Identification of deregulated signaling networks based on gene expression data.

Ioannis N. Melas
European Bioinformatics Institute, UK

May 7, 2014

1 Introduction

Herein we present an approach, based on the Integer Linear Programming (ILP) formulation published in Mitsos et al. [3], for the identification of signaling networks that best fit a measured gene expression and/or proteomic signature. In more detail, assuming we are given (i) a Prior Knowledge Network (PKN) which represents the protein connectivity according to literature (e.g. from online pathway databases - KEGG, Reactome, Pathway Commons etc.), (ii) prior knowledge of transcription regulation (e.g. from Transfac), (iii) a gene expression dataset capturing how a cell/tissue type of interest responds on the gene expression level, and optionally (iv) a proteomic dataset showing protein expression/activation in the interrogated cell type; we identify subsets of the PKN and the regulon that most probably yielded the measured gene/protein signatures. At the basis of our approach lies the ILP formulation published by Mitsos et al. [3], modified at key points to negotiate the complexity of whole proteome networks and to also address the identification of optimal perturbation nodes in the PKN (i.e. the nodes in the network where the signaling process initiated yielding the measured gene expressions). As a case study we have computed cell-type specific models by training the Human Signaling Network (HSN) by Zaman et al. [6] to the GDSC basal gene expression data [5], and predicted drug response based on how central drug targets are to the inferred cell-type specific topologies.

Bellow we first discuss the mathematical basis of our approach and then, how it is employed to construct cell-type specific signalling models capable of predicting drug response. At the end of this report we discuss similarities and differences of our approach to others that tackle the same problem.

2 ILP formulation

2.1 Modeling signal transduction and edge removals - Basic definitions and core formulation

We assume that we are given a signaling network G defined as a set of reactions $i = 1, ..., n_r$ and a set of species (i.e. nodes) $j = 1, ..., n_s$. Each reaction i is an ordered pair of species of the form $S_i \rightarrow T_i$, where $S_i, T_i \in \{j = 1, ..., n_s\}$ are the source and target species respectively.

Moreover, the sign of i is denoted with $\sigma_i \in \{-1, 1\}$, distinguishing between activations ($\sigma_i = 1$) and inhibitions ($\sigma_i = -1$).

We also define a set of experiments $k = 1, ..., n_e$. Where in each experiment a set of species are perturbed $B_j^k \in \{0, 1\}$ and a set of species are measured $m_j^k \in \{0, 1\}$.

Moreover, we introduce the following variables: $x_j^k \in \{0, 1\}$; $j = 1, ..., n_s$; $k = 1, ..., n_e$ to denote the activation state of species j in experiment k. $z_i^k \in \{0, 1\}$; $i = 1, ..., n_r$; $k = 1, ..., n_e$ to denote the activity of reaction i in experiment k. And $y_i \in \{0, 1\}$ to model the removal of reaction i from G.

To model signal transduction from one species to the next we use the following rules:

1. A reaction i will take place in experiment k ($z_i^k = 1$) if it is present ($y_i = 1$) and the source node of the reaction i is active ($x_j^k = 1$; $j = S_i$).

2. A node j will be active in experiment k ($x_j^k = 1$) if (i) at least one of the activation reactions leading to j is active ($\exists i : \sigma_i = 1 \,\&\, T_i = j \,\&\, z_i^k = 1$) and none of the inhibition reactions leading to j are active ($\nexists i : \sigma_i = -1 \,\&\, T_i = j \,\&\, z_i^k = 1$); else if (ii) node j is perturbed directly ($B_j^k = 1$). In any other case node j will be inactive ($x_j^k = 0$).

Aim of the formulation is the optimal selection of reactions i and input nodes j via the y_i and B_j^k variables respectively, to construct a signaling network that perfectly fits the measured gene expressions, while minimizing its size. On this front the following constraints are introduced:

$$\sum |m_{j,k} - x_{j,k}| = 0;\ j = 1, ..., n_s; k = 1, ..., n_e \tag{1}$$

$$z_i^k = x_j^k + y_i - 1;\ j = S_i;\ i = 1, ..., n_r;\ k = 1, ..., n_e \tag{2}$$

$$x_j^k \leq 1 - z_i^k;\ j = T_i;\ \sigma_i = -1;\ i = 1, ..., n_r;\ k = 1, ..., n_e \tag{3}$$

$$x_j^k \geq z_i^k - \sum_{\substack{i'=1,...,nr: \\ j=Ti \& \sigma i' = -1}} z_i';\ j = T_i;\ \sigma_i = 1;\ i = 1, ..., n_r;\ k = 1, ..., n_e \tag{4}$$

$$x_j^k \leq \sum_{\substack{i=1,...,nr: \\ j=Ti \& \sigma i = 1}} z_i^k + B_j^k;\ j = 1, ..., n_s;\ k = 1, ..., n_e \tag{5}$$

Where, constraint (1) forces the resulting network to perfectly fit the measured gene expressions, and the rest of the constraints enforce rules *(i)* and *(ii)*.

