Inferring the role of transcription factors in regulatory networks

Philippe Veber, Carito Guziolowski, Michel Le Borgne, Ovidiu Radulescu, Anne Siegel

Inferring the role of transcription factors in regulatory networks

P. Veber [a], C. Guziolowski [a], M. Le Borgne[b], O. Radulescu[a,c], A. Siegel[d]

[a] Centre INRIA Rennes Bretagne Atlantique, IRISA, Rennes, France [b]Université de Rennes 1, IRISA, Rennes, France [c], Université de Rennes 1, IRMAR, Rennes, France [d] CNRS, UMR 6074, IRISA, Rennes, France

ABSTRACT

Background Expression profiles obtained from multiple perturbation experiments are increasingly used to reconstruct transcriptional regulatory networks, from well studied, simple organisms up to higher eukaryotes. Admittedly, a key ingredient in developing a reconstruction method is its ability to integrate heterogeneous sources of information, as well as to comply with practical observability issues: measurements can be scarce or noisy. The purpose of this work is (1) to build a formal model of regulations among genes; (2) to check its consistency with gene expression data on stress perturbation assays; (3) to infer the regulatory role of transcription factors as inducer or repressor if the model is consistent with expression profiles; (4) to isolate ambiguous pieces of information if it is not.

Results We validate our methods on *E. Coli* network with a compendium of expression profiles. We investigate the dependence between the number of available expression profiles and the number of inferred regulations, in the case where all genes are observed. This is done by simulating artificial observations for the transcriptional network of *E. Coli* (1529 nodes and 3802 edges). We prove that at most 40,8% of the network can be inferred and that 30 distinct expression profiles are enough to infer 30% of the network on average. We repeat this procedure in the case of missing observations, and show that our approach is robust to a significant proportion of unobserved genes. Finally, we apply our inference algorithms to *S. Cerevisiae* transcriptional network, and demonstrate that for small scale subnetworks of *S. Cerevisiae* we are able to infer more than 20% of the regulations. For more complex networks, we are able to detect and isolate inconsistencies between experimental sources and a non negligible portion of the model (15% of all interactions).

Conclusions Our approach does not require accurate expression levels, nor times series. Nevertheless, we show both on real and artificial data that a relatively small number of perturbation experiments are enough to determine a significant portion of regulatory effects. This is a key practical asset compared to statistical methods for network reconstruction. In addition, we illustrate the capability of our method to validate networks. We conjecture that inconsistencies we detected might be good candidates for further experimental investigations.

Contact philippe.veber@irisa.fr

1 INTRODUCTION

A central problem in molecular genetics is to understand the transcriptional regulation of gene expression. A transcription factor (TF) is a protein that binds to a typical domain on the DNA and influences transcription. Depending on the type of binding site, on the distance to the coding regions and on the presence of other molecules that also bind to the DNA, the effect can either be a repression or an activation of the transcription. Finding which gene is controlled by which TF is a reverse engineering problem, usually named *network reconstruction*. This question has been approached over the past years by various groups.

A first approach to achieve this task consists in expanding information spread in the primary literature. A number of important databases that take protein and regulatory interactions from the literature and curate them have been developed [1, 2, 3, 4, 5]. For the bacteria *E. Coli*, RegulonDB is a dedicated database that contains experimentally verified regulatory interactions [6]. For the budding yeast (*S. Cerevisiae*), the Yeast Proteome Database contains amounts of regulatory information [7]. Even in this latter case, the amount of available information is not sufficient to build a reasonably accurate model of transcriptional regulation. It is nevertheless an unavoidable starting point for network reconstruction.

The alternative to the literature-curated approach is a data-driven approach. This approach is supported by the availability of high-throughput experimental data, including microarray expression analysis of deletion mutants (simple or more rarely double non-lethal knockouts), over expression of TF-encoding genes, protein-protein interactions, protein localization or chIP-chip experiments coupled with promoter sequence motifs. We may cite several classes of methods: perturbations and knock-outs, microarray analysis of promoter binding (chIP-chip), sequence analysis, various microarray expression data analysis such as correlation, mutual information or causality studies, Bayesian networks, path analysis, information-theoretic approaches and ordinary differential equations [8, 9, 10].

In short, most available approaches so far are based on a probabilistic framework, which defines a probability distribution over the set of models. Then, an optimization algorithm is applied in order to determine the most likely model given by the data. Due to the size of the inferred model, the optimal model may be a local but not a global optimal. Hence, errors can appear and no consensual model can be produced. As an illustration, a special attention has been paid to the reconstruction of *S. Cerevisiae network* from chIP-chip data and protein-protein interaction networks [11]. A first regulatory network was obtained with promoter sequence analysis methods [12, 13]. Non-parametric causality tests proposed some previously undetected transcriptional regulatory motifs [14]. Bayesian analysis also proposed transcriptional networks [15, 16]. Though, the results obtained with the different methods do not coincide and a fully data-driven search is in general subject to overfitting and to unfiability [17].

In regulatory networks, an important and nontrivial physiological information is the regulatory role of transcription factors as inducer of repressor, also called *the sign of the interaction*. This information is needed if one wants to know for instance the physiological effect of a change of external conditions or simply deduce the effect of a perturbation on the transcription factor. While this can be achieved for one gene at a time with (long and expensive) dedicated experiments, probabilistic methods such as Bayesian models [18] of path analysis [19, 20] are capable to propose models from high-throughput experimental data. However, as for the network reconstruction task, these methods are based on optimization algorithms to compute an optimal solution with respect to an interaction model.

In this paper, we propose to use formal methods to compute the sign of interactions on networks for which a topology is available. By doing so, we are also capable of validating the topology of the network. Roughly, expression profiles are used to constrain the possible regulatory roles of transcription factor, and we report those regulations which are assigned the same role in all feasible models. Thus, we over-approximate the set of *feasible* models, and then look for *invariants* in this set. A similar idea was used in [21] in order to check the consistency of gene expression assays. We use a deeper formalisation and stronger algorithmic methods in order to achieve the inference task.

We use different sources of large-scale data: gene expression arrays provide indications on signs of interactions. When not available, ChIP-chip experiments provide a sketch for the topology of the regulatory network. Indeed, microarray analysis of promoter binding (ChIP-chip) is an experimental procedure to determine the set of genes controlled by a given transcription factor in given experimental conditions [22]. A particularly interesting feature of this approach is that it provides an *in vivo* assessment of transcription factor binding. On the contrary, testing affinity of a protein for a given DNA segment *in vitro* often results in false positive binding sites.

The main tasks we address are the following:

1. Building a formal model of regulation for a set of genes, which integrates information from ChIP-chip data, sequence analysis, literature annotations;
2. Checking its consistency with expression profiles on perturbation assays;
3. Inferring the regulatory role of transcription factors as inducer or repressor if the model is consistent with expression profiles;
4. Isolating ambiguous pieces of information if it is not.

Both, probabilistic approaches and our formal approach mainly aim to deal with incomplete knowledge and experimental noise. However, statistical methods usually require a minimal number of samples (about a hundred), because they explicitly model the distribution of experimental noise. In practice it is feasible but very costly to obtain enough expression profiles to apply them. In contrast, our approach may be used even with less perturbation experiments (some tens) at hand, which makes it a suitable alternative when statistical methods cannot be applied.

Additionally, since our predictions are consensual with *all* profiles and since they are not based on heuristics, our methods are well designed to validate networks inferred with probabilistic methods, and eventually identify the location of inconsistencies.

The paper is organized as follows. Sec. 2 briefly introduces the mathematical framework which is used to define and to test the consistency between expression profiles and gene networks. In Sec. 3 we apply our algorithms to address three main issues.

- We illustrate and validate our formal method on the transcriptional network of *E. Coli* (1529 nodes and 3802 edges), as provided in RegulonDB [6], together with a compendium of expression profiles [9]. We identified 20 inconsistent edges in the graph.
- We investigate the dependence between the number of available observations and the number of inferred regulations, in the case where all genes are observed. This is done by simulating artificial observations for the transcriptional network of *E. Coli*. We prove that at most 40,8% of the network can be inferred and that 30 perturbation experiments are enough to infer 30% of the network on average. By studying a reduced network, we also comment about the complementarity between our approach and detailed analysis of times series using dynamical modeling.
- We repeat this procedure in the case of missing observations, and estimate how the proportion of unobserved genes affects the number of inferred regulations. With these two situations we also demonstrate that our approach is able to handle networks containing thousands of genes, with several hundreds of observations.
- We apply our inference algorithms to *S. Cerevisiae* transcriptional network, in order to assess their relevance in real conditions. We demonstrate that for small scale subnetworks of *S. Cerevisiae* we are able to infer more than 20% of the roles of regulations. For more complex networks, we are able to detect and isolate inconsistencies (also called ambiguities) between expression profiles and a quite important part of the model (15% of all the interactions).

The last two sections discuss the results we obtained, and give more details on the algorithmic procedures.

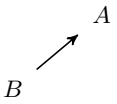
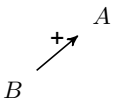
2 APPROACH

2.1 Detecting the sign of a regulation and validating a model

Our goal is to determine the regulatory action of a transcription factor on its target genes by using expression profiles. Let us illustrate our purpose with a simple example.

We suppose that we are given the topology of a network (this topology can be obtained from ChIP-chip data or any computational network inference method). In this network, let us consider a node A with a single predecessor. In other words, the model tells that the protein B acts on the production of the gene coding for A and no other protein acts on A.

Independently, we suppose that we have several gene expression arrays at our disposal. One of these arrays indicates that A and B simultaneously increase during a steady state experiment. Then, the *common sense* says that B must have been as activator of A during the experiment. More precisely, protein B cannot have inhibited

Model	Expression profiles	Prediction
	B increases C increases	The action from B to A is an activation.
Model	Expression profiles	Prediction
	B increases C decreases	Model and data are ambiguous (also called *incompatible*).

gene A, since they both have increased. We say that the model *predicts* that the sign of the interaction from B to A is positive.

This naive rule is actually used in a large class of models, we will call it the *naive inference rule*. When several expression profiles are available, the predictions of the different profiles can be compared. If two expression profiles predict different signs for a given interaction, there is a *ambiguity* or *incompatibility* between data and model. Then, the ambiguity of the regulatory role can be attributed to three factors: (1) a complex mechanism of regulation: the role of the interaction is not constant in all contexts, (2) a missing interaction in the model, (3) an error in the experimental source.

Algorithm: Naive Inference algorithm

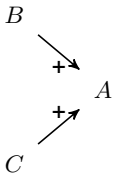
Input:
A network with its topology
A set of expression profiles

Output:
a set of predicted signs
a set of ambiguous interactions

For all Node A with exactly one predecessor B
if *A and B are observed simultaneously* **then return** *prediction* $sign(B \to A) = sign(A) * sign(B)$
if $sign(B \to A)$ *was predicted different by another expression profile* **then return** *Ambiguous arrow* $B \to A$

Let us consider now the case when A is activated by two proteins B and C. No more natural deduction can be done when A and B increase during an experiment, since the influence of C must be taken into account. A *model* of interaction between A, B and C has to be proposed. Probabilistic methods estimate the most probable signs of regulations that fit with the theoretical model [18, 23].

