Inferring large gene networks from microarray data: a constraint-based approach

Liviu Badea
National Institute for Research in Informatics
8-10 Averescu Blvd. Bucharest, Romania
badea@ici.ro

Abstract

We apply a constraint-based Bayesian network inference algorithm to the problem of discovering the network of genes involved in four types of lung carcinoma using microarray gene expression data. The large number of variables (892), the small sample size (73 – typical for current microarray technology), as well as the noisy data require the ability to reconcile possibly unreliable conditional independence tests producing mutually inconsistent results. Our improved constraint-based algorithm QFCI is especially suited for inferring the global gene network structure (even in the presence of unknown hidden variables) rather than just fragmentary high-scoring substructures. Moreover, QFCI was able to reconstruct a plausible substructure of the 'small cell' subtype involving an expression profile typical for neuroendocrine differentiation.

1 Introduction and motivation

The functioning of biological systems at a molecular level is an ideal candidate for AI knowledge discovery, due to the complexity and heterogeneity of the entities and mechanisms involved (transcriptional regulation, post-translational modifications, protein to protein interactions, etc). Although most fundamental biological mechanisms have been uncovered by the painstakingly slow experimental procedures of biologists, achieving a global understanding of biological systems requires automated discovery approaches. On the experimental side, microarray technology has enabled the simultaneous measurement of the expression levels of virtually all genes of an organism. Such data acquisition methods of unprecedented scope have to be complemented by suitable discovery algorithms, able to infer not just shallow associations, correlations or clusters, but also causal influence relations among genes. For example, these would be very helpful for understanding the mechanisms of complex diseases, such as cancer, for which numerous microarray studies have been performed.

The most basic data analysis tools applied to microarray datasets involve (supervised or unsupervised) *clustering* of genes and/or expression profiles (samples). Although clustering is very useful for tracking down groups of genes with similar expression profiles or possibly well correlated with a phenotypic manifestation, it is unable to infer the precise (causal) mechanisms involving these genes. Plausible mechanisms could sometimes be guessed for the genes that were previously known, but very frequently existing knowledge turns out to be too fragmentary to make out the details of the mechanism.

A more ambitious approach would try to employ knowledge discovery methods more refined than clustering. The probabilistic as well as noisy nature of the data makes Bayesian network inference a promising candidate. This setting may even allow the inference of causal relationships between genes, possibly from observational data only [5],[9].

In this paper, we address the problem of inferring a large network of genes involved in four types of lung carcinoma from the microarray gene expression data of Garber et al. [4]. The Garber dataset contains measurements of the expression levels of 23,000 cDNA clones (corresponding to 17,108 unique genes) in 73 tissue samples (from patients with adenocarcinoma (AC), squamous cell carcinoma (SCC), small cell (SCLC) and large cell lung cancer (LCLC), as well as normal tissue). 918 cDNA clones, corresponding to 835 unique genes whose expression varied significantly across the tissue samples were selected by Garber et al. and standard average-linkage hierarchical clustering revealed a faithful match with the morphological classification of the tumors (while allowing an even more refined subdivision of AC into 3 distinct subgroups).

But whereas clustering only produces groups of co-expressed genes, our goal consists in discovering the finer underlying structure of their interactions.

The idea of inferring Bayesian network models from gene expression data has been explored before [10],[3]. Still, the specificities of our problem make applying existing approaches particularly difficult.

First, most causal inference algorithms were designed to deal with tens to a few hundreds of variables (e.g. TETRAD [9], BNPowerConstructor [1]), while microarray data contain measurements for thousands to tens of thousands of genes. Our setting involves 892 variables (884

clones from the Garber selection as well as 8 additional discrete variables representing the disease subtypes).

Second, we need to be able to deal with a potentially very *large number of hidden (latent/unobserved) variables* (although most of the genes with significant differential expression were included, these only cover about 3% of the whole genome – still there are also other potentially relevant hidden factors, like post-translational modifications or protein to protein interactions).

Third, current microarray data is extremely *noisy* and provides *too few samples* (in our case just N=73) to allow the reliable reconstruction of a single model with a high statistical significance. Sophisticated approaches, e.g. using Bayesian Model Averaging [2] have been proposed to deal with network structure in a Bayesian manner, especially when there might be many models (usually exponentially many) with a non-negligible posterior. Decoupling the computationally intractable problem[1] of evaluating the Bayesian posterior probability of certain structural network features into two easier sub-problems (evaluating in a closed form the probability for a given variable order and summing over a sample of all possible orders [2]) alleviates but doesn't completely solve the computational difficulty, especially due to the sheer number of variables in our problem (≈1000).