Moreover, the following objective function is used to enforce the minimization of the resulting network.

$$min \quad \sum \alpha x_{j,k} + \sum \beta B_j^k \tag{6}$$

Where, α and β are user defined constants.

2.2 Removal of feedback loops from the PKN

Next, we address the removal of feedback loops from the PKN. Positive feedback loops break the causality of signal transduction, allowing signal flow to be generated without an external perturbation. For example, for node j to be active ($x_{j,k} = 1$), it either has to be directly perturbed $B_{j,k} = 1$, or be activated by an upstream reaction i, such that $j = T_i$ and $z_{i,k} = 1$. However, if n nodes form a positive cycle (a cycle where all reactions are positive), then one node

will be able to activate the next all the way around the cycle, without the need for an external perturbation (or an incoming interaction transitively connected to a perturbation). To counter this, we introduce variables $d_{j,k} \geq 0$ to represent the distance of node j from a perturbed node in experiment k. If node j is not connected to a perturbed node, then $d_{j,k} = 0$, else $d_{j,k} > 0$. For node j to be active, $d_{j,k} > 0$ has to hold true. If $d_{j,k} = 0$, then $x_{j,k} = 0$. The distance of node j has to be greater than all of its upstream nodes at least by one (to enforce that the distance grows the further away from the stimuli we move), unless, the upstream reactions are not active (i.e. $z_{i,k} = 0$). Finally, the distance of any given node cannot be bigger than the total number of reactions in the PKN. The above may be formulated using linear constraints in the following manner:

$$\begin{aligned} & x_{j,k} \leq d_{j,k} \\ & d_{Ti} \geq d_{Si} + 1 + (z_{i,k} - 1)M \\ & d_{j,k} \leq n_r \end{aligned} \tag{7}$$

Where M is a very large number. If there is an active cycle in the PKN, then constraints (7) will force the distance $d_{j,k}$ to grow indefinitely and the the last constraint will set the ILP infeasible. Thus, constraints (7) prohibit the algorithm from activating all reactions in a cycle at once.

2.3 Calculating the significance of individual nodes/reactions in the solution via a network flow algorithm

The objective function (6) together with constraints (1)-(5) and (7) form a Mixed Integer Program (MIP) that aims to identify minimum subsets of the PKN that perfectly fit a measured gene expression profile. Assuming we have computed a network G^{sol}, optimal solution of the above-mentioned MIP, we are now facing the problem of calculating the significance of every individual node/reaction in the solution.

In more detail, G^{sol} is a directed network that originates at the perturbed nodes $B_{j,k} = 1$ (as these were computed by the ILP algorithm), goes through several signaling layers and the layer of transcription factors, and terminates at the activated genes ($m_{j,k} = 1$). However, not all nodes and reactions play the same role in regulating gene expression. There might be pathways in G^{sol} that serve to fit only a few of the measured gene expressions and others that are essential for fitting a large portion of it. Thus, the problem of systematically calculating the significance of each node/reaction in G^{sol} arises. To this effect we employ a network flow algorithm. We introduce a new quantity $Fz_{i,k} \geq 0$ representing signal flow through reaction i. One unit of Fz is equivalent to the activation of a single gene. Thus,

$$Fz_{i,k} = 1 \,;\, \forall i : j = T_i \,\&\, m_{j,k} = 1 \tag{8}$$

Implying, that the signal flow Fz through a reaction that leads to an expressed gene equals to one. On the other hand, signal flow to other terminal nodes of the network equals to zero:

$$Fz_{i,k} = 0 \,;\, \forall i : j = T_i \text{ is terminal node } \,\&\, m_{j,k} \neq 1 \tag{9}$$

Where a node j is terminal if and only if a reaction exists where j is a target node, but no reactions exist where j is source ($\exists i : j = T_i \,\&\, \nexists i' : j \in S'_i$).

Moreover, we introduce a conservation law applied at the internal (not terminal) nodes of G^{sol} to force that incoming flow to a node j has to equal the outgoing flow:

$$\sum_{i \in IN_j} Fz_{i,k} = \sum_{i \in OUT_j} Fz_{i,k} \tag{10}$$

where, IN_j is the set of reactions that have j as a target: $\{i : j = T_i\}$, and OUT_j is the set of reactions that have j as a source: $\{i : j = S_i\}$.

