	0.32	0.39
IkB	0.84	0.90	0.61	0.77	0.10
STAT3	1.00	1.00	0.94	1.00	0
GSK3	0.29	0.29	0.16	0.39	0.52
CREB	0.26	0.19	0.23	0.16	0.29
c-Jun	0.48	0.55	0.58	0.58	0.64
Hsp27	0.58	0.65	0.71	0.45	0.74
Filtered Models					
Input/Output	IL1β	IL4	G-CSF	IFNγ	SDFα
MEK1/2	0.19	0.19	0.19	0.26	0.32
IkB	0.84	0.87	0.61	0.74	0.10
STAT3	1.00	1.00	0.94	1.00	0
GSK3	0.26	0.13	0.13	0.29	0.39
CREB	0.19	0.13	0.19	0.06	0.26
c-Jun	0.42	0.45	0.52	0.48	0.52
Hsp27	0.55	0.52	0.61	0.39	0.71

Table B.4: Experimentally verified and computationally predicted transcription factor binding sites in relevant genes. Genes were queried in BioBase TRANSFAC [90, 91] for experimentally verified or computationally predicted transcription factor binding sites and the March 2006 (NCBI36/hg18) assembly of UCSC Genome Bioinformatics (http://genome.ucsc.edu/) for computationally predicted transcription factor binding sites. Those binding sites listed below were included either because they were binding sites of phosphorylated proteins measured or transcription factors known to be modulated by phosphorylated proteins measured [18, 120, 110]. Notes: STAT1 and STAT3 have similar binding motifs (BioBase TRANSFAC). C/EBPα is phosphorylated by GSK3 [120]. TAL1α is phosphorylated by Erk1 [18] and Akt [110]. Phosphorylation by Akt inhibits TAL1α repressor activity [110]. Key: * denotes transcription factor binding site (BioBase TRANSFAC). $^{\#}$ denotes transcription factor binding site in the vicinity of the gene (BioBase TRANSFAC). $^{+}$ denotes site computationally predicted (UCSC Genome Bioinformatics).

Gene	Transcription Factor Binding Site
IL4	IRF1^{*+}, IRF2*
IL1β	NFκB^{+}, CREB*, C/EBPβ^{*}, STAT1*
G-CSF	NFκB^{*+}, C/EBPα^{*}, C/EBPβ^{*}, TAL1α/E47^{+}
SDF1α	STAT^{+}, TAL1α/E47^{+}, c-Jun$^{\#}$, STAT1$^{\#}$
IFNγ	IRF1^{+}, IRF2^{+}, STAT3*, NFκB*, CREB*, c-Jun*
IL6	NFκB*, AP-1*, c-Jun*, C/EBPβ^{*}, CREB^{+}, IRF1^{+}, IRF2^{+}
IRF1	STAT^{*+}, AP-1^{+}, NFκB*
IRF2	CREB^{+}, IRF1*

Appendix C

Supporting Information for Chapter 3

C.1 Comparison of cFL and other modeling formalisms

Numerous frameworks have been proposed by ourselves and others to formally train a biological network to data (e.g. artificial neural networks [35], probabilistic graphical models [160, 79], and logic networks [102, 124]). Our approach here is distinct from these because we base our models solely on prior knowledge using reasonable default parameters. While other frameworks could potentially be used in this manner, we argue that cFL logic models are a more attractive means for quickly and efficiently constructing a reliable model because they use logic operations that relate naturally to a linguistic description and yield interpretable results.

The observation that conclusions drawn from cFL models are abstract and species values must be considered relative to those of other species in the model points to a limitation of the technique. Thus, if the goal of a study is to predict an absolute parameter of a system (i.e. most effective K_d of a drug, recommended dose, etc.), one should use a modeling approach that is able to directly relate to physical properties (such as differential equations). However, a mechanistic differential equation model requires more precise knowledge of both the mechanisms and parameters governing system behavior. While default parameters could be assumed for DEs as we exemplified for our cFL models here, estimates for DE parameters must be at least approximately correct because different parameters regimes can yield systems with very different behavior [73]. Thus, we conclude that while cFL models are limited in that they can only make qualitative predictions, they require less precise knowledge of the system, making them an attractive alternative to mechanistic DEs.

Because the quantities resulting from cFL models are abstract, one could raise the question of whether modeling with ostensibly simpler Boolean or discrete logic would be sufficient for the analysis we present here. Indeed, cFL models use traditional AND, OR, and NOT gates to specify the topology of a network, such that tools developed for either analysis are readily interchangeable. However, the use of cFL is justified for several reasons. First, discrete models lack transfer functions such that analyses similar to that shown in main text Figures 3-6C and D could not easily be performed with a discrete model. Furthermore, analysis with cFL is no more difficult than one with discrete logic because of the simplicity of the cFL formalism and ease of specifying a model and its transfer functions in Q2LM. Moreover, cFL modeling allows one to explore the effects of the amount of perturbation, different implementations of perturbations, and the effect of noise in the transfer function parameters. Such explorations allow one to ascertain whether the predictions are robust to variations of the model, which if confirmed, increases confidence in their reliability.

C.2 Supplemental Experimental Methods

To validate the Q2LM model, we measured the ability of wild-type granulocyte colony-stimulating factor (gGCSF) or a mutant form (G43, D113H mutation) to promote hematopoiesis in 5-fluorouracil (5FU)-treated mice, similar to previously reported methods [100, 136]. Briefly, B6D2F1 mice (Jackson Laboratories) were divided into seven groups of five mice each (n=35). One group of animals served as a control group and received no 5FU or colony stimulating factor. The rest of the groups were treated with 150 mg/kg 5FU for 24 hours prior to treatment with colony stimulating factor (gGCSF or G43, injected i.p in phosphate buffered saline, supplemented with 0.1% BSA) for 9 days. Daily doses of 25 or 50 microg/kg of gGCSF or 25, 50, or 100 microg/kg were administered separately to a group of animals. After dosing of colony stimulating factor was completed, animals were sacrificed and blood collected by cardiac puncture. After hemolysing red blood cells using a standard lysis solution (10 mM potassium bicarbonate, 150 mM ammonium chloride, 0.1 mM EDTA, pH 8.0), white blood cells were concentrated and cell count performed with a Coulter counter (Beckman Coulter Instruments). Results are expressed in Main Text Figure 3-6 as an average cell count plus standard deviation.

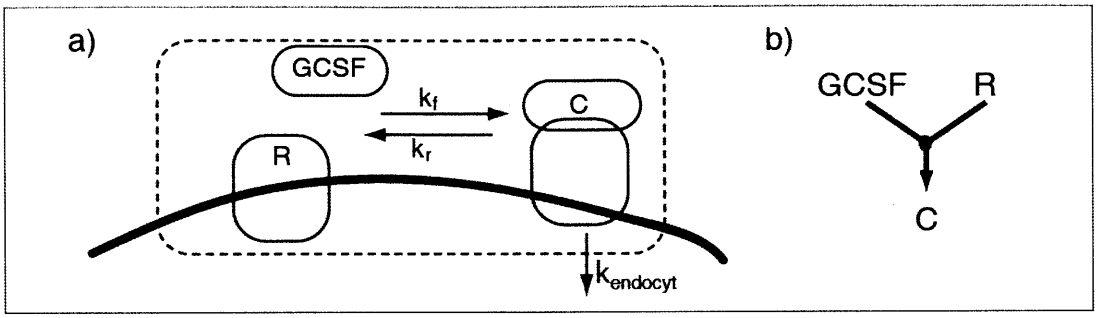
C.3 Relationship between cFL and mechanistic ODEs

C.3.1 Introduction

While others have explored the relationship between stoichiometric maps and logic gates [71, 131], these derivations point out the relationship between logic models and ordinary differential equations (ODEs) based on mass balances. A mass balance is a basic engineering concept based on the law of conservation of mass. A mass balance simply translates the statement "the rate of change of a species' mass in a defined system equals the rate of entry plus the rate of generation minus the rate of consumption and the rate of exit" into an ordinary differential equation.

The first major concept used in the derivations below is that the updating scheme in a logic model simulation is analogous to steady-state solution of an ODE. In the simulation of a logic model, each node is updated based solely on its input nodes'

Figure C-1: Binding of GCSF to its recepter. (a) graphical depiction for development of the mass balance (b) logic gate representation of this interaction



states at the previous time step and the concept of time is not considered. In an ODE framework, this is akin to evaluating each species as if it were at psuedo-steady state.

The figures depicting these systems (Figures C-1, C-3, and C-6) also point out important distinctions between the interpretation of an mechanistic ODE and that of a logic model. In the graphics that motivate development of an ODE, an arrow generally indicates that the molecular species at the 'head' or 'input' of the arrow undergoes some change (i.e. is internalized, becomes bound, gets degraded, etc.). However, in a logic model, these arrows indicate only that the value of one species affects the value of another. Thus, we should understand the arrows in logic models not as indication of what happens to the input species, but rather as indications that the value of the input species (as determined by other nodes) results in some change to the output species' value.

C.3.2 Receptor Binding

In a mechanistic ODE modeled with mass action kinetics, a mass balance on bound receptor $[C]$ depicted in Figure C-1 can be written with Equation C.1:

$$\frac{d[C]}{dt} = k_f[R][GCSF] - k_r[C] - k_{endocyt}[C] \tag{C.1}$$

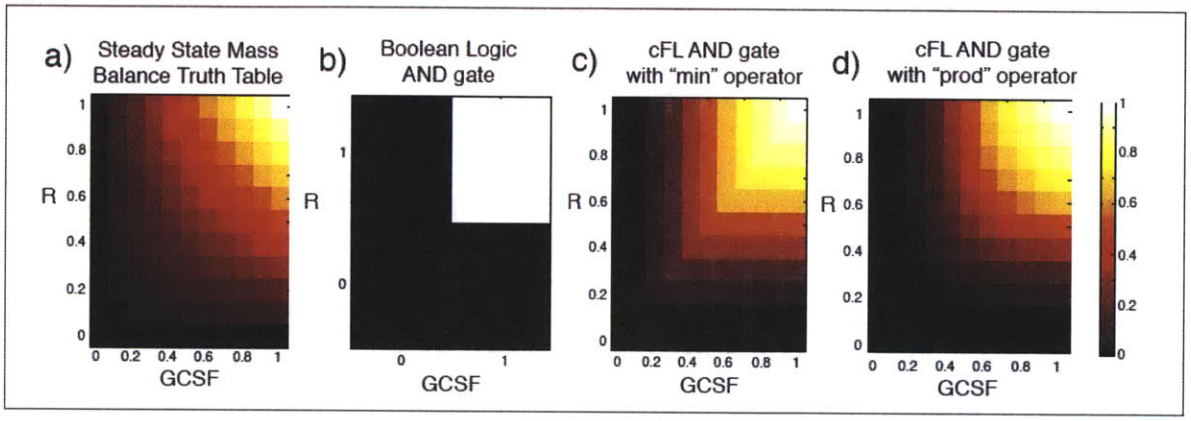
Due to the relationship between the updating scheme of a logic model and steady state described in Section C.3.1, we set the derivative in Equation C.1 to zero and solve for the steady state value of [C] (C_{SS}).

$$0 = k_f R_{SS} GCSF_{SS} - k_r C_{SS} - k_{endocyt} C_{SS} \tag{C.2}$$

$$C_{SS} = \frac{k_f}{k_r + k_{endocyt}} R_{SS} GCSF_{SS} \tag{C.3}$$

From Equation C.3, we note that the pseudo-steady state value of [C] (C_{SS}) is a function of the product of R_{SS} and $GCSF_{SS}$. This dependence is plotted as a heat map in Figure C-2a. From this plot, it is clear that this relationship between

Figure C-2: Heat map of C_{SS} as a function of varying amounts of $GCSF_{SS}$ and R_{SS}. 'Truth tables' of binding as modeled by (a) Equation C.3, (b) Boolean logic AND gate, (c) cFL AND gate evaluated with the *min* operator, and (d) cFL AND gate evaluated by the *prod* operator. For evaluation of Equation C.3 values of Receptor and GCSF were considered to be scaled between zero and one. For cFL evaluation, transfer functions with a gain of 1, EC_{50} of 0.5, and hill coefficient of 3 were assumed. The truth table in (a) could be directly replicated with cFL by using a linear transfer function with slope = 1 and intercept = 0.



species corresponds to an AND gate in logic terms. A heat map of the Boolean logic AND gate truth table (Figure C-2b) is a very abstract representation, but the cFL AND gates shown (Figure C-2 c and d) demonstrate a closer relationship to the 'biochemical truth table.' Thus, we see that the AND gate relating pNR and $bloodGCSF$ to $pNboundGCSF$ is directly related to the steady-state solution of the mechanistic ODE based on mass action kinetics. Additionally, the mass-balance concept of "losing" $[C]$ due to endocytosis does not change the functional form of the relationship of its steady state value to R_{SS} and $GCSF_{SS}$.

C.3.3 Receptor Degradation

To model endosomal degradation, we do not explicitly model all of the processes that occur mechanistically (endocytosis of both bound and unbound receptors, disassociation, etc). Rather, we use the abstract concept of "Substance" ($Subst$) shown in Figure C-3 to lump all processes into one mass balance (equation C.4):

$$\frac{dSubst}{dt} = Subst_{in} - Subst_{degraded} - Subst_{recycled} \tag{C.4}$$

Using a "fraction degraded" constant (f_{deg}) to relate $Subst_{in}$ and $Subst_{degraded}$, we obtain

$$\frac{dSubst}{dt} = Subst_{in} - f_{deg}Subst_{in} - Subst_{recycled} \tag{C.5}$$

Again, we use the steady state description and set the derivative in Equation C.5 to zero and solve for the steady state value of $Subst_{recycled}$.

$$Subst_{recycled} = Subst_{in}(1 - f_{deg}) \tag{C.6}$$

From Equation C.6, we again note that the steady state value of $Subst_{recycled}$ is a function of the product of $Subst_{in}$ and $(1 - f_{deg})$. In logic terms, the product again corresponds to an AND gate truth table where "1−" in $1 - f_{deg}$ indicates inhibition (Figure C-4). Thus, the AND NOT gate relating $pNboundGCSF$ and $pNdegGCSF$ to $pNrecGCSF$ is related to the steady-state solution of an abstracted ODE describing the mass balance of these entities. In this case, the mass-balance concept of "losing" substance due to degradation did change the functional form of the relationship of the amount of substance recycled because this 'loss' is reflected in the logic gate by the inhibition of recycling by degradation.

Figure C-3: Degradation of bound receptor. (a) graphical depiction for development of the mass balance (b) logic gate representation of this interaction

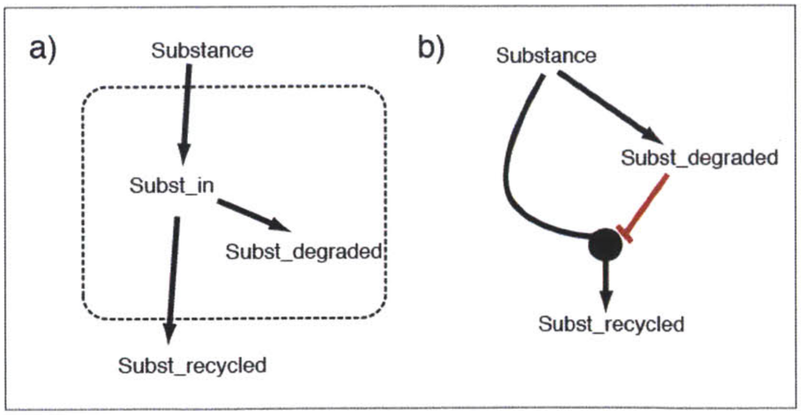
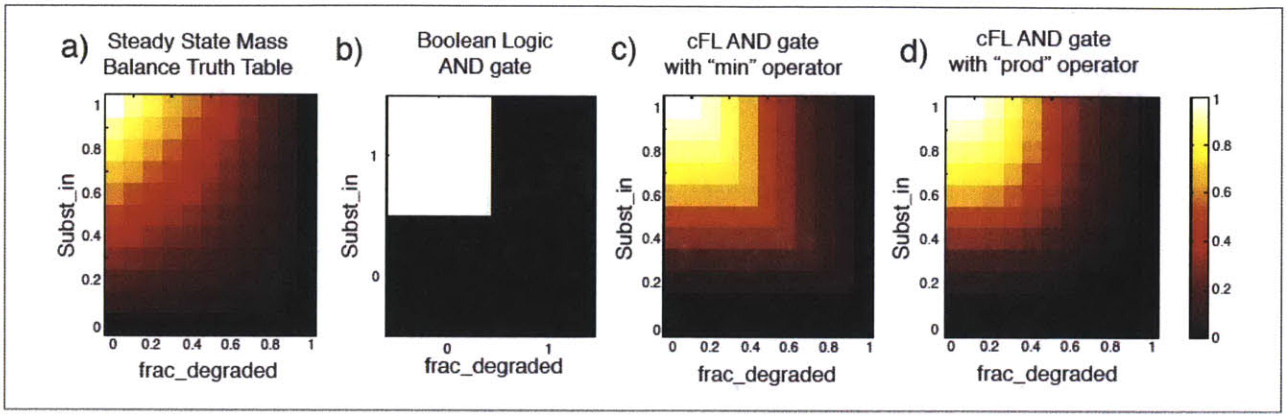
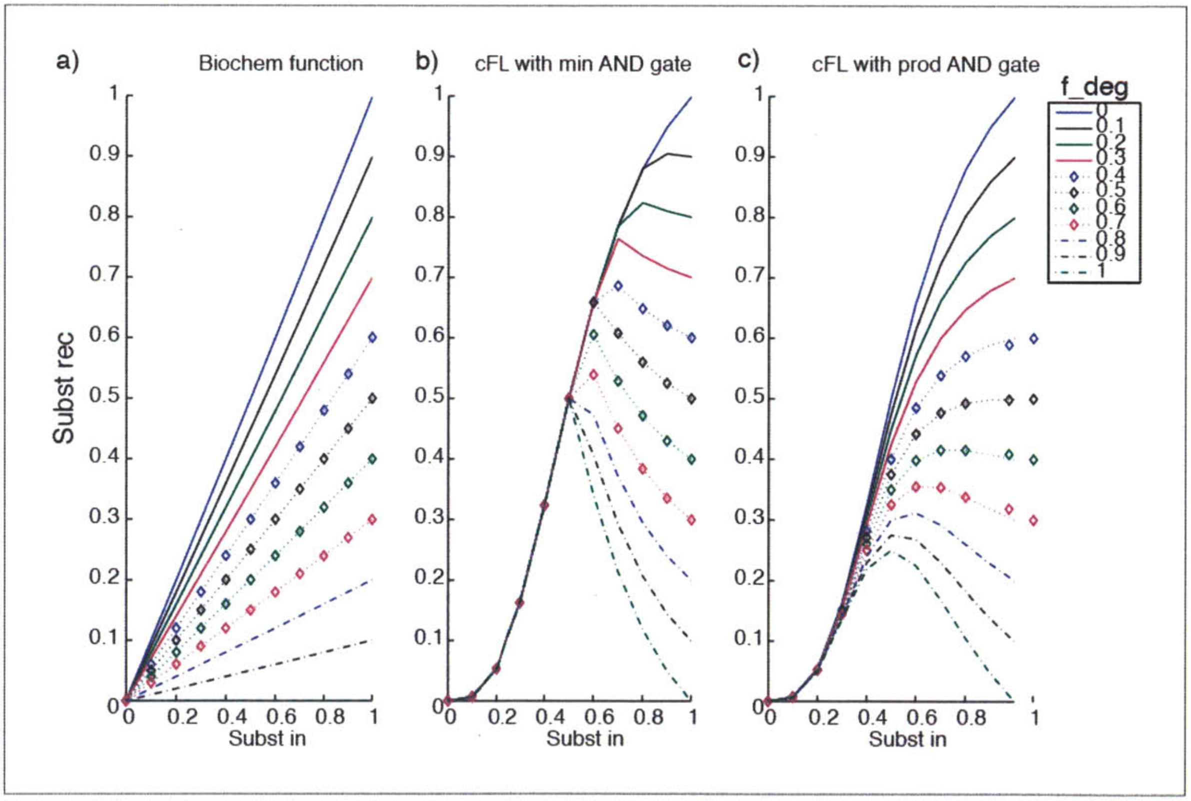


