BIOINFORMATICS

Vol. 30 ISMB 2014, pages i130–i138
doi:10.1093/bioinformatics/btu285

Functional association networks as priors for gene regulatory network inference

Matthew E. Studham[1,2,*], Andreas Tjärnberg[1,2], Torbjörn E.M. Nordling[1,3], Sven Nelander[3] and Erik L. L. Sonnhammer[1,2,4]

[1]Stockholm Bioinformatics Centre, Science for Life Laboratory, SE-171 65 Solna, Sweden, [2]Department of Biochemistry and Biophysics, Stockholm University, SE-106 91 Stockholm, Sweden, [3]Department of Immunology, Genetics and Pathology, Uppsala University, Rudbeck Laboratory, SE-751 05 Uppsala, Sweden and [4]Swedish eScience Research Center, SE-100 44 Stockholm, Sweden

ABSTRACT

Motivation: Gene regulatory network (GRN) inference reveals the influences genes have on one another in cellular regulatory systems. If the experimental data are inadequate for reliable inference of the network, informative priors have been shown to improve the accuracy of inferences.

Results: This study explores the potential of undirected, confidence-weighted networks, such as those in functional association databases, as a prior source for GRN inference. Such networks often erroneously indicate symmetric interaction between genes and may contain mostly correlation-based interaction information. Despite these drawbacks, our testing on synthetic datasets indicates that even noisy priors reflect some causal information that can improve GRN inference accuracy. Our analysis on yeast data indicates that using the functional association databases FunCoup and STRING as priors can give a small improvement in GRN inference accuracy with biological data.

Contact: matthew.studham@scilifelab.se

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

1 INTRODUCTION

Gene regulatory network (GRN) inference determines causal influences in gene networks and is useful for understanding regulation, usually at the transcriptional level, which can hypothetically lead to effective modification of regulatory networks. GRN inference has been studied extensively over the past decade as described in the following reviews (Hecker *et al*., 2009; Lecca and Priami, 2013; Penfold and Wild, 2011; Tegnér and Björkegren, 2007). In GRNs the nodes are genes and the edges are influences, annotated with a direction and signed strength. These networks are normally constructed using transcriptomic data from experiments in which all of the genes in the network of interest have been perturbed, often with RNAi knockdowns. Gene expression is profiled either in a time series or when the system has reached a steady-state.

A plethora of inference methods have been developed and are based on information theory (Altay and Emmert-Streib, 2010; Faith *et al*., 2007; Margolin *et al*., 2006), Boolean networks (Haider and Pal, 2012; Layek *et al*., 2011; Wang *et al*., 2012), Bayesian networks (Djebbari and Quackenbush, 2008; Husmeier and Werhli, 2007; Yu *et al*., 2004) and ordinary differential equations (ODEs; Gardner *et al*., 2003; Gustafsson and Hörnquist, 2010; Yip *et al*., 2010). A subset of the methods based on a ODE description formulates the inference as a convex programming problem (Julius *et al*., 2009; Kulkarni *et al*., 2012; Zavlanos *et al*., 2011). The Dialogue on Reverse Engineering Assessment and Methods (DREAM) (Marbach *et al*., 2012; Penfold and Wild, 2011; Prill *et al*., 2010; Stolovitzky *et al*., 2009) and other benchmarking studies (Bansal *et al*., 2007; Geier *et al*., 2007; Hache *et al*., 2009) have shown that although many methods perform better than random, there is a lot of room for improvement.

It is difficult to determine the true GRN for a biological system because even if major characteristics such as transcription factor binding are known, subtle influences may not be well understood. To avoid this problem, many benchmarking studies use synthetic data where the true GRN is known and the accuracy of inference methods can be analysed. GeneNetWeaver (GNW; Schaffter *et al*., 2011) generated synthetic networks and datasets for three of the DREAM competitions and this program uses nonlinear dynamical models of transcription and translation. Another synthetic data generation program, GeneSpider (Tjärnberg *et al*., 2014, manuscript in preparation), uses a linear dynamical model of transcription. These two programs were used in our study to generate synthetic data.

Prior knowledge may be incorporated into the inference method in order to improve accuracy and can also increase efficiency by reducing the search space. Researchers have begun to explore these possibilities by using pathways (Bonneau *et al*., 2006; Husmeier and Werhli, 2007), transcription factor binding (Gevaert *et al*., 2007; Gustafsson and Hörnquist, 2010; Shih and Parthasarathy, 2012), protein–protein interactions (Shih and Parthasarathy, 2012), gene ontology (Pei and Shin, 2012), epigenetics (Chen *et al*., 2013) and literature (Djebbari and Quackenbush, 2008; Julius *et al*., 2009; Layek *et al*., 2011). These studies incorporate the prior in different ways, but for inference methods which minimize a penalty function, the prior knowledge is often quantified as the ‘unlikelihood’ of a link, and this value is multiplied by the sparsity term in the penalty function (Christley *et al*., 2009; Greenfield *et al*., 2013; Gustafsson and Hörnquist, 2010). The prior value has also been discretized to be positive, negative or zero and used as a constraint in an optimization (Julius *et al*., 2009; Kulkarni *et al*., 2012; Zavlanos *et al*., 2011). Although there have been several studies, none of them has been good enough to become a widespread standard.

A comprehensive, user-friendly prior can be constructed using functional association data: undirected, confidence-weighted

*To whom correspondence should be addressed.

values (between 0 and 1) indicating the possibility of an interaction between two genes. One good place to find such data is in functional association databases, which aggregate heterogeneous experimental data and output a confidence score describing the probability of a functional linkage between two proteins. FunCoup (Schmitt *et al.*, 2014) and the Search Tool for the Retrieval of Interacting Genes (STRING) (Szklarczyk *et al.*, 2011) are two examples of such databases which aggregate data from literature, protein interactions, genomics, orthology, coexpression and subcellular localization in order to calculate the probability of a functional association. Although these databases' pairwise confidence scores were not created to be priors for GRN inference, they may contain enough information to improve GRN accuracy.