Finally, we introduce an additional constraint to force that only nodes/reactions that are conserved in the solution may transduce flow.

$$Fz_{i,k} \leq M\, x \tag{11}$$

Where M is an arbitrarily big number.

Equations (8)-(11) consist a network flow problem that can be coupled to the MIP as defined by the objective function (6) and constraints (1)-(5) and (7) to complete the formulation. Solving it to optimality will yield both the optimized network topology G^{sol} and the signal flow $Fz_{i,k}$ through the conserved interactions. The signal flow $Fz_{i,k}$ will provide an estimate of the significance of each node/reaction in the solution in terms of how many gene expressions it facilitates.

2.4 Additions to the formulation: Integration of proteomic/phenotypic data

In the formulation of section 2 we illustrated how gene expression data can be leveraged to identify subsets of the PKN that may be functional in the interrogated cell type. The gene expression data shape the topology of the resulting network via equation (1), where we force the perfect fit of the data. However in this manner, data is only available at terminal nodes of the network (the expressed genes). In cases where other types of data are available, such as proteomic data, these can also be incorporated in the training process to improve model predictions.

For the following we assume a matching proteomic dataset $p_{j,k}$ is available (a dataset measured under the same experimental conditions as the gene expression dataset). Where, $p_{j,k} = 1$ if node j in the *phosphoproteomic* level is active/expressed in experiment k, $p_{j,k} = 0$ if node j in the *phosphoproteomic* level is NOT active/expressed in experiment k, and $p_{j,k} = NaN$ if we are not sure of the protein activity/expression of node j in experiment k. Then, we can add the following constraint to the formulation of section 2.

$$x_{j,k} = p_{j,k}\,; \quad j : p_{j,k} \neq NaN \tag{12}$$

In this way we force the algorithm to use nodes that are found to be expressed in the proteomic level (assuming that over-expressed nodes may play an important role in signal transduction). In similar fashion, if sequencing (SNP) or other data is available for the interrogated cell line, these can also be incorporated to e.g. prohibit the algorithm from using nodes that are deleted.

3 Case study: Identification of deregulated pathways based on basal gene expressions, and prediction of drug response in the GDSC cell lines

As a case study we address the construction of specific models for the GDSC cell lines and attempt to predict drug response based on how central drug targets are to the inferred cell-type specific topologies. As a PKN we use the network by Zaman et al. [6], and we train it to basal gene expression data by Yang et al. [5]. For predicting drug response and thus validating the predictive power of the computed cell-type specific models, we use the drug sensitivity data, also, by Yang et al. [5].

3.1 Construction of the PKN

As a prior knowledge of protein connectivity in the signaling level we use the Human Signaling Network (HSN) published by Zaman et al. [6]. The HSN includes 62,737 interactions and 6,291 nodes (average node degree 10), and its diameter equals to 12. Before further processing we remove physical (binding) interactions (a total of 21,259 interactions) since they lack directionality and sign. Moreover, we remove self feedback loops (feedback loops with only one node) and interactions of multiple signs between two nodes. For example if node A both activates and inhibits node B (via two independent reactions), then one of the two interactions is removed before further processing of the network. Subsequently, in an attempt to further prune the PKN and remove obscure reactions, we screen all interacting protein pairs against the canonical pathways collection of the MSigDB [4]. For every interaction (protein pair) we calculate a score equal to the number of instances where both proteins are included in the same canonical pathway, implying that proteins that are encountered often in the same canonical pathways have a more significant functional relationship. We set a threshold and conserve in the PKN only reactions that scored above that threshold. In the analysis that follows, we conserved reactions whose source and target proteins are encountered at least once in the same canonical pathway.

Finally, the HSN is merged with prior knowledge of transcription regulation as compiled by Iorio et al. in [2]. The regulon by Iorio et al. includes information on the target genes of 152 Transcription Factors (TFs). 110 of those are also present in the HSN. Transcriptomic information is available for a total of 5,714 genes. The union of the HSN with the regulon by Iorio et al. consists the PKN used for all the analysis presented herein. It includes a total of 14,075 interactions and 7,206 nodes. The procedure for constructing the PKN is illustrated in figure 1.

3.2 Compilation of the training dataset

The basal gene expressions of the GDSC cell lines were used as a training dataset. The normalised data (RMA normalised) were downloaded from `http://www.cancerrxgene.org` and the zscores were computed across all included cell lines. To discretize the dataset into ON/OFF states, a threshold was introduced at two standard deviations above which the corresponding gene expressions where considered active (ON). Gene expression values less than that are considered inactive (OFF).

