Robust Signal-Structure Reconstruction

Vasu Chetty, David Hayden, Jorge Goncalves, and Sean Warnick.

CDC 2013
Florence, Italy
December 11, 2013
What is this talk about?

- **Key Question**: How can the network reconstruction process be improved?
What is this talk about?

• **Key Question**: How can the network reconstruction process be improved?

• **Outline**
  – Background
    • System Structure
    • Dynamical Structure Functions
    • Signal Structure
    • Network Reconstruction
  – Results
    • Non-target specific networks
    • Faster network reconstruction algorithm
    • Improved model selection procedure
System Structure

- Two common system representations:
  - Transfer function: represented graphically using the interconnection pattern representing closed-loop paths from inputs to outputs.
  - State space realization: represented graphically using the complete computational structure representing the physical interconnection of states.
• FACT: Without other system assumptions, only the transfer function structure can be recovered from (sufficiently exciting) input-output data.
Network realization is the process of determining a system’s complete computational structure given data. Note that the realization process is ill-posed, even when looking for a minimal realization.
Dynamical Structure Functions

I/O Methods
- e.g. correlation, mutual information
- Weak characterization of system structure

Dynamical Structure Functions
- Structurally more informative than TF
- Better identifiability conditions than SS

State Space Methods
- e.g. some LMI and Bayesian techniques
- Strong assumptions for reconstruction
Dynamical Structure Functions

**Data**  
- Identification

**Models**  
- Transfer Function

**Transfer Function**  
- $G(A,B,C,D)$

**Identification**  
- $Dynamical\ Structure\ Function\ (Q,P)$

**Dynamical Structure Functions**

- $Y = QY + PU$
- $G = (I-Q)^{-1}P$

**I/O Methods**

- e.g. correlation, mutual information
- Weak characterization of system structure

**State Space Methods**

- e.g. some LMI and Bayesian techniques
- Strong assumptions for reconstruction
Signal-structure is the graphical representation of a system’s dynamical structure function (DSF).
Signal-structure is the graphical representation of a system’s dynamical structure function (DSF).

Network reconstruction is the process of determining a system’s signal-structure given data and, as with the network realization process, the network reconstruction process is ill-posed.
Signal-Structure

State Space Realization

\[
\begin{align*}
\dot{x}_1 &= 1 0 1 0 & x_1 &= 0 0 \\
\dot{x}_2 &= 0 2 1 1 & x_2 &= 0 0 & u_1 \\
\dot{x}_3 &= 1 0 3 0 & x_3 &= 1 0 & u_2 \\
\dot{x}_4 &= 0 0 0 3 & x_4 &= 0 1 \\
y_1 &= 1 0 0 0 & x_2 \\
y_2 &= 0 1 0 0 & x_3 \\
\end{align*}
\]

Complete Computational Structure

![Complete Computational Structure Diagram]

\[y_1 \rightarrow x_3 \rightarrow y_2 \rightarrow x_4 \rightarrow u_2\]
Signal-Structure

State Space Realization

\[
\begin{align*}
\dot{x}_1 &= 1 \ 0 \ 1 \ 0 \quad x_1 \quad 0 \ 0 \\
\dot{x}_2 &= 0 \ 2 \ 1 \ 1 \quad x_2 + 0 \ 0 \quad u_1 \\
\dot{x}_3 &= 1 \ 0 \ 3 \ 0 \quad x_3 \quad 1 \ 0 \quad u_2 \\
\dot{x}_4 &= 0 \ 0 \ 0 \ 3 \quad x_4 \quad 0 \ 1 \\
y_1 &= 1 \ 0 \ 0 \ 0 \quad x_2 \\
y_2 &= 0 \ 1 \ 0 \ 0 \quad x_3 \\
\end{align*}
\]

Complete Computational Structure

\[
\begin{align*}
\begin{bmatrix} y_1 \\ y_2 \\ x_3 \\ x_4 \end{bmatrix} &= \begin{bmatrix} 0 & 0 \\ 1 & (s+2)(s+3) \\ 1 & (s+2)(s+3) \\ 1 & (s+2)(s+3) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \\
\end{align*}
\]

Dynamical Structure Function

\[
\begin{align*}
\begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} &= \begin{bmatrix} 0 & 0 \\ \frac{1}{(s+2)(s+3)} & 0 \end{bmatrix} \begin{bmatrix} Y_1 \\ Y_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{s^2+4s+2} \\ \frac{1}{(s+2)(s+3)} \\ \frac{1}{(s+2)(s+3)} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \\
\end{align*}
\]
Signal-Structure