The pairwise confidence scores from functional association databases can be used to create an undirected, confidence-weighted likelihood matrix that can be easily incorporated as a prior into a GRN inference method. Bonneau *et al.* (2006) used first-generation functional association databases Prolinks (Bowers *et al.*, 2004) and Predictome (Mellor *et al.*, 2002) as part of a gene biclustering algorithm but not explicitly in the network inference. To our knowledge no one has extensively studied the potential of undirected, unsigned, confidence-weighted networks as priors for GRN inference.

In this study we generated synthetic datasets of steady-state expression data and functional association-like priors, assumed a dynamical systems model, and used a convex optimization-based inference method. We compared the accuracy of the GRN inferences with and without the priors to determine if and when undirected, unsigned, confidence-weighted networks improve GRN inference. We also explored a few different experimental (perturbation) designs to see if they had an impact on the prior's usefulness. Finally, we applied our method to a yeast dataset and used FunCoup and STRING to generate priors to see if they can improve network inference with biological data.

2 METHODS

2.1 Regulatory model

Our model is based on system identification concepts common in engineering and similar to the models in Gardner *et al.* (2003), Julius *et al.* (2009) and Zavlanos *et al.* (2011). When the regulatory network is near a steady-state it can be approximated by the linear dynamical system:

$$\begin{aligned} \dot{x} &= Ax + p \\ y &= x + \varepsilon, \end{aligned} \tag{1}$$

where $x \in \mathbb{R}^n$ are actual transcript differences between a perturbed and an unperturbed initial state for n genes in an experiment, $p \in \mathbb{R}^n$ are exogenous perturbations of the n genes, $A \in \mathbb{R}^{n\times n}$ is the network model in which each element $a_{ij} \in \mathbb{R}, \forall i, j$ describes the regulatory influence of gene j on gene i, $y \in \mathbb{R}^n$ are the measurements of the transcript differences and $\varepsilon \in \mathbb{R}^n$ represents measurement noise in a single experiment. When gene expression is measured at steady-state (Crampin *et al.*, 2004) and multiple perturbation experiments are combined we find that

$$Y = -A^{-1}P + \varepsilon, \tag{2}$$

where $Y \in \mathbb{R}^{n\times m}$ is the steady-state gene expression matrix, $P \in \mathbb{R}^{n\times m}$ is the perturbation matrix and ε is the noise matrix for a system with n genes and m experiments. We will avoid underdetermined problems and only focus on situations where $m \geq n$. In such a network, RNA decay is confounded with self-regulation. Normally in a stable system $a_{ii} < 0, \forall i$.

2.2 Inference method

Our network inference method is formulated as a convex optimization problem (Boyd and Vandenberghe, 2004), similar to methods in (Julius *et al.*, 2009; Zavlanos *et al.*, 2011). We used a numerical cutoff based on the reduced precision of the optimization solver to identify zero and non-zero values. The optimization problem is shown below.

$$\text{minimize}_{\text{A}} \ \|AY + P\|_F + \zeta \sum_i \sum_j (1 - \text{w}_{ij})|a_{ij}|$$

Initially, without considering a prior, our goal is to fit the model and ensure a level of sparsity which ignores effects caused by noise. This first term deals with the model fit by minimizing the sum of the residuals $\|AY + P\|_F$. The second term encourages sparsity and agreement with the prior: $\zeta \sum_i \sum_j (1 - w_{ij})|a_{ij}|$ where $W \in \mathbb{R}^{n\times n}, 0 \leq w_{ij} \leq 1 \ \forall i, j$ is an undirected, unsigned, confidence-weighted prior network and $\zeta \in \mathbb{R}_+$ (zeta) is the regularization parameter. This term is similar to the incorporation of the prior in Christley *et al.* (2009) and Gustafsson and Hörnquist (2010). Without a prior, the cross-optimization procedure described in (Tjärnberg *et al.*, 2013) can be used to set the regularization parameter, ζ. However, this procedure was not created for models in which the prior is incorporated into the sparsity term. With no proven method to set the parameter, all sparsity levels are considered, from a diagonal-only network (only RNA decay) through a fully connected network.

2.3 Synthetic data analysis

Five 20-gene true networks were initially generated using GNW (Schaffter *et al.*, 2011). In order to create realistic networks, we used a subset of yeast interactions (provided by GNW), and there were at least 10 regulators in each network. These initial networks were unsigned, so we randomly assigned a positive or negative sign to the non-zero links. Then the link strengths were discretized to values {–1, 0, 1} and we made sure that the self-interactions had a discretized strength of -1 to represent RNA decay. In general the networks were sparse, with an average sparsity level of 83.45%, or ~66 non-zero links.

There were three experimental designs: single-20, double-20 and double-40. For the single-20, each gene was knocked-down once and the number of experiments, m, is equal to the number of genes, n ($m = n = 20$). For the double-20, each experiment perturbed two genes: all were knockdowns except in one experiment one gene is overexpressed. The number of experiments equaled the number of genes ($m = n = 20$). For the double-40, each experiment knocked-down two genes and the number of experiments was double the number of genes ($m = 2n = 40$). All experiments were unique within each design and the strength of each perturbation was set to 0.5, positive in overexpressions and negative in knockdowns.

Given the true network and experimental design, we used GNW (in a way independent of the network generation) and GeneSpider (GSP; Tjärnberg *et al.*, 2014, manuscript in preparation) to generate gene expression data. Both network generators add random numbers from a Gaussian distribution to simulate measurement errors. We therefore perform a Monte Carlo simulation using five 'replicates' of each dataset. Each generator created five expression matrices for each true network and experimental design. We created a total of 150 datasets (2 generators × 5 replicates × 5 true networks × 3 experimental designs).

We define single-to-noise ratio (SNR) as the smallest signal (measured by the singular value) in the gene expression matrix divided by the largest signal in the error (Nordling, 2013):

$$\mathrm{SNR} = \frac{\sigma_n(Y)}{\sigma_1(\ \varepsilon)} \tag{3}$$

This is a conservative SNR and it is motivated by the fact that network inference is an inverse problem, where the smallest signal is very important because it affects the largest signal in the inverse. The SNR of GNW-generated data (median 0.00717, range [6.01×10^{-8}, 0.137]) was significantly lower than the SNR of the data generated by GSP (median 0.409, range [0.052, 2.06]). An $\mathrm{SNR} < 1$ indicates that the largest noise signal obscures the smallest expression signal, as was the case for most of our datasets. Therefore most of our datasets would be considered to have low information content and could use the help of a prior.