Finally, the interpretation of the resulting structure by a human expert critically depends on providing a *single representation of all the alternative models* (such as the partial graph produced by a constraint-based structural inference method [5],[9]) rather than an explicit enumeration of a very large number of models, or the display of just the high-scoring *fragments* of the network (instead of providing the whole network, possibly with confidence factors in its various structural features).

While Bayesian model averaging over the space of structures (or even just variable orders) is computationally intractable in our setting with current technology, *constraint-based methods*, which do not search the space of models, seem an appealing alternative. (Scoring-based procedures also have difficulties in dealing with latent/ hidden variables, especially whenever the positions of these variables in the structure are unknown.)

2 An improved constraint-based algorithm

Algorithms like IC* of Pearl and Verma [5] or the more efficient Fast Causal Inference (FCI) algorithm of Spirtes et al. [9] start with a completely connected network and simply use conditional independence (CI) tests to find separators for edges representing indirect influences. Finally, edge endpoints are placed based on the separators found. IC* and FCI are thus very close our problem's requirements, since:

- they are able to produce a single representation of the (partial) knowledge inferable from the data
- they are able to deal with latent (unobserved) variables
- FCI is fast enough to deal with the about 1000 variables (genes) involved in the Garber dataset.

However, they still have certain important drawbacks w.r.t. this particular problem: as they construct causal structures by *categorical inference* based on the results of conditional independence tests, they are sensitive to the high amount of noise in the microarray data as well as to the small sample size (N=73). And although the independence tests used are thus not completely reliable, the algorithm does not provide any quantitative measure of the confidence in the various features inferred.

In this paper we show how constraint-based methods can be made more robust when dealing with small and noisy samples. We then show that our improved QFCI algorithm can be successfully applied to the Garber dataset. Our approach is based on three key ingredients.

First, quantitative information regarding the reliability of the conditional independence tests can be used to quantify our *confidence* in the structural features of the model that are *directly* based on these tests (such as colliders).

Second, we *combine* such confidence factors during logical inference (constraint propagation) to estimate our confidence in *derived* structural features (such as edge endpoints).

Finally, since the small sample size may support several potentially conflicting models, we provide means for coping with such inconsistencies by:

- strengthening the collider and non-collider tests of FCI while preserving its efficiency, and by
- eliminating the remaining inconsistencies (anomalies) as well as all the features inferred from these.

We refer to [5] for the basic notions on Bayesian networks. The output of our QFCI algorithm described below will be a *Partial Ancestral Graph* (PAG, or PDAG in Pearl's terminology), which is a concise representation of an entire equivalence class of graph models. Unlike standard PAGs, ours have confidence factors attached to the undirected edges, as well as to directed edge endpoints.

In the following, we use the notations of [9] for describing PAGs. Briefly, edges can have three kinds of *endpoints* in a PAG: '–', '>' and 'o'. We also use the additional meta-symbol '∗' that stands for any of the three kinds of endpoints.

An '–' endpoint at Y for an edge X ∗–– Y denotes the fact that Y is an ancestor of X in every graph of the equivalence class represented by the PAG, while an '>' endpoint at X for X ∗–> Y means that Y is *not* an ancestor of X. Finally, an 'o' endpoint places no restriction on the ancestor relationships. (See [9] for more details.)

A *collider* is a structure of the form X ∗–> Y <–∗ Z. A collider is called *unshielded* iff X and Z are not adjacent in the PAG.

In the following, we present a constraint-based causal inference algorithm, QFCI, which aims at improving the robustness of the FCI algorithm in the face of noise and small sample sizes. Although our measures of confidence

[1] intractable in our setting (for approximately 1000 variables).

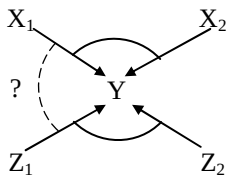
are entirely heuristic, applying model averaging methods in such a setting with about 1000 variables seems out of the question.