3.3 Training cell-type specific models

The ILP formulation of section 2 is used to identify functional subsets of the PKN that best fit the basal gene expressions of the GDSC cell lines. The significance of every node/reaction in the solution is also calculated according to the methodology in subsection 2.3. The procedure is implemented as follows. First, for every cell line we calculate 200 optimal solutions (when these exist). These are network models that best fit the gene expression data of the corresponding cell line. An average signal flow through each node/reaction in the solution is also calculated as an estimate of its significance in the regulation of gene expression. Next, we repeat the process but after scrambling the data, to obtain an average topology (and corresponding signal flows) for the randomised dataset. Finally, we calculate the log ratio of the average signal flow in the solution versus that obtained by the randomised dataset. The log ratio represents

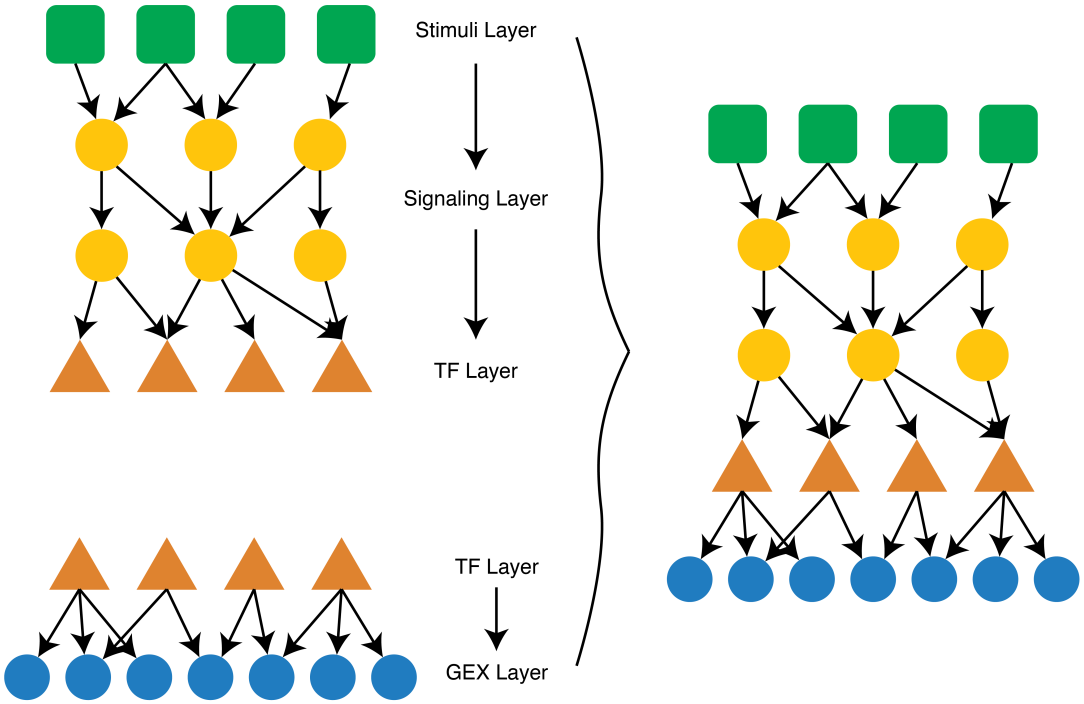

Figure 1: Construction of the PKN

how central an arbitrary node j is for regulating gene expression in that specific cell line. For example, assume signal flow through node A equals to 0.5 for a specific cell line, while signal flow through the same node A equals to 0.1 for a randomised dataset. The drastic increase of signal flow though A (as dictated by the gene expression profile of that cell line), compared to the randomised data, implies that node A plays a major role in regulating gene expression in the respective cell line. On the other hand, assume signal flow through an arbitrary node B is 0.01 for a specific cell line, while signal flow through that node equals to 0.1 for the randomised data. The drastic decrease in signal flow through B implies that this node plays very little role in regulating gene expression in that cell line.

Implementing the above procedure we compute subsets of the PKN that best fit the gene expression data for the interrogated cell line and also calculate the significance of every node/reaction in the PKN in regulating gene expression.