Synthetic priors (undirected, confidence-weighted network matrices) were generated to have confidence score distributions similar to those found in FunCoup. The non-zero links were approximated with a modified exponential decay distribution with an average confidence score of 0.85 and the zero links were approximated with a gamma distribution with an average confidence score of 0.4 (Thomas Schmitt, unpublished data). These distributions were sampled to create the prior matrix. The initial symmetric prior matrix $C \in \mathbb{R}^{n \times n}$, $0 \le c_{ij} \le 1\, \forall\, i, j$ was adjusted to create the final prior matrix W:

$$W \in \mathbb{R}^{n \times n},\ w_{ij} = \left\{ \begin{matrix} \frac{9}{10} c_{ij} & i \neq j \\ 1 & i = j \end{matrix} \right\} \tag{4}$$

This adjustment is necessary to avoid full confidence values (ones) in off-diagonal elements, thereby ensuring that the prior is soft evidence. Ones were assigned to the diagonal to represent RNA decay. We tested priors with different accuracy, i.e. different levels of agreement with the true network. A non-zero link in the prior was deemed accurate if there was also a non-zero link (of any direction and sign) at the same position in the true network. A zero link in the prior was deemed accurate if there were no non-zero links (of any direction and sign) at the same position in the true network. We created priors which were 50, 60, 70, 80, 90 and 100% accurate, and the accuracy applied to both zero and non-zero links. Since functional association priors do not cover self-interactions, we did not count these (diagonal elements) in the accuracy. There was an element of randomness in the prior generation, so we created five ‘replicate’ prior matrices for each level of accuracy and true network, resulting in a total of 150 synthetic functional association priors (6 accuracy levels × 5 true networks × 5 replicates). A naïve prior $W = I$, containing only the RNA decay links, was also created to act as a control in the analysis.

2.4 Yeast data analysis

We used a publicly available dataset (GEO:GSE4654) from Hu *et al.* (2007) containing transcriptional profiles from 263 transcription factor knockout strains in *Saccharomyces cerevisiae*. The yeast strains, derived from BY4741, were sampled in the mid-log phase (Hu *et al.*, 2007). Although there were 263 genes, we only used 173 in our analysis because some data points were missing and not all the genes were represented in our gold standard network. Our final gene expression matrix contained 173 genes and 173 experiments.

The gold standard network was derived from the Yeastract database (Teixeira *et al.*, 2014). We obtained 187 856 activation/inhibition interactions, of which 2910 were relevant to our 173 genes in the knockout experiments.

Functional association priors were constructed from FunCoup (Schmitt *et al.*, 2014) and STRING (Szklarczyk *et al.*, 2011). On the FunCoup website we searched for the network using the default settings except: 0.1 confidence threshold (the lowest possible threshold), *S.cerevisiae* species, and 0 expansion depth. The search was done using FunCoup version 3.0 on January 14, 2014. On the STRING website we used the multiple names search, protein interactors and a zero required confidence score. This search was done using STRING version 9.1 on January 15, 2014. Both priors were adjusted according to Equation (4) in the previous section, and the final FunCoup and STRING priors had 2685 and 6555 links, respectively.

2.5 Inferences

We used the CVX package in MATLAB to implement the GRN inferences. CVX iterates until the precision cutoff (10^{-4}) is reached. In the synthetic analysis, we inferred networks for each experimental design and dataset and prior and sparsity level combination, which resulted in 1.7 million inferences (150 datasets × 5 priors × 6 accuracies × 381 sparsity levels). We used a search procedure to modify the regularization parameter to obtain inferences for all sparsity levels. In the rare situation in which a sparsity level was unreachable (by modifying the regularization parameter) the inference accuracy was assumed to be the average of the accuracies from the adjacent sparsity levels. Often the same sparsity level was reached with different parameter values; in this case we used the average inference accuracy in the results.

For the yeast network, which was much larger than the synthetic networks, time constraints did not allow us to use the same sparsity level search procedure. Instead we used intervals for the regularization parameter, approximately evenly spaced in logarithmic space, which resulted in 22 378 inferred networks for each prior. These inferences covered more than 25% of all possible sparsity levels: 8196 using the naïve prior, 7911 using the FunCoup prior and 7740 using the STRING prior. For sparsity levels with no inferred network, the accuracy was assumed to be the average of accuracies from adjacent sparsity levels.

The resulting inferred networks' interaction strengths were discretized (values {–1, 0, 1}) for evaluation.

2.6 Evaluation

For the synthetic analysis, the inference accuracy was calculated as the proportion of links that were equal in the true discretized network and inferred discretized network. The accuracy from the inferences using the naïve prior were subtracted from the accuracy from the inferences using the FunCoup-simulated prior in order to determine the improvement achieved by using the functional association prior. We also performed an alternative evaluation considering only true non-zero links.

For the yeast analysis we used a similar procedure except we only considered true non-zero links when evaluating accuracy because the Yeastract network contains validated links, but not necessarily validated non-links.

2.7 Functional association prior accuracy estimation

The accuracy of the FunCoup and STRING priors used in the yeast data analysis was estimated with respect to the Yeastract network. Since the functional association priors do not have signed links, sign was ignored. In order to make this prior accuracy analogous to the prior accuracy used in the synthetic analysis, only half of the off-diagonal links were evaluated. Although there are a total of 2910 off-diagonal links in the Yeastract network, 154 of them are symmetrical, so we only considered 2756 links. The values in the prior matrices needed to be discretized to differentiate links from non-links. We did two prior accuracy estimations. In the first estimation, all non-zero values $w_{ij} > 0, j > i$ are considered links. In the second estimation all values at or above a threshold $w_{ij} \ge 0.5, j > i$ are considered links.