Employing a two-valued logic for combining the results of conditional independence tests in noisy domains may lead to inconsistencies, or *anomalies*. In fact, we have observed the occurrence of anomalies not only in microarray datasets (such as the Garber lung carcinoma study [4], the *Rosetta Compendium* of yeast microarray experiments and the *Spellman yeast cell cycle data*), but also in synthetic data. The most important type of anomaly observed was a so-called "*collider anomaly*", which is due to the inconsistencies between different colliders at a given node *Y*.

Recall that FCI recognizes colliders as follows: for non-adjacent X and Z, X *–* Y *–* Z is a collider iff $Y \notin \text{Sep}(X,Z)$ (where $\text{Sep}(X,Z)$ is the first separating set found for X and Z: $X \perp Z \mid \text{Sep}(X,Z)$).

Definition (collider anomaly). Two unshielded colliders detected by the FCI algorithm X_1 *–> Y <–* X_2 ($Y \notin \text{Sep}(X_1,X_2)$) and Z_1 *–> Y <–* Z_2 ($Y \notin \text{Sep}(Z_1,Z_2)$) are *inconsistent* w.r.t. the current set of separators Sep (or short, Sep-*inconsistent*) iff $\exists i,j \in \{1,2\}$ such that X_i and Z_j are not adjacent and X_i *–> Y <–* Z_j is not a collider w.r.t. Sep, i.e. $Y \in \text{Sep}(X_i,Z_j)$.

As can be seen in the following Figure, a collider anomaly appears whenever a pair of arrowheads from different colliders (such as X_1 *–> Y <–* Z_1) doesn't form a collider according to Sep.



Example. Since the true structure of the gene network under study is unknown, we have first tried our algorithms on a randomly generated synthetic network of 40 variables (of which 3 were latent) and 35 edges. The main purpose of this trial was the study of collider anomalies. Samples of sizes N=1000 and N=73 (the latter being equal to the sample size of the Garber dataset) were generated from this network and used to study the influence of N on the undirected (Markov) skeleton of the inferred network, as well as on the edge endpoints (in the directed network).

An example of a collider anomaly in our synthetic network involved the colliders *X7 –> X32 <–* *X22* (*X32* $\notin$ Sep(*X7*,*X22*)=$\emptyset$) and *X7 –> X32 <–* *X36* (*X32* $\notin$ Sep(*X7*,*X36*)={*X39*}) for which *X22 –> X32 <–* X36 is not a collider w.r.t. Sep since *X32* $\in$ Sep(*X22*,*X36*) ={*X32*}.

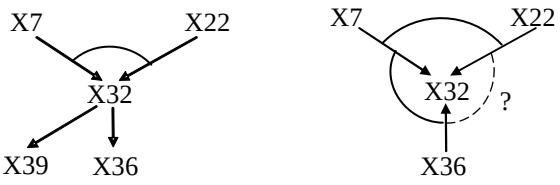


Figure 2. (a) The true graph (b) The collider anomaly

In other words, we have to place an arrow *X22 –> X32* (because *X7 –> X32 <–* *X22* is a collider w.r.t. Sep) and an arrow *X36 –> X32* (since *X7 –> X32 <–* *X36* is also a collider w.r.t. Sep), but these two arrows are inconsistent since *X32* $\in$ Sep(*X22*,*X36*).

As can be seen by looking at the true graph in Figure 2(a), the inconsistency was due in this case to wrongly recognizing *X7 –> X32 <–* *X36* as a collider based on Sep(*X7*,*X36*)={*X39*} which does not contain *X32*. The fact that Sep records only a single separator set (among potentially many others) makes the collider recognition rule of FCI sensitive to errors in the independence test. In this specific case, the error in Sep(*X7*,*X36*) was due to a type-II error in the test $X7 \perp X36 \mid X39$, which succeeded (p-value=0.728 > α=0.05, for N=1000) despite the fact that *X39* does *not* d-separate *X7* from *X36*.

Since it would be very inefficient to recompute all the separators of *X7* and *X36*, we strengthen the FCI collider test by double checking whether adding *X32* to the current separator Sep(*X7*,*X36*) makes *X7* and *X36* dependent: $X7 \not\perp X6 \mid X39,X32$. (If *X32* were a true collider, conditioning on it would d-connect *X7* and *X36*.) If however, *X7* and *X36* remain independent, we cannot safely declare *X32* a collider.