Complete Computational Structure

Signal-Structure

\[
\begin{align*}
\frac{1}{s^2 + 4s + 2} & \quad \frac{1}{(s+2)(s+3)} \\
\frac{1}{(s+2)(s+3)} & \quad \frac{1}{(s+2)(s+3)}
\end{align*}
\]
Signal-Structure

Graphical Dual of Signal Structure

Complete Computational Structure

Signal-Structure

\[
\frac{1}{s^2 + 4s + 2} \quad \frac{1}{(s+2)(s+3)}
\]

\[
\frac{1}{(s+2)(s+3)} \quad \frac{1}{(s+2)(s+3)}
\]

\[
1 \quad 1
\]

\[
(s + 2)(s + 3) \quad (s + 2)(s + 3)
\]

\[
1 
\]

\[
(s + 2)(s + 3) 
\]

\[
\frac{1}{s^2 + 4s + 2} 
\]

\[
\frac{1}{(s+2)(s+3)} 
\]

\[
\frac{1}{(s+2)(s+3)} 
\]
Signal-Structure

Graphical Dual of Signal Structure

Complete Computational Structure

Signal-structure is not the graphical dual of subsystem structure.
Network Reconstruction


\[(I - Q)^{-1} P = G\]

\[\begin{bmatrix} I & G^T \end{bmatrix} \begin{bmatrix} P^T \\ Q^T \end{bmatrix} = G^T\]

\[\begin{bmatrix} I & 0 & 0 & G_{-1}^T & 0 & 0 \\ 0 & \ddots & 0 & 0 & \ddots & 0 \\ 0 & 0 & I & 0 & 0 & G_{-p}^T \\ & & & & \mathbf{L} & \end{bmatrix} \begin{bmatrix} \vec{p} \\ \vec{q} \\ z \end{bmatrix} = \vec{g}\]
Network Reconstruction

\[(I - Q)^{-1} P = G\]

\[
\begin{bmatrix}
I & G^T
\end{bmatrix}
\begin{bmatrix}
P^T \\
Q^T
\end{bmatrix} = G^T
\]

\[pm + (p^2 - p)\]

\[p - \text{number of outputs}\]

\[m - \text{number of inputs}\]
Network Reconstruction

\[ z = Tx \]

\[ pm + (p^2 - p) \leq pm \]

\[
\begin{pmatrix}
I & 0 & 0 & G^T_{-1} & 0 & 0 \\
0 & \ddots & 0 & 0 & \ddots & 0 \\
0 & 0 & I & 0 & 0 & G^T_{-p} \\
\end{pmatrix}
\begin{pmatrix}
\vec{p} \\
\vec{q} \\
z
\end{pmatrix}
= \tilde{\vec{g}}
\]
Network Reconstruction

\[ z = Tx \]

\[ [LT]x = \hat{g} \]

Theorem: The dynamical structure function of a system characterized by transfer function \( G \) can be reconstructed if and only if:

1. The matrix \( LT \) is injective
2. \( \hat{g} \) is in the range of \( LT \)
Network Reconstruction

\[ z = Tx \]

\[ [LT]x = \tilde{g} \]

- Target specificity means the \( P \) matrix is diagonal, which is a sufficient condition for the matrix \([LT]\) to be injective.
What is this talk about?

• **Key Question**: How can the network reconstruction process be improved?

• **Outline**
  – **Background**
    • System Structure
    • Dynamical Structure Functions
    • Signal Structure
    • Network Reconstruction

  – **Results**
    • Non-target specific networks
    • Faster network reconstruction algorithm
    • Improved model selection procedure

\[
LT x = g
\]
Robust Network Reconstruction

• When noise is present in a system, the non-target specific reconstruction method can fail.
Robust Network Reconstruction

• When noise is present in a system, the non-target specific reconstruction method can fail.

• Iterate the network reconstruction process over a candidate set of Boolean structures and use a metric to determine structural distance for each structure.
Robust Network Reconstruction

• When noise is present in a system, the non-target specific reconstruction method can fail.

• Iterate the network reconstruction process over a candidate set of Boolean structures and use a metric to determine structural distance for each structure.

• Fully-connected networks have more parameters to better fit the data, so an information criterion can be used to penalize model complexity.
Robust Network Reconstruction

• When noise is present in a system, the non-target specific reconstruction method can fail.