Our point of view is different: we introduce a *basic rule* that shall be checked by every interactions. This rule tells that **any variation of A must be explained by the variation of at least one of its predecessors**. Biologically, this assumes that the nature of differential gene expression of a given gene is likely to affect the differential expression in other genes. Even if this is not universally true, this can be viewed as a crude approximation of the real event. In previous papers, we introduced a formal framework to justify this basic rule under some reasonable assumptions. We also tested the consistency between expression profiles and a graphical model of cellular interactions. This formalism will be here introduced in an informal way ;

Model	Expression profiles	Prediction
	B decreases C decreases	A decreases

its full justification and extensions can be found in the references [24, 25, 26, 27].

In our example, the basic rule means that if B and C activate A, and both B and C are known to decrease during a steady state experiment, A cannot be observed as increasing. Then A is *predicted* to decrease. More generally, in our approach, we use the rule as a constraint for the model. We write constraints for all the nodes of the model and we use several approaches in order to solve the system of constraints. From the study of the set of solutions, we deduce which signs are surely determined by these rules. Then we obtain *minimal obligatory conditions* on the signs, instead of *most probable signs* given by probabilistic methods. Notice that by construction, our constraints coincide with probabilistic models in the predictions of the naive inference algorithm.

2.2 A formal approach

Consider a system of n chemical species $\{1, \ldots, n\}$. These species interact with each other and we model these interactions using an *interaction graph* $G = (V, E)$. The set of nodes is denoted by $V = \{1, \ldots, n\}$. There is an edge $j \to i \in E$ if the level of species j influences the production rate of species i. Edges are labeled by a sign $\{+, -\}$ which indicates whether j activates or represses the production of i.

In a typical stress perturbation experiment a system leaves an initial steady state following a change in control parameters. After waiting long enough, the system may reach a new steady state. In genetic perturbation experiments, a gene of the cell is either knocked-out or overexpressed; perturbed cells are then compared to wild cells. Most high-throughput measurements provide the ratio between initial and final levels, like in expression arrays for instance. In many experimental settings, the numerical value is not accurate enough to be taken "as it is". The noise distribution may be studied if enough measurements are available. Otherwise, it is safer to rely only on a qualitative feature, such as the order of magnitude, or the *sign of the variation*. Let us denote by $sign(X_i) \in \{+, -, \mathbf{0}\}$ the sign of variation of species i during a given perturbation experiment, and by $sign(j \to i) \in \{+, -\}$ the sign of the edge $j \to i$ in the interaction graph.

Let us fix species i such that there is no positive self-regulating action on i. For every predecessor j of i, $sign(j \to i) * sign(X_j)$ provides the sign of the *influence* of j on the species i. Then, we can write a constraint on the variation to interpret the rule previously stated: *the variation of species i is explained by the variation of at least one of its predecessors in the graph.*

$$sign(X_i) \approx \sum_{j \to i} sign(j \to i) sign(X_j). \qquad (1)$$

When the experiment is a genetic perturbation the same equation stands for every node that was not genetically perturbed during the

experiment and such that all its predecessors were not genetically perturbed. If a predecessor X_M of the node was knocked-out, the equation becomes

$$sign(X_i) \approx -sign(M \to i) + \sum_{j \to i, j \neq M} sign(j \to i) sign(X_j). \quad (2)$$

The same holds with $+sign(M \to i)$ when the predecessor X_M was overexpressed. There is no equation for the genetically perturbed node.

The *sign algebra* is the suitable framework to read these equations [26]. It is defined as the set $\{\mathbf{+}, \mathbf{-}, \mathbf{?}, \mathbf{0}\}$, provided with a sign compatibility relation $\approx$, and arithmetic operations $+$ and $\times$. The following tables describe this algebra:

$\mathbf{+}+\mathbf{-}=\mathbf{?}$	$\mathbf{+}+\mathbf{+}=\mathbf{+}$	$\mathbf{-}+\mathbf{-}=\mathbf{-}$	$\mathbf{+}\times\mathbf{-}=\mathbf{-}$	$\mathbf{+}\times\mathbf{+}=\mathbf{+}$	$\mathbf{-}\times\mathbf{-}=\mathbf{+}$
$\mathbf{+}+\mathbf{0}=\mathbf{+}$	$\mathbf{0}+\mathbf{0}=\mathbf{0}$	$\mathbf{-}+\mathbf{0}=\mathbf{-}$	$\mathbf{+}\times\mathbf{0}=\mathbf{0}$	$\mathbf{0}\times\mathbf{0}=\mathbf{0}$	$\mathbf{-}\times\mathbf{0}=\mathbf{0}$
$\mathbf{?}+\mathbf{-}=\mathbf{?}$	$\mathbf{?}+\mathbf{+}=\mathbf{?}$	$\mathbf{?}+\mathbf{?}=\mathbf{?}$	$\mathbf{?}\times\mathbf{-}=\mathbf{?}$	$\mathbf{?}\times\mathbf{+}=\mathbf{?}$	$\mathbf{?}\times\mathbf{?}=\mathbf{?}$
$\mathbf{?}+\mathbf{0}=\mathbf{?}$	$\mathbf{?}\times\mathbf{0}=\mathbf{0}$				

$\mathbf{+}\not\approx\mathbf{-}$	$\mathbf{+}\approx\mathbf{0}$	$\mathbf{-}\approx\mathbf{0}$	$\mathbf{?}\approx\mathbf{+}$	$\mathbf{?}\approx\mathbf{-}$	$\mathbf{?}\approx\mathbf{0}$

Even if the sign compatibility relation $\approx$ provides a rule for the $\mathbf{0}$ value, we are not able to infer with our approach regulations of sign $\mathbf{0}$. This limitation is because the sign of an arrow in an interaction graph is only restricted to be $\{\mathbf{+}, \mathbf{-}\}$, thus we do not generate an equation for products which have no variation during a specific experiment.

For a given interaction graph G, we will refer to the *qualitative system* associated to G as the set made up of constraint (1) for each node in G. We say that node variations $X_i \in \{\mathbf{+}, \mathbf{-}, \mathbf{0}\}$ are *compatible* with the graph G when they satisfy all the constraints associated to G using the sign compatibility relation $\approx$.

With this material at hand, let us come back to our original problem, namely to infer the regulatory role of transcription factors from the combination of heterogeneous data. In the following we assume that :

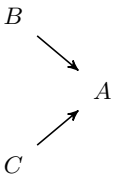
- The interaction graph is either given by a model to be validated, or built from chIP-chip data and transcription factor binding site searching in promoter sequences. Thus, as soon as a transcription factor j binds to the promoter sequence of gene i, j is assumed to regulate i. This is represented by an arrow $j \to i$ in the interaction graph.
- The regulatory role of a transcription factor j on a gene i (as inducer or repressor) is represented by the variable S_{ji}, which is constrained by Eqs. (1) or (2).
- Expression profiles provide the sign of the variation of the gene expression for a set of r steady-state perturbation or mutant experiments. In the following, x_i^k will stand for the sign of the observed variation of gene i in experiment k.

Our inference problem can now be stated as finding values in $\{\mathbf{+}, \mathbf{-}\}$ for S_{ji}, subject to the constraints :

$$\text{for all } (1 \leq i \leq n),\ (1 \leq k \leq r),$$
$$i \text{ not genetically perturbed in the } k\text{-th experiment}$$
$$\begin{cases} x_i^k \approx \sum_{j \to i} S_{ji} x_j^k \text{ if no genetic perturbations on all nodes } j \\ x_i^k \approx -S_{Mi} + \sum_{j \to i, j \neq l} S_{ji} x_j^k \text{ if knocked-out node } M \\ x_i^k \approx S_{Mi} + \sum_{j \to i, j \neq l} S_{ji} x_j^k \text{ if overexpressed node } M \end{cases} \quad (3)$$

Most of the time, this inference problem has a huge number of solutions. However, some variables S_{ji} may be assigned the same value in *all* solutions of the system. Then, the recurrent value assigned to S_{ji} is a logical consequence of the constraints (3), and a prediction of the model. We will refer to these inferred interaction signs as *hard components* of the qualitative system, that is, sign variables S_{ji} that have the same value in all solutions of a qualitative system (3). When the inference problem has *no solution*, we say that the model and the data are *inconsistent* or *ambiguous*.

Let us illustrate this formulation on a very simple (yet informative) example. Suppose that we have a system of three genes A, B, C, where B and C influence A. The graph is shown in Table 1. Let us say that for this interaction graph we obtained six experiments, in each of them the variation of all products in the graph was observed (see Table 1). Using some or all of the experiments provided in Table 1 will lead us to a different qualitative system, as shown in Table 2, hence to different inference results. The process of inference for this example can be summarized as follows: starting from a set of experiments we generate the qualitative system of equations for our graph, studying its compatibility we will be able to set values for the signs of the regulations (edges of the interaction graph), but only we will infer a sign if in all solutions of the system the sign is set to the same value. Following with the example, in Table 2 we illustrate this process showing how the set of inferred signs of regulation varies with the set of experiments provided.



Stress perturbation expression profile	x_A	x_B	x_C
e_1	+	+	+
e_2	+	+	−
e_3	−	+	−
e_4	−	−	−
e_5	−	−	+
e_6	+	−	+

Table 1. Interaction graph of three genes A, B, C, where B and C influence A. Table with the variation of genes A,B, and C observed in six different stress perturbation experiments.

2.3 Algorithmic procedure

When the signs on edges are known (i.e. fixed values of S_{ji}) finding compatible node variations X_i is a NP-complete problem [26]. When the node variations are known (i.e. fixed values of X_i) finding the signs of edges S_{ji} from X_i can be proven NP-complete in a very similar way. Though, we have been able to design algorithms that perform efficiently on a wide class of regulatory networks. These algorithms predict signs of the edges when the network topology and

the expression profiles are compatible. In case of incompatibility, they identify ambiguous motifs and propose predictions on parts of the network that are not concerned with ambiguities.

The general process flow is the following (see Sec. 6 for details):

Step 1 Sign Inference

Divide the graph into motifs (each node with its predecessors). For each motif, find sign valuations (see Algorithm 1 in Sec.6) that are compatible with all expression profiles. If there are no solutions, call the motif *Multiple Behaviors Module* and remove it from the network.

Solve again the remaining equations and determine the edge signs that are fixed to the same value in all the solutions. These fixed signs are called *edge hard components* and represent our predictions.

Step 2 Global test/correction of the inferred signs

Solutions at previous step are not guaranteed to be global. Indeed, two node motifs at step 1 can be compatible separately, but not altogether (with respect to all expression profiles). This step checks global compatibility by solving the equations for each expression profile. New *Multiple Behavior Modules* can be found and removed from the system.

Step 3 Extending the original set of observations

Once all conflicts removed, we get a set of solutions in which signs are assessed to both nodes and edges. *Node hard components*, representing inferred gene variations can be found in the same way as we did for edges. We add the new variations to the set of observations and return to step 1. The algorithm is iterated until no new signs are inferred.

Step 4 Filtering predictions

In the incompatible case, the validity of the predictions depends on the accuracy of the model and on the correct identification of the MBMs. The model can be incomplete (missing interactions), and MBMs are not always identifiable in an unique way. Thus, it is useful to sort predictions according to their reliability. Our filtering parameter is a positive integer k representing the number of different experiments with which the predicted sign is compatible. For a filtering value k, all the predictions that are consistent with less than k profiles are rejected.