Figure C-4: Heat map of $Subst_{recycled}$ as a function of varying amounts of $Subst_{in}$ and f_{deg}. 'Truth tables' of recycling and degradation as modeled by (a) Equation C.6, (b) Boolean logic AND gate, (c) cFL AND gate evaluated with the *min* operator, and (d) cFL AND gate evaluated by the *prod* operator. Equations evaluated as described in Figure C-2



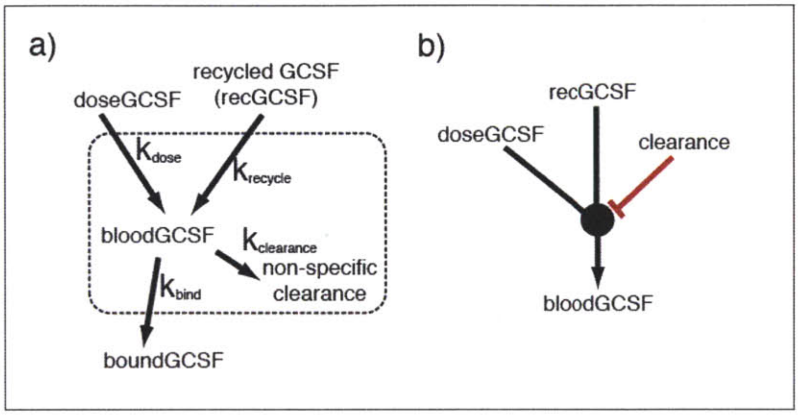
In Equation C.5, we assumed that "fraction degraded" was a constant for the derivation of Equation C.6. However, in Figure C-3b, it is clear that the $Subst_{degraded}$ species depends on $Subst_{in}$ in the logic description. Thus, when we plot the level of $Subst_{recycled}$ as a function of only $Subst_{in}$, we notice dissimilarities in the resulting relationship (Figure C-5). These dissimilarities are caused by the fact that the amount of faction of substance degraded is dependent on $Subst_{in}$ in the logic case, while it is a constant in Equation C.6. It is unclear which assumption is correct in the actual biological system, and we can model the relationship specified by Equation C.6 in a logic model by not having an additional 'degradation' species and instead modelling the logic as a direct interaction between $Subst_{in}$ and $Subst_{recycled}$ the 'gain' of the transfer function relating them analogous to $1 - f_{deg}$. However, this further level of abstraction hinders our ability to alter the 'degradation' species directly. Additionally, it is unclear if f_{deg} is actually independent of the amount of substance presence in the biological setting. Nonetheless, we repeated the work presented in the main text and found that the interpretation of the results is the same regardless of the logic description used for the endosomal degradation process.

Figure C-5: $Subst_{recycled}$ as a function of varying amounts of $Subst_{in}$ modeled by (a) Equation C.6, (b) cFL AND gate evaluated with the *min* operator, and (c) cFL AND gate evaluated by the *prod* operator. In the derivation based on mass balances (a), the variable f_{deg} is considered a constant. In those derived from the logic gate, we vary the gain of the transfer function relating $Subst_{in}$ and $Subst_{degraded}$ and consider this gain to be f_{deg}



C.3.4 Amount of GCSF in the blood: an example where the logic gate and mass-balance are not analogous

Figure C-6: Processes affecting GCSF in the blood. (a) graphical depiction for development of the mass balance (b) logic gate representation of this interaction



We now turn to an example where the relationship between the proper logic gate and mass-balance based ODE is not analogous. A simplified mass balance for GCSF in the blood (Figure C-6) is shown in Equations C.7 - C.8. The 'logic' of the summation in Equation C.8 would normally be an OR gate. However, in the construction of our logic model, we found that an AND gate was necessary to correctly model the logic of the bloodGCSF species because presence or absence of all of the input species to this gate can limit the value of bloodGCSF. In this case the mass balance and logic gate are not analogous. Perhaps one clue that they will not be directly related lies in Equations C.7 - C.8. In these equations, the two terms denoting the 'appearance' of GCSF in the blood were not dependent on a species. Rather, they were further abstracted and given as rate constants independent of other species. Additionally, the contribution of 'binding' is abstracted and modeled as simply a lumped rate rather than including the biochemical steps of association and dissociation. This abstraction at the level of rates of processes serves as an indication that this mass balance does not describe relationships between species we include in our logic model and bloodGCSF, and thus, it will not be directly relatable to our logic model.

$$\frac{dGCSF_{Blood}}{dt} = k_{dose} + k_{rec} - k_{bind}GCSF_{Blood} - k_{clearance}GCSF_{Blood} \tag{C.7}$$

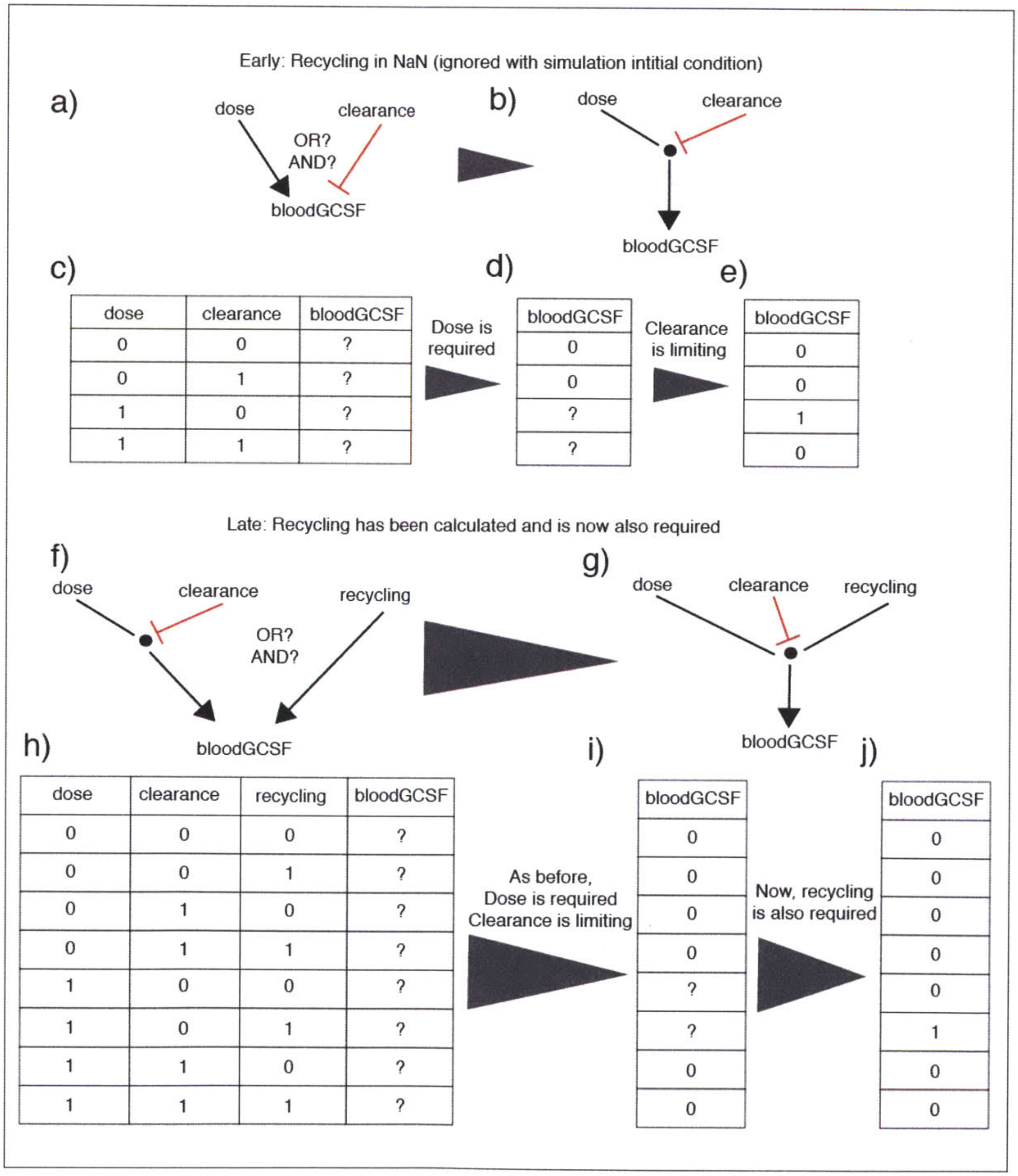
$$GCSF_{Blood,SS} = \frac{k_{dose} + k_{rec}}{k_{clearance} + k_{bind}} \tag{C.8}$$

In order to correctly deduce the logic describing the bloodGCSF species, we instead turn to truth tables describing how we believe the species to relate to other species' values. We first recognize that initially, only the dose and clearance values determine the value of bloodGCSF (because at the beginning of the simulation, recycling has not yet been calculated and is thus the initial value of Not-A-Number). Thus, we will initially determine how bloodGCSF depends on the dose and clearance species (Figure C-7a) by examining the truth table for the dependence of bloodGCSF on limiting values (i.e. zero and one) of each input species (Figure C-7c). We first note that dose is required for bloodGCSF to be 'on'. Thus, we deduce that when dose is zero, bloodGCSF is also zero (Figure C-7d). Next, we note that bloodGCSF is limited by clearance. Thus, we fill in the remaining two entries for the truth table (Figure C-7e). This gate corresponds to an AND NOT gate (Figure C-7b).

Next, we consider how bloodGCSF will depend on recycling after its value has been calculated (Figure C-7f and h). The dependence on dose and clearance remains the same, so we can fill in many entries in the truth table (Figure C-7i). Finally, we note that recycling is now required for bloodGCSF to remain 'on'. Thus, we can fill in the remaining two entries of the truth table (Figure C-7j) and ascertain that the recycling species should be an input to the AND gate (Figure C-7g).

From this example, we see the importance of considering both the interactions between the species as well as how the species will be treated during simulation. As it is sometimes difficult to anticipate all potential factors that should be considered, we emphasize the importance of model validation at the onset of a project as well as repeatedly returning to plots of how the species' values evolve in the course of simulation to check that no artifacts have arisen. For the GCSF example, during the course of analysis, we found that the inclusion of an additional node, 'bodyGCSF' (Main Text Figure 3-5b), was necessary in order to ensure that the model behaved properly under a few conditions where boundGCSF was fixed as a stimulation perturbation (Main Text Figure 3-6b). Such cases underscore the importance of model validation and highlight the benefit of being able to easily 'follow the logic' during model simulation to enable facile model troubleshooting.

Figure C-7: Determining the logic controlling the bloodGCSF node.


Early: Recycling in NaN (ignored with simulation intitial condition)
a)
dose
clearance
OR?
AND?
bloodGCSF
b)
dose
clearance
bloodGCSF
c)
d)
e)
Dose is required
Clearance is limiting
Late: Recycling has been calculated and is now also required
f)
dose
clearance
recycling
OR?
AND?
bloodGCSF
g)
dose
clearance
recycling
bloodGCSF
h)
i)
j)
As before, Dose is required Clearance is limiting
Now, recycling is also required

c)

dose	clearance	bloodGCSF
0	0	?
0	1	?
1	0	?
1	1	?

d)

bloodGCSF
0
0
?
?

e)

bloodGCSF
0
0
1
0

h)

dose	clearance	recycling	bloodGCSF
0	0	0	?
0	0	1	?
0	1	0	?
0	1	1	?
1	0	0	?
1	0	1	?
1	1	0	?
1	1	1	?

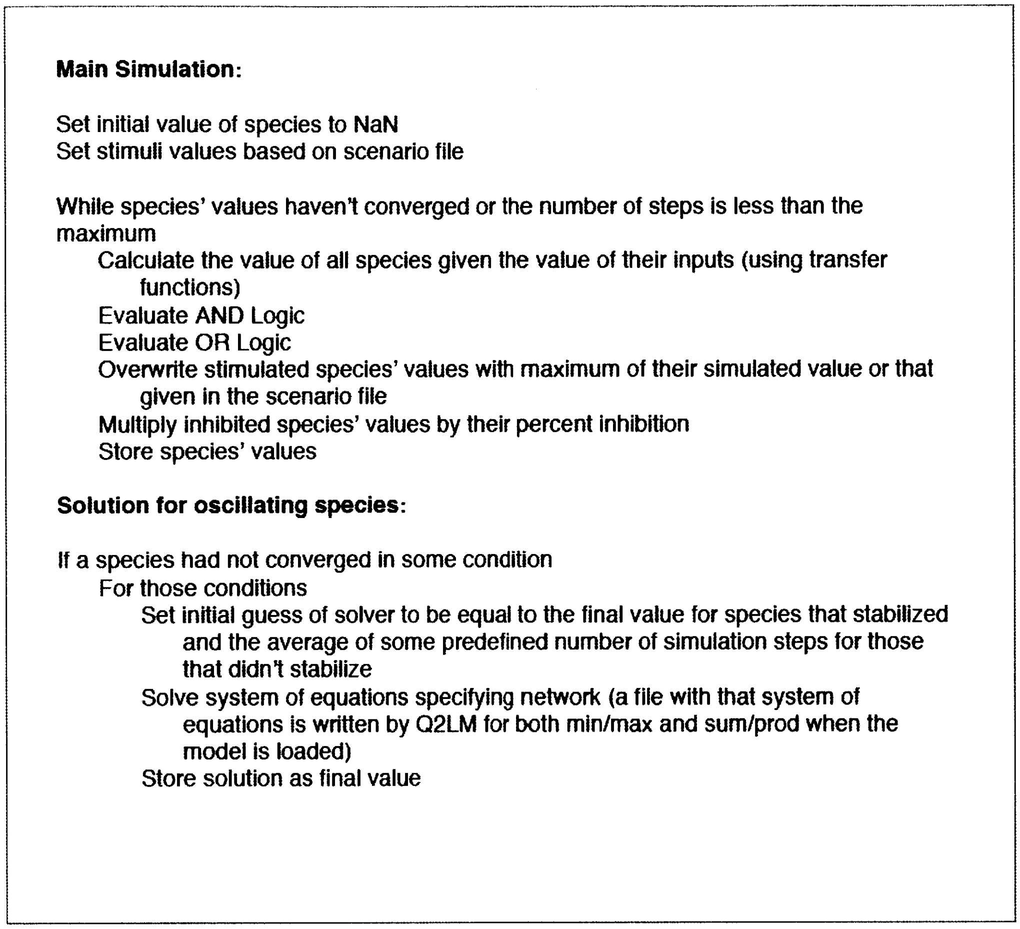
i)

bloodGCSF
0
0
0
0
?
?
0
0

j)

bloodGCSF
0
0
0
0
0
1
0
0

Figure C-8: Simulation Procedure Pseudo-Code



```
Main Simulation:

Set initial value of species to NaN
Set stimuli values based on scenario file

While species' values haven't converged or the number of steps is less than the maximum
    Calculate the value of all species given the value of their inputs (using transfer functions)
    Evaluate AND Logic
    Evaluate OR Logic
    Overwrite stimulated species' values with maximum of their simulated value or that given in the scenario file
    Multiply inhibited species' values by their percent inhibition
    Store species' values

Solution for oscillating species:

If a species had not converged in some condition
    For those conditions
        Set initial guess of solver to be equal to the final value for species that stabilized and the average of some predefined number of simulation steps for those that didn't stabilize
        Solve system of equations specifying network (a file with that system of equations is written by Q2LM for both min/max and sum/prod when the model is loaded)
        Store solution as final value
```

C.4 Simulation Procedure for Determining Steady State Value of Oscillating Species

Figure C-8 describes the procedure developed to calculate the steady state of oscillating species by solving a system of equations. To solve the system of nonlinear equations for cases when species' values are observed to oscillate, we use the fsolve function in MATLAB. This function requires a default initial guess for species values. Depending on the value of the initial guess, the solver will return one of multiple possible roots. In order to return the root corresponding to the steady state of the simulation, the initial guess for each species is determined from the simulated values. Basing the initial guess on simulated species' value is key, as the solution to the equations using a default initial guess not based on simulation results can vary greatly depending on the default initial guess chosen.

