Network Reconstruction

Ulf Leser
Content

- Network reconstruction
  - Boolean models
  - Correlation-Based Approaches: REVEAL / ARACNE
  - Example
- Quantitative network reconstruction
Networks

How do we know? Network reconstruction
Approaches to Network Reconstruction

- By many, many small-scale experiments
- By mathematical modeling from high-throughput data sets
- By evolutionary inference from model organisms
- By curation from the literature (see first bullet)
Reconstruction from Indirect High-Throughput Data

• Network reconstruction, re-engineering, inference, ...

• Idea: Derive network from indirect observations
  – **Network**: Links and their effect (strength, activation, ...)
    • We usually assume the players (genes, metabolites, ...) to be given
  – **Observation**: High-throughput measurements
    • Here: Transcriptome, microarrays, RNA-Seq
  – **Indirect**: We try to infer physical causality by correlation of expression intensities

• Warning: All current methods are **highly reductionist**
Reconstruction from Indirect High-Throughput Data

- Quantitative time-resolved network inference: Infer intensities of activities over time
  - Very complicated
- **Dynamic networks**: Synchronize time and discretize activity
  - Nodes get one of two states: active / inactive
  - Edges determine how states propagate through the network
  - Propagation proceeds in synchronized steps
  - Current states determine future states of connected nodes
Boolean Networks

• Definition

A Boolean Network is a set of nodes $V$ with

- Every node has an associated Boolean state (on/off)
- Every node is labeled with a Boolean function over the states of nodes

• Visualization

- We map a BN $V$ into a digraph $G=(X,Y)$ by:
  - $X = V$
  - $Y = \{ (v,w) \mid v, w \in X \text{ and } w \text{ is part of the boolean function of node } v \}$
- $G$ has less information than $B$
  - Boolean formulas cannot be derived from $G$

\[ f_A(B) = B \]
\[ f_B(A, C) = A \text{ and } C \]
\[ f_C(A) = \text{not } A \]
Boolean Network for Biology

- Vertices = genes
- Boolean formulas: Interplay of other genes necessary to activate (regulate) a node
- An edge (v,w) visualizes an effect of v on w
- Simplistic: No cofactors, no cellular context, no binding affinity, no time, no kinetics, ...
Static Boolean Networks

- **Definition**

  A *state* of a Boolean Network is a labelling of all nodes with TRUE or FALSE. A state $S$ of a Boolean Network is called **consistent**, when the state of every node equals the value of its boolean function.

- **Remarks**
  - Not very interesting – nothing ever changes
  - Not every BN has a consistent state (e.g. $f_A(B) = B$, $f_B(A) = \text{NOT } A$)

\[
\begin{align*}
  f_A(B) &= \text{not } B \\
  f_B(A,B) &= A \text{ and not } C \\
  f_C(B) &= B
\end{align*}
\]
Network Dynamics

- **Definition**
  A *Dynamic Boolean Network (DBN)* is a Boolean network where every node $v$ is assigned a sequence of states $v_0, v_1, v_2, \ldots$ such that the state of $v_t$ with $t>0$ equals the value of the Boolean function of $v$ applied to the states $w_{t-1}$ of all incoming nodes $w$ of $v$. The initial states at $t=0$ are arbitrary.

- **Remarks**
  - Models the state of every gene over time
  - States at time point $t$ only depend on states at time point $t-1$
    - No buffering, slow/fast reactions ...
  - **Deterministic**: Given all states at a time $t$, any state at any later time point can be uniquely determined
Example

Example: Changes over Time

<table>
<thead>
<tr>
<th>genes time</th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Network Analysis

- Many things can be analyzed using DBN
- For instance, an attractor is a (set of) states towards which a subset of the network states converge
  - Point attractor: State which cannot be left any more
  - Cyclic attractor: A series of states which will repeat forever
  - Every DBN must have at least one attractor, as the number of network states is finite – we must “repeat” after at most $2^{|V|}$ steps
  - Number / shape of attractors depend largely on size of network and complexity of Boolean functions
- However, we want to reconstruct networks
Network Reconstruction

- Assume we know all genes, but not their relationships
- Assume that the states of genes only depend on (the states of) the other genes in the past
- Assume we observe the states of n genes over m time points (a matrix S; the observations)
- Can we re-engineer the Boolean function of every gene given a sequence of states?