• Iterate the network reconstruction process over a candidate set of Boolean structures and use a metric to determine structural distance for each structure.

• Fully-connected networks have more parameters to better fit the data, so an information criterion can be used to penalize model complexity.
Systems with Target Specificity

• Noise model with target specificity:

Y. Yuan, G.-B. Stan, S. Warnick, and J. Goncalves, “Robust dynamical network structure reconstruction,” Automatica, Special Issue on Systems Biology, vol. 47, no. 6, pp. 1230 – 1235, 2011
Systems with Target Specificity

• Noise model with target specificity:

\[ \Delta = GX - I \]

where \( X = P^{-1}(I - Q) \).
Systems with Target Specificity

• Noise model with target specificity:

$$\Delta = GX - I$$

where $X = P^{-1}(I - Q)$.

• Non-diagonal zero elements of $X$ correspond to those in $Q$ since $X_{ij} = P_{ii}^{-1}Q_{ij}$ for $i \neq j$. 
Systems with Target Specificity

• Noise model with target specificity:

\[ \Delta = GX - I \]

where \( X = P^{-1}(I - Q) \).

• Non-diagonal zero elements of \( X \) correspond to those in \( Q \) since \( X_{ij} = P_{ii}^{-1}Q_{ij} \) for \( i \neq j \).

• For non-target specific networks, \( P \) may not be invertible. So, we require a different noise model.
Our First Contribution

• Noise model without target specificity:
Our First Contribution

• Noise model without target specificity:

\[ \Delta = Y - \begin{bmatrix} Q & P \end{bmatrix} \begin{bmatrix} Y \\ U \end{bmatrix} \]

• Then, the structural distance for a given Boolean structure is:

\[ \delta = \| Y - \begin{bmatrix} Q & P \end{bmatrix} \begin{bmatrix} Y \\ U \end{bmatrix} \| \]

assuming \( U \) is diagonal, meaning that each input is perturbed individually when collecting data.
Our First Contribution

• We can rewrite the structural distance for a given Boolean structure as:

\[ \delta = \|w - Mx\|_2 \]

where \( w = \text{vec}(Y) \), \( x = \text{vec}([Q \ P]) \), and \( M = \)

\[
\begin{pmatrix}
\text{vec}(Y)' & \cdots & 0 & \text{vec}(U)' & \cdots & 0 \\
\vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\
0 & \cdots & \text{vec}(Y)' & 0 & \cdots & \text{vec}(U)' \\
\end{pmatrix}
\]

• \( \text{vec}(X) \) is a column vector of the transpose of the rows of the matrix \( X \) stacked on each other.
• \( x' \) is the conjugate transpose of \( x \).
Our First Contribution: Example
Our First Contribution: Example
Robust Network Reconstruction

• When noise is present in a system, the non-target specific reconstruction method can fail.

• Iterate the network reconstruction process over a candidate set of Boolean structures and use a metric to determine structural distance for each structure.

• Fully-connected networks have more parameters to better fit the data, so an information criterion can be used to penalize model complexity.
Robust Network Reconstruction

• Each Boolean structure can be represented by a vector the length of the unknowns, with 1s if the edge exists and 0s if there is no edge.
Robust Network Reconstruction

- Each Boolean structure can be represented by a vector the length of the unknowns, with 1s if the edge exists and 0s if there is no edge.
Robust Network Reconstruction

- Without noise

- With noise
Robust Network Reconstruction

IMPORTANT: All subsets of the correct structure have zero score when no noise is present.

• Without noise

![Diagram showing structural "distance" and subsets (0,0), (0,1), (1,0), (1,1)].

- Correct structure with zero structural distance
- Full structure "overfitting"
Robust Network Reconstruction

- Without noise

  structural "distance"
  
  \( (0) \)
  \( (0) \)
  \( (1) \)
  \( (1) \)

  correct structure
  zero structural distance
  full structure "overfitting"

- With noise

  structural "distance"
  
  \( (0) \)
  \( (0) \)
  \( (1) \)
  \( (1) \)

  correct structure
  full structure smallest structural distance
Possible Boolean Structures

• Brute force: look at all possible structures. Time Complexity: $O(2^n)$.

Y. Yuan, G.-B. Stan, S. Warnick, and J. Goncalves, “Robust dynamical network structure reconstruction,” Automatica, Special Issue on Systems Biology, vol. 47, no. 6, pp. 1230 – 1235, 2011
Possible Boolean Structures

• Brute force: look at all possible structures. Time Complexity: $O(2^n)$.