Appendix D

Supporting Information for Chapter 4

D.1 Supplementary Figures

Figure D-1: Larger view of prior knowledge network in Figure 4-2

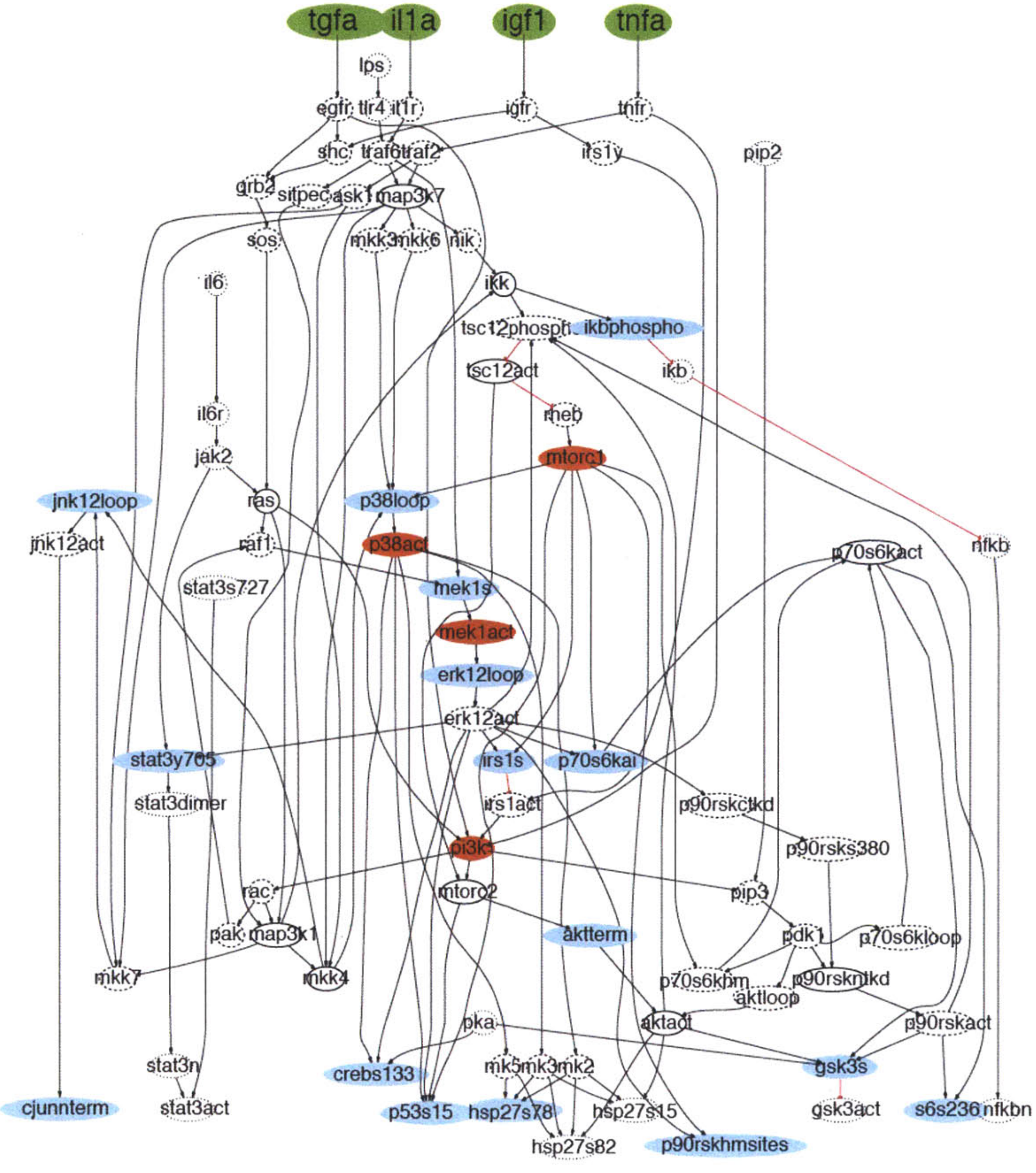


Figure D-2: Larger view of data in Figure 4-2.

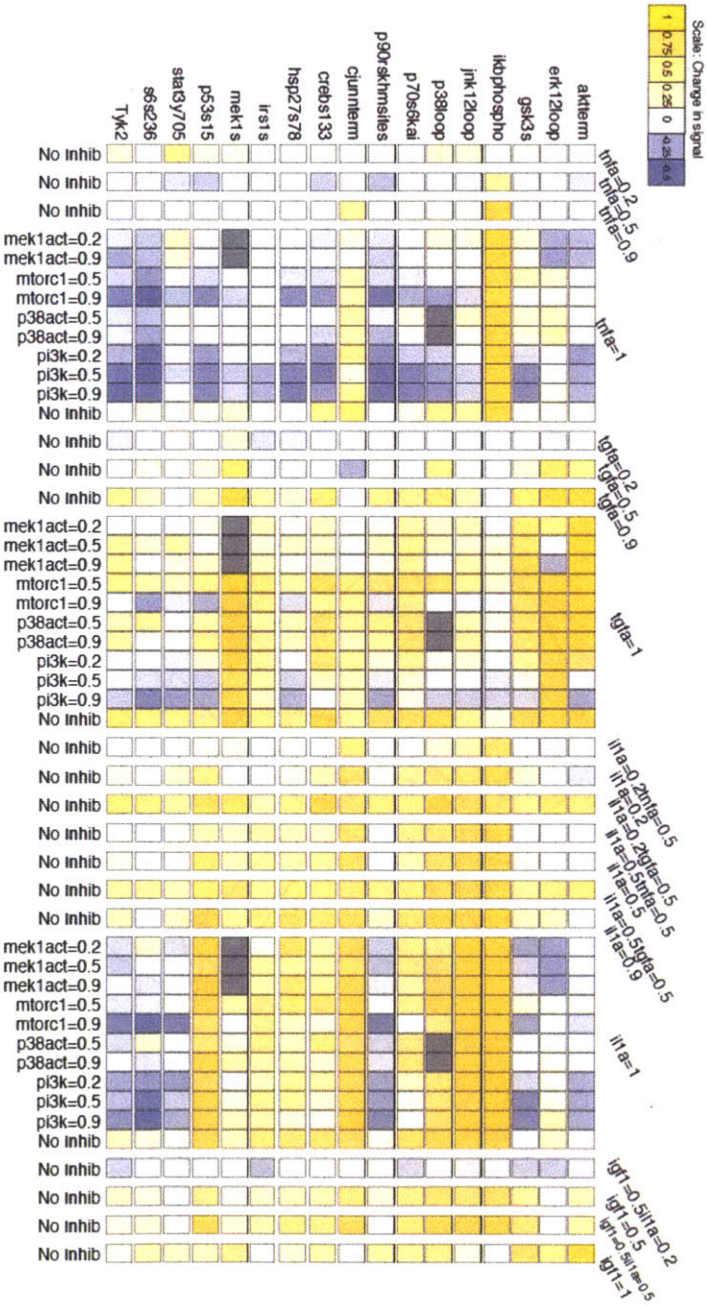


Figure D-3: Larger view of trained models in Figure 4-2

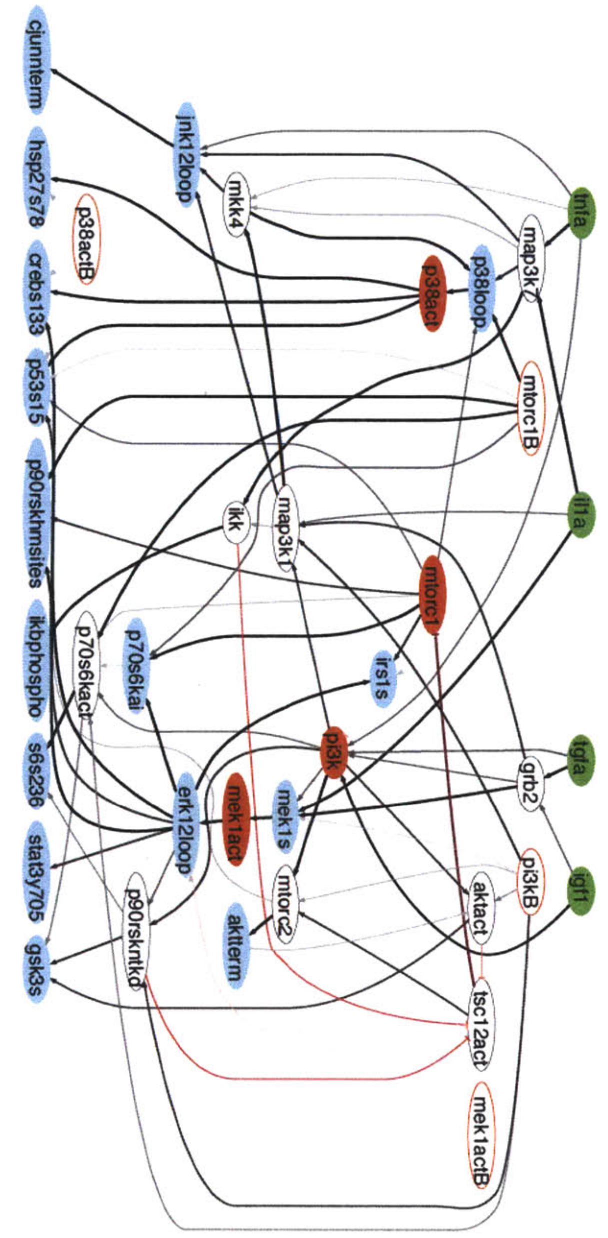


Figure D-4: Larger view of fit in Figure 4-2 Normalized data is black line, average model fit is blue line, and individual model predictions are pink lines

.

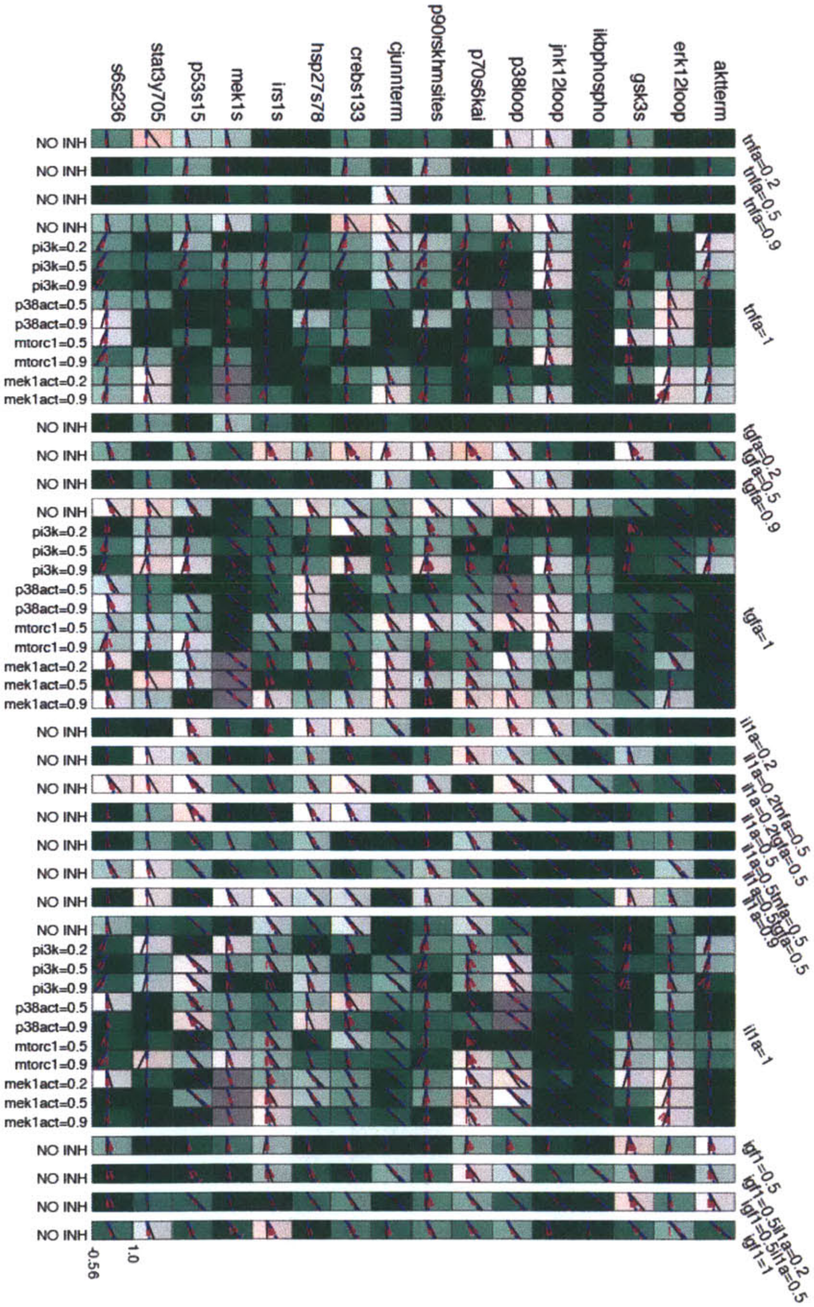


Figure D-5: Dependence of kept models' features on solution pool size. (a) Of 993 models trained to data in Supplementary Figure D-2, only the 120 best fit models were used for further analysis. (b) To determine the affect of pool size on features of kept models, smaller pools were randomly chosen from the full pool and the affect on MSE range of kept models calculated. The plot indicated that, while the kept models would be better fit for larger pools, the effect was incremental after a pool size of 700. The variability in structure was calculated as the standard deviation in each 'bit' indicating if a specific interaction was kept. This result indicates that the topological variance decreased slightly at a pool size of 700, likely because enough solutions had been obtained such that a structural variant that resulted in a slightly better fit was a major component of the best-fit 120 models. We also investigated the influence of keeping more 'well-fit' models with a fixed total pool size of 993. We found that, for the number used in this work (120), only a slight increase in MSE was observed, whereas a much larger increase would be observed in more models were kept, as expected.

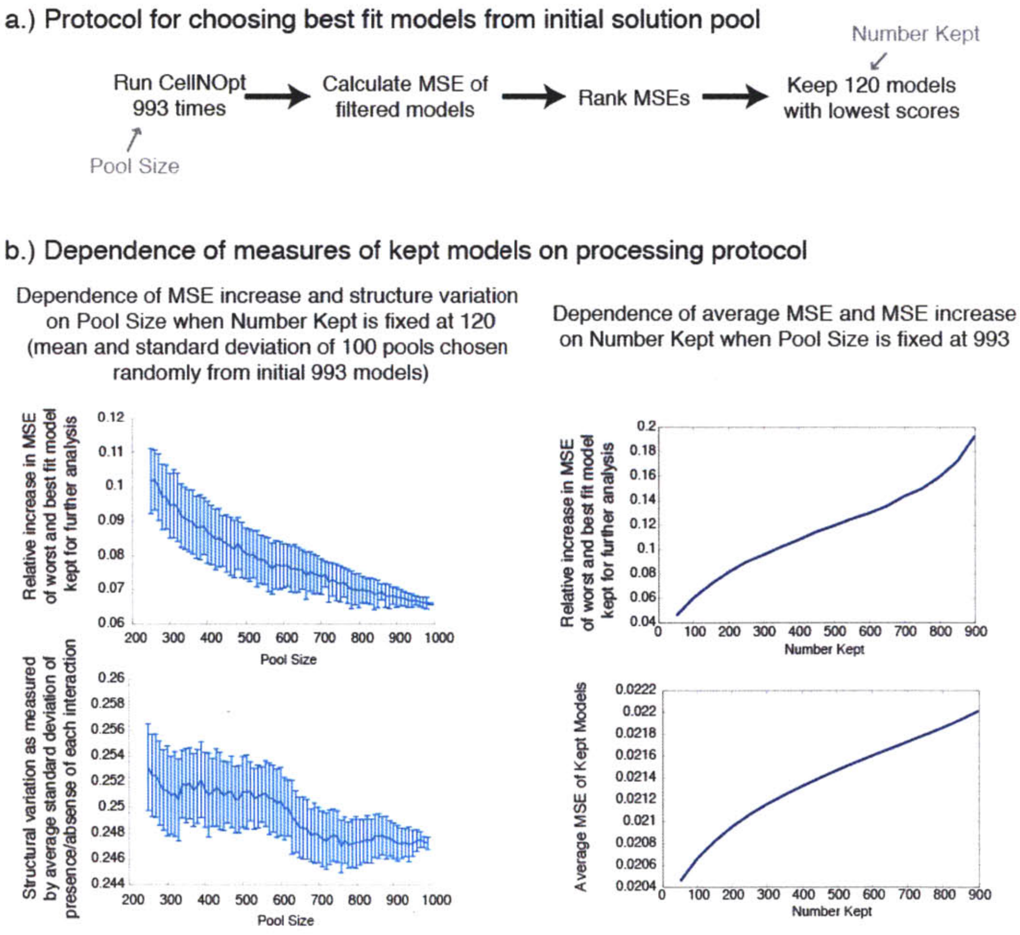


Figure D-6: Results of 10-fold cross validation in which the data was divided into ten random subsets and the optimization procedure performed to obtain a family of at least 350 models from training data comprising nine of the ten subsets; the remaining subset was considered a test set. The fit of these families of models to their respective training and test sets was then plotted as a function of the selection threshold. As expected, on average the ability of the trained models to fit the test sets was slightly worse than, but comparable to, the ability to fit the training sets ($R^2 = 0.79$ for the training set and $R^2 = 0.71$ for the test set), providing a first indication that the models were predictive.