3 RESULTS

In the synthetic analysis we generated 5 ‘true’ networks and 150 expression datasets (covering 3 experimental designs) using non-linear (GNW) and linear (GSP) generation methods. We used these datasets along with 150 priors (covering 6 different accuracy levels), and completed over 1.7 million inferences to determine if and when a functional association prior improves GRN inference. Since we were unable to find a method to optimally set our sparsity parameter, we evaluated the inference accuracy over all sparsity levels (except self-interactions were always non-zero). There were five networks used in the analysis, and since their individual results were similar, we have only shown the combined results. Also, there was never a perfect inference; the best inference recovered 99% of the links, so there was always room for improvement. A perfect prior never resulted in a perfect inference because of noise and the fact that these priors are symmetrical (i.e. that do not give interaction direction) and our true networks were not symmetrical. In the results below, improvement is defined as the inference accuracy percentage of the method using the simulated functional association prior minus the inference accuracy percentage of the method using the naïve prior.

3.1 Accurate priors improve performance

If a functional association prior is accurate enough (i.e. enough non-zero links in the true network are represented by undirected non-zero links in the prior) then inference is improved over virtually all sparsity levels. Figure 1 shows the levels of prior accuracy that resulted in improved GRN inference for datasets generated by GNW and GSP. It should be noted that we used two different dataset generators to ensure that we have a diversity of synthetic data, not to explicitly compare the two generators. As shown in Figure 1A, a 70% accurate prior clearly improved inference for GNW-generated data and in Figure 1D a 90% accurate prior clearly improved inference for GSP-generated data.

A similar overall improvement profile is also seen when only considering true non-zero links (Supplementary Fig. S1). In this situation, the magnitude of improvement is more dramatic but the accuracy level at which the prior achieves improvement is almost exactly the same as when considering all links.

Figure 2 shows the improvement over all sparsity levels for these two types of generated datasets using these prior accuracies. The most improvement is seen at moderate sparsity levels. The GNW inference improvement profile is relatively uniform, while the GSP inference improvement profile was clearly skewed toward the sparse end, indicating that the prior was helpful in determining which links to keep in a sparse network. For both dataset generators, if the prior was not accurate enough then the resulting inferred network is worse than when using a naïve prior.

3.2 Better improvement when using data generated using noisy, nonlinear model

A comparison of Figure 1 parts (A) and (B) shows that a functional association prior improves inferences for GNW-generated data (from a noisy, nonlinear model) much more than for GSP-generated data (from a less noisy, linear model) if the actual sparsity level is unknown; this is shown by the difference in mean (dark blue) or median (light blue) boxes at the same prior accuracy. If the sparsity level is known (green boxes) then the GSP-generated results showed a larger improvement if the prior accuracy is 90 or 100%.

A less accurate prior showed a greater tendency to result in worse inference for GSP-generated data, as seen for the 50% accurate prior. The GNW-generated data were also noisier based on over 68 000 inference profiles (i.e. dataset/prior combinations). There appears to be a negative relationship between SNR and improvement (Supplementary Fig. S2).

3.3 Experimental design did not significantly affect improvement

There were three experimental designs: single-20, double-20 and double-40. For single-20, each gene was knocked-down once and the number of experiments, m, is equal to the number of genes, n ($m = n = 20$). For the double-20, each experiment perturbed two genes: all were knockdowns except in one experiment where one gene was overexpressed. The number of experiments equaled the number of genes ($m = n = 20$). For the double-40, each experiment knocked-down two genes and the number of experiments was double the number of genes ($m = 2n = 40$). All experiments were unique within each design.

The results were similar for the three experimental designs (Supplementary Fig. S3). However, the three different designs did not have as much overlap for the GSP-generated data. Here the double-20 showed the most improvement, followed by the single-20, and finally the double-40. There was still overlap, and this difference can be explained by differences in SNR which are discussed in the following section. The double-20 was the noisiest, then the single-20, and the double-40 was the least noisy.

3.4 Application to yeast network

We applied our method to a yeast dataset (Hu *et al.*, 2007) with 173 genes and 173 experiments, using FunCoup and STRING as priors, and the Yeastract database (Teixeira *et al.*, 2014) as a gold standard. Using only the naïve prior, the maximum inference accuracy is only 49.52% of the gold standard links, so there is plenty of room for improvement.

Figure 3 shows the improvement over all sparsity levels for the two functional association priors when compared to the naïve prior. In Figure 3A the FunCoup prior is helpful for a large range from ~19 000 links and sparser, except for one small spot ~4000 links. For networks with more than 19 000 links the FunCoup prior lowers the inference accuracy. In Figure 3A the maximum improvement is 1.10%, the minimum is −0.61% and the average is 0.23%. These percentages equate to roughly 32, −18 and 7 links, respectively. The STRING prior is shown in Figure 3B. For networks with more than 19 000 links there is unlikely to be improvement, but inference of sparser networks is improved using this prior. In Figure 3B the maximum improvement is 1.31%, the minimum is −0.44% and the average is 0.45%. These percentages equate to roughly 38, −13 and 13 links, respectively.

3.5 Accuracy of FunCoup and STRING priors

In an attempt to quantify their accuracy, the FunCoup and STRING priors were compared to the Yeastract network. In order to make these prior accuracies analogous to our