3.4 Predicting drug response

Drug response is predicted based on how significant drug targets are to the optimised, cell-type specific topologies. In more detail, the procedure described in the previous section led to the identification of cell-type specific signaling topologies that best fit the measured gene expression data, and also to the calculation of signal flow through the nodes/reactions in the PKN, providing an estimate of how significant they are in gene regulation. Here, we calculate a score for every drug in the drug sensitivity dataset that equals to the total amount of signal flow it is expected to disrupt based on the prior knowledge of its targets. For example consider Erlotinib, an EGFR inhibitor, and the optimised cell-type specific model for one of the GDSC

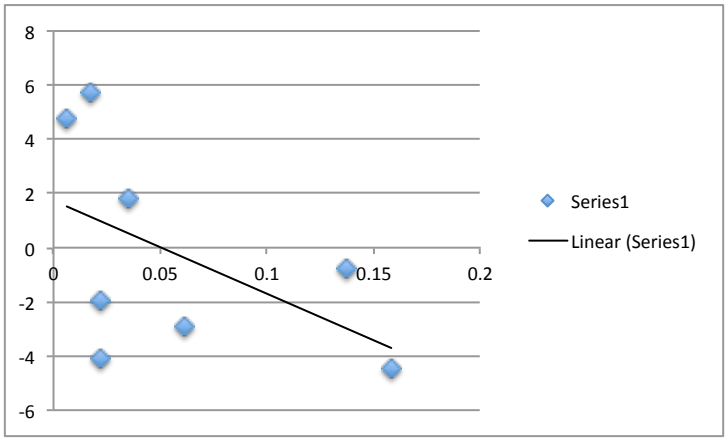

Figure 2: Prediction of drug response

cell lines, for which the signal flow through EGFR equals to 0.3. The expected disruption of signal flow caused by Erlotinib will equal to 0.3. If a drug has more than one targets then the expected disruption will equal the sum of all signal flows through its targets in the optimised model. We expect the more a drug disrupts signal flow in-silico for a given cell line, the stronger the effect of that drug to be in-vitro for that cell line. The in-vitro effect of the drug is quantified by the drugs IC50 value.

In this case study, the drug sensitivity data by Yang et al. [5] were used, including the IC50 values of about 140 drugs on the panel of the GDSC cell lines. Prior knowledge of the drug targets is obtained from `http://www.cancerrxgene.org`, where a curated list has been made available. Otherwise, a resource such as ChEMBL could be used to extract that information.

In figure 2 we plot the correlation between measured IC50 values and the score of each drug (as described above) for a sample cell line of the GDSC panel. With blue dots we have plotted the different drugs whose targets overlap with the optimised topology that was found to regulate gene expression for the respective cell line. The x-axis corresponds to the drug score and the y-axis corresponds to the measured IC50 values. As shown in figure 2, there is a negative correlation between IC50 values and the score of each drug. This implies that the more central drug targets are to the optimised network model (large score for this drug) the lowest the IC50 values, thus, the bigger the effect of this drug in-vitro. The relationship between drug score and IC50s is qualitative (small R^2 value) as the drugs may have other undocumented targets, or a drug may block pathways that are not related to survival/growth, thus inhibition of these pathways will not increase apoptosis. However, it is clear that the most potent drugs for this cell line are the ones whose targets play a significant role in the signal transduction of the optimised model.

4 Comparison with other approaches

In this section we conceptually compare the proposed approach to others that address the same or similar problems. More particularly, there are two methods that have been published over the past few years that tackle the identification of signaling pathways based on measured gene expression signatures. The first by Huang et al. [1] and the second by Zarringhalam et al. [7].

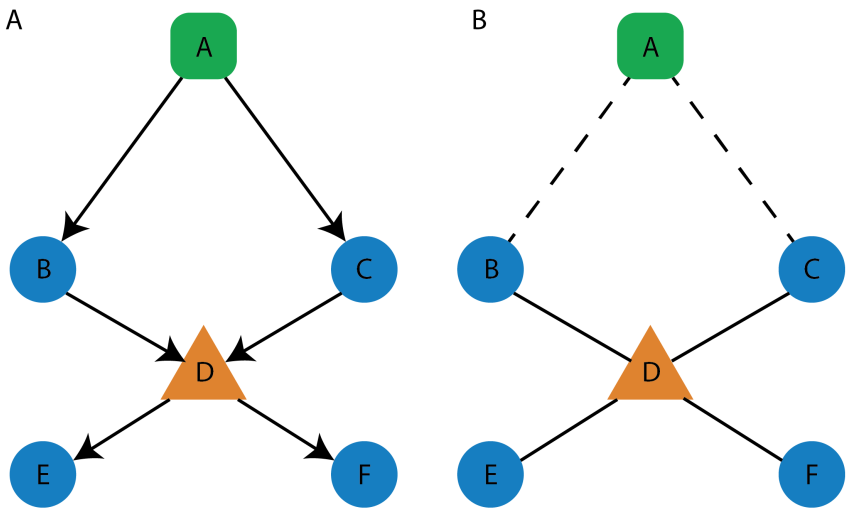

Figure 3: Comparison of our approach with the one by Huang et al.