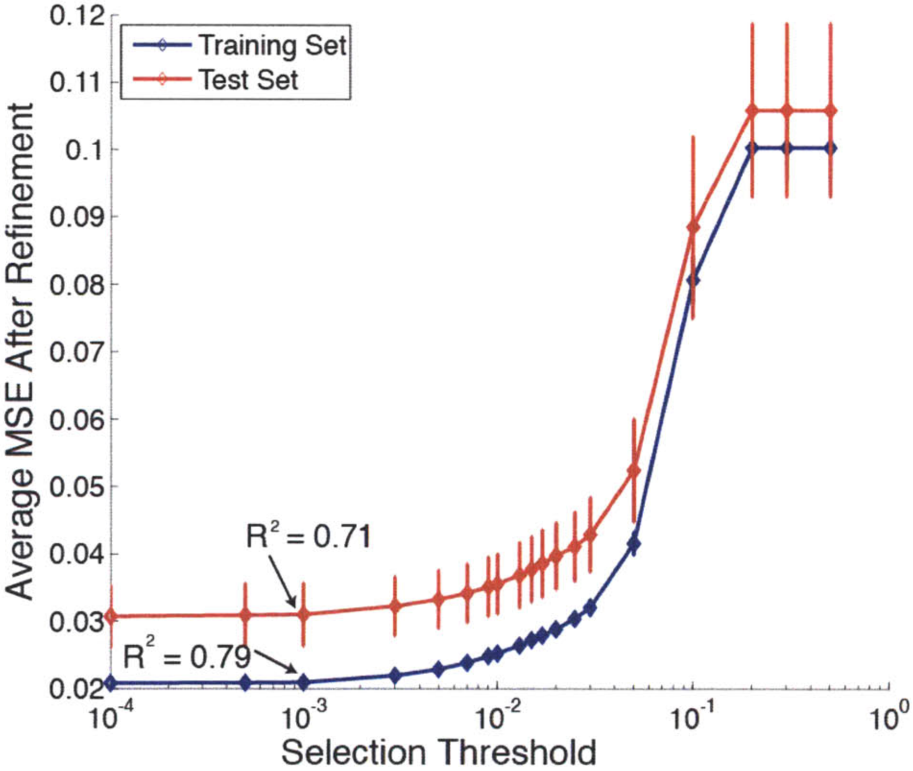


Figure D-7: Experimental data of phosphorylation of protein signals 30 minutes post stimulation. Relative fold change of signal post stimulation compared to basal value. Relative fold change has been scaled such that the maximum for each signal is one. Yellow and blue coloring scales according to magnitude of increase or decrease, respectively.

.

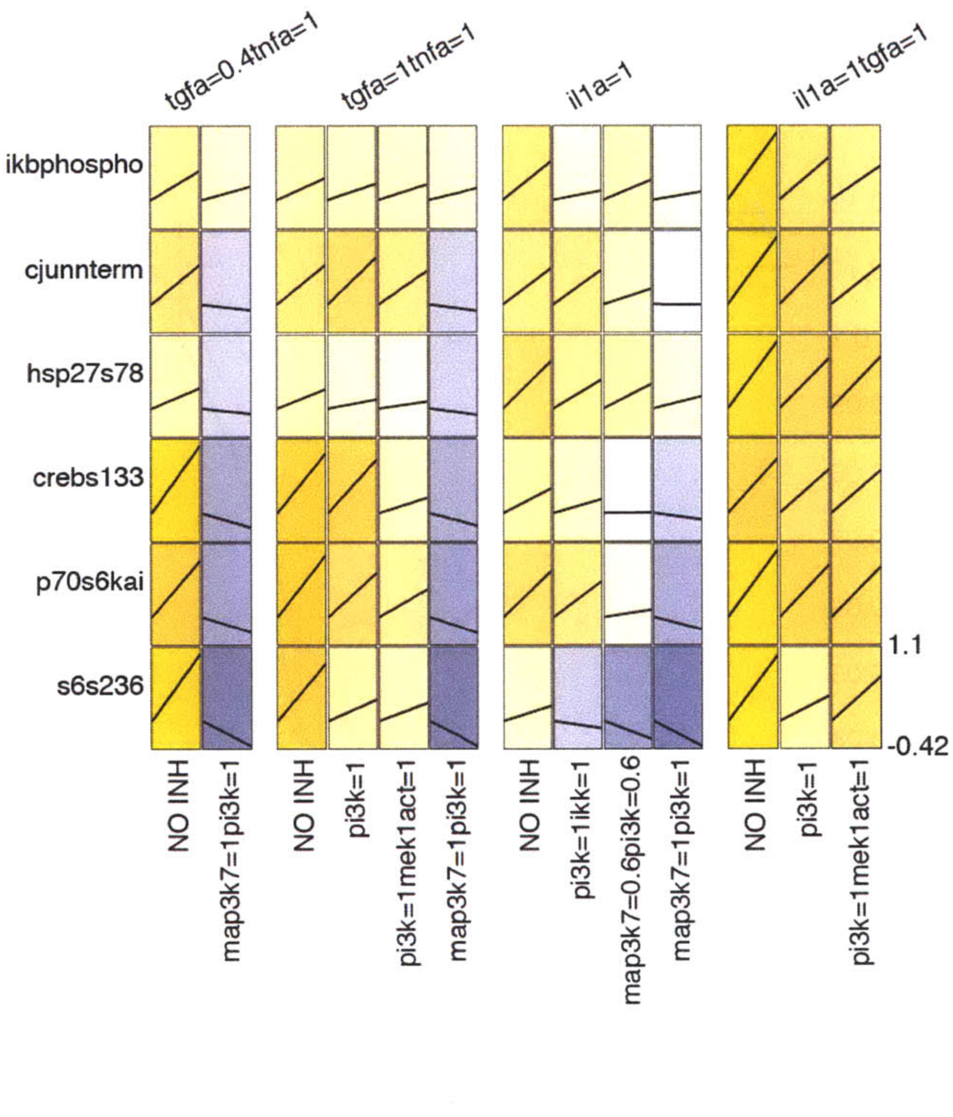


Figure D-8: Trained Models' Structures clustered based on Hamming distance

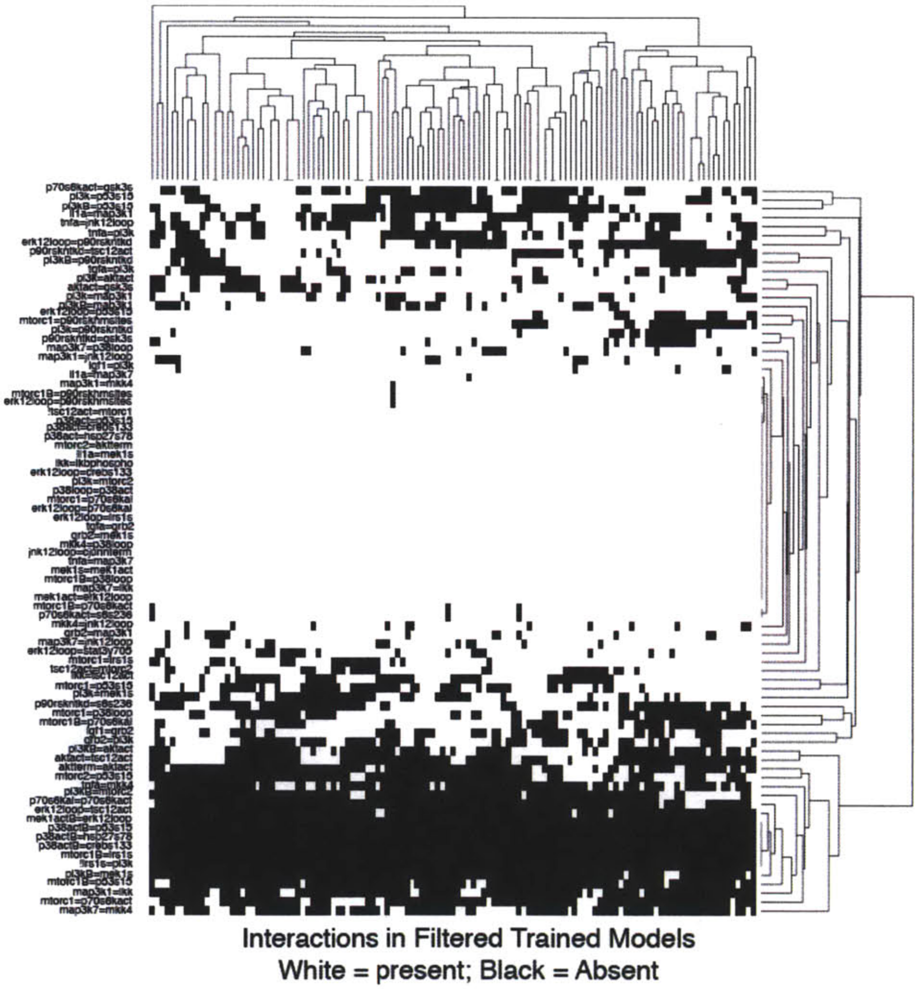


Figure D-9: Correlation between interaction presence/absense in trained filtered Models' Structures. NaNs have been replace with zero for visualization.

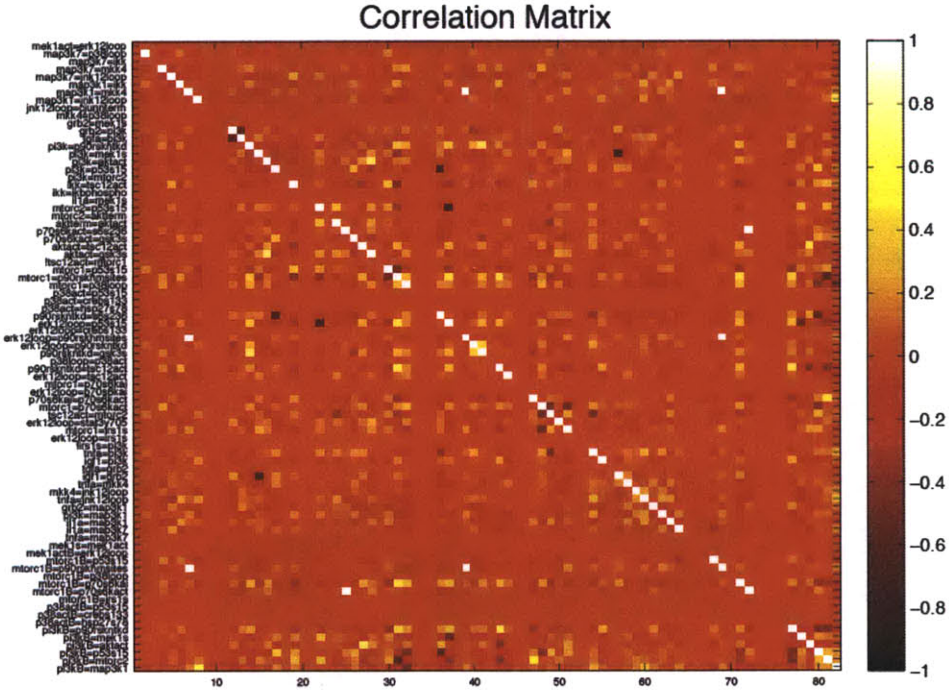


Figure D-10 *(facing page)*: (a) All metrics for evaluating how well constrained models were. (b) Examples of 'good' and 'bad' metrics. Metrics are distributions indicating either how constrained the model topology (as measured by the average of the bitstings cross models for both *unprocessed* and *filtered* models), parameters (as measured by inner quartile range (iqr) of summary 'sensitivity parameter', g, n, or EC_{50}), or predictions (as measured by iqr) were. Negligilbe differences were observed for distributions of the topology and while some of the iqr of the individual parameters were different, few were statistically significant as evaluated by a Mann-Whitney test. For the predictions, all were statistically significantly different. Thus, we conclude that any additional data over that of single ligand doses was not helpful in further constraining the models in terms of topology and parameters, although having at least ligand doses was helpful in constraining the predictions.

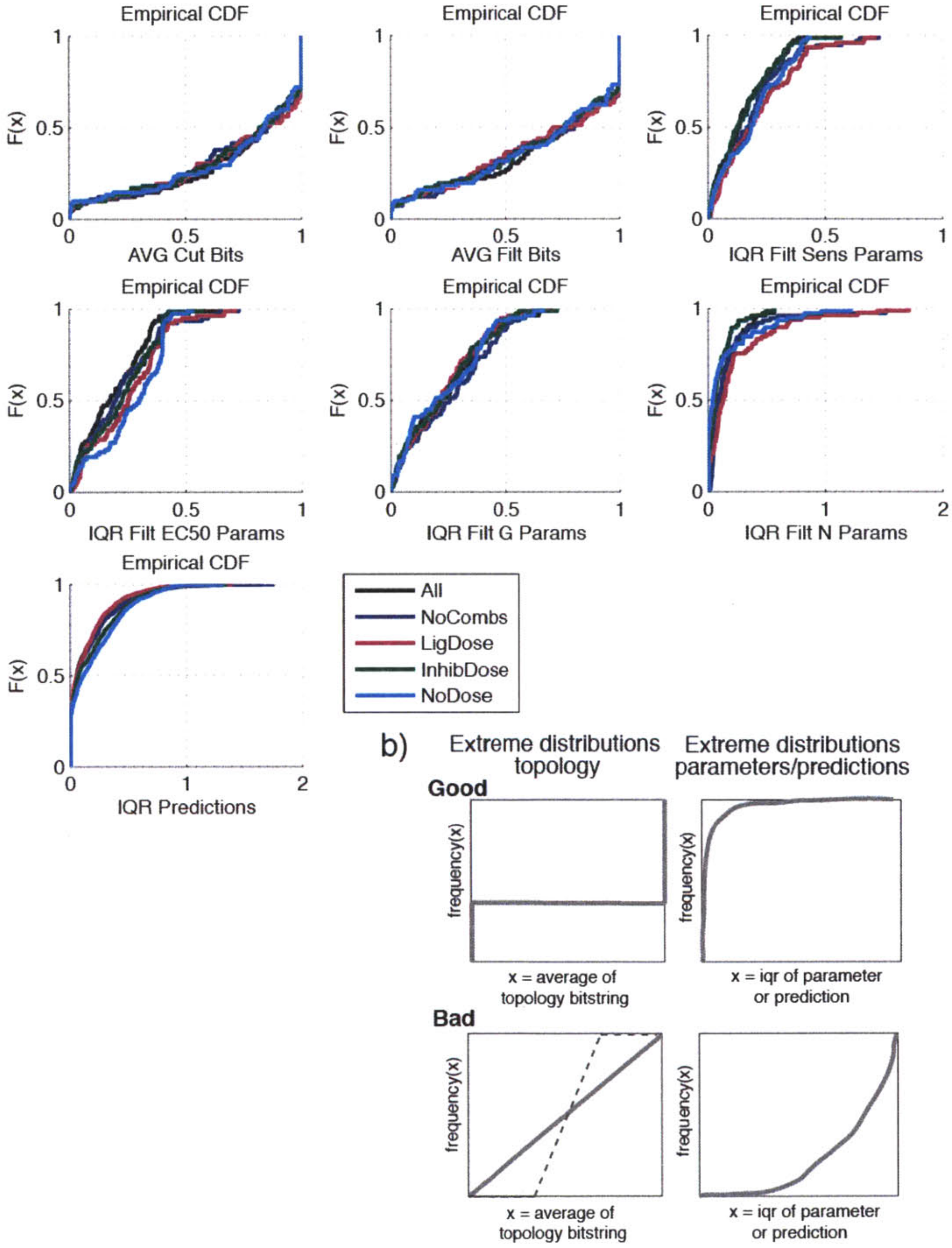

Empirical CDF
F(x)
AVG Cut Bits
Empirical CDF
F(x)
AVG Filt Bits
Empirical CDF
F(x)
IQR Filt Sens Params
Empirical CDF
F(x)
IQR Filt EC50 Params
Empirical CDF
F(x)
IQR Filt G Params
Empirical CDF
F(x)
IQR Filt N Params
Empirical CDF
F(x)
IQR Predictions
0
0.5
1
2
All
NoCombs
LigDose
InhibDose
NoDose
b)
Extreme distributions topology
Extreme distributions parameters/predictions
Good
frequency(x)
x = average of topology bitstring
x = iqr of parameter or prediction
Bad
frequency(x)
x = average of topology bitstring
x = iqr of parameter or prediction

Figure D-11: Original Training Conditions Necessary to Constrain Models - In Silico Results