Definition (strong collider test).
For *X* *–* *Y* *–* *Z*, *Y* passes the *strong collider test* iff $Y \notin \text{Sep}(X,Z)$ and $X \not\perp Z \mid \text{Sep}(X,Z) \cup \{Y\}$, while
Y passes the *strong non-collider test* iff $Y \in \text{Sep}(X,Z)$ and $X \not\perp Z \mid \text{Sep}(X,Z) \setminus \{Y\}$.

The strong non-collider test is dual to the strong collider test: we double check whether removing *Y* from the separator Sep(*X*,*Z*) makes *X* and *Z* dependent (as it should if *Y* were not a collider). If it doesn't, we refrain from declaring *Y* a non-collider.

Collider anomalies that are removed by the stronger definition of (non)collider are called *reducible*. The others are called irreducible.

Definition (irreducible collider anomaly). An *irreducible collider anomaly* is a pair of strong colliders X_1 *–> Y <–* X_2 and Z_1 *–> Y <–* Z_2 such that X_i *–> Y <–* Z_j is a strong non-collider for some $i,j \in \{1,2\}$.

QFCI

1. Initialize the undirected graph by computing unconditional independencies

for all pairs of variables X,Y
  perform the unconditional independence test $X \perp Y$ and set $p_u(X,Y)$ to its p-value[2] and $p(X,Y) = p_u(X,Y)$[3]
  if $p_u(X,Y) < \alpha$ (the test failed w.r.t. the significance level α)
    add an undirected edge X o–o Y to the PAG
  else ($p_u(X,Y) \geq \alpha$, i.e. the test succeeded)
    set $\text{Sep}(X,Y) = \emptyset$

2. Refine the undirected graph by conditional independence tests

for $k = 1..k_{\max}$ (consider conditioning sets of increasing size)
  for all undirected edges X o–o Y (in *decreasing* order of their labels $p_u(X,Y)$, i.e. increasing order of the associated unconditional correlations)
    let $N = \text{neighbors}(X) \cup \text{neighbors}(Y)$ [4]
    if $|N| \geq k$
      for all subsets $S \subseteq N$ of size k (constructed by adding k nodes $Z \in N$ to S in *increasing* order of their minimal p-labels[5] $min\{p_u(Z,X), p_u(Z,Y)\}$)
        perform the conditional independence test $X \perp Y \mid S$ and let p be its p-value
        if $p \geq \alpha$ (the test succeeded, i.e. S is a separator)
          delete the undirected edge X o–o Y
          set $\text{Sep}(X,Y) = S$ and $p(X,Y) = p$
          break
        else if $p > p(X,Y)$ then set $p(X,Y) = p$
          (i.e. set $p(X,Y)$ to the maximal p-value of the $X \perp Y \mid S$ CI tests performed so far)

3. Search for potential colliders and non-colliders

for all variables Y
  for all pairs X,Z of *non-adjacent* neighbors of Y
    if X *–* Y *–* Z passes the *strong collider test*
      add the *positive* assertion
        X *–> $Y \wedge Z$ *–> Y : cf
      with confidence factor
      $cf = p(X,Z)(1-p_d)(1-p(X,Y))(1-p(Y,Z))$, where $p_d = \text{p\_value}(X \perp Z \mid \text{Sep}(X,Z) \cup \{Y\}) < \alpha$ is the p-value of the failed independence test performed during the strong collider test[6]
    else if X *–* Y *–* Z passes the *strong non-collider test*
      add the *negative* assertion
        $\neg$ (X *–> $Y \wedge Z$ *–> Y) : cf
      with confidence factor
      $cf = p(X,Z)(1-p_d)(1-p(X,Y))(1-p(Y,Z))$, where $p_d = \text{p\_value}(X \perp Z \mid \text{Sep}(X,Z) \setminus \{Y\}) < \alpha$ is the p-value of the failed independence test performed during the strong non-collider test

4. Eliminate collider anomalies

for all pairs of positive assertions
  X_1 *–> $Y \wedge X_2$ *–> Y : cf_1 and Z_1 *–> $Y \wedge Z_2$ *–> Y : cf_2
  if there exists a negative assertion
    $\neg$(X_i *–> $Y \wedge Z_j$ *–> Y) : cf for some $i,j \in \{1,2\}$
    remove these positive and negative assertions

5. Constraint propagation of assertions

repeat
  propagate assertions (using the propagation rules below)
until no more propagations are possible
remove inconsistencies