The inference process then generates three results:

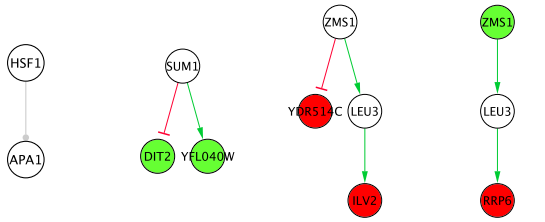
1. *A set of multiple behavior modules (MBM)*, containing interactions whose role was unclear and generated incompatibilities. We have identified several types of MBMS:
   - Modules of TypeI: these modules are composed of several direct regulations of the same gene. These modules are detected in the Step 1 of the algorithm. Most of the MBMs of Type I are made of only one edge like illustrated in Fig. 1, but bigger examples exist.
   - Modules of Type II, III, IV: these modules are detected in Steps 2 or 3, hence, they contain either direct regulations from the same gene or indirect regulations and/or loops. Each of these regulations represent a consensus of all the experiments, but when we attempt to assess them globally, they lead to contradictions. The indices II-IV have no topological meaning, they label the most frequent situations illustrated in Fig. 1.
2. *A set of inferred signs*, meaning that all expression profiles fix the sign of an interaction in a unique way.
3. *A reliability ranking of inferred signs*. The filtering parameter k used for ranking is the number of different expression profiles that validate a given sign.

On computational ground, the division between Step 1 (which considers each small motif with all profiles together) and Step 2 (which considers the whole network with each profile separately) is necessary to overcome the memory complexity of the search of solutions. To handle large-scale systems, we combine a model-checking approach by decision diagrams and constraint solvers (see details in Sec. 6).

Since our basic rule is a crude approximation of real events, we expect it to produce very robust predictions. On the other hand, a regulatory network is only a rough description of a reaction network. For certain interaction graphs, not a single sign may be inferred even

Experiments used	Qualitative system	Replacing values from experiments	Compatible solutions (S_{BA}, S_{CA})	Inferred signs (identical in all solutions)
$\{e_1\}$	$x_A^1 \approx S_{BA}x_B^1 + S_{CA}x_C^1$	$(+) \approx S_{BA} \times (+) + S_{CA} \times (+)$	$(+,+)$ $(+,-)$ $(-,+)$	$\emptyset$
$\{e_1, e_2\}$	$x_A^1 \approx S_{BA}x_B^1 + S_{CA}x_C^1$ $x_A^2 \approx S_{BA}x_B^2 + S_{CA}x_C^2$	$(+) \approx S_{BA} \times (+) + S_{CA} \times (+)$ $(+) \approx S_{BA} \times (+) + S_{CA} \times (-)$	$(+,+)$ $(+,-)$	$\{S_{BA} = +\}$
$\{e_1, e_2, e_3\}$	$x_A^1 \approx S_{BA}x_B^1 + S_{CA}x_C^1$ $x_A^2 \approx S_{BA}x_B^2 + S_{CA}x_C^2$ $x_A^3 \approx S_{BA}x_B^3 + S_{CA}x_C^3$	$(+) \approx S_{BA} \times (+) + S_{CA} \times (+)$ $(+) \approx S_{BA} \times (+) + S_{CA} \times (-)$ $(-) \approx S_{BA} \times (+) + S_{CA} \times (-)$	$(+,+)$	$\{S_{BA} = +, S_{CA} = +\}$

Table 2. Sign inference process. In this example variables are only the roles of regulations (signs) in an interaction graph , the variations of the species in the graph are obtained from the set of six experiments described in Table 1. For different sets of experiments we do not infer the same roles of regulations. We observe in this example that if we take into account experiments $\{e_1, e_2, e_3\}$, our qualitative system will have three constrains and not all valuations of variables S_{BA} and S_{CA} satisfy this system according to the sign algebra rules. As we obtain unique values for these variables in the solution of the system, we consider them as inferred.



[Type I] [Type II] [Type III] [Type IV]

Figure 1. Classification of the multiple behavior modules (MBM). These modules are some of the MBM found in the global regulatory network of *S. Cerevisiae* extracted from [11]. Green and red interactions correspond to inferred activations and repressions respectively. Genes are colored by their expression level during certain experiment (green: more than 2-fold expression, red: less than 2-fold repression) (a) **Type I** modules are composed by direct regulations of one gene by its predecessors. Sources of the conflict in this example are: Heat shock 21°C to 37°C [28], and Cells grown to early log-phase in YPE [29]. (b) **Type II** The genes in this module have the same direct predecessor. Explanation: The interaction among *SUM1* and *YFL040W* is inferred at the beginning of the inference process, as an activation while among *SUM1* and *DIT2* is inferred as an inhibition. During the *correction* step, expression profile related to YPD Broth to Stationary Phase [28], shows that these two genes: *YFL040W* and *DIT2* overexpress under this condition. Resulting impossible to determine the state (overexpressed or underexpressed) of *SUM1*, we mark this module as a MBM. (c) **Type III** The genes in this module share a predecessor, but not the direct one. Source of the conflict: Diauxic shift [30]. (d) **Type IV** The predecessor of one gene is the successor of the other. Source of the conflict: Heat Shock 17°C to 37°C [28].

with a high number of experiments. In Sec. 3, we comment the maximum number of signs that can be inferred from a given graph.

3 RESULTS

In perturbation experiments, gene responses are observed following changes of external conditions (temperature, nutritional stress, etc.) or following gene inactivations, knock-outs or overexpression. When expression profile is available for all the genes in the network we say that we have a *complete profile*, otherwise the profile is *partial* (data is missing). The effect of gene deletions is modelled as the one of inactivations, which is imposing negative gene variations. Thus, we may say that we deal with perturbation experiments that do not change the topology of the network. An experiment in which topology is changed would be to record the effect of stresses on mutants; this possibility will be discussed elsewhere.

In order to validate our formal approach, we evaluate the percentage of the network that might be recovered from a reasonable number of perturbation experiments. We first provide theoretical limits for the percentage of recovered signs. These limits depend on the topology of the network. For the transcriptional network of *E. Coli*, these limits are estimated first by a deterministic and then by a statistical algorithm. The statistical approach uses artificial random data. Then we combine expression profiles with a publicly available structure of *E. Coli* network, and compute the percentage of recovered signs. Finally, we combine real expression profiles with chIP-chip data on *S. Cerevisiae*, and evaluate the percentage of recovered signs in a real setting.

On computational ground, we check that our algorithms are able to handle large scale data, as produced by high-throughput measurement techniques (expression arrays, chIP-chip data). This is demonstrated in the following by considering networks of more than several thousand genes.

3.1 Stress perturbation experiments: how many do you need ?

For any given network topology, even when considering all possible experimental perturbations and expression profiles, there are signs that can not be determined (see Table 2). Sign inference has thus a theoretical limit that we call *theoretical percentage of recovered signs*. This limit is unique for a given network topology. If only some perturbation experiments are available, and/or data is missing, the percentage of inferred signs will be lower. For a given number N of available expression profiles, *the average percentage of recovered signs* is defined over all sets of N different expression profiles compatible with the qualitative constraints Eqs. (1) and (2).

In this section, we calculate and comment the theoretical and the average percentages of recovered signs for the transcriptional network of *E. Coli*.

We first validate our method on the *E. Coli* network. We build the interaction graph corresponding to *E. Coli* transcriptional network, using the publicly available RegulonDB [6] as our reference. For each transcriptional regulation $A \rightarrow B$ we add the corresponding arrow between genes A and B in the interaction graph. This graph will be referred to as the *unsigned interaction graph*.

From the unsigned interaction graph of *E. Coli*, we build the *signed interaction graph*, by annotating the edges with a sign. Most of the time, the regulatory action of a transcription factor is available in RegulonDB. When it is unknown, or when it depends on the level of the transcription factor itself, we arbitrarily choose the value + for this regulation. This provides a graph with 1529 nodes and 3802 edges, all edges being signed. The signed interaction graph is used to generate complete expression profiles that simulate the effect of perturbations. More precisely, a perturbation experiment is represented by a set of gene expression variations $\{X_i\}_{i=1,\ldots,n}$. These variations are not entirely random: they are constrained by Eqs.(1) and (2). Then we forget the signs of network edges and we compute the qualitative system with the signs of regulations as unknowns.

The *theoretical maximum percentage of inference* is given by the number of signs that can be recovered assuming that expression profiles of *all* conceivable perturbation experiments are available. We computed this maximum percentage by using constraint solvers (the algorithm is given in Sec. 6). We found that *at most* 40.8% of the signs in the network can be inferred, corresponding to $M_{max} = 1551$ edges.

However, this maximum can be obtained only if all conceivable (much more than 2^{50}) perturbation experiments are done, which is not possible. We performed computations to understand the influence of the number of experiments N on the inference. For each value of N, where N grows from 5 to 200, we generated 100 sets of N random expression profiles. Each time our inference algorithm is used to recover signs. Then, the average percentage of inference is calculated as a function of N. The resulting statistics are shown in Fig. 2.

When the number of experiments (X-axis) equals 1, the value $M_1 = 609$ corresponds to the average number of signs inferred from a single perturbation experiment. These signs correspond to

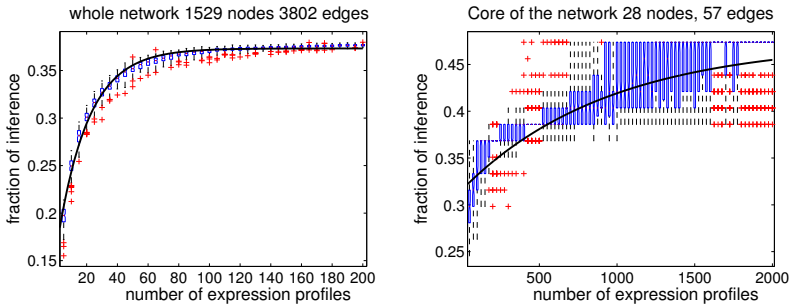


Figure 2. (Both) Statistics of inference on the regulatory network of *E. Coli* from complete expression profiles. The signed interaction graph is used to randomly generate sets of X artificial expression profiles which cover the *whole* network (*complete expression profile*). Each set of artificial profiles is then used with the unsigned interaction graph to recover regulatory roles. X-axis: number of expression profiles in the dataset. Y-axis: percentage of recovered signs in the unsigned interaction graph. This percentage may vary for a fixed number of expression profile in a set. Instead of plotting each dot corresponding to a set, we represent the distribution by boxplots. Each boxplot vertically indicates the minimum, the first quartile, the median, the third quartile and the maximum of the empiric distribution. Crosses show outliers (exceptional data points). The continuous line corresponds to the theoretical prediction $Y = M_1 + M_2(1-(1-p)^X)$, where M_1 stands for the number of signs that should be inferred from any expression profile (that is, inferred by the naive inference algorithm); and M_2 denotes the number of signs that could be inferred with a probability p.
(Left) Statistics of inference for the *whole E. Coli* transcriptional network. We estimate that at most $37,3\%$ of the network can be inferred from a limited number of different complete expression profiles. Among the inferred regulations, we estimate to $M_1 = 609$ the number of signs that should be inferred from any complete expression profile. The remaining $M_2 = 811$ signs are inferred with a probability estimated to $p = 0.049$. Hence, 30 perturbation experiments are enough to infer 30% of the network.
(Right) Statistics of inference for the core of the former graph (see definition of a core in the text). An estimation gives $M_1 = 18$ and $M_2 = 9$ so that the maximum rate of inference is $47,3\%$. Since $p = 0.0011$, the number of expression profiles required to obtain a given percentage of inference is much greater than in the whole network.

single incoming regulatory interactions and are thus within the scope of the naive inference algorithm. We deduce that the naive inference algorithm allows to infer on average 18% of the signs in the network.

