BIOINFORMATICS

Vol. 26 ECCB 2010, pages i596–i602
doi:10.1093/bioinformatics/btq385

Dynamic deterministic effects propagation networks: learning signalling pathways from longitudinal protein array data

Christian Bender[1,*], Frauke Henjes[1], Holger Fröhlich[2], Stefan Wiemann[1], Ulrike Korf[1] and Tim Beißbarth[3]

[1]Department of Molecular Genome Analysis, German Cancer Research Center, 69120 Heidelberg, [2]Department of Algorithmic Bioinformatics, Bonn-Aachen International Center for IT, 53113 Bonn and [3]Department of Medical Statistics, University of Göttingen, 37099 Göttingen, Germany

ABSTRACT

Motivation: Network modelling in systems biology has become an important tool to study molecular interactions in cancer research, because understanding the interplay of proteins is necessary for developing novel drugs and therapies. *De novo* reconstruction of signalling pathways from data allows to unravel interactions between proteins and make qualitative statements on possible aberrations of the cellular regulatory program. We present a new method for reconstructing signalling networks from time course experiments after external perturbation and show an application of the method to data measuring abundance of phosphorylated proteins in a human breast cancer cell line, generated on reverse phase protein arrays.

Results: Signalling dynamics is modelled using active and passive states for each protein at each timepoint. A fixed signal propagation scheme generates a set of possible state transitions on a discrete timescale for a given network hypothesis, reducing the number of theoretically reachable states. A likelihood score is proposed, describing the probability of measurements given the states of the proteins over time. The optimal sequence of state transitions is found via a hidden Markov model and network structure search is performed using a genetic algorithm that optimizes the overall likelihood of a population of candidate networks. Our method shows increased performance compared with two different dynamical Bayesian network approaches. For our real data, we were able to find several known signalling cascades from the ERBB signalling pathway.

Availability: Dynamic deterministic effects propagation networks is implemented in the R programming language and available at http://www.dkfz.de/mga2/ddepn/

Contact: c.bender@dkfz.de

1 INTRODUCTION

Studying the molecular biology of cells and tissues has developed from the investigation of few genes or proteins in one experiment to the analysis of the interplay of many components as a system. Various array techniques have been devised for analysing cellular behaviour on DNA, RNA and protein level that make it possible to generate thousands of measurements in a single experiment. These data can be plugged into *de novo* network reconstruction methods in order to infer regulatory interactions between the measured components. For this purpose, several approaches have been developed in the past.

Bayesian Networks (BN; Heckerman, 1996) have been frequently used to reconstruct gene regulatory networks from RNA expression experiments (Friedman *et al.*, 2000; Segal *et al.*, 2005) as well as causal protein–protein relationships for intensity data from protein quantification (Sachs *et al.*, 2005). The latter is an example where directed perturbations of several measured proteins were performed in order to resolve the structure of the underlying interactions. External interventions can be introduced by multiple means, for example, changing environmental conditions, applying drugs or using gene silencing methods such as RNA interference. Another example for BN usage under perturbation conditions is given in Pe'er *et al.* (2001).

Besides BNs, there are several related approaches to infer networks from perturbation data. Markowetz *et al.* (2005) derived networks after knocking out specific genes by analysing expression patterns in the discretized gene expression measurements. Fröhlich *et al.* (2008) extended this approach to perform inference on non-discretized expression levels. Tegner *et al.* (2003) suggested iterative perturbation of the system in order to reveal the underlying network structure. They modelled perturbations as a linear combination of inputs and inferred weights for the pairwise node to node influences. Nelander *et al.* (2008) improved this idea by using nonlinear perturbation effects and modelled the interaction behaviour of a number of components after several single and combinatorial perturbations.

Time resolved measurements provide insight into the dynamical behaviour of the system and do not restrict modelling to a 'snapshot' of the system's state. A suitable approach for network inference from time resolved data are dynamic Bayesian networks (DBN), a family of reconstruction methods including Boolean network models, state-space models or regression models (Akutsu *et al.*, 1999; Imoto *et al.*, 2002; Lébre, 2009; Murphy and Mian, 1999; Rau *et al.*, 2010).

While these methods model the dynamics of the system over time, they do not model perturbation effects directly. However, Geier *et al.* (2007) studied reverse engineering methods on simulated data for time courses and external perturbations and came to the conclusion that additional perturbation of the system is beneficial. So methods that explicitly include perturbations in the modelling approach for time course analysis are still needed. In addition, most of the current network reconstruction methods are tailored to the analysis of gene regulatory networks based on gene expression data from microarray experiments. Rather few studies deal with the signalling flow between proteins based on the analysis of protein activation and abundance coupled with intervention effects. Fröhlich *et al.* (2009) developed a network inference method for protein networks after knockdown of the measured components that allows

*To whom correspondence should be addressed.

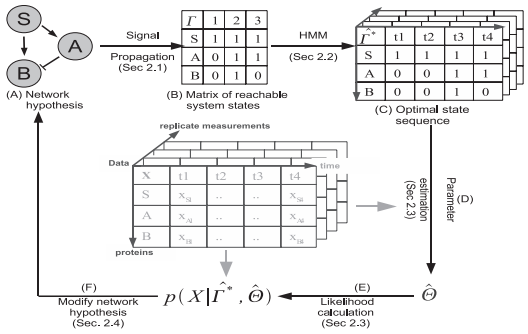

Fig. 1. Overview of the approach: given a network hypothesis (**A**), we generate a set of reachable system states by applying a fixed signal propagation scheme (**B**) which in effect reduces the number of possible system states. An optimal path through these reachable system states over time is identified by an HMM (**C**). Using the series of system states from the HMM, model parameters for two Gaussian distributions for each protein (one for active, one for passive) are estimated (**D**) and a total likelihood of our measurements given the network and model parameters is calculated (**E**). We use this likelihood score in a GA in order to optimize the overall score for an evolving population of candidate networks (**F**) and generate a final network from this population, after we found convergence in the GA.

time series measurements, too. But their method treats each time point as independent measurement and does not model the time-dependent behaviour of the system explicitly. However, using only few perturbations and gathering information on the signal flow through longer time series would be desirable, too.