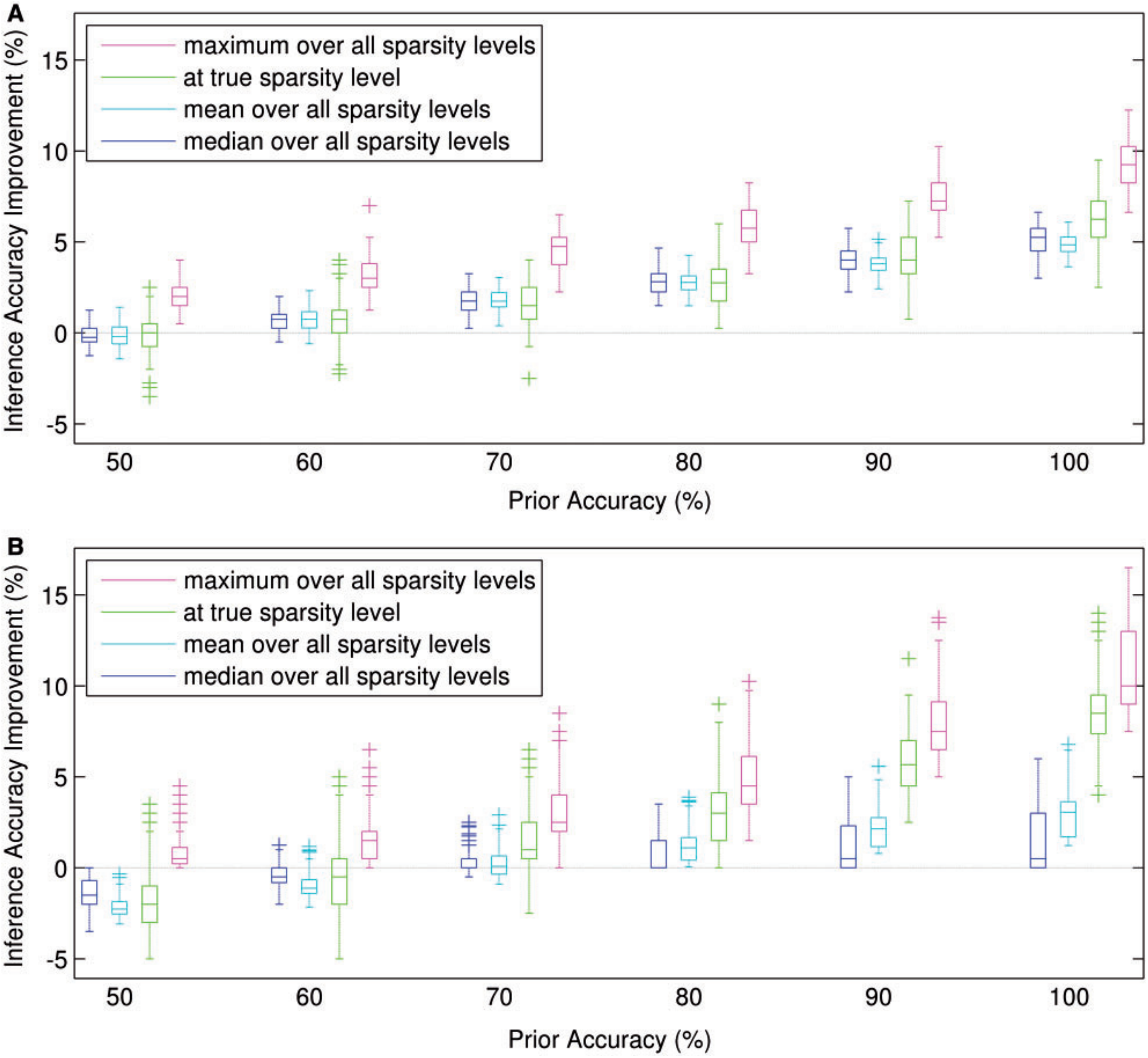

Fig. 1. Inference improvement and prior accuracy. As the prior gets more accurate, the GRN inference improvement increases. At each prior accuracy level, 125 inferences are averaged and the accuracies over the sparsity levels are aggregated using the median (dark blue), mean (light blue), true sparsity level (green) and maximum (magenta) inference improvement. The results from the GNW-generated data are shown in (**A**) and the GSP-generated data in (**B**)

synthetic prior accuracies, we did not count both directions of symmetrical links. Therefore our Yeastract network contained 2756 links (2910 minus 154 symmetrical links).

When estimating the accuracy for the functional association priors, we had to discretize the values to differentiate links from non-links. When all non-zero values are considered links, the FunCoup prior contained 1256 links, 594 of which were in the Yeastract network, so it covered ~22% of the validated links. The STRING prior contained 3191 links, 1414 of which were in the Yeastract network, so it covered 51% of the validated links.

When a confidence score threshold of 0.5 is used, FunCoup gives us 263 links, 130 of which are in common with the Yeastract network, and STRING has 1155 links, of which 548 are in the Yeastract network. With this threshold, FunCoup and STRING covered 5% and 20% of the validated links, respectively.

4 DISCUSSION

Our results show that use of a functional association prior matrix can improve GRN inference accuracy. The prior needs to be at least 70% accurate in order to show a clear improvement over most sparsity levels based on our testing of synthetic data. However, our testing on a yeast dataset indicates that the prior accuracy can be much lower and still result in a small