<table>
<thead>
<tr>
<th>genes</th>
</tr>
</thead>
<tbody>
<tr>
<td>time</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

S
Example

\[
\begin{align*}
  f_A(B) &= \text{not } B \\
  f_B(A,B) &= A \text{ and not } B \\
  f_C(B) &= B
\end{align*}
\]

\[
\begin{array}{c|ccc}
  \text{genes} & A & B & C \\
  \text{time} & 0 & 1 & 1 & 0 \\
               & 1 & 0 & 0 & 1 \\
               & 2 & 1 & 0 & 0 \\
               & 3 & 1 & 1 & 0 \\
               & 4 & 0 & 0 & 1 \\
               & 5 & \ldots & \ldots & \ldots
\end{array}
\]
Formal Problem

- **Definition**
  
  *Let $S_t$, $0 \leq t \leq m$, be the vector of all observed states of all genes at time point $t$. A DBN $G$ with functions $f_1, \ldots, f_n$, $n = |V|$, is called
  - consistent with $S_t$ iff $S_t = [f_1(S_{t-1}), f_2(S_{t-1}), \ldots f_n(S_{t-1})]$
  - consistent with $S$ iff it is consistent for all $S_t$, $1 \leq t \leq m$*

- **The Boolean network reconstruction problem**
  *Given an observation $S$ over a set $V$, find a DBN $G$ that is consistent with $S$.*

- **Remark**
  - Reconstruction means finding the functions $f_1, \ldots, f_n$
Solutions

- Clearly, there are many observations $S$ for which no consistent $G$ exists
  - Recall that DBN are deterministic
  - Imagine $S_t$, $S_{t+1}$ and $S_u$, $S_{u+1}$ with $S_t = S_u$ but $S_{t+1} \neq S_{u+1}$
- Also, there are many observation $S$ for which more than one consistent $G$ exists
- Every time point narrows the options for $G$ – the longer $S$, the (monotonically) less consistent $G$’s exist
Optimal Networks

- Definition
  - For a DBN $G$, let $\text{size}(G)$ be the total number of variables (edges) appearing in the Boolean functions of $G$
  - A DBN $G$ is minimal for observation $S$, if $G$ is consistent with $S$ and there is no $G'$ which is also consistent with $S$ and $\text{size}(G') < \text{size}(G)$

- Remark
  - Parsimony assumption: Small models are better
  - Thus, the smallest network is the best – functions are as simple as possible, nothing is inferred that is not enforced by the data
  - Not necessarily unique
Naïve Algorithm

- Exhaustive naïve algorithm for finding minimal networks
- **Very complex** (AND, OR, NOT, no paranthesis)
  - k=1: 2n functions
  - k=2: 2*2n*2n=O(n²) functions
  - ...
  - General: $O(2^{2k-1}n^k)$ functions

```plaintext
N = V;
for k = 1...|V|  # length of functions
    for every n in N  # all unexplained nodes
        test all functions f of size k for n on S;
        if f is consistent for n on S
            N := N \ n;  # n is explained
            Add f to network;
        end if;
    end for;
end for;
```
Pros and Cons

• Application (transcriptome data)
  – Perform time-series gene expression experiments
  – Brutally discretize each measurement: Genes are on or off
  – Reconstruct DBN

• Pros: Simple

• Cons
  – Binary values are not capturing reality
  – Nature has no synchronized time or reactions
  – No quantification ("it needs 2*A and one B to regulate C")
  – Only small networks are solvable
  – No unique solutions
  – ...
Content