In this study, we set up a framework for reconstructing signalling networks from time course measurements after external perturbation (both inhibitory and stimulating). Figure 1 shows an outline of the proposed workflow. Networks are represented as directed cyclic graphs with distinct edge types for activating and inhibiting interactions. We model signalling dynamics by a Boolean signal propagation mechanism defining state transitions for a given network structure. An optimal state transition series is found in a hidden Markov model (HMM; Durbin *et al.*, 1998) and the fit of the data to such a transition matrix is assessed by our proposed likelihood score. Network structure search is performed in a genetic algorithm (GA) optimizing the overall likelihood of a population of candidate networks. Our method shows good performance for reconstructing signalling networks from artificial data and outperforms two current DBN approaches.

As an application of the algorithm, we used protein phosphorylation measurements for 16 ERBB signalling-related phosphoproteins. The data were generated on reverse phase protein arrays (RPPA; Loebke *et al.*, 2007) in the human breast cancer cell line HCC1954, which overexpresses the ERBB2 receptor that is associated with reduced disease-free and overall survival in breast cancer patients (Slamon *et al.*, 1987, 1989). Note that dynamic deterministic effects propagation networks (DDEPN) was developed for protein phosphorylation data, but in principle is also applicable to other types of high-throughput data, e.g. for RNA microarrays. We stimulated the cells with two ligands [epidermal growth factor (EGF) and heregulin (HRG)], both as separate and combined stimulation experiments. All three experiments were combined to infer a signalling network which was compared with current literature knowledge. DDEPN was able to identify several well-known signalling chains from the MAPK and AKT signalling pathways, some of which originally were found in ERBB2 overexpressing cells. This shows the ability of our method to identify meaningful interactions from experimental proteomics data.

2 SYSTEM AND METHODS

2.1 Modelling the dynamics of the system

Let $V=\{v_i : i\in 1,\ldots,N\}$ be the set of nodes representing proteins and $\Phi = V\times V \rightarrow \{0,1,2\}$ an adjacency matrix defining a network, where 0 means no edge, 1 activation and 2 inhibition between two nodes. The signal flow through a given network of proteins is represented as a matrix $\Gamma=\{\gamma_{ik}\in\{0,1\} : i\in 1,\ldots,N, k\in 1,\ldots,M\}$, which contains a series of possible system states $\gamma_{\mathbf{k}}=\{\gamma_i : i\in 1,\ldots,N, \gamma_i\in\{0,1\}\}$. These are vectors of activation states for each node at a time step k. Define $0<M\leq 2^N$ as number of reachable system states, determined as soon as a state is repeated during the signal propagation. Each perturbation is seen as an external influence which is included as a node into the network and whose state is constantly active (i.e. 1).

Starting at the stimuli nodes, the status of all children is subsequently determined. A child is active if at least one parent connected by an activation edge is active and all parents connected via inhibition edges are inactive in the preceding step. For example, in the matrix Γ shown in Figure 1, the state γ_{B2} of protein B at Step 2 is determined by $\gamma_{B2}=\gamma_{S1}\wedge\neg\gamma_{A1}=1\wedge 1=1$ (where '$\neg$' is the logical negation which is used whenever a parent is connected via an inhibitory edge).

A formal description of the signal propagation follows: given a set of nodes V and a network Φ. Define $S\subseteq V$ as the set of input stimuli and consider the network Φ as fixed for the propagation. We derive the state matrix Γ that comprises all M reachable state vectors $\gamma_{\mathbf{k}}$ for the given network. The propagation is stopped at a step M, if $\exists k\leq M$, such that $\gamma_{\mathbf{k}}=\gamma_{\mathbf{M}}$, i.e. if one of the preceding states is found a second time.

All stimuli nodes $s\in S$ are active in all steps, i.e. $\gamma_{sk}=1\,\forall k$, and all other nodes are initialized to be 0 in the first step, i.e. $\gamma_{v_i 1}=0\,\forall v_i\in V\setminus S$. Let $pa(v_i)$ be the set of all parents of a node v_i and ϕ_{wv_i} an edge from a node w to v_i. For any status k and protein v_i, define

$$E^+_{k-1}(v_i)=\{\gamma_{wk-1} : \phi_{wv_i}=1, \forall w\in pa(v_i)\}$$

$$E^-_{k-1}(v_i)=\{\gamma_{wk-1} : \phi_{wv_i}=2, \forall w\in pa(v_i)\}$$

as the sets of states of parental nodes of v_i in step $k-1$, connected by activating edges (E^+_{k-1}) and connected by inhibiting edges (E^-_{k-1}). An entry γ_{v_ik} in Γ is then determined by:

$$\gamma_{v_ik}=\left(\bigvee_{e^+\in E^+_{k-1}(v_i)} e^+\right)\wedge\neg\left(\bigvee_{e^-\in E^-_{k-1}(v_i)} e^-\right) \quad (1)$$

This procedure reduces the maximal number of columns in the system state matrix Γ from 2^N to $M\leq 2^N$. However, the states in Γ do not necessarily correspond to the actual measured time points in the data. In general, it is expected that a different number of reachable states than time points is found. For example, in the hypothetical case that the system remains in a constant state, only one state would be present Γ. Thus, we have to find a series of system states that is consistent with the measured experimental data and represents expected dynamics under our given network hypothesis, which is described in the next section.