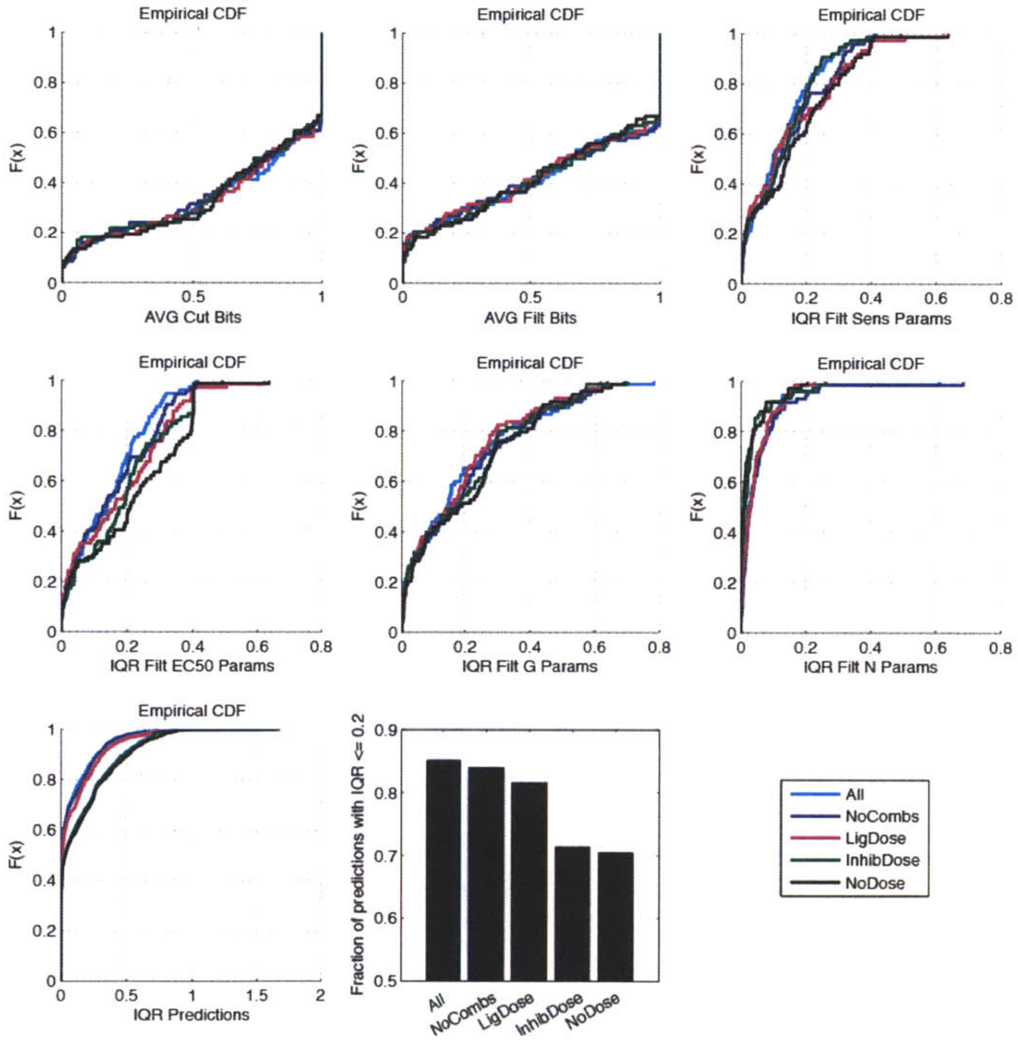


Figure D-12: Influence of Model Pair Threshold Parameter on Experimental Design

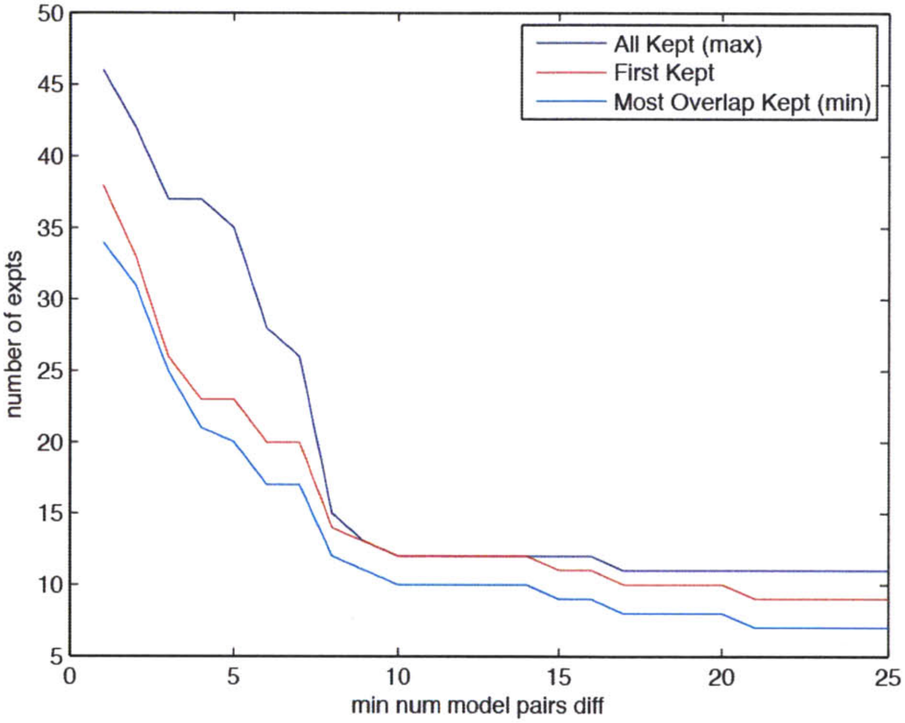


Figure D-13: Experimental Design Validation Data. Because the 'Booleanizer' method was not general enough for this data set, relative fold change of signal post stimulation compared to basal value was scaled by dividing each signal by an empirically determined value representing the average of the maximum and minimum increase of the signal under 'maximally' stimulating conditions (TGFα=1 and IL1α=1 and any inhibition). Any value greater than one was set equal to one and these values used for subsequent model training. Yellow and blue coloring scales according to magnitude of increase or decrease, respectively.

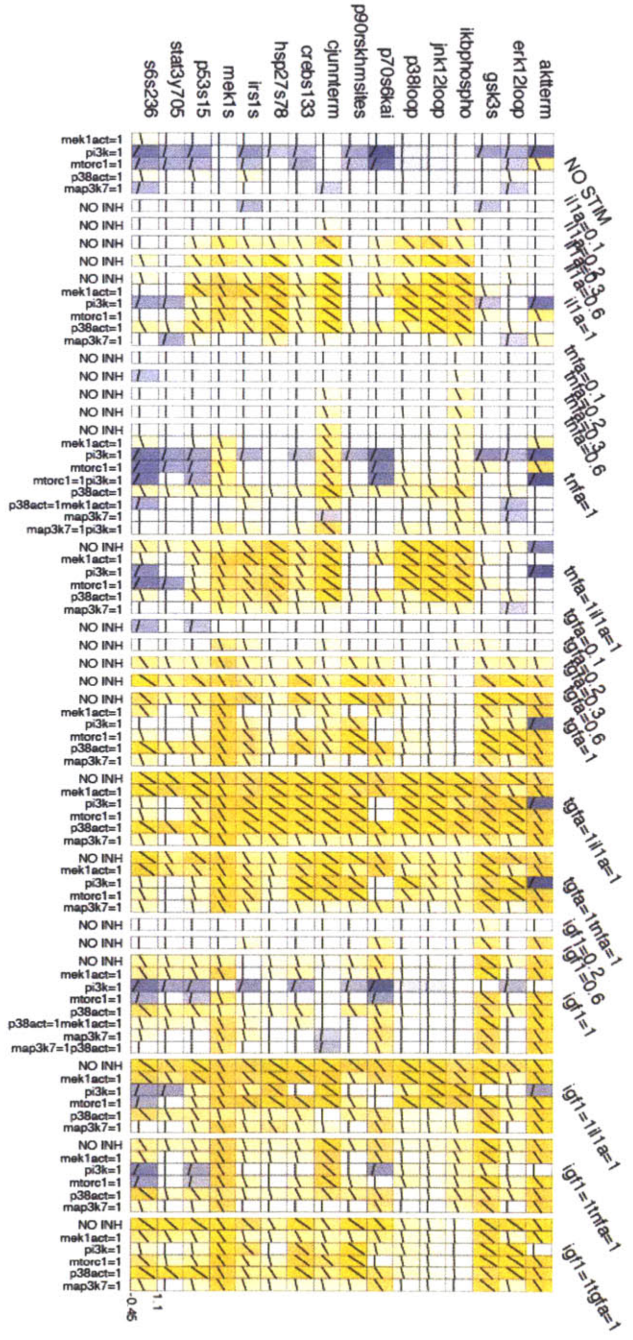
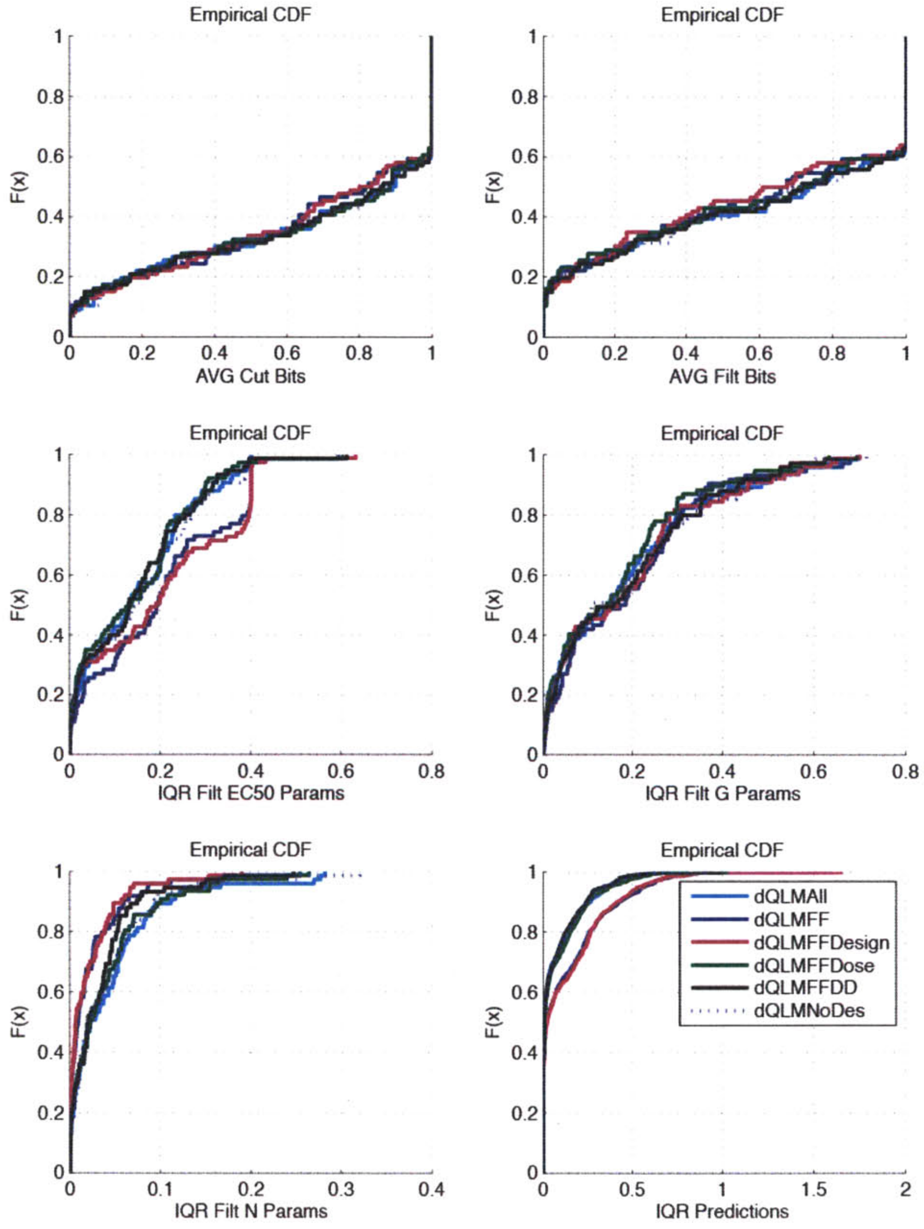


Figure D-14: All distributions (as in Supplementary Figure D-10) for *in silico* investigation of effectiveness of designed conditions. We found that models trained to the various subsets did not differ significantly in how constrained the model topologies were. A few subsets did differ in terms of how constrained predicted species values were, and these subsets seemed to also feature more constrained EC_{50} parameters but *less* constrained Hill coefficients.



D.2 Supplementary Tables

Table D.1: Hypotheses suggested by gates removed during CellNOpt-cFL analysis

Hypothesis	**Evidence in cFL Models**	**Evidence in data**
Map3k1 → Iκk crosstalk is inconsistent with the data.	Map3k1 → Iκk gate is present in only 13% of filtered models	cJun and not Iκb are phosphorylated upon growth factor stimulation. Removal of this crosstalk allows the two pathways to be decoupled.
TSC is mainly phosphorylated via by Iκk or p90RSK	Erk → TSC1/2 and Akt → TSC1/2 gates are present in few unprocessed models	IL1α stimulation results in Iκk activation, which then results in MTORC1-dependent phosphorylation of p70s6k (inhibition of Mek did not ablate IL1α-induced phosphorylation). Activation of TSC1/2 through p90RSK allows for independent modulation of this pathway through PI3K
Inhibition of Mek and p38 did not affect basal phosphorylation levels	Mek1actB and p38actB were not connected to downstream species	Few data values are negative upon Mek and p38 inhibition
Inhibition of MTORC1 and PI3K affected basal phosphorylation levels of several signals	mtorc1B and pi3kB were connected to several downstream species in many models	Several signals have negative values upon MTORC1 and PI3K inhibition

Table D.2: Combination conditions predicted to differentiate between trained models (predicted to distinguish between 5116 of 5151 model pairs)

Stimuli	Inhibitors
IGF1 = 0.6	Map3k7 + p38
IGF1 = 0.6	MTORC1 + Mek
IGF1 = 0.8	Map3k7 + PI3K
IGF1 = 0.8	PI3K
IGF1 = 1	p38
TGFα = 0.2	PI3K + Mek
TGFα = 0.4	PI3K + p38
TGFα = 0.8	PI3K + p38
TGFα = 0.4 + IL1α = 0.4	p38

Table D.3: Combination conditions predicted to distinguish between trained models where only full stimulation and inhibition conditions were considered (predicted to distinguish between 5041 of 5151 model pairs).

Stimuli	Inhibitors
IL1α	Map3k7
IL1α	PI3K + Mek
TNFα	Map3k7 + Mek
IGF1	MTORC1 + Mek
IGF1	PI3K + Mek
TGFα	Map3k7 + PI3K
TGFα	p38 + PI3K
TGFα	MTORC1 + PI3K

Appendix E

Constrained fuzzy logic to link signaling and transcriptional regulation

E.1 Motivation

This thesis focussed on using cFL to model activation of signal transduction networks. These signal transduction networks result in phenotypic changes through interactions with cytoskelatal proteins, secretion and cleavage of autocrine and paracrine cytokines, and activation of transcription factors. Transcription factors form complexes that bind to DNA, resulting in alteration in transcribed genes. The transcribed genes are then translated into proteins, which act to change cellular phenotype. In this appendix, we present a proof of principle study for the use of logic models to determine consistency between an expanded prior knowledge network linking signaling proteins to the transcription factors hypothesized to be responsible for a measured transcriptional response and condition-specific data describing activation of signaling proteins and hypothesized transcription factors.

E.2 Data

HepG2 cells were stimulated with various inflammatory and growth ligands and prosphorylation of several downstream signaling proteins measured by BioPlex after 30 minutes (Figure E-1). Additionally, gene expression after stimulation with the same ligands for 4 hours was measured with mRNA-seq (Figure E-2).

Figure E-1: Signaling Data

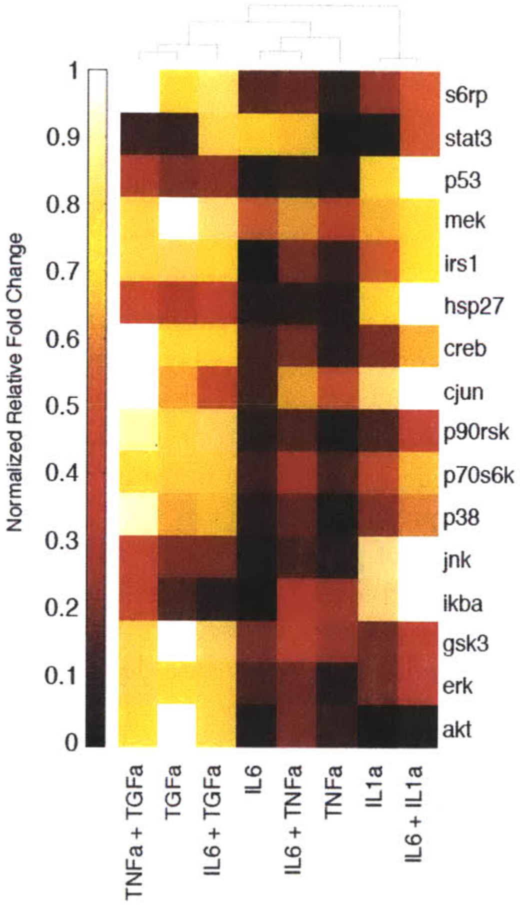
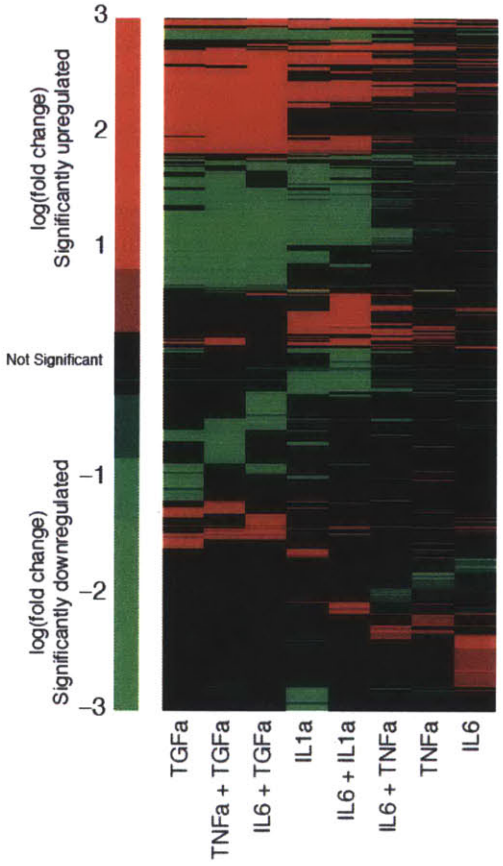