The worst-case *complexity* of the algorithm is exponential in the number of variables, because in principle it has to consider all subsets of variables as conditioning sets. Fortunately however, genetic networks typically have small in- and out-degrees k, so that in practice the run-time is dominated by the independence tests conditional on size 1 subsets ($X \perp Y \mid S$, $|S|=1$) and thus is bounded above by $O(n^3)$, where n is the number of variables. This upper bound is only attained for networks of variables that cannot be separated by *unconditional* independence tests, case in which phase 1 results in a network with $O(n^2)$ edges. In such cases, phase 2 will initially start with nodes having n direct neighbors each, so that $O(n^3)$ conditional independence tests $X \perp Y \mid S$ ($|S|=1$) may be performed in the worst case. In practice, it is essential to reduce the number of direct neighbors of nodes as quickly as possible. This is achieved by our ordering heuristic which tries to separate the pairs of variables (X,Y) in increasing order of their unconditional correlation $|r_u(X,Y)|$. This heuristic assumes that (unconditionally) less correlated variables will be easier to separate conditionally. Scheduling independence tests that are more likely to succeed earlier reduces node neighborhoods as quickly as possible, thereby reducing the number of candidate neighbors in the later phases.

Quantitative information is also used in phase 2 when exploring potential separator sets S for a pair of nodes (X,Y). Variables Z with a higher (unconditional) correlation with one of X or Y are more likely to be true neighbors (as opposed to just temporary neighbors at this stage of the algorithm[7]).

---

[2] $p_u(X,Y)$ will be used later to quantify the degree of *unconditional* correlation of X with Y.

[3] $p(X,Y)$ will be the largest p-value of a *conditional* independence test performed so far on X and Y: $p(X,Y) = max_S \text{ p\_value}(X \perp Y \mid S)$. We use $p(X,Y)$ to quantify our confidence in the undirected edge X*–*Y.

[4] For simplicity, we do not reproduce here the more complex determination of a complete set of candidate separators used in FCI (based on *Possible-D-Sep*), which might not be reliable for small samples.

[5] i.e. in decreasing order of their maximal unconditional correlations $max\{|r_u(Z,X)|, |r_u(Z,Y)|\}$.

[6] Note that $p(X,Z) = \text{p\_value}(X \perp Z \mid \text{Sep}(X,Z)) \geq \alpha$ and $p(X,Y) = max_S \text{ p\_value}(X \perp Y \mid S) < \alpha$ (similarly, $p(Y,Z) < \alpha$).

[7] Recall that initially, nodes may be connected to many more other nodes than their direct neighbors.

The search for colliders in phase 3 employs the strong collider and non-collider tests. But since even these stricter tests may not eliminate all collider anomalies, we need to explicitly remove the colliders involved in such anomalies.

To allow a more precise evaluation of the results, the discovery of potential colliders and non-colliders produces assertions labeled by *confidence factors* (based on quantitative information from the independence tests).

Definition (assertions). *Assertions* can be either *positive*

$X \ast\!\!\rightarrow Y \wedge Z \ast\!\!\rightarrow Y : cf$ (p2)

$X \ast\!\!\rightarrow Y : cf$ (p1)

or negative

$\neg (X \ast\!\!\rightarrow Y \wedge Z \ast\!\!\rightarrow Y) : cf$ (n2)

$\neg X \ast\!\!\rightarrow Y : cf$ (n1)

Assertions of the form (p2), (p1), or (n1) are called *definite*, while those of the form (n2) are called *disjunctive* (since they are equivalent to $\neg X \ast\!\!\rightarrow Y \vee \neg Z \ast\!\!\rightarrow Y : cf$).

A positive assertion of the form (p2) means that we are confident with degree *cf* that both arrowheads at *Y* ($X \ast\!\!\rightarrow Y$ and $Z \ast\!\!\rightarrow Y$) should appear in the partial graph. A negative assertion of the form (n2) means that the arrowheads $X \ast\!\!\rightarrow Y$ and $Z \ast\!\!\rightarrow Y$ cannot both appear in the partial graph (with confidence *cf*).

Collider anomalies are inconsistencies in the assertions. Under the usual assumptions (such as faithfulness and the representability of the observed JPD by a single graph model), the most likely explanation for such inconsistencies is the small sample size (73 in our application), which cannot exclude several potentially conflicting models. As already argued above, the very large number of variables (around 1000) entails a huge number of such high probability models, which makes their analysis by a human expert impossible. We therefore require the synthesis of a *single representation of these alternative models* (the PAG), which is analysable by an expert.