2.2 HMM for searching the optimal sequence of system states

Let $t\in 1,\ldots,T$ denote the index for the time point and $r\in 1,\ldots,R$ denote the index for the replicated measurements. Our measured data are recorded in a data matrix $X=\{x_{itr} : i\in 1,\ldots,N, t\in 1,\ldots,T, r\in 1,\ldots,R\}$. The true

sequence of reachable system states is represented in an unknown matrix $\Gamma^* = \{\gamma^*_{itr} : i \in 1,\ldots,N, t \in 1,\ldots,T, r \in 1,\ldots,R\}$. Each entry in Γ^* represents the state of a node i at time point t and corresponds to each measurement x_{itr}, where replicate measurements indexed by r are assumed to have the same state. We omit the index r for notational simplicity for the rest of this section, but the reader should be aware that optimization in the HMM is done by multiplying all replicate emission probabilities for determining the entries in the Viterbi matrix [as shown in Equation (4)].

Intuitively, Γ^* provides a classification of measurements into measurements coming from an active state and those from an inactive state. We infer an estimate $\hat{\Gamma}^*$ for Γ^* by using an HMM $H = (W, \Gamma, A, e)$. Here, W represents the range of possible values for observations, i.e. all positive real-valued intensities generated by the array scanning software (in our case $[0, 2^{16}-1]$). Γ is the set of possible states, as derived in Section 2.1. A is a matrix of transition probabilities for the system states. We refer to e as the emission probability $e(\mathbf{x_t}) = p(\mathbf{x_t}|\hat{\boldsymbol{\gamma}}^*_t, \hat{\Theta})$ [Equation (4)], where $\hat{\Theta}$ is the matrix of estimated model parameters [Equation (3)]. e corresponds to the likelihood of observing data point $\mathbf{x_t}$ given its state $\hat{\boldsymbol{\gamma}}^*_t$. Note that $\mathbf{x_t}$ is a column in the measurement matrix X, i.e. a vector of intensity values.

We use the Viterbi training algorithm (Durbin *et al.*, 1998) to find an optimal sequence of system states and optimize the transition matrix A as well as the parameter matrix $\hat{\Theta}$. We initialize $\hat{\Gamma}^*$ by sampling random states from Γ, while preserving the order of the states, and the transition matrix A to uniform probabilities for all state transitions. We estimate model parameters $\hat{\Theta}$ depending on $\hat{\Gamma}^*$ [Equations (2) and (3)]. Now $\hat{\Gamma}^*$ is updated using the HMM and the procedure iterated until convergence, as described in Durbin *et al.* (1998). This yields the final state matrix estimate $\hat{\Gamma}^*$ used for the likelihood calculation, described in the next section.

2.3 Likelihood model

For calculation of emission probabilities in Section 2.2 as well as computation of the total network likelihood in the structure search (Section 2.4), we set up a likelihood score that describes the probability of observing measurements under our model, represented by the network hypothesis. Given a state matrix estimate $\hat{\Gamma}^*$, each measurement x_{itr} for protein i, time point t and replicate r comes from an 'active' normal distribution $\mathcal{N}(\mu_{i1}, \sigma_{i1})$, if its state $\hat{\gamma}^*_{itr} = 1$, and from a 'passive' normal distribution $\mathcal{N}(\mu_{i0}, \sigma_{i0})$, if $\hat{\gamma}^*_{itr} = 0$:

$$x_{itr} \sim \begin{cases} \mathcal{N}(\mu_{i0}, \sigma_{i0}), \textit{ if } \hat{\gamma}^*_{itr} = 0 \ (\textit{passive}) \\ \mathcal{N}(\mu_{i1}, \sigma_{i1}), \textit{ if } \hat{\gamma}^*_{itr} = 1 \ (\textit{active}) \end{cases} \tag{2}$$

The parameters of each distribution for one protein are obtained as unbiased empirical mean and SD of all measurements for this protein in the given class. This yields the parameter matrix:

$$\hat{\Theta} = \{\hat{\theta}_{i0}, \hat{\theta}_{i1}\} = \{(\hat{\mu}_{i0}, \hat{\sigma}_{i0}), (\hat{\mu}_{i1}, \hat{\sigma}_{i1})\} \, \forall i \in 1,\ldots,N \tag{3}$$

Now we can write the likelihood for a data point x_t as:

$$\begin{aligned} p(\mathbf{x_t}|\Phi) &= p(\mathbf{x_t}|\hat{\boldsymbol{\gamma}}^*_t, \hat{\Theta}) \\ &= \prod_{i=1}^{N} \prod_{r=1}^{R} p(x_{itr}|\hat{\boldsymbol{\theta}}_{i\hat{\gamma}^*_{itr}}) \end{aligned} \tag{4}$$

The total likelihood for a network hypothesis Φ can be written as:

$$\begin{aligned} p(X|\Phi) &= p(X|\hat{\Gamma}^*, \hat{\Theta}) = \prod_{t=1}^{T} p(\mathbf{x_t}|\hat{\boldsymbol{\gamma}}^*_t, \hat{\Theta}) \\ &= \prod_{t=1}^{T} \prod_{i=1}^{N} \prod_{r=1}^{R} p(x_{itr}|\hat{\boldsymbol{\theta}}_{i\hat{\gamma}^*_{itr}}) \end{aligned} \tag{5}$$

2.4 Network structure search

The previous two sections dealt with the assessment of a single network hypothesis. However, the aim of our method is to optimize the network structure with respect to the network likelihood, so a suitable network structure search strategy has to be chosen. We use a GA as sampling-based technique for network structure search that optimizes a whole population of candidate networks. Studies of Wahde and Hertz (2000) and Spieth *et al.* (2006) show the usefulness of evolutionary strategies for network reconstruction. We evolve a population of networks in parallel by selection and mutation of the individuals. Selection should choose the fittest individuals and mutations should be beneficial for the overall fitness of all networks. Further, we allow 'communication' between the networks in form of crossovers. To avoid overfitting by inclusion of too many edges in the networks, we use the Bayesian information criterion (BIC; Schwarz, 1978) as fitness score, which penalizes higher numbers of edges and is calculated from the likelihood [Equation (5)]:

$$\mathrm{BIC} = -2\log(p(X|\Phi)) + K\log(n)$$

where K is the number of edges in Φ and n is the number of data points in X. This will result in sparse network structures.