Figure E-2: mRNA-Seq Data



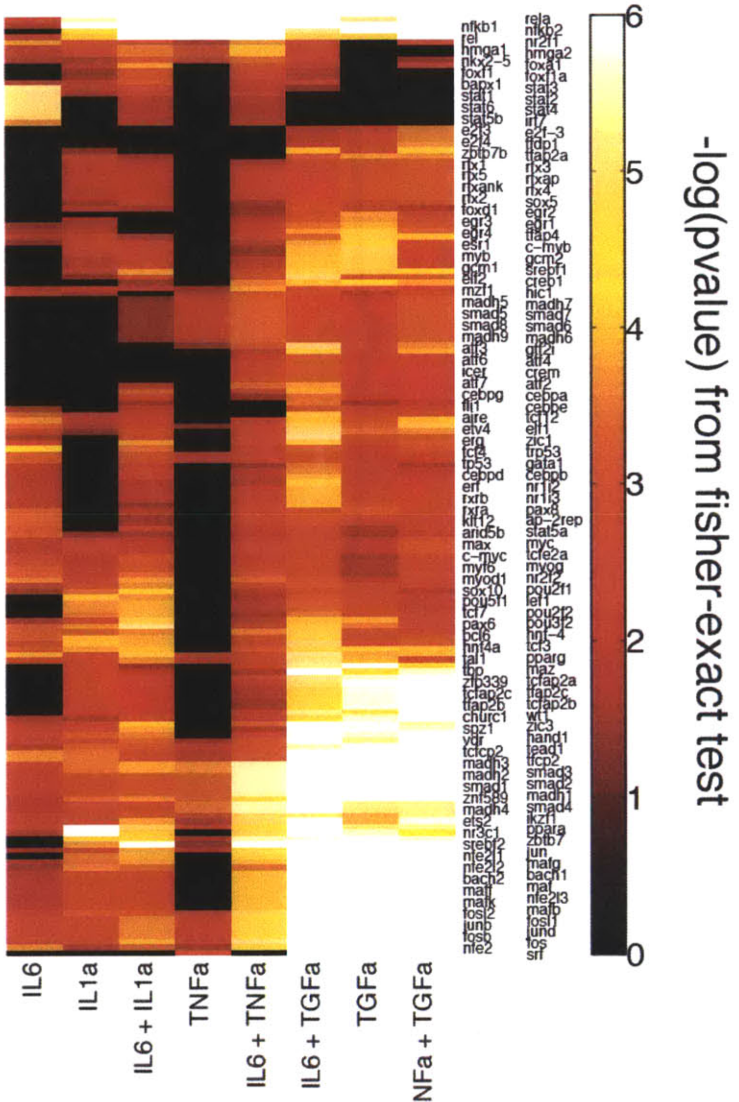
E.3 Results

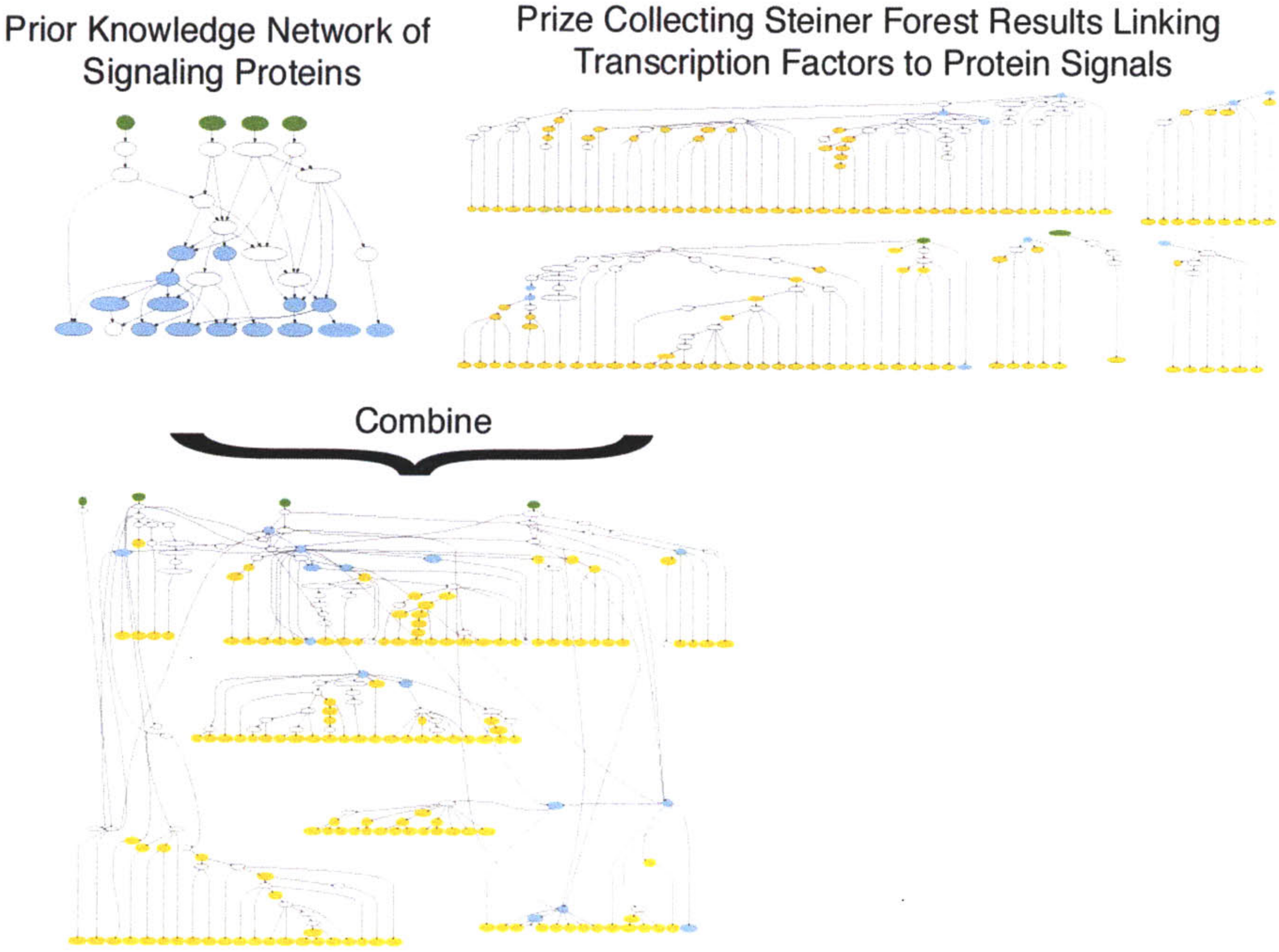
For each stimulation condition, transcription factors (TFs) responsible for the observed gene expression changes were hypothesized by calculating over-representation of predicted transcription factor targets in the differentially expressed genes (Figure E-3) with a Fisher Exact test and Benjamini-Hochberg multiple hypothesis correction. Predicted TF targets were obtained by determining if the gene contained the TF binding motif in DNase hypersensitive regions 5kb upstream and 1kb downstream the transcription start site. DNase hypersensitivity data was obtained from ENCODE for un-stimulated HepG2 cells [22]. TF binding motifs were obtained from TRANSFAC [90, 91].

In order to link the hypothesized transcription factors to the measured protein signals, the Prize Collecting Steiner Forest (PCSF) algorithm developed in the Fraenkel lab was used [54, 150]. Briefly, this algorithm seeks to connect detected proteins with edges from a protein protein interaction network by balancing the penalty of excluding nodes with the cost of including additional edges. In this work, we used the log-2 transformed score of the edge in the STRING protein-protein interaction network as edge costs [142]. We used the maximum across conditions, log-2 transformed multiple-hypothesis corrected p-value from the Fisher Exact test as node penalties for the transcription factors. The root ('dummy') node was connected to measured signals and stimulated receptors such that the algorithm sought to create the most parsimonious network possible that connected TFs to measured signals or stimulated receptors.

The PCSF result was combined with our PKN of the signaling network (Figure E-4) and the resulting enhanced PKN trained to the data of signaling response and transcription factors hypothesized to be responsible for the observed gene expression changes (Figure E-5).

Figure E-3: Hypothesized Transcription Factors




Prior Knowledge Network of Signaling Proteins

Prize Collecting Steiner Forest Results Linking Transcription Factors to Protein Signals

Figure E-4: Enhanced Prior Knowledge Network

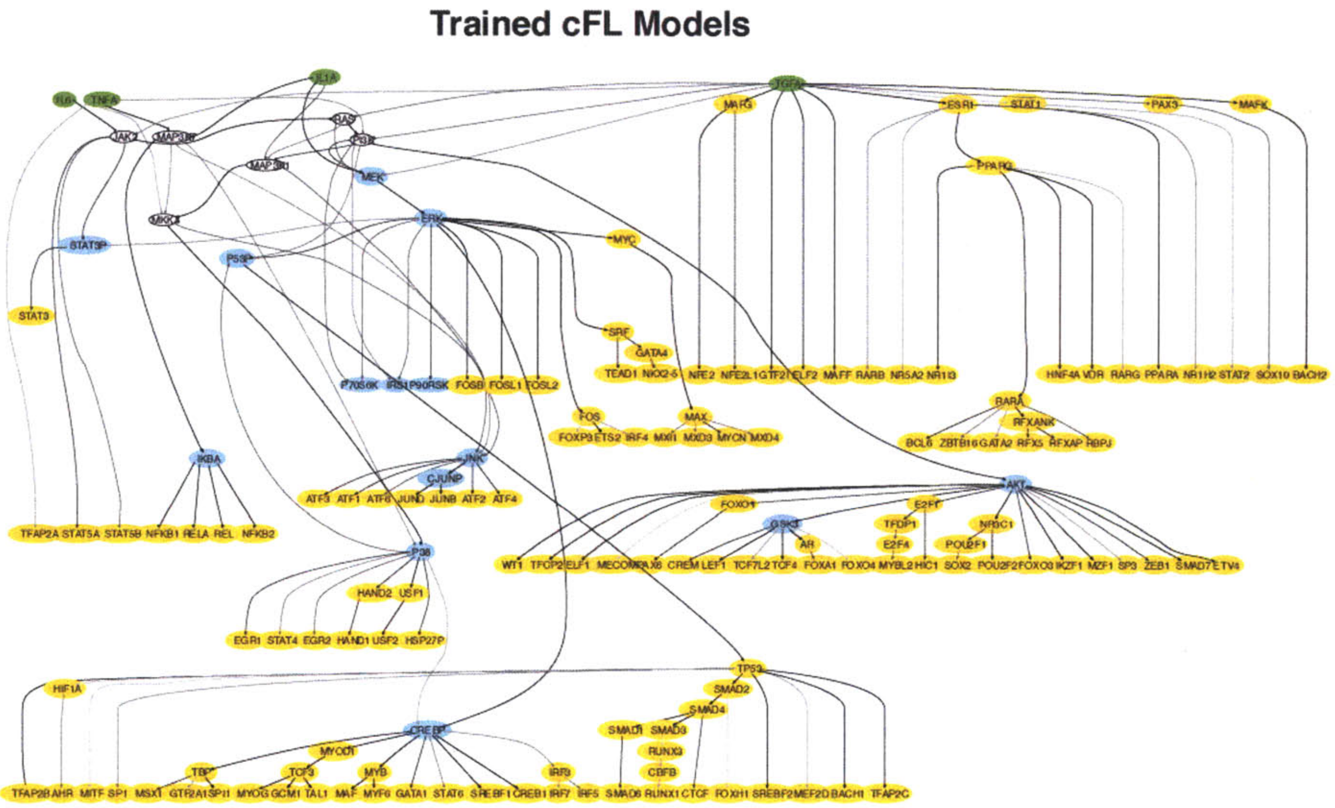

Trained cFL Models

Figure E-5: Trained cFL models linking signaling and transcriptional regulation

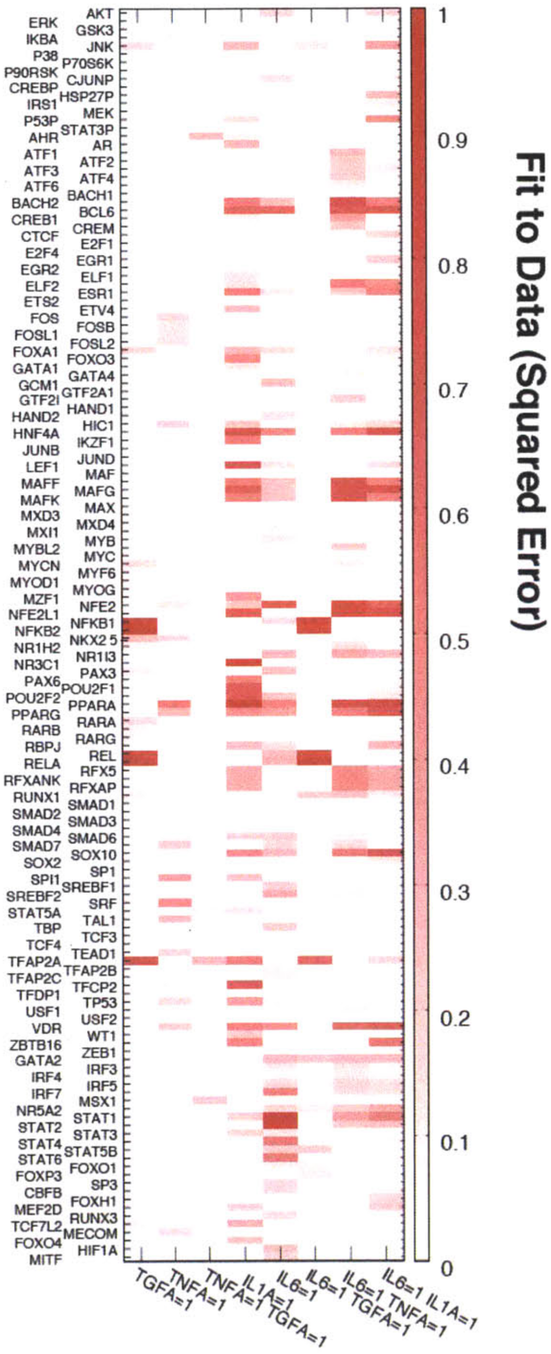
The results of the cFL training analysis indicate that, although most edges in our enhanced PKN were not inconsistent with the data (i.e. most edges are either grey or back in trained cFL models; Figure E-5), they were insufficient to describe the data (i.e. the fit to the data indicates much systematic error; Figure E-6).

However, by examining the fit the data more closely (Figure E-6), we are better able to determine what condition-specific overrepresentation of transcription factor targets was consistent with our general picture. For example, the relatively low error in proteins that form AP1 dimers (i.e. jun, atf, creb, and fos) indicates that they were activated by the expected pathways. The error in NFκB overrepresentation in conditions with TGFα stimulation indicates that either this was a false positive, or TGFα activated the NFκB pathway through mechanisms not included in our prior knowledge network. Finally, the improved fit despite similar overrepresentation patterns of the Bach1 and Elf1 transcription factors over Bach2 and Elf2 indicate that the placement of Bach1 and Elf1 in the network was more consistent with condition-specific data than that of Bach2 and Elf2.

The analysis presented here suffered from an apparent lack of specificity in the TF target lists when calculating over-representation of TF targets in the differentially expressed genes such that many over-represented TFs were false positives. More specific lists could be obtained by collection of differential DNase hypersensitivity data or incorporation of additional data types and sources in the designation of a gene as a TF target. Alternatively, the calculation of overrepresentation could be circumvented by linking differentially expressed genes to the protein-protein interaction network through the TF target list and using the enhanced interaction network and gene expression data directly in the PCSF algorithm.

Thus, we conclude that cFL training of a prior knowledge network linking protein and transcriptional regulation is useful in systematically determining if condition specific hypotheses are consistent with our general picture, but further development is necessary to fully evaluate the ability of this methodology to provide additional insight to this type of analysis.

Figure E-6: Fit of cFL models linking signaling and transcriptional regulation



Bibliography

[1] A. Abdi, M.B. Tahoori, and E.S. Emamian. Fault diagnosis engineering of digital circuits can identify vulnerable molecules in complex cellular pathways. *Science Signaling*, 1(42):ra10, —2008—.

[2] B. B. Aggarwal, A. B. Kunnumakkara, K. B. Harikumar, S. R. Gupta, S. T. Tharakan, C. Koca, S. Dey, and B. Sung. Signal transducer and activator of transcription-3, inflammation, and cancer: how intimate is the relationship? *Ann N Y Acad Sci*, 1171:59–76, —2009—.

[3] B. Aldridge, J. Saez-Rodriguez, J. Muhlich, P. Sorger, and D. A. Lauffenburger. Fuzzy logic analysis of kinase pathway crosstalk intnf/egf/insulin-induced signaling. *PLoS Comput Biol*, 5(4):e1000340, —2009—.

[4] B. B. Aldridge, J. M. Burke, D. A. Lauffenburger, and P. K. Sorger. Physicochemical modelling of cell signalling pathways. *Nat Cell Biol*, 8(11):1195–203, —2006—.

[5] 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. *Mol Cell Proteomics*, 9(9):1849–65, —2010—.

[6] R. Anjum and J. Blenis. The rsk family of kinases: emerging roles in cellular signalling. *Nat Rev Mol Cell Biol*, 9(10):747–58, —2008—.

[7] KL Auer, J Contessa, S Brenz-Verca, L Pirola, S Rusconi, G Cooper, A Abo, MP Wymann, RJ Davis, M Dirrer, and P Dent. The ras/rac1/cdc42/sek/jnk/c-jun cascade is a key pathway by which aganists stimulate dna synthesis in primary cultures of rat hepatocytes. *Mol Biol Cell*, 9:561–573, —1998—.

[8] A. Bauer-Mehren, L. I. Furlong, and F. Sanz. Pathway databases and tools for their exploitation: benefits, current limitations and challenges. *Mol Syst Biol*, 5:290, —2009—.

[9] CP Blobel. Adams: Key components in egfr signaling and development. *Nat Rev Mol Cell Biol*, 6:32, —2005—.

[10] W.J. Bosl. Systems biology by the rules: hybrid intelligent systems for pathway modeling and discovery. *BMC Syst. Biol.*, 1(13), —2007—.

[11] S. Braunewell and S. Bornholdt. Superstability of the yeast cell-cycle dynamics: ensuring causality in the presence of biochemical stochasticity. *J. Theo. Bio.*, 245:638–643, —2007—.

[12] 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 Comput. Biol.*, 6(3):e1000702, —2010—.

[13] MS Carro, WK Lim, MJ Alvarez, RJ Bollo, X Zhao, EY Snyder, EP Sulman, SL Anne, F Doetsch, H Colman, A Lasorella, K Aldape, A Califano, and A Iavarone. The transcriptional network for mesenchymal transformation of brain tumours. *Nature*, 463(779):318–25, —2010—.

[14] C. Chaouiya, E. Remy, B. Mosse, and D. Thieffry. Qualitative analysis of regulatory graphs: A computations tool based on a discrete formal framework. pages 119–126, —2003—.

[15] M. Chaves, E. D. Sontag, and R. Albert. Methods of robustness analysis for boolean models of gene control networks. *IEEE Proc-Syst Bio*, 153:154–167, —2006—.

[16] J Chen, L Sam, Y Huang, Y Lee, J Li, Y Liu, HR Xing, and YA Lussier. Protein interaction network underpins concordan prognosis among heterogeneous breast cancer signatures. *J Biomed Inform*, 43(3):385–96, —2010—.

[17] W.W. Chen, B. Schoeberl, P.J. Jasper, M. Niepel, U.B. Nielsen, D.A. Lauffenburger, and P.K. Sorger. Input-output behavior of erbb signaling pathways as revealed by a mass action model trained against dynamic data. *Mol. Syst. Biol.*, 5(239), —2009—.

[18] J. Cheng, M. Cobb, and R. Baer. Phosphorylation of the tal1 oncoprotein by the extracellular-signal-regulated protein kinase erk1. *Molec. Cell Biol.*, 13:801–808, —1993—.