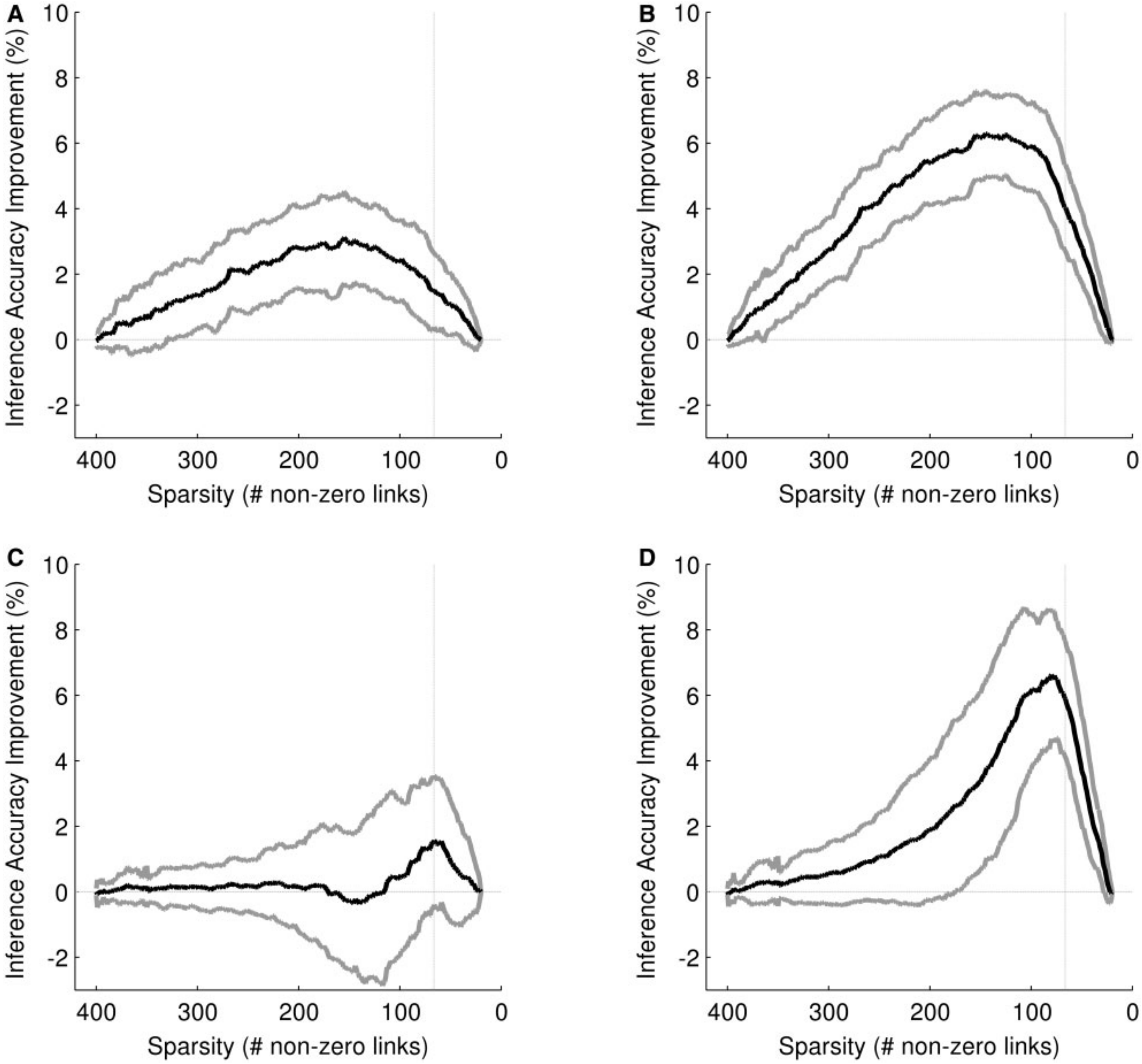

Fig. 2. Prior improves inference over almost all sparsity levels. For all plots above, the inference accuracy improvement is shown over all sparsity levels. The average improvement is shown as the black line and the gray line is one SD from the average. The vertical dotted gray line shows the average true sparsity level of the five synthetic networks. (**A**) GNW-generated data, single-perturbation design with 70% prior accuracy, (**B**) GNW-generated data, single-perturbation design with 90% prior accuracy, (**C**) GeneSpider-generated data, single-perturbation design with 70% prior accuracy and (**D**) GeneSpider-generated data, single-perturbation design with 90% prior accuracy. Parts (A), (B) and (D) show that the average improvement can be positive over almost all sparsity levels

improvement over most sparsity levels. It is important to note, however, that we consider all possible links in the synthetic analysis and only gold standard links in the yeast analysis.

This 70% level of prior accuracy is at odds with several inference prior studies which assert that even an inaccurate prior can aid in GRN inference. Greenfield *et al.* (2013) show that even if their prior consists of more than 90% erroneous links they can still accurately recover a GRN. Although their prior incorporation is similar to ours (they multiply unlikelihood times the strength in the sparsity term) their inference method is different and they limit the possible regulators to transcription factors. In our model any gene can influence any other gene, regardless of its known molecular function. Christley *et al.* (2009) were also able to work with an inaccurate prior but they used an extra parameter (set by cross-validation) to weight the prior information so an inaccurate prior would simply be given less weight than an accurate one.

These methods, as well as ours, can be seen as picking the model, from the set of all models that cannot be rejected based on the recorded data, that minimizes the objective function