2.4.1 GA specification A population $P = \{\Phi_j : j \in 1,\ldots,p\}$ of p networks, a crossover (selection) rate q $(1-q)$ and mutation rate m with $q, m \in [0;1]$ are given. During selection we choose a fraction $\lfloor (1-q)p \rfloor$ individuals with probability proportional to their fitness. We require that BICs of selected networks are smaller than the median of the BICs of all individuals in the population, mimicing a simple greedy search, but leaving the possibility for selecting suboptimal moves. The selected individuals are added to the next generation population P'.

For crossing over we choose $\lfloor \frac{qp}{2} \rfloor$ random pairs from P, again proportional to each individuals' fitness. To perform crossing over of two networks, each network adjacency matrix is represented as a vector (simply attaching all columns to each other) and two point crossover is performed for these vectors. The modified individuals are added to P' if their BICs are smaller than the median BIC for all individuals in P'. In case that after crossover the size of the modified population P' is smaller than p, we add as many random individuals from P to P', such that the population size stays constant.

Finally, we perform mutation of $\lfloor mp \rfloor$ networks chosen from the new population P'. For each selected network a random edge is drawn and its type is changed randomly to one of the remaining types. As an example, given an edge $\phi_{vw} = 2$, it can be either changed to $\phi'_{vw} = 1$ or $\phi'_{vw} = 0$. Mutations are allowed if the fitness of the individual improves by introducing the mutation.

These three steps are repeated until a prespecified number of iterations (usually 1000) have been run or the median of all BICs in the population does not change for 10 times in a row. At the end of the GA, the population of candidate networks is combined into a final network by including each edge that occurs in more than a prespecified fraction of all networks in the population (usually 50% if not stated explicitly).

2.5 Data generation and preprocessing for HCC1954 RPPA data

The human breast cancer cell line HCC1954 was cultivated as recommended by ATCC and cells were split three times per week. For stimulation experiments, cells were seeded in 6-well plates, cultivated for 24 h and serum-starved in phenol red-free medium for additional 24 h. EGF (Sigma, Steinheim, Germany) and HRG (Biovision, Mountain View, CA, USA) were added to the cells to a final concentration of 5 nM. After times 0, 4, 8, 12, 16, 20, 30, 40, 50 and 60 min, medium was replaced by ice-cold PBS and plates were put on ice. Afterwards, PBS was aspirated and cells were harvested by manual scraping in 40 µl lysis buffer [M-PER (Pierce, Bonn, Germany), Complete Mini, PhosSTOP (Roche, Mannheim, Germany)]. Cells were lysed for 20 min at 4 °C. After centrifugation, total protein concentration was determined using the BCA method (Pierce, Bonn, Germany) and all samples were adjusted to the same protein concentration. Prior to printing, samples were mixed with Tween-20 to a final concentration of 0.05%. Three biological replicates were generated at three different days. The samples were printed in triplicate onto nitrocellulose coated glass slides [Oncyte;

Table 1. Proteins and phosphorylation sites used in the RPPA analysis

Protein	Phosphosite	Protein	Phosphosite	Protein	Phosphosite
AKT	S473	EGFR	Y1068	ERBB2	Y1112
ERBB3	Y1289	ERBB4	Y1162	ERK1/2	T202,Y204
GSK3	Y279,Y216	MEK1/2	S217,S221	MTOR	S2448
p38	T180,Y182	p70S6K	T389	PDK1	S241
PKCα	S657,Y658	PLCγ	S1248	PRAS	T246
SRC	Y416				

Grace-Biolabs (Bend, OR, USA)] with a contact spotter [2470 Arrayer; (Aushon Biosystems, billerica, MA, USA)] using 180 µm pins. Slides were blocked in 50% Odyssey Blocking Buffer LI-COR (Lincoln, NE, USA) in PBS containing 5 mM sodium fluoride and 1 mM vanadate. Primary antibodies were diluted 1:300 in antibody diluent with background reducing components (Dako, Glostrup, Denmark). Alexa 680 labelled secondary antibodies (Molecular Probes, Darmstadt, Germany) were diluted 1:5000 in PBS (+0.2% NP-40, 0.02% SDS + 0.5% BSA). After drying, arrays were scanned using the Odyssey Infrared Imaging System (LI-COR, Lincoln, NE, USA) and signal intensities were determined with GenePix Pro 5.0 (Molecular Devices, Sunnyvale, CA, USA). Sample normalisation was done using Fast Green FCF dye (see Loebke *et al.*, 2007; Luo *et al.*, 2006) to account for different protein concentrations in each spot on the array. Replicate time courses were centred around their common mean to remove systematic shifts in the intensities. Sixteen antibodies for specific phosphorylation sites were used to obtain signal intensities of phosphorylated protein. A list of the proteins and phosphorylation sites is shown in Table 1. The antibodies were obtained from the following companies: ERBB4 and GSK3 from Epitomics (Burlingame, CA, USA), ERBB2 from Millipore (billerica, MA, USA), MEK1/2 from Sigma, PKCα from Abcam (Cambrige, UK) all others from Cell Signaling (Beverly, MA, USA).