While some anomalies disappear when using our stronger (non)collider test, the remaining irreducible ones need to be eliminated by removing the conflicting assertions (phase 4).

The remaining assertions, which are now guaranteed to be consistent, are subsequently propagated in phase 5.

Propagation (for example of $\neg(X \ast\!\!\rightarrow Y \wedge Z \ast\!\!\rightarrow Y)$ and $Z \ast\!\!\rightarrow Y$) can produce definite (unary) negative assertions of the form $\neg X \ast\!\!\rightarrow Y$, which can be automatically converted to $X \ast\!\!-\!\!- Y$ (recall that an '>' arrowhead into *Y* means that *Y* is *not* an ancestor of *X*, while an '–' endpoint says that *Y* is an ancestor of *X*). But in the absence of hidden *selection* variables, we cannot have edges with '–' endpoints at both ends, so X $\ast\!\!-\!\!-$ Y could be immediately turned into X <–– Y. Unfortunately, placing new '<' arrowheads may lead to new inconsistencies,[8] for example involving $U \ast\!\!\rightarrow X <\!\!-\!\!- Y$ and the negative assertion (non-collider) $\neg(U \ast\!\!\rightarrow X <\!\!-\!\!\ast Y)$. To make things even more complicated, the arrow $X <\!\!-\!\!- Y$ may propagate another arrow, for example $V \mathbf{<\!\!-\!\!-} X <\!\!-\!\!- Y$ *before* the discovery of the inconsistency with $U \ast\!\!\rightarrow X$.

As all assertions involved in inconsistencies must be eliminated (in the case of the $X <\!\!-\!\!- Y$ arrow, we'll simply remove the arrowhead at *X* obtaining X o–– *Y*), we have to keep track of the inferences (propagations) made from these assertions, in order to enable their removal (as they are based on inconsistent premises).

In our case, removing the arrowhead at *X* in $X <\!\!-\!\!- Y$ will have to invalidate the $V <\!\!-\!\!- X$ arrow as well (of course, only if $V <\!\!-\!\!- X$ has no other "justification").

More generally, we attach a "justification" to each assertion, representing the successive insertions of arrowheads (for avoiding $X -\!\!-\!\!- Y$ edges) that have lead to placing the current arrowhead.

Definition (justification of an assertion). The *justification of a primitive assertion* (i.e. an assertion generated in phase 3 and based on CI tests) is empty. The justification of a *derived assertion* (i.e. an assertion propagated in phase 5) is a set of atomic labels $j = \{l_1, l_2, ..., l_n\}$ representing arrowheads placed for avoiding $X -\!\!-\!\!- Y$ edges.

We use the notation $A : cf :: j$ for an assertion *A* with justification *j* (empty justifications can be omitted).

An ATMS could be used to manage assertions and their justifications. But the *propagation rules* in our domain are very simple due to the very constrained form of assertions (in the following, *a* and *b* stand for edge arrowheads of the form $X \ast\!\!\rightarrow Y$):

$$
\begin{array}{rcl}
a \wedge b : cf & \Rightarrow & a : cf, \quad b : cf \\
a : cf_a :: j_a,\ \neg(a \wedge b) : cf & \Rightarrow & \neg b : cf_a \cdot cf :: j_a \\
\neg(X \ast\!\!\rightarrow Y) : cf :: j & \Rightarrow & X \ast\!\!-\!\!- Y : cf :: j \\
X \ \mathbf{o}\!\!-\!\!- Y : cf :: \boldsymbol{j} & \Rightarrow & Y \mathbf{-\!\!-\!\!>} X : cf :: \boldsymbol{j} \cup \{\boldsymbol{l_i}\}
\end{array}
$$

(with l_i a new atomic label).

An *arrowhead inconsistency* is treated by the rule:

$a :: j_1,\ \neg a :: j_2 \ \Rightarrow$ remove_inconsistency(j_1, j_2)

which deletes all assertions $A :: j$ with justifications containing j_1 or j_2: $j \supseteq j_1$ (but only if $j_1 \neq \emptyset$) or $j \supseteq j_2$ (but only if $j_2 \neq \emptyset$).

Finally, after all inconsistencies have been removed, we aggregate the confidence factors for edge endpoints as follows (since a given edge endpoint can be supported by several assertions with different confidence factors):

$cf(a) = max\{cf_i \mid a : cf_i\}$.