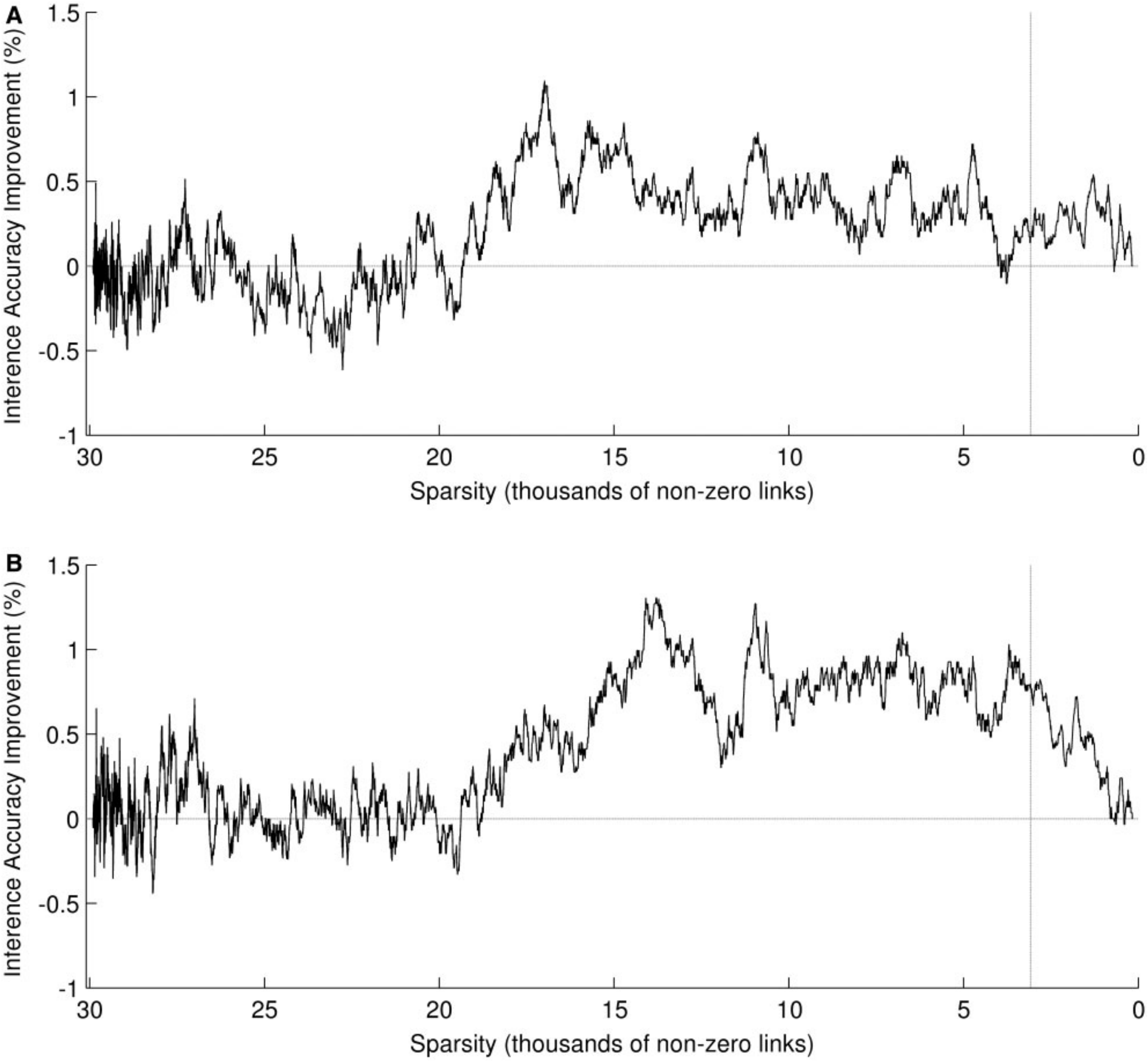

Fig. 3. FunCoup (**A**) and STRING (**B**) priors improve yeast network inference for most sparsity levels. The plots show inference accuracy improvement over almost all sparsity levels for the yeast network with 173 genes using the FunCoup and STRING priors. The most fully connected network had 29 906 non-zeros and the sparsest network had 173 non-zeros (all self-interactions). Only the 2910 off-diagonal links in the Yeastract network were considered. Only about a quarter of the sparsity levels were actually inferred; the accuracies for the other sparsity levels were estimated based on those inferences. The vertical line shows the Yeastract network sparsity

based on the prior. The ability to test the hypothesis made by the prior depends on the informativeness of the recorded data. If the data were very informative then the prior would not be helpful nor needed and in that case the prior has no influence.

The fact that the prior improved inferences based on the GNW-generated data much more than the corresponding inferences based on GeneSpider-generated data might be explained by the differences in the two generators. We used GeneSpider and a linear model to generate datasets, while GNW has nonlinearities built in to its dataset generation. Our inference method is based on a linear dynamical system, so it follows that it is easier for it to recover a network from data created with a linear model. Thus the inference with the naïve prior works better on the GSP-generated data compared to the GNW-generated data, and we just do not need the functional association prior as much in that case. Another explanation for the discrepancy between the inference of GNW- and GSP-generated data could be due to the differences in SNR. GNW data had a lower signal than GSP data (Supplementary Fig. S2), and it is logical that the naïve prior would do worse (and thus increase the improvement) when there is a low SNR. The SNR of some datasets generated by GNW is so low that it is questionable that they are informative for network inference (Tjärnberg *et al*., 2013).

The yeast data consists of expression changes caused by knockout of each of the 173 genes. A successful gene knockout alters the topology of the regulatory network because the corresponding node and all of its links are removed. Strictly speaking, this implies that we are trying to infer the wild-type steady-state network based on data recorded from 173 different knockout steady-state networks, which should be questioned. A topology change can be seen as a nonlinear transformation, so it is in general also questionable if a linear model can be used. However, in this case the number of data points equals the number of parameters in the network model so the data can always be explained using a linear model, which motivates why we, following the parsimony principle, use one. In principle, every indirect path through genes that are not included in the model should show up in the inferred model (Nordling, 2013). Nonetheless, we only included direct links among the 173 genes that were in the Yeastract gold standard network.

We therefore verified that a linear model with the topology given by this gold standard can explain the input–output relationship. Actually, such a model can explain 99.5% of the variation in the recorded data. One should bear in mind that the dominating 20 components explain more than 75% of the total variation and that the gene expression matrix is ill-conditioned (condition number above 2000), so the dataset is not sufficiently informative for complete network inference (Nordling, 2013). On the other hand, if it was informative enough then the prior would not be needed and it would not be an interesting test case. The lack of information is likely to in part explain why the prior, despite being inaccurate, leads to a small improvement.

Functional association priors from FunCoup (Alexeyenko *et al*., 2011) or STRING (Szklarczyk *et al*., 2011) might be useful in GRN inference if these priors capture enough causal information. These functional association databases do a good job of aggregating heterogeneous experimental data, which makes them convenient, but many of the associations (e.g. coexpression) are the result of correlation and not necessarily causation. Since we estimate the prior accuracies of FunCoup and STRING to be well below the 70% threshold for our yeast analysis, it seems unlikely that these priors reflect enough causal information for clear improvement over most sparsity levels. However, our yeast analysis also shows, for certain sparsity ranges, that using FunCoup and/or STRING can result in small inference improvement.

ACKNOWLEDGEMENTS

The authors would like to thank Thomas Schmitt for information about FunCoup, including the distribution of confidence values; Yeastract researchers for providing activation and inhibition information; and Richard Bonneau for helpful suggestions.

Funding: This work was supported by SciLifeLab and the Swedish strategic research program ESSENCE.

Conflict of Interest: none declared.

REFERENCES

Alexeyenko,A. *et al.* (2011) Comparative interactomics with Funcoup 2.0. *Nucleic Acids Res.*, **40**, 1–8.

Altay,G. and Emmert-Streib,F. (2010) Inferring the conservative causal core of gene regulatory networks. *BMC Syst. Biol.*, **4**, 132.

Bansal,M. *et al.* (2007) How to infer gene networks from expression profiles. *Mol. Syst. Biol.*, **3**, 78.

Bonneau,R. *et al.* (2006) The Inferelator: an algorithm for learning parsimonious regulatory networks from systems-biology data sets de novo. *Genome Biol.*, **7**, R36.

Bowers,P.M. *et al.* (2004) Prolinks: a database of protein functional linkages derived from coevolution. *Genome Biol.*, **5**, R35.

Boyd,S.P. and Vandenberghe,L. (2004) *Convex Optimization.* 7th edn. Cambridge University Press, New York.

Chen,H. *et al.* (2013) Integrating epigenetic prior in dynamic Bayesian network for gene regulatory network inference. In: *IEEE Symposium Series on Computational Intelligence in Bioinformatics and Computational Biology*.