3 RESULTS AND DISCUSSION

3.1 Simulations

3.1.1 Generation of simulation data Given a number of nodes and a number of input stimuli, we generated networks as follows: starting at the input stimuli, we sampled outgoing activation edges until all nodes were connected and added 20% of the number of activating edges as inhibitions to retrieve fully connected networks. This ensured that all nodes could be reached by a stimulus signal and that feed forward and feed back loops were included in the network.

Given such a sampled network, a data matrix X (as defined in Section 2.2) was constructed. We refer to parameters *nstim* as the number of distinct input stimuli and *cstim* as the number of stimulus combinations. Each stimulus gives rise to a separate experiment, so for each stimulus a separate state matrix was constructed by our effect propagation. A state transition matrix for each stimulus was built up by sampling T columns with replacement from each state matrix, while the order of the states was preserved. Each column in the state transition matrix was repeated R times to generate replicates. Finally, all state matrices were attached to get the total state matrix Γ, and all state transition matrices were attached to generate Γ^*.

Then, for each time point, replicate and node a measurement x_{itr} was sampled from two Gaussian distributions, either from $x_{itr} \sim \mathcal{N}(1200,400)$ if $\gamma^*_{itr}=0$ or from $x_{itr} \sim \mathcal{N}(2000,1000)$ if $\gamma^*_{itr}=1$. The parameters for the Gaussians (mean and variance) were chosen similar to the observed measurements in our real data. We chose $T=10$ and $R=9$ as number of time points and number of replicates for the simulations.

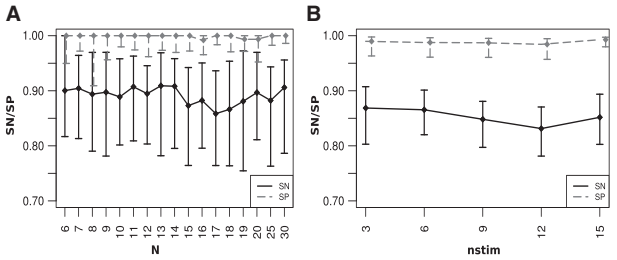

Fig. 2. Performance of state recovery for increasing number of nodes N (**A**) and number of stimuli *nstim* (**B**).

3.1.2 Recovering the true state sequence We tested how good the HMM from Section 2.2 is able to recover a true state sequence Γ^*. For this purpose we sampled networks for increasing number of nodes and performed the effect propagation from Section 2.1 for different numbers of input stimuli. Γ^* matrices were sampled 100 times for each network and stimulus combination, and for each Γ^* data was generated as described in Section 3.1.1. We performed the HMM state sequence search for all data matrices and compared the resulting state transition matrix $\hat{\Gamma}^*$ with the corresponding reference Γ^* in terms of sensitivity SN$=$(TP/(TP+FN)) and specificity SP$=$(TN/(TN+FP)), counting the true and false occurrences of the entries in $\hat{\Gamma}^*$. Figure 2 shows that the recovery performance stays constantly high at average values of around SN$=0.84$ and SP$=0.95$ for networks up to 30 nodes, and around SN$=0.83$ and SP$=0.97$ for increasing the number of input stimuli *nstim*. Hence, given an unknown series of system states, the HMM is able to identify the correct states, even for bigger networks with up to 30 nodes.

3.1.3 Performance of structure search We sampled random networks and generated intensity measurements as described in Section 3.1.1. For network comparisons, we counted the number of truly inferred edges (TP), truly not inferred edges (TN), erroneously inferred edges (FP) and erroneously not inferred edges (FN). Note that now we counted edges in the network, and not entries in the state matrix as in the previous section. Network reconstructions were done for artificial networks of size $N=10$ with population sizes from $p \in \{100,250,500\}$, $q=0.3$ and $m=0.8$. Also increasing numbers of different input stimuli were compared. We chose $nstim \in \{1,2\}$ and $cstim \in \{0,1\}$. To measure the performance, the GA was run for 25 sampled networks, each time with the maximum number of generations set to 1000. The edge inclusion threshold for the final network (Section 2.4) was varied in $[0;1]$, and the respective final network for each given threshold was compared with the original net, yielding SN and SP values for the generation of receiver operator characteristic (ROC) curves and area under curve (AUC) values.

Figure 3 shows that the reconstruction performance was limited for the case of $nstim=1, cstim=0$ and increasing population size $p=100$, $p=250$ and $p=500$ (AUCs $0.57, 0.6, 0.61$), while a slight increase could be found for the bigger population size. This is expected because the use of a bigger population ensures broader sampling of the network search space. However, true increase in performance was reached when including two distinct

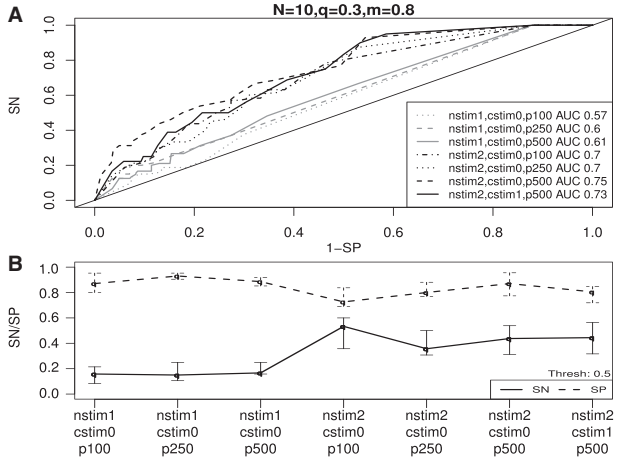

Fig. 3. (**A**) ROC curves and AUCs for different settings of input (*nstim*) and combinatorial stimuli (*cstim*) and population sizes (*p*). SN and SP were calculated as average of each 25 network reconstructions with network size of $N=10$. (**B**) Example SN and SP plot for $th=0.5$ for all settings. For $p=500$, SP was high at ~ 0.83, while SN increased from ~ 0.17 to ~ 0.4. This shows, that DDEPN found edges with strong support from the data with low FP rates. The increase in SN for bigger population sizes shows, that broader sampling of the network search space yielded better inference results.