• Network reconstruction
  – Boolean models
  – Correlation-Based Approaches: REVEAL / ARACNE
  – Example

• Quantitative network analysis
Towards Reality

- There are less complex & more robust algorithms
- REVEAL replaces Boolean functions by mutual information; correlations rather than deterministic switching
- ARACNE is even simpler: Build correlation network and removal some (presumably indirect) correlations
Foundations

- Definition

Let $X, Y$ be two discrete random variables. The **mutual information** $MI(X,Y)$ is defined as

$$MI(X,Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \ast \log \left( \frac{p(x,y)}{p(x) \ast p(y)} \right)$$

- Remark
  - Measures the variable’s mutual dependency
    - Deviation of **observation** $(p(x,y))$ from **expectation** in case of independence $(p(x) \ast p(y))$
    - How much does $x$ determine the state of $y$ (and vice versa)?
    - How helpful is it to know $x$ to know $y$ (and vice versa)?

- Similar measures: Information gain, Pearson correlation, conditional entropy, ...
  - Note: Many are asymmetric
Example

\[ MI(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \cdot \log \left( \frac{p(x, y)}{p(x) \cdot p(y)} \right) \]

<table>
<thead>
<tr>
<th>p(x, y)</th>
<th>( y=0 )</th>
<th>( y=1 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>x=0; ( p(x=0)=0.2 )</td>
<td>0.12</td>
<td>0.08</td>
</tr>
<tr>
<td>x=1; ( p(x=1)=0.8 )</td>
<td>0.48</td>
<td>0.32</td>
</tr>
</tbody>
</table>

\[ MI(X, Y) = 0 \]

\[ MI(X, Y) = 0.24 \]
Two more Facts

- With a little math, we find
  \[ MI(X,Y) = H(X) - H(X|Y) = H(Y) - H(Y|X) \]
  - \( H(X) \): Entropy of \( X \)
  - \( H(X|Y) \): Conditional entropy of \( X \) given \( Y \)
- It follows: \( MI(X,Y) < \min(H(X),H(Y)) \)
  - In case of \( H(X|Y)=0 \) or \( H(Y|X)=0 \), which means that \( X \) (or \( Y \)) completely determines \( Y \) (or \( X \))
  - This defines a maximal value for \( MI(X,Y) \)
- \( MI \) can be extended to sets of three, four, \ldots \ variables
  - Like Boolean functions over three, four, \ldots \ variables
  - Multivariate mutual information
Application

- Assume $m$ observation of $n$ genes
  - Can be $m$ time points, $m$ conditions, $m$ samples, $m$ treatments ...
  - REVEAL has no notion of time
- Discretize expression values to 0 or 1 (again)
- Compute for each gene $X$ $p(X=0)$ and $p(X=1)$; the fraction of observations in which $X$ was 0 / 1
  - Compute for each pair $X,Y$ the probabilities $p(X=0, Y=0)$, ...
  - Compute for each triple $X,Y,Z$ the probabilities ...
  - ...
- Task: Find network such that every node $X$ has the minimal number of incoming edges with maximal mutual information
  - Minimal number of other variables offering maximal explanation
REVEAL Algorithm

\[
N = V;
\text{for } k=1 \ldots |V| \quad \# \text{number of nodes/variables}
\text{    for every } X \text{ in } N \quad \# \text{all unexplained nodes}
\text{        find subset } T=(Y_1, \ldots, Y_k) \text{ with } MI(X,Y_1,\ldots,Y_k) = H(X);
\text{        if } T \text{ exists}
\text{            } N := N \setminus X; \quad \# \text{n is explained}
\text{    end for;}
\text{end for;}
\]

- Very strict: \( Y_1, \ldots, Y_k \) must \textbf{maximally explain} \( X \)
  - Unrealistic – noise, neglected effects, ...
  - Still very high complexity (“all subsets…”)