Comparison with the method by Huang et al.

In the paper by Huang et al. the authors propose an application of the Steiner tree problem for calculating minimum subsets of a protein interaction network (i.e. minimum spanning trees) that join together genes and proteins found to be over expressed in a specific sample (cell line, tissue type etc). In contrast to the methodology we propose herein, Huang et al. use an undirected PPI network making difficult to pinpoint the origin of signal transduction. Moreover, the solution obtained to the Steiner Tree problem is always a fully connected graph, something that is not dictated by the physics of the problem, as in a specific sample many different/disconnected neighbourhoods of the network may be deregulated. Finally, even though the use of an exact algorithm (like the ones available for the Steiner tree problem) will be more efficient than an ILP implementation, formulating the problem as a regular optimisation problem provides a certain degree of flexibility, such as the computation of suboptimal solutions, the computation of signal flow through each node/reaction of the PKN obtaining an estimate of its significance in signal transduction, and the ability to enforce "soft" constraints, for example by maximising signal flow through nodes that are related to a specific phenotype. To illustrate how the methodology by Huang et al. may yield different results even at a very simple case, we have put together the following toy model (see figure 3).

In Figure 3A we plot the PKN, where node A represents a stimulus or generally an input to the system, nodes B and C are measured proteins, node D is a TF, and nodes E, F are measured genes. Assuming that nodes B, C, E and F are deregulated in the specific cell line or tissue sample, both approaches will try to identify minimum subsets of the PKN that join them together. Our methodology, taking into account the directionality of the PKN, will pinpoint as an origin of the signal transduction node A and conserve all included reactions in the solution to propagate signal to nodes B, C, E and F. On the other hand, the approach by Huang et al. will disregard directionality and identify a minimum spanning tree of the underlying PPI network as shown in figure 3B. However, in the solution of figure 3B it is not clear how signal propagates from one node to the next, and moreover, node D that appears to be the centre of the hub does not affect nodes B and C, it only affects nodes E and F (B and C are upstream of D). This implies, that if one was advising figure 3B to identify optimal drug candidates to

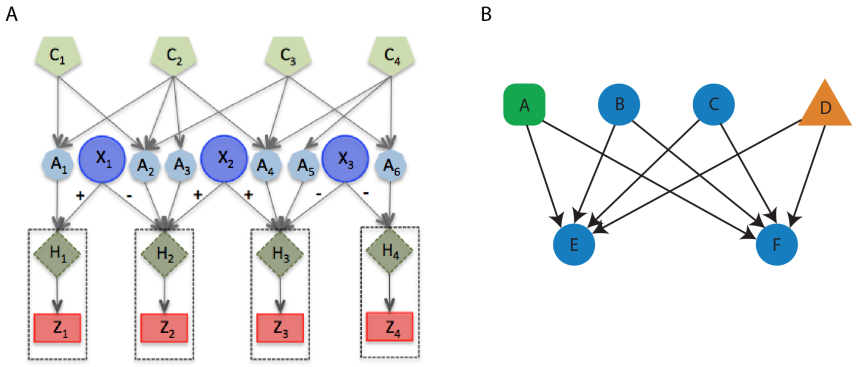

Figure 4: A. Schematic of the method by Zarringhalam et al. (obtained from [7]). B. Transforming the network of figure 3 into a DAG according to Zarringhalam et al.

block the deregulated nodes B, C, E and F, he would choose to inhibit node D, leaving nodes B and C unaffected. On the other hand, inhibiting node A that we identified as the source of the signalling process will effectively block all deregulated nodes.

Comparison with the method by Zarringhalam et al.

In the paper by Zarringhalam et al. the authors propose a bayesian inference approach to identify the source of differential gene expression in the signaling network of the interrogated cell/tissue type. Major difference with our approach is that Zarringhalam et al. do not use a PKN in the form that we use it (standard form signaling network from online pathway databases) but instead use a bayesian network that includes the following layers, ordered bottom to top (see also figure 4A) *(i)* Transcript nodes $Z = \{Z_1, ..., Z_m\}$, *(ii)* True state of the transcripts $H = \{H_1, ..., H_m\}$, *(iii)* Regulator nodes $X = \{X_1, ..., X_n\}$, *(iv)* Applicability nodes $A = \{A_1, ..., A_l\}$, and *(v)* Context nodes $C = \{C_1, ..., C_k\}$.