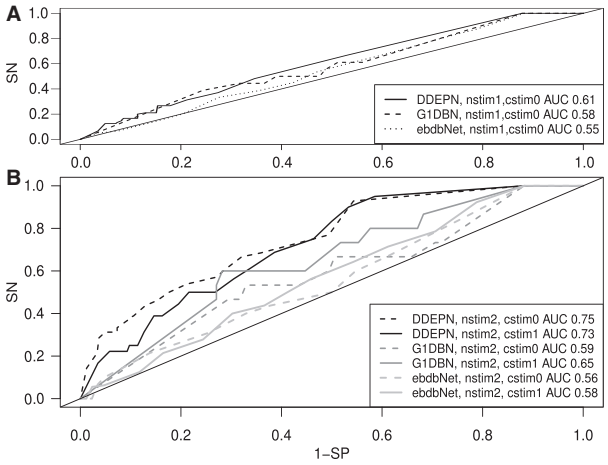

Fig. 4. ROC curves and AUCs for DDEPN network reconstruction compared with G1DBN and ebdbNet. (**A**) For $nstim=1, cstim=0$, a slight improvement of AUCs was observed, and performances were limited for all approaches. (**B**) For $nstim=2, cstim=\{0,1\}$, a clear increase in AUC was found for DDEPN, showing the improved quality of the network reconstructions.

stimuli ($nstim=2$) and further including one stimulus combination ($cstim=1$). Here, the AUCs increased to 0.75 and 0.73, respectively. As before, for higher population sizes the AUCs increased (from 0.7 to 0.75). In Figure 3B, for a fixed threshold $th=0.5$, SN and SP were plotted for each simulation test. In the $nstim=1$ case, SP was high around 0.87, while SN was rather low around 0.17. For $nstim=2$, SN increased to values around 0.4, while SP improved from 0.78 to 0.83 for growing population sizes. This showed that inclusion of multiple stimuli triggering signalling in the network at different input nodes increased the amount of information that could be used to find the signalling connectivity, and thus resulted in better identification of true edges in the network (apparent in the increasing SN values). However, it was also apparent that SN levels were still rather low, so the reconstruction missed quite a number of edges, that should have been found. On the other hand, the high values for SP ensured that inferred edges were those with strong support from the experimental measurements, and thus could be expected to be meaningful. Summarising this, our method was able to recover parts of the original signalling networks but did this with high specificity, meaning that predominantly true edges were found. This makes the method useful for the generation of new interaction hypotheses.

3.2 Comparison with related approaches for network inference

We compared our method with the DBN reconstruction approach G1DBN of Lébre (2009) and to a recent method of Rau *et al.* (2010), called ebdbNet. For network size $N=10$, 25 networks were simulated and the reconstruction performed. We repeated each network reconstruction 100 times and calculated ROCs and AUCs as shown in Section 3.1.3. The results are depicted in Figure 4. For $nstim=1, cstim=0$, DDEPN performed slightly better than G1DBN and ebdbNet (AUCs 0.61 for DDEPN, 0.58 and 0.55 for G1DBN and ebdbNet, respectively). However, the performance was limited in this case for all methods. Using $nstim=2$, DDEPN clearly outperformed G1DBN and ebdbNet, for both $cstim=0$ (AUC = 0.75) and $cstim=1$ (AUC = 0.73). This highlighted the ability of DDEPN to make use of the additional information gained from multiple perturbations. The better performance had its price in terms of computation time. On average, a 10 node network with three input stimuli was reconstructed in around 7000 s using DDEPN, while G1DBN and ebdbNet completed this task in a few seconds. However, the network inferred in DDEPN was derived from a whole population of candidate networks that covers larger portions of the network search space than the other two approaches. Calculation was done on a Quad-Core AMD Opteron(tm) 2.7 GHz machine with 64 GB memory, on which each 14 cores were used in parallel to optimize the population of networks in the GA. Because of the better performance of the reconstruction and the fact that DDEPN was able to infer both activation and inhibition edges, we think this price is worth paying.

We also compared our new approach DDEPN with the related approach deterministic effects propagation networks (DEPNs) of Fröhlich *et al.* (2009), but were not able to infer reasonable networks under the settings applied here. This was for two reasons: first, DEPN was designed for the setting of few time points and many perturbations, i.e. the information on the signal flow is collected through the perturbation of many or even all nodes in the network, so a small number of time points is sufficient. In DDEPN, we only introduced few perturbations and got additional information on the signal flow through a higher number of time points. Therefore, it was not possible to capture the signalling relationships of the components downstream the perturbed nodes with high resolution using DEPN. Second, DEPN cannot model perturbations as stimulation, but only as knockdown. So both methods have specific requirements and cannot be exchanged without care for different datasets. However, DDEPN can also be run with more than two perturbed nodes and has the advantage, that both types of perturbation can be included.

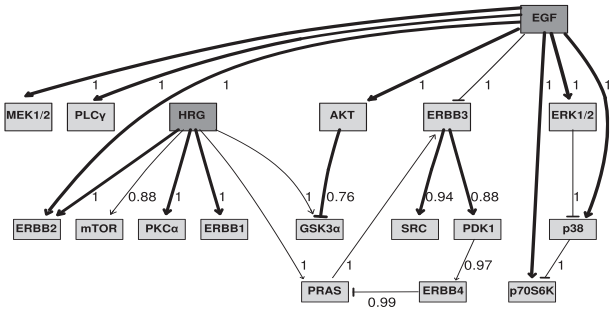

Fig. 5. Network reconstructed from HCC1954 data. Interactions found in the literature are marked as thick lines. Dark nodes mark the input stimuli. The numbers at the edges show the proportion of networks in the final GA population, in which the respective edge was contained.