Surprisingly, by using our method we can significantly improve the naive inference, with little effort. For the whole *E. Coli* network it appears that a few expression profiles are enough to infer a significant percentage of the network. More precisely, 30 different expression profiles may be enough to infer one third of the network, that is about 1200 regulatory roles. Adding more expression profiles continuously increases the percentage of inferred signs. We reach a plateau close to 37,3% (this corresponds to $M = 1450$ signed regulations) for $N = 200$.

The saturation aspect of the curve in Fig. 2 is compatible with two hypotheses. According to the first hypothesis, on top of the M_1 single incoming regulations (that can be inferred with a single expression profile), there are M_2 interactions whose signs are inferred with more than one expression profile. On average, a single expression profile determines with probability $p < 1$ the sign of interactions of the latter category. According to the second hypothesis, the contributions of different experiments to the inference of this type of interactions are independent. Thus, the average number of inferred signs is $M(N) = M_1 + M_2(1-(1-p)^N)$. The two numbers satisfy $M_1 + M_2 < E$ (E is the total number of edges), meaning that there are edges whose signs can not be inferred.

According to this estimate the position of the plateau is $M = M_1 + M_2$ and should correspond to the theoretical maximum percentage of inferred signs M_{max}. Actually, $M < M_{max}$. The difference, although negligible in practice (to obtain M_{max} one has to perform $N > 10^{15}$ experiments) suggests that the plateau has a very weak slope. This means that contributions of different experiments to sign inference are weakly dependent.

The values of M_1, M_2, p estimate the efficiency of our method: large p,M_1,M_2 mean small number of expression profiles needed for inference. For the *E. Coli* full transcriptional network we have $p = 0.049$ per observation. This means that we need about 20 profiles to reach half of the theoretical limit of our approach.

3.2 Inferring the core of the network

Obviously, not all interactions play the same role in the network. The *core* is a subnetwork that naturally appears for computational purpose and plays an important role in the system. It consists of all oriented loops and of all oriented chains leading to loops. All oriented chains leaving the core without returning are discarded when reducing the network to its core. Acyclic graphs and in particular trees have no core. The main property of the core is that if a system of qualitative equations has no solution, then the reduced system built from its core also have no solution. Hence it corresponds to the most difficult part of the constraints to solve. It is obtained by reduction techniques that are very similar to those used in [31] (see details in Sec. 6). As an example, the core of *E. Coli* network only has 28 nodes and 57 edges. It is shown in Fig. 3.

We applied the same inference process as before to this graph. Not surprisingly, we noticed a rather different behavior when inferring signs on a core graph than on a whole graph as demonstrated in Fig. 2. In this case we need much more experiments for inference: sets of expression profiles contain from $N = 50$ to 2000 random profiles.

Two observations can be made from the corresponding statistics of inference. First as can be seen on X-axis, a much greater number of experiments is required to reach a comparable percentage of inference. Correspondingly, the value of p is much smaller than for the full network. This confirms that the core is much more difficult to infer than the rest of network. Second, Fig. 2. displays a much less continuous behavior. More precisely, it shows that for the core,

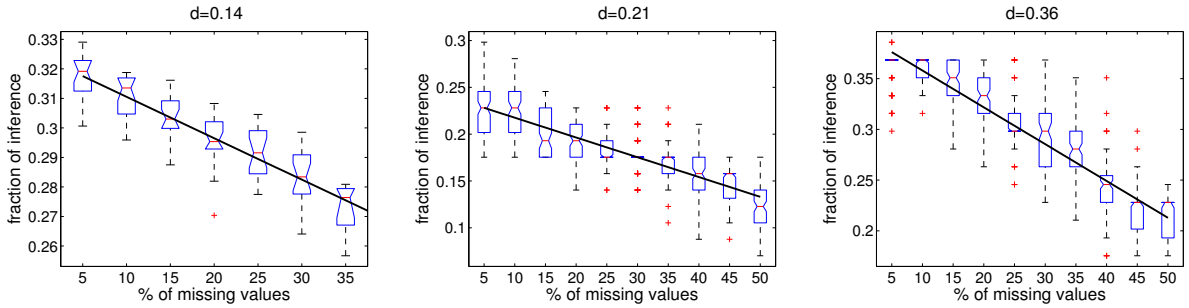


Figure 4. (All) Statistics of inference on the regulatory network of *E. Coli* from *partial* expression profiles. The setting is the same than in Fig. (2), except for the cardinal of an expression profile which is set to a given value, and for the variable on X-axis which is the percentage of missing values in the expression profile. In each case, the dependence between average percentage of inference and percentage of missing values is qualitatively linear. The continuous line corresponds to the theoretical prediction $M_i = M_i^{max} - d * f * M_{total}$, where d is the number of signs interactions that are no longer inferred when a node is not observed, M_i^{max} is the number of inferred interactions for complete expression profiles (no missing values), M_{total} is the total number of nodes and f is the fraction of unobserved nodes.
(Left) Statistics for the whole network (the inference is supposed to be performed from 30 random expression profiles). We estimate $d = 0.14$, meaning that on average, one loses one interaction sign for about 7 missing values.
(Middle) Statistics for the core network (the inference is supposed to be performed from 30 random expression profiles). We estimate $d = 0.21$; the core of the network however is more sensitive to missing data.
(Right) Statistics for the core network (the inference is supposed to be performed from 200 random expression profiles). We estimate $d = 0.35$. Hence, increasing the number of expression profiles increases sensitivity to missing data.

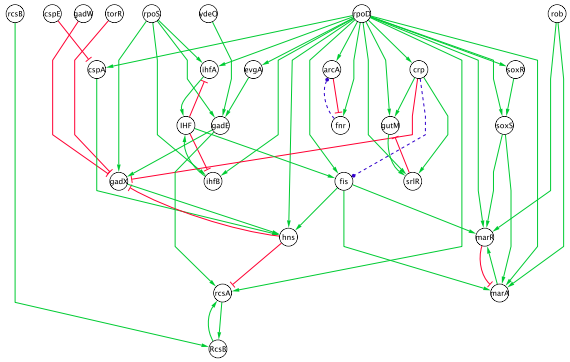


Figure 3. Core of *E. Coli* network. It consists of all oriented loops and of all oriented chains leading to loops. The core contains the dynamical information of the network, hence sign edges are more difficult to infer.

different perturbations experiments have strongly variable impact on sign inference. For instance, the experimental maximum percentage of inference (27 signs over 58) can be obtained already from about 400 expression profiles. But most of datasets with 400 profiles infer only 22 signs.

This suggests that not only the core of the network is more difficult to infer, but also that a brute force approach (multiplying the number of experiments) may fail as well. This situation encourage us to apply experiment design and planning, that is, computational methods to minimize the number of perturbation experiments while inferring a maximal number of regulatory roles.

This also illustrates why our approach is complementary to dynamical modelling. In the case of large scale networks, when an interaction stands outside the core of the graph, then an inference approach is suitable to infer the sign of the interaction. However, when an interaction belongs to the core of the network, then more complex behaviors occur: for instance, the result of a perturbation on the variation of the products might depend on activation thresholds. Then, a precise modelling of the dynamical behavior of this part of the network should be performed [32].

3.3 Influence of missing data

In the previous paragraph, we made the assumption that all proteins in the network are observed. That is, for each experiment each node is assigned a value in $\{+, \mathbf{0}, -\}$. However, in real measurement devices, such as expression profiles, a part of the values is discarded due to technical reasons. A practical method for network inference should cope with missing data.

We studied the impact of missing values on the percentage of inference. For this, we have considered a fixed number of expression profiles ($N = 30$ for the whole *E. Coli* network, $N = 30$ and $N = 200$ for its core). Then, we have randomly discarded a growing percentage of proteins in the profiles, and computed the percentage of inferred regulations. The resulting statistics are shown in Fig. 4.

In both cases (whole network and core), the dependency between the average percentage of inference and the percentage of missing values is qualitatively linear. Simple arguments allow us to find an analytic dependency. If not observing a node implies losing information on d interaction signs, we are able to obtain the following linear dependency $M_i = M_i^{max} - d * f * M_{total}$, where M_i^{max} is the number of inferred interactions for complete expression profiles (no missing values), M_{total} is the total number of nodes, and f is the fraction of unobserved nodes. In order to keep M_{total} non negative, d must decrease with f. Our numerical results imply that the constancy of d and the linearity of the above dependency extend to rather large values of f. This indicates that our qualitative inference method is robust enough for practical use. For the full network we

estimate $d = 0.14$, meaning that on average one loses one interaction sign for about 7 missing values. However, for the same number of expression profiles, the core of the network is more sensitive to missing data (the value of d is larger, it corresponds to lose one sign for about 4.8 missing values). For the core, increasing the number of expression profiles increases d and hence the sensitivity to missing data.

3.4 Application to *E. Coli* network with a compendium of expression profiles

We first validate our method on the *E. Coli* network. We use the compendium of expression profiles publicly available in [9].

For each experimental assay several profiles were available (including a profile for the reference initial state). We processed time series profiles, considering only the last time expression data. For each measured gene, we calculated its average variation in all the profiles of the same experiment. Then, we sorted the measured genes/regulators in four classes: 2-fold induced, 2-fold repressed, non-observed and zero variation, this last class corresponds to genes whose expression did not vary more than 2-fold under an experimental condition. Only the first two classes were used in the algorithm. Obviously this leads us to missing data: there will be edges for which neither the input, nor the output are known. Altogether, we have processed 226 sets of expression profiles corresponding to 68 different experimental assays (over-expression, gene-deletion, stress perturbation).

It appears that the signed network is consistent with only 40 complete profiles of the 68 selected. After discarding the incompatible motifs from the profiles (deleting observations that cause conflicts), 67 profiles remained that were compatible with the signed network. In these 67 expression profiles, 14,47% of the nodes of the network were observed on average as varying. When summing all the observations, we obtained that 9,8% of the edges (input and output) are observed in at least one expression profile. In order to test our algorithm we wipe out the information on edge signs and then try to recover it.

Since the profiles and network were compatible, our algorithm found no ambiguity and predicted 51 signs, i.e. 1,8% of the edges. The naive inference algorithm inferred 43 signs. Hence our algorithm inferred 8 signs, that is 15% of the total of prediction, that were not predicted by the naive algorithm.

Then we applied our algorithm, filtering our inference with different parameters, on the full set of 68 expression profiles including incompatibilities. This time 16% of the network products were observed on average. Several values of the filtering parameter k were used from $k = 1$ to $k = 15$. Without filtering we predicted 183 signs of the network (6,3%), among which 131 were inferred by the naive algorithm. We compared the predictions to the known interaction signs: 77 signs were false predictions (42% of the predictions). A source of the error in the prediction could lie on non-modelled interactions (possibly effects of sigma-factors). Filtering greatly improves our score, allowing us to retain only reliable predictions. Thus, for $k = 15$, we inferred 36 signs, of them, only 3 were incorrect predictions (8% of false prediction). We conclude that filtering is a good way to stronger our predictions even when the model is not precise enough. We illustrate the effect of the filtering process in Fig. 5.