The bayesian approach by Zarringhalam et al. uses gene expression data to train the network of figure 4A and identify the optimal set of regulators that yielded the measured gene expression signature. Where the regulators (layer $X = \{X_1, ..., X_n\}$) are any entities (e.g. signaling molecules, TFs, etc.) that affect gene expression. Note that the connections between regulators and genes may be indirect. We can transform the PKN of figure 3 into a DAG according to Zarringhalam et al. by introducing all transitive reactions to genes E and F. See figure 4B. However, by compressing the PKN into the form of Figure 4B we lose information such as the regulation of transcription factor D by B and C. Moreover, the prior knowledge used by the method of Zarringhalam et al. needs to be much better curated compared to the one used by our approach, since if an ordinary signaling network (as parsed by an online pathway database) is compressed in this manner, an exponential amount of interactions will be introduced (for example the network in Figure 4B includes 8 interactions, while the network of figure 3 includes 6). Thus, of the two methods only ours offers the advantage to take into account the intricate structure of the PKN, while the method of Zarringhalam et al. uses a degenerated version of it.

5 Implementation

The ILP formulation presented in section 2 can be solved using commercial and/or open source solvers. In this work we have developed implementations using CPLEX and GUROBI which are commercial solvers but offer an academic license for free. More particularly the following implementations are available: *(i)* For GUROBI an implementation in C, *(ii)* For CPLEX implementations in C, R and Python (R via the Rcplex package). In the following we document the use of R interface for CPLEX, the other interfaces have a similar workflow. For inquiries please contact Ioannis Melas at `giannis.melas@gmail.com`

documentation of R interface for CPLEX

Installation

Requirements: You definitely need to install CPLEX and R on your computer. This software was developed using IBM-ILOG CPLEX Studio v12.6 and R version 3.1.0. The you need to install Rcplex which is a third party R interface for CPLEX. Rcplex v0.3-1 was used. Finally you need to include all input files and R source in the same directory.

Files

R implementation consists of the following files.

- **"optimize.r"**. R source code; it includes the main functionality of the toolbox. It formulates the ILP problem, calls CPLEX and parses the results.
- **"mainXX.r"**. R source code; XX stands for the version. It parses input files and calls the optimize.R script.
- **"ilp_options.txt"**. options file.
- **"species_merged.txt"**. example species file.
- **"Network.sif"**. example network file.
- **"inputs.txt"**. example inputs file.
- **"measurements.txt"**. example measurements file.

ilp_options.txt

Options file, it includes the following user defined options:

- **"mipgap"**. Relative MIP gap tolerance for CPLEX. CPLEX will stop as soon as it has found a feasible integer solution proved to be within five percent of optimal.
- **"timelimit"**. Sets the maximum time, in seconds, for a call to CPLEX.
- **"number_solutions"**. Sets the maximum number of optimal solutions generated for the solution pool. Suboptimal solutions can also be obtained but not advised.

- **"species_file"**. The name of the species file to be used. Has to be in the same directory as the ilp_options.txt and the R source.
- **"network_file"**. The name of the network file to be used. Has to be in the same directory as the ilp_options.txt and the R source.
- **"inputs_file"**. The name of the inputs file to be used. Has to be in the same directory as the ilp_options.txt and the R source.
- **"measurements_file"**. The name of the measurements file to be used. Has to be in the same directory as the ilp_options.txt and the R source.

species.txt

Tab-delimited file. Includes all species/nodes in the pathway and corresponding indices. The first column corresponds to the index and the second to the protein name. By convention the names of all gene nodes are ending in "_g". No special characters are allowed in the names.

Network.sif

Tab-delimited file. Lists the reactions (edges) of the prior knowledge network. First column contains the source node (or reactant) and the third column contains the target node (or product) in the corresponding edge. Second column contains the type of the interactions. Two types are supported, **"activations"** and **"inhibitions"**. If set to 1, then an activation reaction is assumed, if set to -1, then an inhibition is assumed. If set to 2, then an activation is assumed which cannot be removed by the algorithm (useful for when the user is positive the reaction is functional in the interrogated system). If set to -2, then an inhibition is assumed which cannot be removed by the algorithm. The signaling molecules are represented in terms of their ID, as these are included in the species file.