3.3 Signalling networks in HCC1954 breast cancer cell line

We used DDEPN to reconstruct a signalling network from our data. Parameters were chosen as population size 500, maximum iterations 1000, crossover rate $q=0.3$ and mutation rate $m=0.8$. The inferred network is shown in Figure 5. An edge is shown if it was contained in at least 50% of the networks in the final population ($th=0.5$), allowing only interactions with strong support from the data. We saw several signal cascades in our network that were known from the literature. For example, we inferred the regulation HRG → ERBB1. Olayioye *et al.* (1999) showed that HRG is an activator of the ERBB-Dimers 1/3 and 1/4, which supported this result. Activation of ERBB2 by EGF or HRG could be found in Jones *et al.* (1999) (EGF/HRG → ERBB2/3), which also supported activation of PKCα by HRG through the cascade HRG → ERBB2 → PKCα, since crosstalk between ERBB2 and PKCα in ERBB2 overexpressing breast cancer cells was reported by Magnifico *et al.* (2007). The result was further interesting, since our HCC1954 cells overexpress ERBB2. Kim *et al.* (2009) reported activation of p38 by ERBB2 in ERBB2 overexpressing breast cancer cells, reflected in our activation EGF → p38. The activations of MEK1/2, ERK1/2 and p70S6K by EGF are key elements in the classical MAPK signalling cascade EGF → ERBB1/1 → GRB2 → SOS1 → RAS → RAF1 → MEK1/2 → ERK1/2 → p70S6K. EGF → ERBB1/1 → PLCγ was shown by Kim *et al.* (1990), which demonstrated the relevance of the activation EGF → PLCγ in our network. Further EGF → AKT ⊣ GSK3α is found in the cascade EGF → ERBB → GRB2 → GAB1 → PI3K → AKT ⊣ GSK3α.

More hypothetical interactions included the inferred SRC activation (ERBB3 → SRC), interpreted as activation of SRC by ERBB2 (see e.g. Luttrell *et al.*, 1994; Mao *et al.*, 1997; Xian *et al.*, 1997) through the ERBB2/3 heterodimer. Finally, PDK1 activation by receptor tyrosine kinases was shown by Cohen *et al.* (1997) in insulin signalling. We found the activation ERBB3 → PDK1, which supported the hypothesis that the cascade ERBB1/3 → PI3K → PIP3 → PDK1 (see Oda *et al.*, 2005; Vanhaesebroeck *et al.*, 1997) might also play a role in cancer-related signal transduction processes.

All of these inferred and literature confirmed interactions had high support by our data (occurrence in >75% of all networks in the final population, see edge labels in Figure 5). Our findings showed that literature knowledge was reproduced well by our method and in addition allowed for discussion of the newly inferred interactions. However, there were cases, where interactions would have been expected, but were not found in the network. For example, in the classical MAPK cascade, MEK1/2 phosphorylates ERK1/2 directly. In our network, the interaction between MEK1/2 and ERK1/2 was not found, but only direct activations of the two proteins by EGF. The reason was that we only measured phosphorylation at time points 8 and 12 min. Activation of MEK1/2 and ERK1/2 is expected around 10 min after stimulation, but in our data we saw peaks for both proteins at the 12 min time point. Thus, we could not resolve this cascade at a higher resolution. Another problem arised when proteins of a signalling cascade were not measured on the array, as seen, for example, for several of the components in the MAPK cascade (e.g. RAS, RAF, etc.). So even if a direct edge between two proteins is found, it has to be carefully assessed whether this edge is a direct influence or an indirect interaction over multiple intermediate steps. Our data only represents the abundance of phosphorylated protein in the cells, which might increase or decrease in response to a ligand. All interactions from such data are abstract influences between two proteins that have to be validated in further experiments. However, considering these kind of caveats and performing careful interpretation of the results makes our method suitable for the generation of reasonable hypotheses on signalling cascades.

4 CONCLUSION

In this work, we showed a novel approach for the reconstruction of signalling networks from high-throughput proteomics data generated on RPPAs. The phosphorylation of 16 proteins related to ERBB signalling in human breast cancer was measured after three different stimulations (EGF, HRG, EGF+HRG). We devised a method that describes the signalling dynamics in a discretized way and infers the most likely series of activation states for all proteins at each time point given a candidate network by using an HMM. A likelihood model was set up to describe the goodness of fit of our measurements for a particular candidate network. To find a best fitting network, the space of possible network hypotheses is searched by a GA. We tested our method on simulated data and found good performance for the reconstruction of given networks, and improved performance over two other DBN approaches that were suitable for analysing our kind of data. Finally, we used our method to infer a signalling network from real protein phosphorylation measurements, generated by RPPAs for cell lysates from the human breast cancer cell line HCC1954. We successfully identified parts of signalling cascades, as they could be found in the literature, in particular from the MAPK and AKT signalling cascades, with some interactions originally found in ERBB2 overexpressing breast cancers (e.g. HRG→PKCα). As new technologies such as RPPAs arise that make parallel measurement of larger numbers of proteins feasible, the need for suitable methods for the analysis of this kind of data is apparent. Our method aims precisely at this niche and gives an example for upcoming systems proteomics methodology.

ACKNOWLEDGEMENTS

We thank Anika Joecker, Maria Fälth, Stephan Gade, Marc Johannes and Alexander Kerner for proofreading the manuscript. We also thank Dirk Ledwinka for technical support.

Funding: The Helmholtz Alliance on Systems Biology network SB-Cancer, through the German Federal Ministry of Education and Science (BMBF) project BreastSys in the platform Medical Systems Biology; the network IG Cellular Systems Genomics (01GS0864) in the platform NGFNplus; the Clinical Research Group 179 through the German Research Foundation.