- Practical modifications
  - Only require \(|MI(X,Y_1,\ldots,Y_k) - H(X)| < \varepsilon\)
  - Set a \textbf{maximal} \( k \) and find best explanation with \( \leq k \) edges
ARACNE

- Fast variation of REVEAL which (a) considers each pair in isolation and (b) gives up model minimality
- Idea
  - Compute mutual information between all pairs of genes
    - This gives a complete network
  - Remove edges where $|\text{MI}(X,Y)-H(X)| > \varepsilon$
    - $\varepsilon$ can be estimated from the distribution of MI – created at random?
    - Do not consider composite effects – all $Y$ in isolation
  - Remove certain indirect effects (“data processing inequalities”)
Data Processing Inequalities

Imagine with strong effects of A on B and B on C; will appear as

But if we find which edge most likely is an artifact?

- Assumption: If $\text{MI}(X,Z) \leq \min(\text{MI}(X,Y),\text{MI}(Y,Z))$, then the correlation between $X$-$Z$ is an indirect effect and removed
- Procedural: In every triangle, remove the smallest edge
  - But in which order should triangles be visited?
Content

• Network reconstruction
  – Boolean models
  – Correlation-Based Approaches: REVEAL/ ARACNE
  – Example
• Quantitative network analysis
Reconstructing the Mammalian Clock

- DA Sven Lund, 2015
- Data
  - ~630 rather unspecific arrays from GEO
  - Compared to two time-resolved clock-specific experiments
- Reconstruction quality of three algorithms
  - Aracne, Bayes Networks, Time-Delay Aracne
Results

- Filtering of ARACNE reduces recall a lot, while precision increases only marginally.
- Data set size outweighs specificity – reconstruction about as good using many untargeted arrays or using fewer targeted arrays.

<table>
<thead>
<tr>
<th>Kennzahl</th>
<th>Verfahren</th>
<th>TP</th>
<th>TN</th>
<th>FP</th>
<th>FN</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bar{x}$</td>
<td>Pearson</td>
<td>53.75</td>
<td>20.00</td>
<td>41.00</td>
<td>21.25</td>
<td>0.72</td>
<td>0.57</td>
</tr>
<tr>
<td>$s$</td>
<td>Pearson</td>
<td>4.979</td>
<td>8.718</td>
<td>8.718</td>
<td>4.979</td>
<td>0.068</td>
<td>0.070</td>
</tr>
<tr>
<td>$\bar{x}$</td>
<td>Bayes</td>
<td>36.00</td>
<td>33.50</td>
<td>27.50</td>
<td>39.00</td>
<td>0.48</td>
<td>0.57</td>
</tr>
<tr>
<td>$s$</td>
<td>Bayes</td>
<td>12.739</td>
<td>10.282</td>
<td>12.739</td>
<td>10.282</td>
<td>0.170</td>
<td>0.020</td>
</tr>
<tr>
<td>$\bar{x}$</td>
<td>ARACNE</td>
<td>18.88</td>
<td>48.00</td>
<td>13.00</td>
<td>56.13</td>
<td>0.25</td>
<td>0.59</td>
</tr>
<tr>
<td>$s$</td>
<td>ARACNE</td>
<td>5.515</td>
<td>3.47</td>
<td>3.47</td>
<td>3.47</td>
<td>0.017</td>
<td>0.017</td>
</tr>
</tbody>
</table>