Christley,S. *et al.* (2009) Incorporating existing network information into gene network inference. *PLoS One*, **4**, e6799.

Crampin,E.J. *et al.* (2004) Mathematical and computational techniques to deduce complex biochemical reaction mechanisms. *Prog. Biophys. Mol. Biol.*, **86**, 77–112.

Djebbari,A. and Quackenbush,J. (2008) Seeded Bayesian Networks: constructing genetic networks from microarray data. *BMC Syst. Biol.*, **2**, 57.

Faith,J.J. *et al.* (2007) Large-scale mapping and validation of Escherichia coli transcriptional regulation from a compendium of expression profiles. *PLoS Biol.*, **5**, e8.

Gardner,T.S. *et al.* (2003) Inferring genetic networks and identifying compound mode of action via expression profiling. *Science*, **301**, 102–105.

Geier,F. *et al.* (2007) Reconstructing gene-regulatory networks from time series, knock-out data, and prior knowledge. *BMC Syst. Biol.*, **1**, 11.

Gevaert,O. *et al.* (2007) A framework for elucidating regulatory networks based on prior information and expression data. *Ann. N. Y. Acad. Sci.*, **1115**, 240–248.

Greenfield,A. *et al.* (2013) Robust data-driven incorporation of prior knowledge into the inference of dynamic regulatory networks. *Bioinformatics*, **29**, 1060–1067.

Gustafsson,M. and Hörnquist,M. (2010) Gene expression prediction by soft integration and the elastic net-best performance of the DREAM3 gene expression challenge. *PLoS ONE*, **5**, e9134.

Hache,H. *et al.* (2009) Reverse engineering of gene regulatory networks: a comparative study. *EURASIP J. Bioinform. Syst. Biol.*, **2009**, 617281.

Haider,S. and Pal,R. (2012) Boolean network inference from time series data incorporating prior biological knowledge. *BMC Genomics*, **13**, S9.

Hecker,M. *et al.* (2009) Gene regulatory network inference: data integration in dynamic models–a review. *Biosystems*, **96**, 86–103.

Hu,Z. *et al.* (2007) Genetic reconstruction of a functional transcriptional regulatory network. *Nat. Genet.*, **39**, 683–687.

Husmeier,D. and Werhli,A.V. (2007) Bayesian integration of biological prior knowledge into the reconstruction of gene regulatory networks with Bayesian networks. *Comput. Syst. Bioinformatics Conf.*, **6**, 85–95.

Julius,A. *et al.* (2009) Genetic network identification using convex programming. *IET Syst. Biol.*, **3**, 155–166.

Kulkarni,V.V. *et al.* (2012) Gene regulatory network modeling using literature curated and high throughput data. *Syst. Synth. Biol.*, **6**, 69–77.

Layek,R.K. *et al.* (2011) From biological pathways to regulatory networks. *Mol. Biosyst.*, **7**, 843–851.

Lecca,P. and Priami,C. (2013) Biological network inference for drug discovery. *Drug Discov. Today*, **18**, 256–264.

Marbach,D. *et al.* (2012) Wisdom of crowds for robust gene network inference. *Nat. Methods*, **9**, 796–804.

Margolin,A.A. *et al.* (2006) ARACNE: an algorithm for the reconstruction of gene regulatory networks in a mammalian cellular context. *BMC Bioinformatics*, **7** (**Suppl. 1**), S7.

Mellor,J.C. *et al.* (2002) Predictome: a database of putative functional links between proteins. *Nucleic Acids Res.*, **30**, 306–309.

Nordling,T.E.M. (2013) Robust inference of gene regulatory networks: system properties, variable selection, subnetworks, and design of experiments. Doctoral Thesis. KTH Royal Institute of Technology, Stockholm, Sweden.

Pei,B. and Shin,D.-G. (2012) Reconstruction of biological networks by incorporating prior knowledge into Bayesian network models. *J. Comput. Biol.*, **19**, 1324–1334.

Penfold,C. and Wild,D. (2011) How to infer gene networks from expression profiles, revisited. *Interface Focus*, **1**, 857–870.

Prill,R.J. *et al.* (2010) Towards a rigorous assessment of systems biology models: the DREAM3 challenges. *PLoS One*, **5**, e9202.

Schaffter,T. *et al.* (2011) GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods. *Bioinformatics*, **27**, 2263–2270.

Schmitt,T. *et al.* (2014) FunCoup 3.0: database of genome-wide functional coupling networks. *Nucleic Acids Res.*, **42**, D380–D388.

Shih,Y.-K. and Parthasarathy,S. (2012) A single source k-shortest paths algorithm to infer regulatory pathways in a gene network. *Bioinformatics*, **28**, i49–i58.

Stolovitzky,G. *et al.* (2009) Lessons from the DREAM2 Challenges. *Ann. N. Y. Acad. Sci.*, **1158**, 159–195.

Szklarczyk,D. *et al.* (2011) The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored. *Nucleic Acids Res.*, **39**, D561–D568.

Tegnér,J. and Björkegren,J. (2007) Perturbations to uncover gene networks. *Trends Genet.*, **23**, 34–41.

Teixeira,M.C. *et al.* (2014) The YEASTRACT database: an upgraded information system for the analysis of gene and genomic transcription regulation in *Saccharomyces cerevisiae*. *Nucleic Acids Res.*, **42**, D161–D166.

Tjärnberg,A. *et al.* (2013) Optimal sparsity criteria for network inference. *J. Comput. Biol.*, **20**, 398–408.

Wang,G. *et al.* (2012) Process-driven inference of biological network structure: feasibility, minimality, and multiplicity. *PLoS One*, **7**, e40330.

Yip,K.Y. *et al.* (2010) Improved reconstruction of in silico gene regulatory networks by integrating knockout and perturbation data. *PLoS One*, **5**, e8121.

Yu,J. *et al.* (2004) Advances to Bayesian network inference for generating causal networks from observational biological data. *Bioinformatics*, **20**, 3594–3603.

Zavlanos,M.M. *et al.* (2011) Inferring stable genetic networks from steady-state data. *Automatica*, **47**, 1113–1122.