We notice that the inference rate is much more lower in this case than the theoretical inference rate predicted in Sec. 3.3. This shows

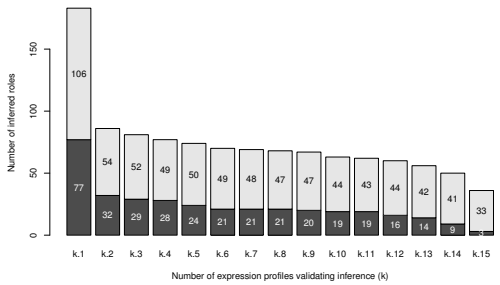


Figure 5. Results of the inference algorithm on *E. Coli* network from a compendium of 68 expression profiles. The profiles were not globally coherent. With no filtering, there are 42% of false predictions. With filtering – keeping only the sign predictions confirmed by k different sets of expression profiles – the rate of false prediction decreases to 8%.

that when the percentage of observation is very low (as it is the case here), the sign-inference process is very dependent from the type of available expression profiles. To overcome this problem, we should take into account more stress perturbation experiments and less genetic perturbation experiments.

Our algorithm also detected ambiguous modules in the network. There are 10 modules of typeI (i.e. single incoming interactions) in the network. Among these interactions, 5 are also stated as ambiguous by the naive algorithm. There are also 6 modules of typeII and III, which are not detected by the naive inference algorithm. All ambiguities are shown in Fig. 6. The list of experimental assays that yields to ambiguities on each interaction is given in the Supplementary Web site. Notice that in RegulonDB, only two of these interactions are annotated with a double-sign, i.e. they are known to have both repressor and inducer effect depending on external condition. On the other 18 interactions belonging to an ambigous module, this analysis shows that there exist non-modelled interactions that balance the effects on the targets.

3.5 A real case: inference of signs in *S. Cerevisiae* transcriptional regulatory network

We applied our inference algorithm to the transcriptional regulatory network of the budding yeast *S. Cerevisiae*. Let us here briefly review the available sources that can be used to build the unsigned regulatory network. The experimental dataset proposed by Lee *et al.* [11] is widely used in the network reconstruction literature. It is a study conducted under nutrient rich conditions, and it consists of an extensive chIP-chip screening of 106 transcription factors. Estimations regarding the number of yeast transcription factors that are likely to regulate specific groups of genes by direct binding to the DNA vary from 141 to 209, depending on the selection criteria. In follow up papers of this work, the chIP-chip analysis was extended to 203 yeast transcription factors in rich media conditions and 84 of these regulators in at least one environmental perturbation [12]. Analysis methods were refined in 2005 by MacIsaac *et al.*[13]. From the same chIP-chip data and protein-protein interaction networks, non-parametric causality tests proposed some previously undetected transcriptional regulatory motifs [14]. Bayesian analysis also proposed transcriptional networks [15, 16, 10]. Here we selected four

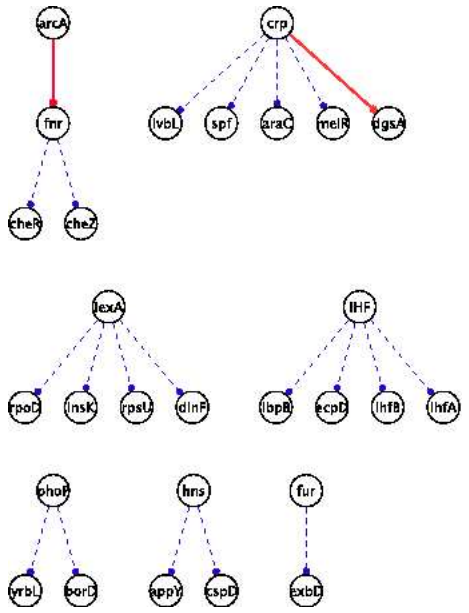


Figure 6. Interactions in the regulatory network of *E. Coli* that are ambiguous with a compendium data of expression profiles [9]. For each interaction, there exist at least two expression profiles that do not predict the same sign on the interaction. In this subnetwork, only 2 interactions (red edges) are annotated with a double-sign in RegulonDB.

of these sources. All networks are provided in the Supplementary Web site.

(A) The first network consists in the core of the transcriptional chIP-chip regulatory network produced in [11]. Starting from the full network with a p-value of 0.005, we reduced it to the set of nodes that have at least one output edge. This network was already studied in [31]. It contains 31 nodes and 52 interactions.

(B) The second network contains all the transcriptional interactions between transcription factors shown by [11] with a p-value below 0.001. It contains 70 nodes and 96 interactions.

(C) The third network is the set of interactions among transcription factors as inferred in [13] from sequence comparisons. We have considered the network corresponding to a p-value of 0.001 and 2 bindings (83 nodes, 131 interactions).

(D) The last network contains all the transcriptional interactions among genes and regulators shown by [11] with a p-value below 0.001. It contains 2419 nodes and 4344 interactions.

3.5.1 Inference process with gene-deletion expression profiles
We first applied our inference algorithm to the large-scale network (D) extracted from [11] using a panel of expression profiles for 210 gene-deletion experiments [40]. The information given by this panel is quite small, since $1,6\%$ of all the products in the network is on average observed, and 12% of the edges (input and output) of the network are observed in at least one expression profile. Using this data, we obtain 162 regulatory roles.

We validated our prediction with a literature-curated network on Yeast [41]. We found that among the 162 sign-predictions, 12 were referenced with a known interaction in the database, and 9 with a good sign.

Gene-deletion expression profiles were used so we could compare our results to path analysis methods [23, 20] since the latter can only be applied to knock-out data (http://chianti.ucsd.edu/idekerlab/). Other sign-regulation inference methods need either other sources of gene-regulatory information (promoter binding information, protein-protein information), or time-series data to be performed [15, 18, 10].

Before comparing our inference results to the work of Yeang *et al.*, we tested the compatibility between their inferred network with the 210 gene-deletion experiments. We obtained that their network was incompatible with 28 of the 210 experiments. The comparison of both results showed us that the method of Yeang *et al.* infers 234 roles of widely connected paths, while our method infers 162 roles in the branches of the network. Both results intersect on 17 interactions, and no contradiction in the inferred role was reported. An illustration of these results is given in the Supplementary Web site.

This suggests that our approach is complementary to path analysis methods. Our explanation is the following: In [23, 20], network inference algorithms identify probable paths of physical interactions connecting a gene knockout to genes that are differentially expressed as a result of that knockout. This leads to search for the smallest number of interactions that carry the largest information in the network. Hence, inferred interactions are located near the core

Experiment Identifier	Description	Reference
E1	Diauxic Shift	[30]
E2	Sporulation	[33]
E3	Expression analysis of Snf2 mutant	[34]
E4	Expression analysis of Swi1 mutant	[34]
E5	Pho metabolism	[35]
E6	Nitrogen Depletion	[28]
E7	Stationary Phase	[28]
E8	Heat Shock from 21°C to 37°C	[28]
E9	Heat Shock from 17°C to 37°C	[28]

Experiment Identifier	Description	Reference
E10	Wild type response to DNA-damaging agents	[36]
E11	Mec1 mutant response to DNA-damaging agents	[36]
E12	Glycosylation defects on gene expression	[37]
E13	Cells grown to early log-phase in YPE (Rich medium with 2% of Ethanol)	[29]
E14	Cells grown to early log-phase in YPG (Rich medium with 2% of Glycerol)	[29]
E15	Titratable promoter alleles - Ero1 mutant	[38]

Table 3. List of genome expression experiments of *S. Cerevisiae* used in the inference process. Experiments contain information on steady state shift and their curated data is available in SGD (Saccharomyces Genome Database) [39].

Interaction network	Nodes	Edges	Average number of observed nodes	In/Out observed simulnat.	Inferred signs	MBM Int. TypeI	MBM Int. Type II,III,IV	Total Inf. rate	Predictions of the naive algorithm
(A) Core of Lee transcriptional network [11, 31]	31	52	28%	46	11 (21.1%)	3 (5.7%)	0	26.8%	11%
(B) Extended Lee transcriptional network [11]	70	96	26%	70	29 (30.2%)	7 (7.2%)	0	37.4%	15,6%
(C) Inferred network [12, 13] threshold = 0.001 ; bindings=2	83	131	33%	91	21 (16%)	4 (3%)	0	19%	11%
(D) Global transcriptional network [11] p-value = 0.001	2419	4344	30%	2270	631 (14.5%) 198 (4.5%) filter k=3	281 (6.5%) no filter	463 (11%)	32%	13.9%

Table 4. Budding yeast transcriptional regulatory networks on which the sign inference algorithm was applied. For each network 14 or 15 different expression profiles were used for calculating the inference. The set of observations provided by one expression profile, was composed by at least two expressed/repressed (ratio over/under 2-fold) genes of the network. The *Input/Output observed simultaneously* column, is an indicator of the maximum possible number of sign-inferred interactions. There are three different inference results: *Inferred signs*, signs fixed in a unique way by all experiments, *MBM Interactions of TypeI*, the set of non-repeated interactions that belong to all the multiple behavior modules of TypeI detected, and *MBM Interactions of TypeII,II,IV*, the number of non-repeated interactions belonging to MBM of Type II,III,IV. For all the inference results a percentage concerning the total number of edges of the network, is calculated. The *Total inference rate* represents the percentage of the total number of edges that was inferred (inferred signs plus interactions in MBM). It is compared to the results of the naive algorithm.

of the network (even though not exactly in the core). On the contrary, as we already detailed it, the combinatorics of interaction in the core of the network is too intricate to be determined from a few hundreds of parse expression profiles with our algorithm, and we concentrate on interactions around the core.

3.5.2 Inference with stress perturbation expression profiles In order to overcome the problem raised by the small amount of information contained in [40], we have selected stress perturbation experiments. This data corresponds to curated information available in SGD (Saccharomyces Genome Database) [39]. When time series profiles were available, we selected the last time expression array. Therefore, we collected and treated 15 sets of arrays described in Table 3. For each expression array, we sorted the measured genes/regulators in four classes: 2-fold induced, 2-fold repressed, non-observed and zero variation. We were only interested in the expression of genes that belong to any of the four networks we studied. Full datasets are available in the Supplementary Web site.

As for *E. Coli* network, it appeared that all networks (A), (B), (C) and (D) are not consistent with the whole set of expression arrays and ambiguities appeared. We performed our inference algorithm. We identified motifs that hold ambiguities, and we marked them as Multiple Behavior Modules of type I, II and III, as described in Sec. 3.1. The algorithm also generates a set of inferred signs. Then we applied the filtered algorithm (with filter $k = 3$) to the large-scale network (D).

We obtain our total inference rate adding the number of inferred signs fixed in a unique way to the number of non-repeated interactions that belong to all the detected multiple behavior modules and dividing it by the number of edges in the network. In Table 4 we show the inference rate for Networks (A), (B), (C) and (D). Depending on the network, the rate of inference goes from 19% to 37%. Hence, the rates of inference are very similar to the theoretical rates obtained for *E. Coli* network, still with a small number of perturbation experiments (14 or 15).

We validated the inferred interaction by comparing them to the literature-curated network published in [41]. We first obtained that among the 631 interactions predicted when no filtering is applied, 23 are annotated in the network, and seven annotations are contradictory to our predictions. However, among the 198 interactions predicted with a filter parameter $k = 3$, 19 are annotated in the network, and only one annotation is contradictory to our predictions. As in the case of *E. Coli*, we conclude that filtering is a good way to make strong predictions even when the model is not precise enough. We also compared the sign predictions to the predictions of the naive inference algorithm. We found that the naive algorithm usually predicts half of the signs that we obtain. In Fig. 7 we illustrate the inferred interactions for Network (B), that is, the Transcriptional network among transcription factors produced in [11].