Averages over all data sets

<table>
<thead>
<tr>
<th>Kennzahl</th>
<th>Datenquelle</th>
<th>TP</th>
<th>TN</th>
<th>FP</th>
<th>FN</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\bar{x}$</td>
<td>GEO</td>
<td>45.00</td>
<td>26.00</td>
<td>35.00</td>
<td>30.00</td>
<td>0.60</td>
<td>0.57</td>
</tr>
<tr>
<td>$s$</td>
<td>GEO</td>
<td>17.550</td>
<td>16.480</td>
<td>16.480</td>
<td>17.550</td>
<td>0.235</td>
<td>0.034</td>
</tr>
<tr>
<td>$\bar{x}$</td>
<td>Korenčič</td>
<td>35.67</td>
<td>36.22</td>
<td>24.78</td>
<td>39.33</td>
<td>0.48</td>
<td>0.60</td>
</tr>
<tr>
<td>$s$</td>
<td>Korenčič</td>
<td>16.462</td>
<td>12.940</td>
<td>12.940</td>
<td>16.462</td>
<td>0.219</td>
<td>0.037</td>
</tr>
<tr>
<td>$\bar{x}$</td>
<td>Hogenesch</td>
<td>30.89</td>
<td>36.67</td>
<td>24.33</td>
<td>44.11</td>
<td>0.41</td>
<td>0.55</td>
</tr>
<tr>
<td>$s$</td>
<td>Hogenesch</td>
<td>15.648</td>
<td>12.708</td>
<td>12.708</td>
<td>15.648</td>
<td>0.208</td>
<td>0.094</td>
</tr>
</tbody>
</table>
Content

- Network reconstruction
- Quantitative network reconstruction
Networks as Equations

- REVEAL / ARACNE infer relationships based on correlation
- Alternative: Describe states as sets of (linear) equations
  - No discretization
  - Extensibility: Incorporate different types of experiments (“multi-omics” – proteome, binding, epigenetic status, …)
  - Still many limitations: Synchronized time, no kinetics
- We look at one simple approach in between reconstruction and analysis (Schacht et al., 2014)
  - Differentiates between regulators (transcription factors) and regulated entities (genes)
  - Goal: Rank transcription factors by effect strength
    - Which are the most important TFs in this data set?
    - This involves estimating the impact of TF on genes
Approach

- Assume a network $G=(V,E)$, where $V$ consists of a set of transcription factors $T$ and a set of genes $G$
  - Transcription factors regulate genes, but not vice versa
    - We ignore that a TF may regulate TFs (even including itself)
    - Each gene $g$ is regulated by all TFs
      - For efficiency, we can also assume this set to be constrained – “potential regulators”
- Measurements: $m$ observations for $n$ nodes (genes / TFs)
- We model the expression values of all genes as linear combinations of the expression values of its regulating TFs

$$g_{i,s} = \beta_0 + \sum_{t=1}^{\left|T\right|} \beta_t \ast \delta_{t,i} \ast e_{t,s}$$
\[ g_{i,s} = \beta_0 + \sum_{t=1}^{\vert T \vert} \beta_t \cdot \delta_{t,i} \cdot e_{t,s} \]

- \( g_{i,s} \): Expression of gene \( i \) in observation \( s \)
- \( \beta_0 \): Fixed additive offset
- \( \beta_t \): Global activity parameter for transcription factor \( t \)
  - Independent of observation and gene
- \( \delta_{t,i} \): Affinity of TF \( t \) to gene \( I \)
  - E.g. Binding strength to promoter
- \( e_{t,s} \): Expression of TF \( t \) in observation \( s \)
Optimization

- Typically, these (large) systems cannot be solved exactly
- Instead, minimize the error

\[
| g_{i,s} - \left( \beta_0 + \sum_{t=1}^{\vert T \vert} \beta_t \cdot \delta_{t,i} \cdot e_{t,s} \right) | \Rightarrow \min
\]

- ... under a set of constraints
- Several solvers available
Comparison (Trescher & Leser, 2018)

- Comparison of different tools shows very little agreement
- Research question essentially open – which method is best? How can we infer regulatory activity?
Many Other Models

- **Stoichiometric networks**
  - Model the turnover of molecules
    - Especially metabolism
  - Needs to consider enzymatic effects
  - What will a network produce given a certain input?
  - Is a network in flux balance?

- **Kinetic networks**
  - Takes into account reaction rates: How many in what time
    - No linear relationship
  - Leads to systems of differential equations
  - Can predict system behavior in time under realistic assumptions
Further Reading