Two node network: $(0)_0$ $(0)_1$ $(1)_0$ $(1)_1$
Possible Boolean Structures

- Brute force: look at all possible structures.
  Time Complexity: \( O(2^n) \).

Two node network:

\[
\begin{pmatrix}
0 \\
0
\end{pmatrix} \quad \begin{pmatrix}
0 \\
1
\end{pmatrix} \quad \begin{pmatrix}
1 \\
0
\end{pmatrix} \quad \begin{pmatrix}
1 \\
1
\end{pmatrix}
\]

Three node network:

\[
\begin{pmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{pmatrix} \quad \begin{pmatrix}
0 & 1 & 0 \\
0 & 0 & 1 \\
0 & 0 & 0
\end{pmatrix} \quad \begin{pmatrix}
0 & 0 & 0 \\
0 & 1 & 1 \\
0 & 0 & 0
\end{pmatrix} \quad \cdots \quad \begin{pmatrix}
1 \\
1 \\
1
\end{pmatrix}
\]
Possible Boolean Structures

- Brute force: look at all possible structures.
  Time Complexity: $O(2^n)$.

Two node network:

\[
\begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{pmatrix}
\quad \begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{pmatrix}
\quad \begin{pmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1
\end{pmatrix}
\]

Three node network:

\[
\begin{pmatrix}
0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1
\end{pmatrix}
\quad \ldots
\quad \begin{pmatrix}
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1 \\
1
\end{pmatrix}
\]

Combinatoric growth means this method does not scale well.
Possible Boolean Structures

• Polynomial Algorithm: looks at all Boolean structures with 1 link and uses metric to find which one is in the actual system. Then looks at all structures with 2 links, keeping the 1\textsuperscript{st} link, and so on. Time Complexity: $O(n^2)$.

Possible Boolean Structures

• Polynomial Algorithm: looks at all Boolean structures with 1 link and uses metric to find which one is in the actual system. Then looks at all structures with 2 links, keeping the 1st link, and so on. Time Complexity: $O(n^2)$.

First iteration:

\[
\begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix}
\]
Possible Boolean Structures

• Polynomial Algorithm: looks at all Boolean structures with 1 link and uses metric to find which one is in the actual system. Then looks at all structures with 2 links, keeping the 1st link, and so on. Time Complexity: $O(n^2)$.

First iteration:

\[
\begin{pmatrix}
1 \\
0 \\
0 \\
0
\end{pmatrix}
\begin{pmatrix}
0 \\
1 \\
0 \\
0
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
1 \\
0
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
0 \\
1
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
0 \\
0
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
0 \\
0
\end{pmatrix}
\]

WLOG, assume this Boolean structure has the smallest structural distance.
Possible Boolean Structures

- Polynomial Algorithm: looks at all Boolean structures with 1 link and uses metric to find which one is in the actual system. Then looks at all structures with 2 links, keeping the 1st link, and so on. Time Complexity: $O(n^2)$.

First iteration:

Second iteration:

WLOG, assume this Boolean structure has the smallest structural distance.
Possible Boolean Structures

- Polynomial Algorithm: looks at all Boolean structures with 1 link and uses metric to find which one is in the actual system. Then looks at all structures with 2 links, keeping the 1st link, and so on. Time Complexity: $O(n^2)$.

First iteration:

Second iteration:

WLOG, assume this Boolean structure has the smallest structural distance.

Continue until you reach the fully-connected network.

and so on...
Possible Boolean Structures

- Polynomial Algorithm: looks at all Boolean structures with 1 link and uses metric to find which one is in the actual system. Then looks at all structures with 2 links, keeping the 1\textsuperscript{st} link, and so on. Time Complexity: $O(n^2)$.

First iteration:

\[
\begin{pmatrix}
1 \\
0 \\
0 \\
0 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
1 \\
0 \\
0 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
0 \\
1 \\
0 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
0 \\
0 \\
1 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
1
\end{pmatrix}
\]

Second iteration:

\[
\begin{pmatrix}
1 \\
0 \\
0 \\
0 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
1 \\
0 \\
0 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
0 \\
1 \\
0 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
0 \\
0 \\
1 \\
0
\end{pmatrix},
\begin{pmatrix}
0 \\
0 \\
0 \\
0 \\
1
\end{pmatrix}
\]

WLOG, assume this Boolean structure has the smallest structural distance.