(Assertions with confidence factors below a given threshold, e.g. α=0.05, are automatically discarded.)

Note that the rule that orients edges for avoiding the formation of new colliders (rule R1 in [5], or rule G(ii) in [9]): $X \ast\!\!\rightarrow Y \ast\!\!-\!\!\ast Z \ \Rightarrow \ X \ast\!\!\rightarrow Y -\!\!\rightarrow Z$ is a special case of our constraint propagation of assertions (phase 5).

Also note that we do not apply the acyclicity rule (R2 from [5], or G(i) from [9]), since genetic networks are potentially cyclic. However, dealing with both cycles and latent variables is an open research problem, so we do not aim at completeness in the presence of cycles. (The complexity of the CCD algorithm [7] dealing with cycles but only in the absence latent variables –which is anyway too

[8] of course, only in the case of unreliable CI tests.

restrictive in our setting– suggests the extraordinary difficulty in devising a complete algorithm treating both latent variables and feedback.)

3 Results for the Garber dataset

We have run QFCI on the Garber dataset with a significance level α=0.05 for the conditional independence tests. Only 11 *irreducible* collider anomalies were produced (showing on the other hand that there are at least 3^{11} = 177,147 different alternative high-probability models and a similar number of variable orderings, which would represent a serious problem for model averaging approaches). Moreover, 77 of the 109 anomalies encountered by the simple FCI algorithm were removed by our stronger collider test, without affecting the computational efficiency of FCI (i.e. without exhaustively performing all CI tests for the *separable* pairs of variables – which are the majority).

Since expression levels in microarray measurements are continuous variables, we have chosen to employ CI tests based on the Fisher z transform of partial correlations. (Strictly speaking, these tests are only correct if the variables are jointly normal, but they are still often useful for non-normal distributions as well. The alternative of discretizing the variables seems worse, especially in view of the small sample size N=73, as it would further reduce the power of the tests.)

Running QFCI on the Garber dataset produced very encouraging results from a biological viewpoint. Figure 3 depicts the neighbours (up to depth 5) of the discrete variable associated to the '*small cell*' subtype.[10] It is particularly striking that QFCI finds 'small cell' connected to a single gene of unknown function, INSM1 (insulinoma-associated 1), which is known to serve as a marker for lung tumours of neuroendocrine differentiation. In fact, many of the genes in the neighbourhood of INSM1 show a neuroendocrine differentiation profile, and/or involvement in oncogenic transformation, cell fate specification, proliferation, cellular signaling (e.g. growth factor receptors), etc. All of the genes hand-picked by human experts in Garber et al. in their discussion of SCLC are very close neighbors in our network: 7B2 (SGNE1), glutaminyl cyclase (QPCT), L-myc (Hs.92137) and the neuronal differentiation marker achaete-scute homolog (IMAGE: 1416420), while several others appearing in our network have unknown functions and should be further investigated.

Though very encouraging, the results obtained should be regarded with caution, due to the high experimental noise, the small sample size, the potentially very large number of hidden variables and the impossibility of gathering data from single cells (or at least from collections of cells with very low variance) [10]. Still, they could represent a good starting point for elucidating the details of the genetic networks involved in lung carcinoma by enabling much better targeted experiments.

4 Related work and conclusions

Friedman et al. [3] use a scoring-based approach to deal with the *yeast cell cycle data of Spellman et al.* Their *SparseCandidate* algorithm is essentially a *model selection* approach able to cope with the extremely large search space of structures. But unfortunately, for small sample sizes and thereby many alternative models, model selection makes somewhat arbitrary choices between these alternative models and thus only the highest confidence features can be considered reliable guesses (unfortunately, these make up just a small and sparse fragment of the whole network). This problem is however addressed in [2] in the context of model averaging. Our own experiments with the Spellman dataset showed mixed results, which we believe are due to the influence of a variable (gene) from a sample taken at a particular time-point of the cell cycle on another variable *from a different sample* (taken at a different time-point[11]).

Pe'er et al. [6] infer sub-networks of genes from *perturbed* expression profiles taken from the *Rosetta Compendium for yeast* using a greedy hill-climbing scoring-based approach (with similar problems due to the small sample size). The puzzling discrepancies mentioned in [6] may be due to the heterogeneous nature of the Compendium (which is a compilation of experiments performed in different laboratories), as well as to the inability of dealing with latent (hidden) variables.