Conflict of Interest: none declared.

REFERENCES

Akutsu,T. *et al*. (1999) Identification of genetic networks from a small number of gene expression patterns under the Boolean network model. *Pac. Symp. Biocomput.*, 17–28.

Cohen,P. *et al*. (1997) PDK1, one of the missing links in insulin signal transduction? *FEBS Lett.*, **410**, 3–10.

Durbin,R. *et al*. (1998) *Biological Sequence Analysis*, 1 edn. Cambridge University Press, Cambridge.

Friedman,N. *et al*. (2000) Using Bayesian networks to analyze expression data. *J. Comput. Biol.*, **7**, 601–620.

Fröhlich,H. *et al*. (2008) Estimating large scale signaling networks through nested effect models with intervention effects from microarray data. *Bioinformatics*, **24**, 2650–2656.

Fröhlich,H. *et al*. (2009) Deterministic effects propagation networks for reconstructing protein signaling networks from multiple interventions. *BMC Bioinformatics*, **10**, 322.

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

Heckerman,D. (1996) A tutorial on learning with Bayesian networks. *Innovations in Bayesian Networks*, Springer, Berlin/Heidelberg, pp. 33–82.

Imoto,S. *et al*. (2002) Estimation of genetic networks and functional structures between genes by using Bayesian networks and nonparametric regression. *Pac. Symp. Biocomput.*, 175–186.

Jones,J.T. *et al*. (1999) Binding specificities and affinities of EGF domains for ErbB receptors. *FEBS Lett.*, **447**, 227–231.

Kim,I.-Y. *et al*. (2009) Overexpression of ErbB2 induces invasion of MCF10A human breast epithelial cells via MMP-9. *Cancer Lett.*, **275**, 227–233.

Kim,J.W. *et al*. (1990) Tyrosine residues in bovine phospholipase C-gamma phosphorylated by the epidermal growth factor receptor in vitro. *J. Biol. Chem.*, **265**, 3940–3943.

Lébre,S. (2009) Inferring dynamic genetic networks with low order independencies. *Stat. Appl. Genet. Mol. Biol.*, **8**, Article 9.

Loebke,C. *et al*. (2007) Infrared-based protein detection arrays for quantitative proteomics. *Proteomics*, **7**, 558–564.

Luo,S. *et al*. (2006) Quantitation of protein on gels and blots by infrared fluorescence of coomassie blue and fast green. *Anal. Biochem.*, **350**, 233–238.

Luttrell,D.K. *et al*. (1994) Involvement of pp60c-src with two major signaling pathways in human breast cancer. *Proc. Natl Acad. Sci. USA*, **91**, 83–87.

Magnifico,A. *et al*. (2007) Protein kinase calpha determines HER2 fate in breast carcinoma cells with HER2 protein overexpression without gene amplification. *Cancer Res.*, **67**, 5308–5317.

Mao,W. *et al*. (1997) Activation of c-Src by receptor tyrosine kinases in human colon cancer cells with high metastatic potential. *Oncogene*, **15**, 3083–3090.

Markowetz,F. *et al*. (2005) Non-transcriptional pathway features reconstructed from secondary effects of RNA interference. *Bioinformatics*, **21**, 4026–4032.

Murphy,K. and Mian,S. (1999) Modelling gene expression data using dynamic Bayesian networks. *Technical report*, University of California, Berkeley.

Nelander,S. *et al*. (2008) Models from experiments: combinatorial drug perturbations of cancer cells. *Mol. Syst. Biol.*, **4**, 216.

Oda,K. *et al*. (2005) A comprehensive pathway map of epidermal growth factor receptor signaling. *Mol. Syst. Biol.*, **1**, 2005.0010.

Olayioye,M.A. *et al*. (1999) ErbB receptor-induced activation of stat transcription factors is mediated by Src tyrosine kinases. *J. Biol. Chem.*, **274**, 17209–17218.

Pe'er,D. *et al*. (2001) Inferring subnetworks from perturbed expression profiles. *Bioinformatics*, **17** (Suppl. 1), S215–S224.

Rau,A. *et al*. (2010) An empirical Bayesian method for estimating biological networks from temporal microarray data. *Stat. Appl. Genet. Mol. Biol.*, **9**, Article 9.

Sachs,K. *et al*. (2005) Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, **308**, 523–529.

Schwarz,G. (1978) Estimating the dimension of a model. *Ann. Stat.*, **6**, 461–464.

Segal,E. *et al*. (2005) Learning module networks. *J. Mach. Learn. Res.*, **6**, 557–588.

Slamon,D.J. *et al*. (1987) Human breast cancer: correlation of relapse and survival with amplification of the HER-2/neu oncogene. *Science*, **235**, 177–182.

Slamon,D.J. *et al*. (1989) Studies of the HER-2/neu proto-oncogene in human breast and ovarian cancer. *Science*, **244**, 707–712.

Spieth,C. *et al*. (2006) Comparing evolutionary algorithms on the problem of network inference. In *Proceedings of the Genetic and Evolutionary Computation Conference*. Seattle, Washington.

Tegner,J. *et al*. (2003) Reverse engineering gene networks: integrating genetic perturbations with dynamical modeling. *Proc. Natl Acad. Sci. USA*, **100**, 5944–5949.

Vanhaesebroeck,B. *et al*. (1997) Phosphoinositide 3-kinases: a conserved family of signal transducers. *Trends Biochem. Sci.*, **22**, 267–272.

Wahde,M. and Hertz,J. (2000) Coarse-grained reverse engineering of genetic regulatory networks. *Biosystems*, **55**, 129–136.

Xian,W. *et al*. (1997) Activation of erbB2 and c-src in phorbol ester-treated mouse epidermis: possible role in mouse skin tumor promotion. *Oncogene*, **14**, 1435–1444.