[19] Y. Chinenov and T. K. Kerppola. Close encounters of many kinds: Fos-jun interactions that mediate transcription regulatory specificity. *Oncogene*, 20(19):2438–52, —2001—.

[20] S. Chowbina, K. A. Janes, S. M. Peirce, and J. A. Papin. Mathematical and computational models in cancer. In Gioeli D., editor, *Targeted Therapies: Mechanisms of Resistance*, Molecular and Translational Medicine, pages 113–126. Humana Press, New York, NY, —2011—.

[21] D.C. Clarke, M.L. Brown, R.A. Erickson, Y. Shi, and X. Liu. Transforming growth factor β depletion is the primary determinant of smad signaling kinetics. *Molec. Cell. Biol.*, 29(9):2443–2455, —2009—.

[22] ENCODE Project Consortium, R.M. Myers, J. Stamatoyannopoulos, M. Snyder, I. Dunham, R.C. Hardison, B.E. Bernstein, T.R. Gingeras, W.J. Kent, E. Birney, and et al. A user's guide to the encyclopedia of dna elements (encode). *PLoS Biol.*, 9(4):e1001046, —2011—.

[23] O Cordon, F Herrera, and A Peregrin. Applicability of the fuzzy operators in the design of fuzzy logic controllers. *Fuzzy Sets Syst*, 86(1):15–41, —1997—.

[24] M.I. Davidich and S. Bornholdt. Boolean network model predicts cell cycle sequence of fission yeast. *PLoS One*, 3(2):e1672, —2008—.

[25] H. de Jong. Modeling and simulation of genetic regulatory systems: a literature review. *J Comput Biol*, 9(1):67–103, —2002—.

[26] H. de Jong, J. Geiselmann, C. Hernandez, and M. Page. Genetic network analyzer: Qualitative simulation of genetic regulatory networks. *Bioinformatics*, 19(3):336–344, —2003—.

[27] A DiCara, A. Garg, B. DeMicheli, I. Xenarios, and L. Mendoza. Dynamic simulation of regulatory networks using squad. *BMC Bioinformatics*, 26(8), —2007—.

[28] L Ding, G Getz, DA Wheeler, ER Mardis, MD McLellan, K Cibulskis, C Sougnez, H Greulich, DM Muzny, MB Morgan, L Fulton, RS Fulton, Q Zhang, MC Wendl, MS Lawrence, DE Larson, K Chen, DJ Dooling, A Sabo, AC Hawes, H Shen, SN Jhangiani, LR Lewis, O Hall, Y Zhu, T Mathew, Y Ren, J Yao, SE Scherer, K Clerc, GA Metcalf, B Ng, A Milosavljevic, ML Gonzalez-Garay, JR Osborne, R Meyer, X Shi, Y Tang, DC Koboldt, L Lin, R Abbott, TL Miner, C Pohl, G Fewell, C Haipek, H Schmidt, BH Dunford-Shore, A Kraja, SD Crosby, CS Sawyer, T Vickery, S Sander, J Robinson, W Winckler, J Baldwin, LR Chirieac, A Dutt, T Fennell, M Hanna, BE Johnson, RC Onofrio, RK Thomas, G Tonon, BA Weir, X Zhao, L Ziaugra, MC Zody, T Giordano, MB Orringer, JA Roth, MA Spitz, II Wistuba, B Ozenberger, PJ Good, AC Chang, DG Beer, MA Watson, M Ladanyi, S Broderick, A Yoshizawa, WD Travis, W Pao, MA Province, GM Weinstock, HE Varmus, SB Gabriel, ES Lander, RA Gibbs, M Meyerson, and RK Wilson. Somatic mutations affect key pathways in lung adenocarcinoma. *Nature*, 455(7216):1069–75, —2008—.

[29] F. Eduati, J. De Las Rivas, B. Di Camillo, G. Toffolo, and J. Saez-Rodriguez. Integrating literature-constrained and data-driven inference of signalling networks. *submitted.*

[30] A Ergun, CA Lawrence, MA Kohanski, TA Brennan, and JJ Collins. A network biology approach to prostate cancer. *Molecular Systems Biology*, 3(82), —2007—.

[31] A. Faure, A. Naldi, C. Chaouiya, and D. Thieffry. Dynamical analysis of a generic boolean model for the control of the mammalian cell cycle. *Bioinformatics*, 22(14):e124–31, —2006—.

[32] J Fisher and T Henzinger. Executable cell biology. *Nature Biotechnology*, 25(11):1239–1249, —2007—.

[33] J. Fitzgerald and A. Lugovskoy. Rational engineering of antibody therapeutics targeting multiple oncogene pathways. *mAbs*, 3(3):299–309, —2011—.

[34] R. Franke, M. Muller, N. Wundrack, E. D. Gilles, S Klamt, T. Kahne, and M. Naumann. Host-pathogen systems biology: logical modeling of hepatocyte growth factor and helicobacter pylori induced c-met signal transduction. *BMC Syst Biol*, 2(4), —2008—.

[35] N. Friedman. Inferring cellular networks using probabilistic graphical models. *Science*, 303:799–805, —2004—.

[36] A. Garg, A. Di Cara, I. Xenarios, L. Mendoza, and G. De Micheli. Synchronous versus asynchronous modeling of gene regulatory networks. *Bioinformatics*, 24(17):1917–25, —2008—.

[37] I. Gat-Viks and R. Shamir. Refinement and expansion of signaling pathways: The osmotic response network in yeast. *Genome Research*, —2007—.

[38] S. Gaudet, K. A. Janes, J. G. Albeck, E. A. Pace, D. A. Lauffenburger, and P. K. Sorger. A compendium of signals and responses triggered by prodeath and prosurvival cytokines. *Mol Cell Proteomics*, 4(10):1569–90, —2005—.

[39] L. Glass and S.A. Kauffman. The logical analysis of continuous, non-linear biochemical control networks. *J. Theo. Bio.*, 39:103–129, —1973—.

[40] A.G. Gonzalez, A. Naldi, L. Sanchez, D. Thieffry, and C. Chaouiya. Ginsim: a software suite for the qualitative modelling, simulation, and analysis of regulatory networks. *BioSystems*, 84(2):91–100, —2006—.

[41] S. Gupta, S. S. Bisht, R. Kukreti, and S. Jain. Boolean network analysis of a neurotransmitter signaling pathway. *J Theor Biol*, —2007—.

[42] P Hajek. Metamathematics of fuzzy logic. —1998—.

[43] P Hajek. What is mathematical fuzzy logic. *Fuzzy Sets Syst*, 157(5):597–603, —2006—.

[44] PC Heinrich, I Behrmann, S Haan, HM Hermanns, G Muller-Newen, and F Schaper. Principles of interleukin (il)-6-type cytokine signalling and its regulation. *Biochem. J.*, 374:1–20, —2003—.

[45] R. Heinrich, B. G. Neel, and T. A. Rapoport. Mathematical models of protein kinase signal transduction. *Mol Cell*, 9(5):957–70, —2002—.

[46] A. Heinrichs, E. Kritikou, B. Pulverer, and M. Raftopoulou. Systems biology: a user's guide. —2006—.

[47] T. Helikar, J. Konvalina, J. Heidel, and J.A. Rogers. Emergent decistion-making in biological signal transduction networks. *Proc Natl Acad Sci U S A*, 105(6):1913–1918, —2008—.

[48] T. Helikar and J.A. Rogers. Chemchains: a platform for simulation and analysis of biochemical networks aimed to laboratory scientists. *BMC Systems Biology*, 3, —2009—.

[49] B. S. Hendriks, L. K. Opresko, H. S. Wiley, and D. Lauffenburger. Quantitative analysis of her2-mediated effects on her2 and epidermal growth factor receptor endocytosis: distribution of homo- and heterodimers depends on relative her2 levels. *J Biol Chem*, 278(26):23343–51, —2003—.

[50] J. Hess, P. Angel, and M. Schorpp-Kistner. Ap-1 subunits: quarrel and harmony among siblings. *J Cell Sci*, 117(Pt 25):5965–73, —2004—.

[51] CF Huang and JE Ferrell. Ultrasensitivity in the mitogen-activated protein kinase cascade. *Proc Natl Acad Sci U S A*, 93:10078–83, —1996—.

[52] J. Huang and B. D. Manning. A complex interplay between akt, tsc2 and the two mtor complexes. *Biochem Soc Trans*, 37(Pt 1):217–22, —2009—.

[53] S. Huang and D. E. Ingber. Shape-dependent control of cell growth, differentiation, and apoptosis: switching between attractors in cell regulatory networks. *Exp Cell Res*, 261(1):91–103, —2000—.

[54] S.S. Huang and E. Fraenkel. Integrating proteomic, transcriptional, and interactome data reveals hidden components of signaling and regulatory networks. *Sci. Signal.*, 2(81):ra40, —2009—.

[55] Z. Huang and J. Hahn. Fuzzy modeling of signal transduction networks. *Chemical Engineering Science*, 64:2044–2056, —2009—.

[56] P. J. Hunter, E. J. Crampin, and P. M. Nielsen. Bioinformatics, multiscale modeling and the iups physiome project. *Brief Bioinform*, 9(4):333–43, —2008—.

[57] T. Ideker and D. A. Lauffenburger. Building with a scaffold: emerging strategies for high- to low-level cellular modeling. *Trends Biotechnol*, 21(6):255–62, —2003—.

[58] K. A. Janes, J. G. Albeck, S. Gaudet, P. Sorger, D A Lauffenburger, and M.B. Yaffe. A systems model of signaling identifies a molecular basis set for cytokine-induced apoptosis. *Science*, 310(5754):1646–1653, —2005—.

[59] K. A. Janes and D A Lauffenburger. A biological approach to computational models of proteomic networks. *Current Opinion in Chemical Biology*, 10(1):73–80, —2006—.

[60] M. Johannessen, M. P. Delghandi, and U. Moens. What turns creb on? *Cell Signal*, 16(11):1211–27, —2004—.

[61] D. S. Jones, A. P. Silverman, and J. R. Cochran. Developing therapeutic proteins by engineering ligand-receptor interactions. *Trends Biotechnol*, 26(9):498–505, —2008—.

[62] R. S. Jope and G. V. Johnson. The glamour and gloom of glycogen synthase kinase-3. *Trends Biochem Sci*, 29(2):95–102, —2004—.

[63] C Jorgensen and R Linding. Simplistic pathways or complex networks? *Curr Opin Genet Dev*, 20(1):15–22, —2010—.

[64] B.A. Joughin, E. Cheung, R. Karuturi, J Saez-Rodriguez, D.A. Lauffenburger, and E.T. Liu. Cellular signaling networks. In E. Liu and D.A. Lauffenburger, editors, *Systems Biomedicine, Concepts and Perspective.* Elseveier, —2009—.

[65] K Kandasamy, SS Mohan, R Raju, S Keerthikumar, GSS Kumar, AK Venugopal, D Telikicherla, JD Navarro, S Mathivanan, C Pecquet, SK Gollapudi, SG Tattikota, S Mohan, H Padhukasahasram, Y Subbannayya, R Goel, HKC Jacob, J Zhong, R Sekhar, V Nanjappa, L Balakrishnan, R Subbaiah, YL Ramachandra, BA Rahiman, TSK Prasad, JX Lin, JCD Houtman, S Desiderio, JC Renauld, and SN Constantinescu. Netpath: a public resource of curated signal transduction pathways. *Genome Biol*, 11(1):R3, —2010—.

[66] Stuart A Kauffman. Metabolic stability and epigenesis in randomly constructed genetic nets. *J. theor. Biol.*, 22:437–467., —1969—.

[67] M. Kaufman, F. Andris, and O. Leo. A logical analysis of t cell activation and anergy. *Proc Natl Acad Sci U S A*, 96:3894–3899, —1999—.

[68] M. Kaufman, R. Urbain, and R. Thomas. Towards a logical analysis of the immune response. *Journal of Theoretical Biology*, 114(4):527–561, —1985—.

[69] H. A. Kestler, C. Wawra, B. Kracher, and M. Kuhl. Network modeling of signal transduction: establishing the global view. *Bioessays*, 30(11-12):1110–25, —2008—.

[70] S. Klamt, J. Saez-Rodriguez, and E. D. Gilles. Structural and functional analysis of cellular networks with cellnetanalyzer. *BMC Systems Biology, 1:2,* —2007—.

[71] S. Klamt, J. Saez-Rodriguez, J. Lindquist, L. Simeoni, and E. D. Gilles. A methodology for the structural and functional analysis of signaling and regulatory networks. *BMC Bioinformatics*, 7:56, —2006—.

[72] L.B. Kleiman, T. Maiwald, H. Conzelmann, D.A. Lauffenburger, and P.K. Sorger. Rapid phospho-turnover by receptor tyrosine kinases impacts downstream signaling and drug binding. *Mol Cell*, 43(5):723–37, —2011—.

[73] 2nd Klinke, D. J. Signal transduction networks in cancer: quantitative parameters influence network topology. *Cancer Res*, 70:1773–1782, —2010—.

[74] S. Kostenko and U. Moens. Heat shock protein 27 phosphorylation: kinases, phosphatases, functions and pathology. *Cell Mol Life Sci*, 66(20):3289–307, —2009—.

[75] A. Kremling and J. Saez-Rodriguez. Systems biology - an engineering perspective. *J. Biotechnol*, 129:329–51, —2007—.

[76] T. Kuwabara, S. Kobayashi, and Y. Sugiyama. Pharmacokinetics and phyamacodynamics of a recombinant human granulocyte colony-stimulating factor. *Drug Metabolism Reviews*, 28(4):625–658, —1996—.

[77] M Laakso and S Hautaniemi. Integrative platform to translate gene sets to networks. *Bioinformatics*, 26(14):1802, —2010—.

[78] A Lachmann and A Ma'ayan. Lists2networks: integrated analysis of gene/protein lists. *BMC Bioinformatics*, 11(87):87, —2010—.

[79] L. J. Lancashire, C. Lemetre, and G. R. Ball. An introduction to artificial neural networks in bioinformatics–application to complex microarray and mass spectrometry datasets in cancer studies. *Brief Bioinform*, 10:315–329, —2009—.

[80] D. A. Lauffenburger, E. M. Fallon, and J. M. Haugh. Scratching the (cell) surface: cytokine engineering for improved ligand/receptor trafficking dynamics. *Chem Biol*, 5(10):R257–63, —1998—.

[81] F. Li, T. Long, Y. Lu, Q. Ouyang, and C. Tang. The yeast cell-cycle network is robustly designed. *Proc Natl Acad Sci U S A*, 101(14):4781–4786, —2004—.

[82] S. Li, S.M. Assmann, and R. Albert. Predicting essential components of signal transduction networks: a dynamic model of guard cell abscisic acid signaling. *PLoS Biology*, 4(10):1732–1748, —2006—.

[83] J Lim, T Hao, C Shaw, AJ Patel, G Szabo, JF Gual, CJ Fisk, N Li, N Li, A Smolyar, DE Hill, AL Barabasi, M Vidal, and HY Zoghbi. A protein-protein interaction network for human inherited ataxias and disorder of purkinje cell degeneration. *Cell*, 125(4):801–14, —2006—.

[84] R Lu, F Markowetz, RD Unwin, JT Leek, EM Airoldi, BD MacArthur, A Lachmann, R Rozov, A Ma'ayan, LA Boyer, OG Troyanskaya, AD Whetton, and IR Lemischka. Systems-level dynamic analyses of of fate change in murine embryonic stem cells. *Nature*, 462:358–62, —2009—.

[85] A Ma'ayan. Network integration and graph analysis in mammalian molecular systems biology. *IET Systems Biology*, 2(5):206–221, —2008—.

[86] A. MacNamara, C. Terfve, D. Henriques, B.P. Bernabe, and J. Saez-Rodriguez. State-time spectrum of signal transduction logic models. *Submitted.*

[87] Z. Mai and H. Liu. Boolean network-based analysis of the apoptosis network: irreversible apoptosis and stable surviving. *J Theor Biol*, 259(4):760–9, —2009—.

[88] E. Mamdani. Application of fuzzy logic to approximate reasoning using linguistic synthesis. *Proc 6th Internat Symp Mult-Val Logic, IEEE*, 76CH1111-4C:196–202, —1976—.

[89] V Matthews, B Schuster, S Schutze, I Bussmeye, A Ludwig, C Hundhausen, T Sadowski, P Saftig, D Hartmann, KJ Kallen, and Rose-John S. Cellular cholesterol depletion triggers shedding of the human interleukin-6 receptor by adam10 and adam17 (tace). *J Biological Chemistry*, 278(40):38829–38839, —2003—.

[90] V Matys, E Fricke, R Geffers, E Gling, M Haubrock, R Hehl, K Hornischer, AE Kel, OV Kel-Margoulis, DU Kloos, S Land, B Lewicki-Potapov, H Michael, R Mnch, I Reuter, S Rotert, H Saxel, M Scheer, S Thiele, and E Wingender. Transfac: transciptional regulation, from patterns to profiles. *Nucleic Acids Res.*, 31:374–378, —2003—.

[91] V Matys, O Kel-Margoulis, E Fricke, I Liebich, S Land, A Barre-Dirrie, I Reuter, D Chekmenev, M Krull, K Hornischer, N Voss, P Stegmaier, B Lewicki-Potopov, H Saxel, A Kel, and E Wingender. Transfac and its modeule transcompel: transcriptional gene regulation in eukaryotes. *Nucleic Acids Res.*, 34:D108–D110, —2006—.

[92] D. W. Meek and C. W. Anderson. Posttranslational modification of p53: cooperative integrators of function. *Cold Spring Harb Perspect Biol*, 1(6):a000950, —2009—.

[93] L. Mendoza and Alvarez-Buylla. Dynamics of the genetic regulatory netowrk for arabidopsis thaliana flower morphogenesis. *J. Theo. Bio.*, 193:307–319, —1998—.

[94] L. Mendoza, D. Thieffry, and E.R. Alvarez-Buylla. Genetic control of flower morphogenesis in arabidopsis thaliana: a logical analysis. *Bioinformatics*, 15(7/8):593–606, —1999—.

[95] L. Mendoza and I. Xenarios. A method for the generation of standardized qualitative dynamical systems of regulatory networks. *Theoretical biology & medical modelling*, 3:13, —2006—.