The Boolean structure with the lowest structural distance for each iteration (+ zero structure) is added to the candidate set.

and so on...
Our Second Contribution

As with the polynomial algorithm, look at all Boolean structures with 1 link. This time, order them using some metric and then add additional links to this network using the order given. Use some stopping criterion to determine when the network is reconstructed. Time Complexity: $O(n)$.

First iteration:

\[
\begin{pmatrix}
1 \\
0 \\
0 \\
0 \\
0 \\
\end{pmatrix}
\begin{pmatrix}
0 \\
1 \\
0 \\
0 \\
0 \\
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
1 \\
0 \\
0 \\
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
1 \\
1 \\
0 \\
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
1 \\
0 \\
1 \\
\end{pmatrix}
\begin{pmatrix}
0 \\
0 \\
1 \\
0 \\
1 \\
\end{pmatrix}
\]
Our Second Contribution

- As with the polynomial algorithm, look at all Boolean structures with 1 link. This time, order them using some metric and then add additional links to this network using the order given. Use some stopping criterion to determine when the network is reconstructed. Time Complexity: $O(n)$.

First iteration:

\[
\begin{pmatrix}
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 
\end{pmatrix}
\]

WLOG, assume above order for lowest structural distance from left to right.
Our Second Contribution

As with the polynomial algorithm, look at all Boolean structures with 1 link. This time, order them using some metric and then add additional links to this network using the order given. Use some stopping criterion to determine when the network is reconstructed. Time Complexity: $O(n)$.

First iteration:

\[
\begin{pmatrix}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
\end{pmatrix}
\]

Second iteration:

\[
\begin{pmatrix}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 1 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 \\
0 & 0 & 1 & 1 & 1 & 1 \\
\end{pmatrix}
\]

WLOG, assume above order for lowest structural distance from left to right.
Our Second Contribution

- As with the polynomial algorithm, look at all Boolean structures with 1 link. This time, order them using some metric and then add additional links to this network using the order given. Use some stopping criterion to determine when the network is reconstructed. Time Complexity: $O(n)$.

First iteration:

\[
\begin{pmatrix}
0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{pmatrix}
\]

Second iteration:

\[
\begin{pmatrix}
0 & 0 & 0 & 0 & 1 & 1 \\
0 & 1 & 1 & 1 & 1 & 1 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 1 & 1 & 1 \\
1 & 1 & 1 & 1 & 1 & 1 \\
\end{pmatrix}
\]

WLOG, assume above order for lowest structural distance from left to right.

No more iterations required. The above set (+ zero structure) is our candidate set.
Our Second Contribution: Example
Our Second Contribution: Example

Average % Solved

- Polynomial Algorithm, $O(n^2)$
- Our Algorithm, $O(n)$

Noise Variance

%
Our Second Contribution: Example
Robust Network Reconstruction

• When noise is present in a system, the non-target specific reconstruction method can fail.
• Iterate the network reconstruction process over a candidate set of Boolean structures and use a metric to determine structural distance for each structure.
• Fully-connected networks have more parameters to better fit the data, so an information criterion can be used to penalize model complexity.
Model Selection Procedure

• Original Akaike’s penalized overfitting, but still heavily favored the fully connected network in the presence of noise.

$$AIC = 2k + 2ln(\delta)$$

where $\delta$ is the value of the likelihood function we are trying to minimize and $k$ is the number of parameters in the mode.
Our Third Contribution

- Customized algorithm penalizes overfitting more heavily for better results.

$$AIC_{custom} = \frac{\delta}{n} + k$$

where $\delta$ is the value of the likelihood function we are trying to minimize, $k$ is the number of parameters in the model and $n$ is total number of possible links in the network.
Our Third Contribution: Example
Our Third Contribution: Example

Average % Solved

- Red: Original AIC
- Blue: Customized AIC

Noise Variance

%
What is this talk about?

• **Key Question**: How can the network reconstruction process be improved?

• **Outline**
  – Background
    - System Structure
    - Dynamical Structure Functions
    - Signal Structure
    - Network Reconstruction
    - Transfer Function and State Space
    - Intermediate representation
    - Graphical representation of DSF
    - Solve system of equations: $[LT]x = \tilde{\mathbf{y}}$

  – Results
    - Non-target specific networks
    - Increased applicability
    - Faster network reconstruction algorithm
    - Improved speed
    - Improved model selection procedure
    - Improved accuracy