inputs.txt

Tab-delimited file. defines the possible perturbations/inputs of the network. Rows correspond to different experimental scenarios (i.e. samples), columns correspond to inputs of the prior knowledge network (i.e. perturbed nodes). The first row contains the IDs of the corresponding input nodes. the rest of the entries correspond to the imposed activation values of the input nodes. "1" corresponds to up-regulation (activation) of the respective node, "0" corresponds to inactivation (inhibition), "nan" implies the respective node may or may not be perturbed in that sample. Because the optimal perturbations set is calculated by the ILP algorithm to best fit the measured gene expressions, it is advised to the user to include all nodes that are possibly inputs/perturbations to the network (maybe even all nodes in the network) with a NaN value. If the user is certain that a given node is (or is not) perturbed then he may include that node with "1" (or "-1" respectively).

measurements.txt

Tab-delimited file. Specifies the measured gene expression values. Rows correspond to different experimental scenarios (samples), columns correspond to measured signals (gene expressions).

First row contains the IDs of the measured gene expressions. The rest of the entries correspond to the measured expression values. "1" corresponds to over expression (activation) of the respective gene, "0" corresponds to inactivation of the respective gene in that specific sample.

Output

CPLEX computes the optimal values of the variables in the model and exports them into the R workspace in a list named "sol". "sol" has equal number of elements to the computed solutions. Every element of "sol" is a list itself with the following elements:

- "xopt" : the optimal values of model variables (for the corresponding solution).
- "obj" : the objective value of the solution.
- "status" : the termination status of CPLEX.
- "extra" : number of nodes in the tree and value of slack variables.

To facilitate post processing the most informative variables are exported separately and stored in the following lists:

- "xs": the values of $x_{j,k}$.
- "zs": the values of $z_{i,k}$.
- "fxs": the values of $fx_{j,k}$.
- "fzs": the values of $fz_{i,k}$.
- "Bs": the values of $B_{j,k}$.
- "ys": the values of y_i.

References

[1] Shao-shan Carol Huang and Ernest Fraenkel. Integrating proteomic, transcriptional, and interactome data reveals hidden components of signaling and regulatory networks. *Sci. Signal.*, 2(81):ra40, 2009.

[2] Francesco Iorio, Roberta Bosotti, Emanuela Scacheri, Vincenzo Belcastro, Pratibha Mithbaokar, Rosa Ferriero, Loredana Murino, Roberto Tagliaferri, Nicola Brunetti-Pierri, Antonella Isacchi, and Diego di Bernardo. Discovery of drug mode of action and drug repositioning from transcriptional responses. *Proceedings of the National Academy of Sciences*, 107(33):14621–14626, 2010.

[3] Alexander Mitsos, Ioannis N. Melas, Paraskeuas Siminelakis, Aikaterini D. Chairakaki, Julio Saez-Rodriguez, and Leonidas G. Alexopoulos. Identifying drug effects via pathway alterations using an integer linear programming optimization formulation on phosphoproteomic data. *PLoS Comput Biol*, 5:e1000591, 12 2009.

[4] Aravind Subramanian, Pablo Tamayo, Vamsi K. Mootha, Sayan Mukherjee, Benjamin L. Ebert, Michael A. Gillette, Amanda Paulovich, Scott L. Pomeroy, Todd R. Golub, Eric S. Lander, and Jill P. Mesirov. Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proceedings of the National Academy of Sciences of the United States of America*, 102(43):15545–15550, 2005.

[5] Wanjuan Yang, Jorge Soares, Patricia Greninger, Elena J. Edelman, Howard Lightfoot, Simon Forbes, Nidhi Bindal, Dave Beare, James A. Smith, I. Richard Thompson, Sridhar Ramaswamy, P. Andrew Futreal, Daniel A. Haber, Michael R. Stratton, Cyril Benes, Ultan McDermott, and Mathew J. Garnett. Genomics of drug sensitivity in cancer (gdsc): a resource for therapeutic biomarker discovery in cancer cells. *Nucleic Acids Research*, 41(D1):D955–D961, 2013.

[6] Naif Zaman, Lei Li, MariaLuz Jaramillo, Zhanpeng Sun, Chabane Tibiche, Myriam Banville, Catherine Collins, Mark Trifiro, Miltiadis Paliouras, Andre Nantel, Maureen OConnor-McCourt, and Edwin Wang. Signaling network assessment of mutations and copy number variations predict breast cancer subtype-specific drug targets. *Cell Reports*, 5(1):216 – 223, 2013.

[7] Kourosh Zarringhalam, Ahmed Enayetallah, Alex Gutteridge, Ben Sidders, and Daniel Ziemek. Molecular causes of transcriptional response: a bayesian prior knowledge approach. *Bioinformatics*, 29(24):3167–3173, 2013.