[96] Luis Mendoza. A network model for the control and differentiation process in th cells. *BioSystems*, 84:101–114, —2006—.

[97] K. Miller-Jensen, K. A. Janes, J. S. Brugge, and D. A. Lauffenburger. Common effector processing mediates cell-specific responses to stimuli. *Nature*, 448(7153):604–8, —2007—.

[98] A. Mitsos, I. Melas, P. Siminelaki, A. Chairakai, J Saez-Rodriguez, and L. G. Alexopoulos. Identifying drug effects via pathway alteractions usingan interger linear programming optimization formulation on phosphoproteomic data. *PLoS Comp. Biol.*, 5(12), —2009—.

[99] A. Mitsos, I.N. Melas, M.K. Morris, J. Saez-Rodriguez, D.A. Lauffenburger, and L.G. Alexopoulos. Non linear programming (nlp) formulation for quantitative modeling of protein signal transduction pathways. *Submitted.*

[100] M. A. Moore and D. J. Warren. Synergy of interleukin 1 and granulocyte colony-stimulating factor: *in vivo* stimulation of stem-cell recovery and hematopoietic regeneration following 5-fluorouracil treatment of mice. *Proc Natl Acad Sci U S A*, 84:7134–7138, —1987—.

[101] M.K. Morris, I.N. Melas, and J. Saez-Rodriguez. Construction of cell type-specific logic models of signaling networks using cellnetoptimizer. In B. Reisfeld and A. Mayeno, editors, *Computational Toxicology*, Methods in Molecular Biology. Humana Press, New York, NY, —in press—.

[102] M.K. Morris, J. Saez-Rodriguez, D.C. Clarke, P.K. Sorger, and D.A. Lauffenburger. Training signaling pathway maps to biochemical data with constrained fuzzy logic: quantitative analysis of liver cell responses to inflammatory stimuli. *PLoS Comput Biol*, 7(3):e1001099, —2011—.

[103] M.K. Morris, J. Saez-Rodriguez, P.K. Sorger, and D.A. Lauffenburger. Logic-based models for the analysis of cell signaling networks. *Biochemistry*, 49(15):3216–24, —2010—.

[104] M.K. Morris, Z. Shriver, R. Sasisekharan, and D.A. Lauffenburger. Querying quantitative logic models (q2lm) to study intracellular signaling networks and cell-cytokine interactions. *Biotechnol J.*, 7(3):374–386, —2012—.

[105] W Nickel and C Rabouille. Mechanisms of regulated unconventional protein secretion. *Nat Rev Mol Cell Biol*, 10:148, —2009—.

[106] J. Ninomiya, T. Kajino, K. Ono, T. Ohtomo, M. Matsumoto, M. Shiina, M. Mihara, M. Tsuchiya, and K. Masumoto. A resorcylic acid lacton, 5z-7-oxozeaenol, prevents inflammation by inhibiting the catalytic activity of tak1 mapk kinase kinase. *J. Biol. Chem.*, 278(20):18485–18490, —2003—.

[107] V Novak. Which logic is the real fuzzy logic? *Fuzzy Sets Syst*, 157(5):635–641, —2006—.

[108] K. Oda, Y. Matsuoka, A. Funahashi, and H. Kitano. A comprehensive pathway map of epidermal growth factor receptor signaling. *Mol Syst Biol*, 1:2005 0010, —2005—.

[109] K. A. Orlova and P. B. Crino. The tuberous sclerosis complex. *Ann N Y Acad Sci*, 1184:87–105, —2009—.

[110] A. Palamarchuk, A. Efanov, V. Maximov, R. Aquilan, C. Croce, and Y. Pekarsky. Akt phosphorylates tal1 oncoprotein and inhibits its repressor activity. *Cancer Res.*, 65:4515–4519, —2005—.

[111] M Pardo, B Lang, L Yu, H Prosser, A Bradley, MM Babu, and J Choudhary. An expanded oct4 interaction network: implications for stem cell biology, development, and disease. *Cell Stem Cell*, 6(4):382–95, —2010—.

[112] DW Parsons, S Jones, X Zhang, JC Lin, RJ Leary, P. Angenendt, P Mankoo, H Carter, IM Siu, GL Gallia, M Olivi, R McLendon, BA Rasheed, S Keir, T Nikolskaya, Y Nikolsky, DA Buscam, H Tekleab, Diaz LA Jr, J Hartigan, DR Smith, RL Strausber, SK Marie, SM Shinjo, H Yan, GJ Riggins, DD Bigner, R Kearchin, N Papadopoulos, G Parmigiani, B Vogelstein, VE Velculescu, and KW Kinzler. An integrated genomic analysis of human glioblastoma multiforme. *Science*, 321(5897):1807–12, —2008—.

[113] WD Penny, KE Stephan, J Daunizeau, MJ Rosa, KJ Friston, TM Schofield, and AP Leff. Comparing families of dynamic causal models. *PLoS Comput. Biol.*, 6(3):e1000709, —2010—.

[114] A. S. Perelson, A. U. Neumann, M. Markowitz, J. M. Leonard, and D. D. Ho. Hiv-1 dynamics in vivo: virion clearance rate, infected cell life-span, and viral generation time. *Science*, 271(5255):1582–6, —1996—.

[115] W. P. Petros. Pharmacokinetics and administration of colony-stimulating factors. *Pharmacotherapy*, 12(2 Pt 2):32S–38S, —1992—.

[116] N. Philippi, D. Wlter, R. Schlatter, K. Ferreira, M. Ederer, O. Sawodny, J. Timmer, C. Borner, and T. Dandekar. Modleing system states in liver cells: Survival, apoptosis, and their modifications in response to viral infection. *BMC Systems Biology*, 3(97), —2009—.

[117] E. Pieroni, S. de la Fuente van Bentem, G. Mancosu, E. Capobianco, H. Hirt, and A. de la Fuente. Protein networking: insights into global functional organization of proteomes. *Proteomics*, 8(4):799–816, —2008—.

[118] A. Raj and A. van Oudenaarden. Nature, nurture, or chance: Stochastic gene expression and its consequences. *Cell*, 135(2):216–226, —2008—.

[119] S. H. Rhee, A. C. Keates, M. P. Moyer, and C. Pothoulakis. Mek is a key modulator for tlr5-induced interleukin-8 and mip3alpha gene expression in non-transformed human colonic epithelial cells. *J Biol Chem*, 279(24):25179–88, —2004—.

[120] S. Ross, R. Erickson, and O. Hemati, N.and MacDougald. Glycogen synthase kinase 3 is an insulin-regulated c/ebpalpha kinase. *Molec. Cell Biol.*, 19:8433–8441, —1999—.

[121] J. Rual, K. Venkatesan, T. Hao, T. Hirozane-Kishikawa, A. Dricot, N. Li, G. F. Berriz, F. D. Gibbons, M. Dreze, N. Ayivi-Guedehoussou, N. Klitgord, C. Simon, M. Boxem, S. Milstein, J. Rosenberg, D. S. Goldberg, L. V. Zhang, S. L. Wong, G. Franklin, S. Li, J. S. Albala, J. Lim, C. Fraughton, E. Llamosas, S. Cevik, C. Bex, P. Lamesch, R. S. Sikorski, J. Vandenhaute, H. Y. Zoghbi, A. Smolyar, S. Bosak, R. Sequerra, L. Doucette-Stamm, M. Cusick, D. Hill, F. P. Roth, and M. Vidal. Towards a proteome-scale map of the human protein-protein interaction network. *Nature*, 437(7062):1173–8, —2005—.

[122] A Rubartelli, F Cozzolino, M Talio, and R Sitia. A novel secretory pathway for interleukin-1b, a protein lacking a signal sequence. *EMBO J*, 9(5):1503–1510, —1990—.

[123] 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*, 308(5721):523–529, —2005—.

[124] 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*, 5:331, —2009—.

[125] J. Saez-Rodriguez, L.G. Alexopoulos, M. Zhang, M.K. Morris, D.A. Lauffenburger, and P.K. Sorger. Comparing signaling networks between normal and transformed hepatocytes using discrete logical models. *Cancer Res*, 71(16):5400–5411, —2011—.

[126] J. Saez-Rodriguez, A. Goldsipe, J. Muhlich, L. G. Alexopoulos, B. Millard, D. A. Lauffenburger, and P. K. Sorger. Flexible informatics for linking experimental data to mathematical models via datarail. *Bioinformatics*, 24(6):840–847, —2008—.

[127] J. Saez-Rodriguez, S. Mirschel, R. Hemenway, S. Klamt, E. D. Gilles, and M. Ginkel. Visual setup of logical models of signaling and regulatory networks with promot. *BMC Bioinformatics*, 7:506, —2006—.

[128] J. Saez-Rodriguez, L. Simeoni, J. A. Lindquist, R. Hemenway, U. Bommhardt, B. Arndt, U. U. Haus, R. Weismantel, E. D. Gilles, S. Klamt, and B. Schraven.

A logical model provides insights into t cell receptor signaling. *PLoS Comput Biol*, 3(8):e163, —2007—.

[129] O. Sahin, H. Frohlich, C. Lobke, U. Korf, S. Burmester, M. Majety, J. Mattern, I. Schupp, C. Chaouiya, D. Thieffry, A. Poustka, S. Wiemann, T. Beissbarth, and D. Arlt. Modeling erbb receptor-regulated g1/s transition to find novel targets for de novo trastuzumab resistance. *BMC Syst Biol*, 3:1, —2009—.

[130] C Salazar and T Hofer. Multisite protein phosphorylation - from molecular mechanism to kinetic models. *FEBS J*, 276:3177, —2009—.

[131] 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 Comp. Biol.*, 5(8):e1000438, —2009—.

[132] C. A. Sarkar and D. A. Lauffenburger. Cell-level pharmacokinetic model of granulocyte colony-stimulating factor: implications for ligand lifetime and potency in vivo. *Mol Pharmacol*, 63(1):147–58, —2003—.

[133] C. A. Sarkar, K. Lowenhaupt, T. Horan, T. C. Boone, B. Tidor, and D. A. Lauffenburger. Rational cytokine design for increased lifetime and enhanced potency using ph-activated "histidine switching". *Nat Biotechnol*, 20(9):908–13, —2002—.

[134] R. Schlatter, K. Schmich, I.A. Vizcarra, P. Scheurich, T. Sauter, C. Borner, M. Ederer, I. Merfort, and O. Sawodny. On/off and beyond - a boolean model of apoptosis. *PLoS Comp. Biol.*, 5(12), —2009—.

[135] SD Shapira, I. Gat-Viks, BOV Shum, A Dricot, MM de Grace, L Wu, PB Gupta, T Hao, SJ Silver, DE Root, DE Hille, A Regev, and N Hacohen. A physical and regulatory map of host-influenza interactions reveals pathways in h1n1 infection. *Cell*, 139:1255–1267, —2009—.

[136] S. Sheding, J.E. Media, and A. Nakeff. Influence of rhg-csf scheduling on megakaryocytopoietic recovery following 5-fluorouracil-induced hematotoxicity in splenectomized b6d2f1 mice. *Stem Cells*, 16:144–151, —1998—.

[137] I. Shmulevich, E. R. Dougherty, S. Kim, and W. Zhang. Probabilistic boolean networks: a rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, 18(2):261–74, —2002—.

[138] C. A. Sparks and D. A. Guertin. Targeting mtor: prospects for mtor complex 2 inhibitors in cancer therapy. *Oncogene*, 29(26):3733–44, —2010—.

[139] U Stelzl, U Worm, M Lalowski, C Haenig, FH Brembeck, H Goehler, M Stroedicke, M Zenkner, A Schoenherr, S Koeppen, J Timm, S Mintzlaff,

C Abraham, N Bock, S Kietzmann, A Goedde, E Toksoz, A Droege, S Drobitsch, B Korn, W Birchmeier, H Lehrach, and EE Wanker. A human protein-protein interaction network: a resource for annotating the proteome. *Cell*, 122(6):957–68, —2005—.

[140] M Sugeno and M Nishida. Fuzzy control of a model car. *Fuzzy Sets Syst*, 16:103–113, —1985—.

[141] T. Sunami, N. Byrne, R. E. Diehl, K. Funabashi, D. L. Hall, M. Ikuta, S. B. Patel, J. M. Shipman, R. F. Smith, I. Takahashi, J. Zugay-Murphy, Y. Iwasawa, K. J. Lumb, S. K. Munshi, and S. Sharma. Structural basis of human p70 ribosomal s6 kinase-1 regulation by activation loop phosphorylation. *J Biol Chem*, 285(7):4587–94, —2010—.

[142] D. Szklarczyk, A. Franceschini, M. Kuhn, M. Simonovic, A. Roth, P. Minguez, T. Doerks, M. Stark, J. Muller, P. Bork, L.J. Jensen, and C. von Mering. The string database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Res*, 39:D561–8, —2011—.

[143] RC Taylor, M Singhal, DX Daly, J Gilmore, WR Cannon, K Domico, AM White, DL Auberry, KJ Auberry, BS Hooker, G Hurst, JE McDermott, WH McDonald, DA Pelletier, D Schmoyer, and HS Wiley. An analysis pipeline for the inference of protein-protein interaction networks. *In J Data Mining Bioinform*, 3(4):409–30, —2009—.

[144] C. Terfve, A. Cokelaer, D. Henriques, A. MacNamara, M.K. Morris, D.A. Lauffenburger, and J. Saez-Rodriguez. Cellnoptr : a flexible pipeline to model protein signalling networks trained to data using various logic formalisms. *Submitted.*

[145] J. Thakar, M. Pilione, G. Kirimanjeswara, T.T. Harvill, and R. Albert. Modeling systems-level regulation of host immune responses. *PLoS Comp. Biol.*, 3(6):e109, —2007—.

[146] R. Thomas. Boolean formalization of genetic control circuits. *J Theor Biol*, 42(3):563–85, —1973—.

[147] R. Thomas and R. D'Ari. *Biological Feedback.* CRC Press, Boca Raton, —1990—.

[148] R. Thomas and M. Kaufman. Multistationarity, the basis of cell differentiation and memory. i. structural conditions of multistationarity and other nontrivial behavior. *Chaos (Woodbury, NY)*, 11(1):170–179, —2001—.

[149] RM Tong. A control engineering review of fuzzy systems. *Automatica*, 13(6):559–569, —1977—.

[150] N. Tuncbag, A. Braunstein, A. Pagnani, S.S. Huang, J. Chayes, C. Borgs, R. Zecchina, and E. Fraenkel. Simultaneous reconstruction of multiple signaling pathways via the prize-collecting steiner forest problem. In B. Chor, editor, *RECOMB 2012*, pages 287–301, Heidelburg, —2012—.

[151] I. Ulitsky, I. Gat-Viks, and R. Shamir. Metareg: A platform for modeling, analysis, and visualization of biological systems using large-scale experimental data. *Genome Biology*, 9(R1), —2008—.

[152] H. van Dam and M. Castellazzi. Distinct roles of jun : Fos and jun : Atf dimers in oncogenesis. *Oncogene*, 20(19):2453–64, —2001—.

[153] HB Verbruggen and PM Bruijn. Fuzzy control and conventional control: What is (and can be) the real contribution of fuzzy systems? *Fuzzy Sets Syst*, 90(2):151–160, —1997—.

[154] P. Vicini. Multiscale modeling in drug discovery and development: future opportunities and present challenges. *Clin Pharmacol Ther*, 88(1):126–9, —2010—.

[155] D Voet and JG Voet. *Biochemistry*, volume 1. John Wiley & Sons, Inc., 3 edition, —2004—.

[156] B Vogelstein and KW Kinzler. Cancer genes and the pathways they control. *Nature Medicine*, 10(8):789–99, —2004—.

[157] S Watterson, S Marshall, and P Ghazal. Logic models of pathway biology. *Drug Discov Today*, 13(9-10):447–456, —2008—.

[158] JK Westwick, C Weitzel, HL Leffert, and DA Brenner. Activation of jun kinase is an early event in hepatic regeneration. *J. Clin. Invest.*, 95:803–810, —1995—.

[159] D. Wittmann, J. Krumsiek, J. Saez-Rodriguez, D. A. Lauffenburger, S. Klamt, and F. Theis. From qualitative to quantitative modeling. *BMC Syst Biol*, 3:98, —2009—.

[160] P. J. Woolf, W. Prudhomme, L. Daheron, G. Q. Daley, and D. A. Lauffenburger. Bayesian analysis of signaling networks governing embryonic stem cell fate decisions. *Bioinformatics*, 21(6):741–53, —2005—.

[161] M. Wu, X. Yang, and C. Chan. A dynamic analysis of irs-pkr signaling in liver cells: A discrete modeling approach. *PLoS One*, 4(12), —2009—.

[162] Z Yi, M Luo, CA Carroll, ST Weintraub, and LJ Mandarino. Identification of phosphorylation sites in insulin receptor substrate-1 by hypothesis-driven high-performance liquid chromatography-electrospray ionization tandem mass spectrometry. *Anal Chem*, 77(17):5693–5699, —2005—.

[163] D. J. Yoon, C. T. Liu, D. S. Quinlan, P. M. Nafisi, and D. T. Kamei. Intracellular trafficking considerations in the development of natural ligand-drug molecular conjugates for cancer. *Ann Biomed Eng*, 39(4):1235–51, —2011—.

[164] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, —1965—.

[165] R. Zhang, M. Shah, J. Yang, S. Nyland, X. Liu, J. Yun, R. Albert, and T. Loughran. Network model of survival signaling in large granular lymphocyte leukemia. *Proc Natl Acad Sci U S A*, 105(42):16308, —2008—.

[166] C. F. Zheng and K. L. Guan. Activation of mek family kinases requires phosphorylation of two conserved ser/thr residues. *EMBO J*, 13(5):1123–31, —1994—.

[167] Y. Zick. Insulin resistance: a phosphorylation-based uncoupling of insulin signaling. *Trends Cell Biol*, 11(11):437–41, —2001—.

[168] R. Zielinski, P.F. Przyytycki, J. Zheng, D. Zhang, T.M. Przytycka, and J. Capala. The crosstalk between egf, igf, and insulin cell signaling pathways - computations and experimental analysis. *BMC Systems Biology*, 3(88), —2009—.