As mentioned already, the algorithm identified a large number of ambiguities. The exhaustive list of MBM is given in the Supplementary Web site. We notice that MBM of Type I are detected in the four networks; we list the Type I modules of size 2 found for the networks (A), (B) and (C) in Table 5. In contrast, MBM of Type II, III and IV are only detected, in an important number, for Network (D) following the distribution: 85.4% of Type II, 5.3% of Type III and 9.3% of Type IV. In network (D), all the results were obtained after 3 iterations of the inference algorithm. For each MBM, a precise biological study of the species should allow to understand the origin of the ambiguity: error in expression data, missing interaction in the model or changing in the sign of the interaction during the experimentation.

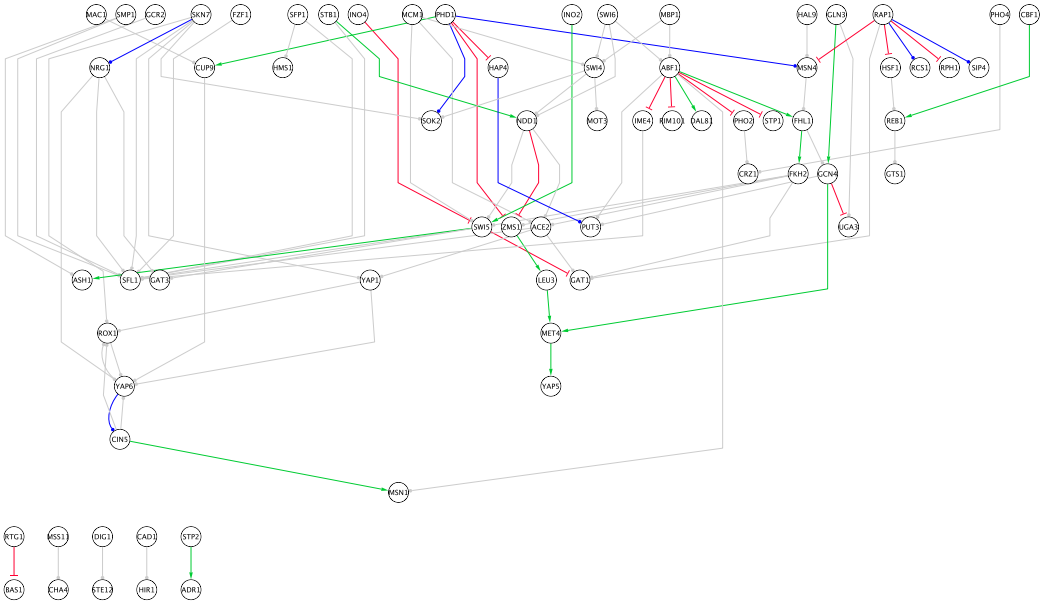


Figure 7. Transcriptional regulatory network among transcription factors (70 nodes, 96 edges) extracted from [11]. A total of 29 interactions were inferred: arrows in green, respectively in red, correspond to positive, respectively negative, interactions inferred; blue arrows correspond to the detected multiple behavior modules of TypeI. Diagram layout is performed automatically using the Cytoscape package [42].

3.6 Contribution of expression profiles to the inference

In order to evaluate the contribution of the 14 experiments used for the inference in the global network provided in [11] (2419 nodes and 4344 arcs), we addressed the following question: assuming that all inferred roles are correct, which is the experiment that causes the suppression of most of the inferred roles? For example, in Fig. 1 expression data related to YPD Broth to Stationary Phase [28], caused the suppression of the inferred interactions of the module of Type II.

We compared the 14 expression profiles according to the MBM of TypeII, III and IV that are detected by using an element of the dataset. MBM of TypeI are not included in this computation, since they do not invalidate any interaction role, as no interaction role is inferred before their detection. The results of this comparison are shown in Fig. 8. The fourth chart illustrates that the real contribution of each expression profile does not depends on the amount of observations.

4 DISCUSSION

In this work we show how a qualitative reasoning framework can be used to infer the role of transcription factor based on expression profiles. The regulatory effect of a transcription factor on its target genes can either be an activation or a repression. Our framework

Interaction network	Actor	Target	Experiment 1	Experiment 2
Core of Lee network	YAP6	CIN5	Expression during Sporulation [33]	YPD Broth to Stationary Phase [28]
	GRF10	MBP1	YPD Broth to Stationary Phase [28]	Mec1 mutant + Heat [36]
	PDH1	MSN4	Nitrogen Depletion [28]	Heat shock 21 to 37 [28]
Extended Lee network	YAP6	CIN5	Expression during Sporulation [33]	YPD Broth to Stationary Phase [28]
	RAP1	SIP4	Expression during Sporulation [33]	Expression during the diauxic shift [30]
	SKN7	NRG1	YPD Broth to Stationary Phase [28]	Expression during the diauxic shift [30]
	PHD1	SOK2	Heat shock 21 to 37 [28]	YPD Broth to Stationary Phase [28]
	RAP1	RCS1	Wild type + Heat [36]	Transition from fermentative to glycerol-based respiratory growth [29]
	PHD1	MSN4	Nitrogen Depletion [28]	Heat shock 21 to 37 [28]
	HAP4	PUT3	Expression during the diauxic shift [30]	Snf2 mutant, YPD [34]
MacIssac inferred network	SWI5	ASH1	Expression regulated by the PHO pathway [35]	YPD Broth to Stationary Phase [28]
	SKN7	NRG1	YPD Broth to Stationary Phase [28]	Nitrogen Depletion [28]
	NRG1	YAP7	Expression regulated by the PHO pathway [35]	Transition from fermentative to glycerol-based respiratory growth [29]
	NRG1	GAT3	Glycosylation [37]	Transition from fermentative to glycerol-based respiratory growth [29]

Table 5. Result of the diagnosis procedure for three networks related to budding yeast *S. Cerevisiae* (core, extended transcriptional networks of Lee, inferred network of MacIsaac). We found ambiguities between single interactions and pairs of data (we call them Multiple Behavior Modules of Type I and size 2). For each ambiguous interaction found, we list two experiments that deduce a different role of interaction among these genes.

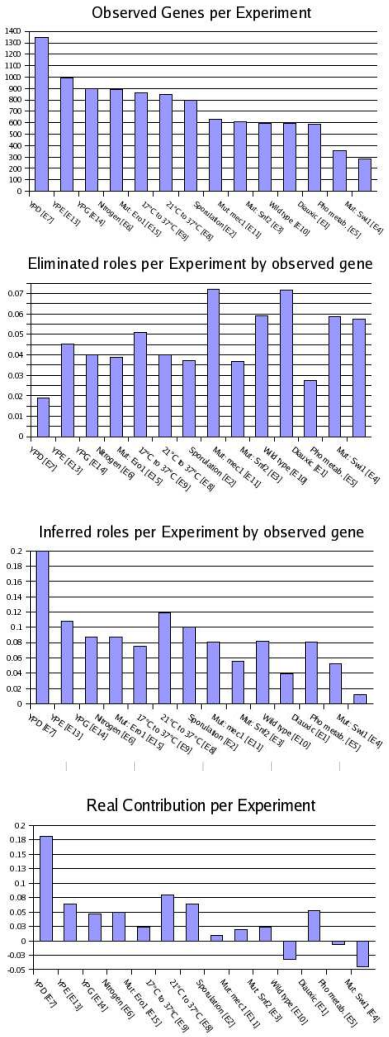


Figure 8. Comparison of 14 experiments used in the sign-inference process for the global transcriptional network in [11] (2419 genes, 4344 interactions). Each experiment has a twofold contribution: it spots inconsistent modules (MBM, that are further excluded from inference) and it predicts interaction roles. Some experiments have more predictive power, just because they include more genes. In order to normalize the predictive power, we divide the percentage of predictions by the percentage of observed nodes. For each experiment we have estimated, from top to down: (First) Number of 2-fold expressed or 2-fold repressed genes. (Second) Percentage of edges in the spotted MBMs of type II,III,IV divided by the percentage of observed nodes. (Third) Percentage of inferred interactions divided by the percentage of observed nodes. (Fourth) Real contribution of each experiment, calculated by subtracting the third quantity (inference) from the second quantity (eliminated inconsistency); negative values correspond to experiments whose main role is to spot ambiguities.

models a single qualitative rule, which basically says that the variation of expression for a gene should be explained by at least one of its regulators.

While intuitive and simple, this rule is sufficient to infer a significant number of regulatory effects from a reasonable amount of expression profiles.

On computational grounds, we designed algorithms that are able to cope with systems consisting of several thousands of genes. Our methods can thus readily be applied to networks and expression data that are produced by current high-throughput measurement techniques.

Inferring the role of transcription factor from expression profile can be seen as a particular case of network reconstruction. Let us now review some of the most relevant approaches in this domain.

Looking for high correlation or mutual information in expression profiles [16, 43] can be used to find interactions among genes. Much progress has been done over the past few years to improve the quality of statistical estimators or to detect indirect correlations, and some promising results were obtained in higher eukaryotes [43]. There remains some open problems however. First, the relation between network structure and correlation is not one to one (inference procedures rely on calculating pseudoinverses of singular matrices). Consequently, many false positive or false negatives exist among the inferred interactions. Moreover, the orientation of the inferred interactions (A acts on B) is impossible to tell if both A and B are transcription factors. Other non-parametric statistical methods are designed to test hypothetical causality relations [14].

Bayesian networks have been widely applied to gene network reconstruction [44]. Though it is limited to the class of acyclic graphs (regulation loops are excluded), the framework of Bayesian networks is attractive because it offers an intuitive, graphical representation of regulatory networks, and a simple way to deal with stochasticity in regulatory networks. This approach is however demanding, both in computational resources and experimental measurements.

Segal and coworkers [15] proposed a probabilistic model to infer transcriptional networks from promoter sequences and gene expression data. They introduce a principled framework to integrate heterogeneous sources of information. Computing the most probable model in this setting requires to solve a hard non linear optimization problem.

Network inference based on ordinary differential equation relates changes in RNA concentration to each other and to an external perturbation [45, 46]. AS ODE's are deterministic, the inferred interactions represent influences and not statistical dependencies as the other methods. It yields signed directed graphs. The main restriction is that it requires knowledge on the perturbed gene in each experiment.

More recently, some methods focused on paths of interactions [19, 20]. Global expression profiles are used to validate models of transcriptional regulation inferred from protein-protein interaction, genome-wide location analysis and expression data. A network inference algorithm identifies probable paths of physical interactions connecting a gene knockout to genes that are differentially expressed as a result of that knockout. These methods are really dependent on the topology of the networks: complex networks in which many competing or alternative paths connect a knockout to differentially expressed genes may be difficult to infer. Then, dynamical Boolean analysis is efficient to infer competing behaviors

on models containing tens of products [20, 31]. The main restriction to this method is that expression profiles have to result from a gene-deletion perturbation.

In this work, we rely on a discrete modeling framework, which consists in calculating an over-approximation of the set of possible observations, by abstracting noisy quantitative values into more robust properties. In contrast, statistical methods deal with experimental noise by explicitly modeling the noise distribution, provided enough measurements are available – which usually means hundreds of independent experiments. Moreover while most methods report the most likely model given the data, we describe the (possibly huge) set of consistent experimental behaviors with a system of qualitative constraints. Then we look for invariants in this set. In the worst case, not a single regulatory effect can be deduced from the set of constraints, whereas computing the most likely model provides with signs for all regulations. However, we expect the inferred regulations to be more robust. Another crucial difference is that the system of constraints might have no solution at all. In combination with a diagnosis procedure, we illustrated how this approach can be a relevant tool for the curation of network databases.