The BNPowerConstructor [1] also employs quantitative information related to the independence tests in a constraint-based algorithm. However, extending it to deal with latent variables or small samples seems extremely difficult, if not inherently impossible (with latent variables, his '*try_to_separate_B*' procedure is no longer complete).

Here we have shown that constraint-based methods can be improved to deal with small and noisy samples by a more sophisticated propagation of constraints followed by a careful elimination of the inconsistencies (anomalies) due to unreliable conditional independence tests. The results on the Garber dataset are in line with the current hypothesis of biologists that the 'small cell' subtype corresponds to a neuroendocrine differentiation profile.

References

[1] J. Cheng, D. Bell. Learning Bayesian networks from data: an efficient approach based on information theory. Proc. 6th ACM Int. Conf. on Inform. and Knowledge Management, 1997.

[2] Friedman N., Koller D. Being Bayesian about Network Structure, Machine Learning, 50:95-126, 2003.

[10] Due to space limitations, we only discuss the results obtained for the 'small cell' subtype.

[11] since transcriptional regulation takes non-negligible time. This non-independence of samples may also affect the results of [3].

[3] Friedman N. et al. Using Bayesian networks to analyze expression data. J. of Comp. Biology, 7(3-4):601-620, 2000.
[4] Garber M.E. et al. Diversity of gene expression in adenocarcinoma of the lung. PNAS, Vol. 98, 13784-9, Nov. 20, 2001.
[5] Pearl, J. Causality: Models, Reasoning, and Inference, Cambridge University Press, 2000.
[6] D. Pe'er, A. Regev, G. Elidan, N. Friedman. Inferring subnetworks from perturbed expression profiles. Bioinformatics, 17(S1):S215-24, 2001.
[7] Richardson, T., Spirtes, P. Automated discovery of linear feedback models, Technical Report CMU-75-Phil.
[8] Spirtes P. Directed Cyclic Graphical Representations of Feedback Models. UAI 1995, 491-498.
[9] Spirtes P., Meek C., Richardson T. An algorithm for causal inference in the presence of latent variables and selection bias, in Computation, Causation and Discovery. C. Glymour and G. Cooper. MIT Press, 1999.
[10] Spirtes, P., Glymour, C., and Scheines, R. Constructing Bayesian Network Models of Gene Expression Networks from Microarray Data, Proc. Atl. Symp. Comp. Biology, 2000.

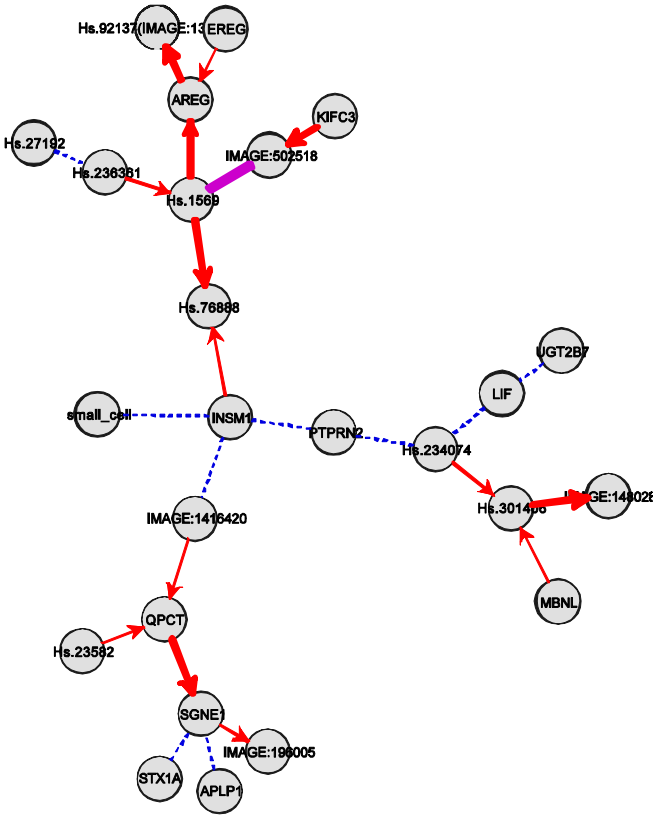


Figure 3: The 'small cell' sub-network. Bold directed edges are of type ––>, simple directed edges o–>, bold undirected edges <–>, and dotted undirected edges o–o.