We compared our inference approach to a naive inference algorithm and path analysis methods introduced in [23, 20]. As detailed above, all other inference methods need additional information to infer the signs of regulations. We found that both our algorithm and path analyses infer non-trivial interactions. Both approaches are complementary: path analyses identify coupled with boolean analysis allows to infer the signs of interactions located in paths that are connected to a large number of targets; whereas our method yields information on paths connected to a quite small number of targets. Another difference is that paths analysis requires gene-deletion perturbation expression profiles, while our method give better results with stress perturbation experiments (though it can be applied to any type of experiment).

Using simulations we investigated the dependence between the number of inferred signs and the number of available observations. Not surprisingly we noticed that the topology of the regulatory graph alone had a strong influence on the estimated relationship. This was illustrated by computing statistics both on a complete regulatory network and on its core, as defined in the Methods section. The complete network is characterized by an over-representation of feedback-free regulatory cascades, which are controlled by a small number of transcription factors. In this setting, the number of inferred signs grows quasi continuously with the number of observations. In contrast, the core network does not obey the simple law "the more you observe, the better": some expression profiles are clearly more informative than others. A challenging sequel to this work deals with experimental planification: given some control parameters, how to find the most informative experiments while keeping their number as low as possible ?

As a practical assessment of our method, we conducted sign inference experiments on *E. Coli* and *S. Cerevisiae*, using curated expression measurements, and regulatory networks either already published or based on chIP-chip data. When expression profiles mostly consisted in genetic perturbations, the inference rate was quite low, even though comparable to the results of paths analysis [20]. When expression profiles consisted in stress perturbation, our inference results corresponded to the theoretical rate of inference. For smaller networks, of about 100 interactions, we were able to infer 20% of the regulatory roles. For bigger networks, of thousands interactions, we were only able to infer the 14%, however, a huge number of inconsistencies (that we called multiple behaviour modules) were detected. Even if we were able to state some corrections over the model or data, all our inferences and corrections proposed depend on the model we worked with. If the orientation sense of some interaction was mistaken, our inferences will be mistaken as well. In our opinion, what is even more relevant than correctly inferring signed regulations among genes is the ability to detect and isolate situations where different data sources are not consistent with each other. Moreover, if we group some of the MBM found according to the common genes they share, it is possible to assign a higher relevance to the correction of some specific interaction or data; in other words, it is possible to choose which of all the interactions is the most inappropriate.

5 CONCLUSION

In this work, we showed that our approach is suitable to infer regulatory roles of transcription factor from a *limited* amount of data. More precisely, we could infer 30-40% of the networks we studied from about 20-30 perturbation expression arrays. We believe that our approach is complementary to previous statistical methods: while qualitative modeling is a less accurate description of regulatory networks, it requires less data in order to make robust predictions. Thus, it is more adapted to situations where diverse but even limited expression profiles (some tens) are available, instead of the large panel of expression profiles usually needed for statistical methods.

We proposed a characterization of sub-networks that are more difficult to infer, called the *core* of a network. We showed on simulated data that in these core networks an unfeasible number of experiments is necessary to infer a small number of signs with high probability. For these core networks, two different strategies may be adopted. The first strategy is to build a more accurate model for these restricted subnetworks, using dynamic modeling techniques (see ([32] for a review), The alternative is to develop experiment design in our qualitative framework: find suitable values for control parameters to infer the maximum number of signs.

Finally, we illustrated another advantage of discrete modeling, namely that models can be submitted to exhaustive verification and diagnosis. As we show it in this paper it is possible to reason on systems with thousands of observations, constraints and variables, and provide intuitive diagnosis representations automatically when expression profiles happen to be ambiguous with the regulation model. As a follow-up to this work, we plan to deepen diagnosis representation, and eventually propose automatic hypothesis generation for the existence of defects.

6 METHODS

Problem statement We consider the set of equations derived from a given interaction graph G:

$$X_i^k \approx \sum_{j \to i} S_{ji} X_j^k \text{ for } 1 \leq i \leq n,\ 1 \leq k \leq r \tag{4}$$

where X_i^k stands for the sign of variation of species i in experiment k, and S_{ji} the sign of the influence of species j on species i. Recall that the graph G itself comes from chIP-chip experiments or sequence analysis. Using expression arrays, we obtain an experimental value for some variables X_i^k, which

will be denoted x_i^k; more generally uppercase (resp. lowercase) letters will stand for variables of the systems (resp. constants **+**, **–** or **0**).

A single equation in the system (4) can be viewed as a predicate $P_{i,k}(X, S)$ where i denotes a node in the graph and k one of the r available experiments. If the value for some variables in the equation is known, the predicate resulting from their instantiation will be denoted $P_{i,k}(X, S)[x^k, s]$.

Our problem can now be stated as follows: given a set of expression profiles $x^1, \ldots, x^r$, decide if the predicate:

$$P(X, S) = \bigwedge_{1 \leq i \leq n,\ 1 \leq k \leq r} P_{i,k}(X, S)[x^k] \tag{5}$$

can be satisfied. If so, find all variables that take the same value in all admissible valuations (so called *hard components* of the system).

Decision diagram encoding In a previous work [26], we showed how the set of solutions of a qualitative system can be computed as a decision diagram [47]. A decision diagram is a data structure meant to represent functions on finite domains ; it is widely used for the verification of circuits or network protocols. Using such a compact representation of the set of solutions, we proposed efficient algorithms for computing solutions of the systems, hard components, and other properties of a qualitative system. Back to our problem, in order to predict the regulatory role of transcription factors on their target genes, it is enough to compute the decision diagram representing the predicate (5), and compute its hard components as proposed in [26]. This approach is suitable for systems of at most a couple of hundred variables. Above this limit, the decision diagram is too large in memory complexity. In our case however, we consider systems of about 4000 variables at most, which is far too large for the above mentioned algorithms.

In order to cope with the size of the problem, we propose to investigate a particular case, when all species are observed, in all experiments. In this case, $i \neq j$ implies that $P_{i,k}(X, S)[x^k]$ and $P_{j,k}(X, S)[x^k]$ share no variables. This means that P may be satisfied if and only if each predicate

$$P_{i,\cdot}(S) = \bigwedge_{1 \leq k \leq r} \exists X\ P_{i,k}(X, S)[x^k] \tag{6}$$

may be satisfied. As a consequence, a variable S_{ji} is a hard component of P if and only if it is a hard component of $P_{i,\cdot}$. $P_{i,\cdot}$ correspond to the constraints which relate species i to its predecessors in G for all experiments. The number of variables in $P_{i,\cdot}$ is exactly the in-degree of species i in G, which is at most 10-20 in biological networks.

As soon as some species are not observed in some experiment, the predicates $P_{i,\cdot}$ share some variables and it is not guaranteed to find all hard components by studying them separately. A brief investigation showed (data not shown) that due to the topology of the graph, most of the equations are not independent any more, even with few missing nodes. Note however, that any hard component of $P_{i,\cdot}$ still is a hard component of P. The same statement holds for

$$P_{\cdot,k}(X) = \bigwedge_{1 \leq i \leq n} \exists S\ P_{i,k}(X, S)[x^k] \tag{7}$$

where $P_{\cdot,k}$ corresponds to the constraints that relate all species in G for a *single* experiment. Relying on this result, we implemented the following algorithm

In practice, this algorithm is very effective in terms of computation time and number of hard components found. However, as already stated, it is not guaranteed to find all hard components of P. This is what motivates the technique described in the next paragraph.

Solving with Answer Set Programming In order to solve large qualitative systems, we also tried to encode the problem as a logic program, in the setting of answer set programming (ASP). While decision diagrams represent the set of *all* solutions, finding a model for a logic program provides *one* solution. In order to find hard components, it is enough to check for each variable V, if there exists a solution such that $V =$ **+** and another solution

Input:
the predicates $P_{i,\cdot}$ and $P_{\cdot,k}$ for all i and k
observed variations x
Output:
a set s of hard components of P

$s \leftarrow \emptyset$
while *True* **do**
  $s' \leftarrow \bigcup_i hard\_components(P_{i,\cdot}[x^k, s])$
  if $s' = \emptyset$ **then return** s
  $s \leftarrow s \cup s'$
  $x' \leftarrow \bigcup_k hard\_components(P_{\cdot,k}[x^k, s])$
  if $x' = \emptyset$ **then return** s
  $x \leftarrow x \cup x'$
end

Algorithm 1: Heuristic for finding hard components in large interaction networks with many expression profiles.

such that $V =$ **–**. The ASP program we used in order to solve the qualitative system is given in supplementary materials. In the following we will denote by $asp\_solve(P)$ the call to the ASP solver on the predicate P. The returned value is an admissible valuation if there is one, or $\perp$ otherwise. The complete algorithm is reported below

Algorithm: Hard components using ASP
Input:
the predicates P
observed variations x
Output:
a set h of hard components of P

$h \leftarrow \emptyset$
$C \leftarrow \{S_{ji} | j \rightarrow i\}$
$s^* \leftarrow asp\_solve(P)$
if $s^* = \perp$ **then return** $\perp$
while $C \neq \emptyset$ **do**
  choose V in C
  $s \leftarrow asp\_solve(P[V = -s^*_V])$
  if $s = \perp$ **then**
    $h \leftarrow \{(V, s_V)\} \cup h$
  else
    delete from C all W in C s.t. any $s^*_W \neq s_W$
  end
end

Algorithm 2: Exact algorithm for finding the set of hard components of P, based on logic programming.

We use `clasp` for solving ASP programs [48], which performs astonishingly well on our data. The procedure described in Algorithm 2 is particularly efficient to find *non* hard components: generating one solution may be enough to prove non hardness of many variables at a time.

To sum up, in order to solve a system of qualitative equations (4) with only partial observations, we use Algorithm 1 first and thus determine most (if not all) hard components. Then, Algorithm 2 is used for the remaining components, which are nearly all non hard.

Reduction technique As mentioned in the Result section, interaction graphs may be reduced in a way that preserves the satisfiability of the associated qualitative system. Consider a graph G with defined signs on its edges. If

some node n has no successor, then delete it from G. Note then, that any solution of the qualitative system associated to the new graph can be extended in a solution to the system associated to G. The same statement holds if one iteratively delete all nodes in the graph with no successor. The result of this procedure is the subgraph of G such that any node is either on a cycle, or has a cycle downstream. We refer to it as the core of the interaction graph.

The core of an interaction graph corresponds to the most difficult part to solve, because extending a solution for the core to the entire graph can be done in polynomial time, using a breadth-first traverse.

Diagnosis for noisy data When working with real-life data, it may happen that the predicate P defined in Eq. (5) cannot be satisfied. This may be due to three (non exclusive) reasons:

- a reported expression data is wrong
- an arrow (or more generally a subgraph) is missing
- the sign on an edge depends on the state of the system

In the third case, the conditions for deriving Eq. (1) are not fulfilled for one node and its qualitative equation should be discarded. This, however, does not affect the validity of the remaining equation.

In all cases, isolating the cause of the problem is a hard task. We propose the following diagnosis approach: as P is a conjunction of smaller predicates, it might happen that some subsets of the predicates are not satisfiable yet. Our strategy is then to find a "small" subsets of predicates which cannot be satisfied. A particularly interesting feature of this approach is that by selecting subsets of $P_{i,\cdot}$ predicates, the result might directly be interpreted and visualized as a subgraph of the original model.

How to determine if a sign can be inferred In section 2, we have seen some examples showing that even when all feasible observations are available, it might not be possible to infer all signs in the interaction graph. Whether or not a sign can be inferred depends on the topology of the graph, but also on the actual signs on interactions. In practice, it is thus impossible to tell from the unsigned graph only if a sign can be recovered. However, it is still interesting to evaluate on fully signed interaction networks which part can be inferred. A trivial algorithm for this consists in explicitly generating all feasible observations and use the algorithms described above. This is unfeasible due to the number of observations.

With the notations introduced above, consider an observation X and sign variables S for an interaction graph. $P_i(X,S)$ denotes the constraint that link the variation of a node i to that of its predecessors given the signs of the interactions. Moreover, the real signs in the graph are denoted by s. For each node i, we build the predicate giving the feasible observations on node i and its predecessors, given the rest of the graph and the real signs s

$$O_i(X) = \exists X_{j\, j\in\{i\}\cup \mathrm{pred}(i)} \bigwedge_{1\leq i\leq n} P_i(X,s)$$

Then, the constraint that we can derive on S variables is: for any observation X that is feasible $P_i(X,S)$ should hold. This constraint is more formally defined by

$$C_i(S) = \forall X O_i(X) \Rightarrow P_i(X,S)$$

Finally, the hard components of C_i are exactly the signs that can be inferred using *all* feasible observations. Let us sum up the procedure:

1. compute $P(X,S) = \bigwedge_{1\leq i\leq n} P_i(X,s)$
2. compute O_i from P and the actual signs s
3. compute C_i, the constraints of signs given all feasible observations
4. compute the hard components of C_i, which are exactly the signs that can be inferred.

If it is not possible to compute $P(X,S)$ (mainly because the interaction graph is too large), we use a more sophisticated approach based on a modular decomposition of the interaction graph. The resulting algorithm can be found in the Supplementary materials.

SUPPLEMENTARY MATERIAL

Inference algorithms and all the results obtained for the *S. cerevisiae* regulatory network can be found at:
www.irisa.fr/symbiose/interactionNetworks/supplementaryInference.html

ACKNOWLEDGMENT

The authors are particularly grateful to B. Kauffman, M. Gebser and T. Schaub from the University of Potsdam for their help on CLASP software. They also wish to thank the referee for their interesting and constructive remarks.

REFERENCES

[1]Zanzoni A, Montecchi-Palazzi L, Quondam M, Ausiello G, Helmer-Citterich M, et al. (2002) MINT: a Molecular INTeraction database. FEBS Lett 513:135–40.
[2]Hermjakob H, Montecchi-Palazzi L, Lewington C, Mudali S, Kerrien S, et al. (2004) IntAct: an open source molecular interaction database. Nucleic Acids Res 32:D452–5.
[3]Peri S, Navarro JD, Kristiansen TZ, Amanchy R, Surendranath V, et al. (2004) Human protein reference database as a discovery resource for proteomics. Nucleic Acids Res 32:D497–501.
[4]Kanehisa M, Goto S, Kawashima S, Okuno Y, Hattori M (2004) The KEGG resource for deciphering the genome. Nucleic Acids Res 32:D277–80.
[5]Ingenuity-Systems (1998). Ingenuity pathways knowledge base. Avalaible: http://www.ingenuity.com.
[6]Salgado H, Gama-Castro S, Peralta-Gil M, Diaz-Peredo E, Sanchez-Solano F, et al. (2006) RegulonDB (version 5.0): Escherichia coli K-12 transcriptional regulatory network, operon organization, and growth conditions. Nucleic Acids Res 34:D394–7.
[7]Reguly T, Breitkreutz A, Boucher L, Breitkreutz BJ, Hon GC, et al. (2006) Comprehensive curation and analysis of global interaction networks in Saccharomyces cerevisiae. J Biol 5:11.
[8]Joyce AR, Palsson BO (2006) The model organism as a system: integrating 'omics' data sets. Nat Rev Mol Cell Biol 7:198–210.
[9]Faith JJ, Hayete B, Thaden JT, Mogno I, Wierzbowski J, et al. (2007) Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles. PLoS Biology 5.
[10]Bansal M, Belcastro V, Ambesi-Impiombato A, di Bernardo D (2007) How to infer gene networks from expression profiles. Mol Syst Biol 3.
[11]Lee TI, Rinaldi NJ, Robert F, Odom DT, Bar-Joseph Z, et al. (2002) Transcriptional regulatory networks in Saccharomyces cerevisiae. Science 298:799–804.
[12]Harbison CT, Gordon DB, Lee TI, Rinaldi NJ, Macisaac KD, et al. (2004) Transcriptional regulatory code of a eukaryotic genome. Nature 431:99–104.
[13]MacIsaac KD, Wang T, Gordon DB, Gifford DK, Stormo GD, et al. (2006) An improved map of conserved regulatory sites for Saccharomyces cerevisiae. BMC Bioinformatics 7:113.
[14]Xing B, van der Laan MJ (2005) A causal inference approach for constructing transcriptional regulatory networks. Bioinformatics 21:4007–13.
[15]Segal E, Shapira M, Regev A, Pe'er D, Botstein D, et al. (2003) Module networks: identifying regulatory modules and their condition-specific regulators from gene expression data. Nat Genet 34:166–76.
[16]Nariai N, Tamada Y, Imoto S, Miyano S (2005) Estimating gene regulatory networks and protein-protein interactions of Saccharomyces cerevisiae from multiple genome-wide data. Bioinformatics 21 Suppl 2:ii206–ii212.
[17]Ferrazzi F, Magni P, Sacchi L, Nuzzo A, Petrovic U, et al. (2007) Inferring gene regulatory networks by integrating static and dynamic data. Int J Med Inform Epub 2007 Sept 6.
[18]S B, Eils R (2005) Inferring genetic regulatory logic from expression data. Bioinformatics 21:2706–13.
[19]Ideker T (2004) A systems approach to discovering signaling and regulatory pathways–or, how to digest large interaction networks into relevant pieces. Adv Exp Med Biol 547:21–30.
[20]Yeang CH, Mak HC, McCuine S, Workman C, Jaakkola T, et al. (2005) Validation and refinement of gene-regulatory pathways on a network of physical interactions. Genome Biol 6:R62.
[21]Gutierrez-Rios RM, Rosenblueth DA, Loza JA, Huerta AM, Glasner JD, et al. (2003) Regulatory network of Escherichia coli: consistency between literature knowledge and microarray profiles. Genome Res 13:2435–2443.

[22]Bulyk ML (2006) DNA microarray technologies for measuring protein-DNA interactions. Curr Opin Biotechnol 17:422–30.

[23]Yeang CH, Ideker T, Jaakkola T (2004) Physical network models. J Comput Biol 11:243–62.

[24]Radulescu O, Lagarrigue S, Siegel A, Veber P, Borgne ML (2006) Topology and static response of interaction networks in molecular biology. J R Soc Interface 3:185–96.

[25]Siegel A, Radulescu O, Borgne ML, Veber P, Ouy J, et al. (2006) Qualitative analysis of the relation between DNA microarray data and behavioral models of regulation networks. Biosystems 84:153–74.

[26]Veber P, Borgne ML, Siegel A, Lagarrique S, Radulescu O (2004/2005) Complex qualitative models in biology: A new approach. Complexus 2:140–151.

[27]Guziolowski C, Veber P, Borgne ML, Radulescu O, Siegel A (2007) Checking consistency between expression data and large scale regulatory networks: A case study. Journal of Biological Physics and Chemistry :In press.

[28]Gasch AP, Spellman PT, Kao CM, Carmel-Harel O, Eisen MB, et al. (2000) Genomic expression programs in the response of yeast cells to environmental changes. Mol Biol Cell 11:4241–57.

[29]Roberts GG, Hudson AP (2006) Transcriptome profiling of Saccharomyces cerevisiae during a transition from fermentative to glycerol-based respiratory growth reveals extensive metabolic and structural remodeling. Mol Genet Genomics 276:170–86.

[30]DeRisi JL, Iyer VR, Brown PO (1997) Exploring the metabolic and genetic control of gene expression on a genomic scale. Science 278:680–6.

[31]Kauffman S, Peterson C, Samuelsson B, Troein C (2003) Random Boolean network models and the yeast transcriptional network. Proc Natl Acad Sci U S A 100:14796–9.

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

[33]Chu S, DeRisi J, Eisen M, Mulholland J, Botstein D, et al. (1998) The transcriptional program of sporulation in budding yeast. Science 282:699–705.

[34]Sudarsanam P, Iyer VR, Brown PO, Winston F (2000) Whole-genome expression analysis of snf/swi mutants of Saccharomyces cerevisiae. Proc Natl Acad Sci U S A 97:3364–9.

[35]Ogawa N, DeRisi J, Brown PO (2000) New components of a system for phosphate accumulation and polyphosphate metabolism in Saccharomyces cerevisiae revealed by genomic expression analysis. Mol Biol Cell 11:4309–21.

[36]Gasch AP, Huang M, Metzner S, Botstein D, Elledge SJ, et al. (2001) Genomic expression responses to DNA-damaging agents and the regulatory role of the yeast ATR homolog Mec1p. Mol Biol Cell 12:2987–3003.

[37]Cullen PJ, Sabbagh WJ, Graham E, Irick MM, van Olden EK, et al. (2004) A signaling mucin at the head of the Cdc42- and MAPK-dependent filamentous growth pathway in yeast. Genes Dev 18:1695–708.

[38]Mnaimneh S, Davierwala AP, Haynes J, Moffat J, Peng WT, et al. (2004) Exploration of essential gene functions via titratable promoter alleles. Cell 118:31–44.

[39]Hong E, Balakrishnan R, Christie K, Costanzo M, Dwight S, et al. (2001). Saccharomyces genome database. Avalaible: http://www.yeastgenome.org/.

[40]Hughes T, Marton M, Jones A, Roberts C, Stoughton R, et al. (2000) Functional discovery via a compendium of expression profiles. Cell 102:109–126.

[41]Nabil Guelzim N, Bottani S, Bourgine P, 2 Képès F (2002) Topological and causal structure of the yeast transcriptional regulatory network. Nature Genetics 31:60–63.

[42]Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, et al. (2003) Cytoscape: a software environment for integrated models of biomolecular interaction networks. Genome Research 13:2498–2504. Avaibility: http://www.cytoscape.org.

[43]Margolin AA, Nemenman I, Basso K, Wiggins C, Stolovitzky G, et al. (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. BMC Bioinformatics 7 Suppl 1:S7.

[44]Friedman N, Linial M, Nachman I, Pe'er D (2000) Using Bayesian networks to analyze expression data. J Comput Biol 7:601–20.

[45]Di Bernardo D, Thomson M, Gardner T, Chobot S, Eastwood E, et al. (2005) Chemogenomic profiling on a genome-wide scale using reverse-engeneered gene networks. Nat Biotechnol 23:377–383.

[46]Bansal M, Della Gatta G, di Bernardo D (2006) Inference of gene regulatory networks and compound mode of action from time course gene expression profiles. Bioinformatics 22:815–822.

[47]Bryan R (1986) Graph-based algorithm for boolean function manipulation. IEEE Transactions on Computers 8:677–691.

[48]Gebser M, Kaufmann B, Neumann A, Schaub T (2007) clasp: A conflict-driven answer set solver. In: Ninth International Conference on Logic Programming and Nonmonotonic Reasoning. Tempe, AZ, USA.