Pinning Control and Robust Controllability of Complex Networks: A Machine Learning Approach

Guanrong CHEN
City University of Hong Kong

Xiaofan WANG, 汪小帆 Shanghai University
Lin WANG, 王琳 Shanghai Jiao Tong University
Yang LOU, 楼洋 City University of Hong Kong
Contents

• Pinning Control of Complex Networks
• Network Controllability
• Robustness of Controllability
• Machine Learning Approach
Motivational Examples
The worm *Caenorhabditis elegans* has 297 nerve cells. The neurons switch one another on or off, and, making 2345 connections among themselves. They form a network that stretches through the nematode’s millimeter-long body.

“How many neurons would you have to commandeer to control the network with complete precision?”

The answer is, on average: 49


Here, control = stimuli
Example 2:

“… very few individuals (approximately 5%) within honeybee swarms can guide the group to a new nest site.”


These 5% of bees can be considered as “controlling” or “controlled” agents

Leader-Followers network
Given a network of dynamical systems (e.g., ODEs)

Given a specific control objective (e.g., synchronization)

Assume: a certain class of controllers (e.g., local state-feedback controllers) are chosen to use

\[
\frac{dx_i}{dt} = f(x_i), \quad x_i \in \mathbb{R}^n
\]

\[
u_i = -\Gamma x_i
\]
Control Problem

Pining Control:

- How many controllers to use?
- Where to “pin” them?

\[ \frac{dx_i}{dt} = f(x_i), \quad x_i \in \mathbb{R}^n \]

\[ u_i = -\Gamma x_i \]

Pinning Control: Our Research Progress


Network Model

Linearly coupled network:

\[
\dot{x}_i = f(x_i) + c \sum_{j=1}^{N} \beta_{ij} H x_j \quad x_i \in R^n \quad i = 1,2,\ldots,N
\]

- General assumption: \( f(.) \) is Lipschitz. Here, it is linear (or linearized):

\[
\dot{x}_i = A x_i + c \sum_{j=1}^{N} \beta_{ij} H x_j \quad x_i \in R^n \quad i = 1,2,\ldots,N
\]

- Coupling strength \( c > 0 \) and \( H \) – input coupling matrix

- Adjacency matrix: \( \begin{bmatrix} \beta_{ij} \end{bmatrix}_{N \times N} \)

If node \( i \) points to node \( j \) (\( j \neq i \)), then \( \beta_{ij} = 1 \); otherwise \( \beta_{ij} = 0 \); and \( \beta_{ii} = 0 \)

For undirected networks, \( \begin{bmatrix} \beta_{ij} \end{bmatrix}_{N \times N} \) is symmetrical; for directed networks, may not be so
How many? Where to pin?

\[ \dot{x}_i = Ax_i + c \sum_{j=1}^{N} \beta_{ij} Hx_j \leftarrow + Bu_i \quad (\text{e.g.,} \quad u_i = -\Gamma x_i) \]

\[ \dot{x}_i = Ax_i + c \sum_{j=1}^{N} \beta_{ij} Hx_j + \delta_i Bu_i \]

\[ \delta_i = \begin{cases} 
1 & \text{if to - control} \\
0 & \text{if not - control} 
\end{cases} \]

Q: How many \( \delta_i = 1 \)? Which \( i \)? \((i = 1,2,\ldots,N) \quad \rightarrow \text{Pinning Control}\)
Controllability Theory
In retrospect, …

MATHEMATICAL DESCRIPTION OF LINEAR DYNAMICAL SYSTEMS*

R. E. KALMAN†

(1930-2016)

Abstract. There are two different ways of describing dynamical systems: (i) by means of state variables and (ii) by input/output relations. The first method may be regarded as an axiomatization of Newton’s laws of mechanics and is taken to be the basic definition of a system.

It is then shown (in the linear case) that the input/output relations determine only one part of a system, that which is completely observable and completely controllable. Using the theory of controllability and observability, methods are given for calculating irreducible realizations of a given impulse-response matrix. In particular, an explicit procedure is given to determine the minimal number of state variables necessary to realize a given transfer-function matrix. Difficulties arising from the use of reducible realizations are discussed briefly.
**System Controllability**

Linear Time-Invariant (LTI) system

\[ \dot{x}(t) = Ax(t) + Bu(t) \]

- \( x \in \mathbb{R}^n \): state vector
- \( u \in \mathbb{R}^p \): control input
- \( A \in \mathbb{R}^{n \times n} \): system matrix
- \( B \in \mathbb{R}^{n \times p} \): control matrix

**Controllable:** The system orbit can be driven by an input from any initial state to any target state in finite time.

State Controllability Theorems

(i) Kalman Rank Criterion

\[ \dot{x}(t) = Ax(t) + Bu(t) \]

The controllability matrix \( Q \) has full row rank:

\[ Q = [B \ AB \cdots A^{n-1}B] \]

(ii) Popov-Belevitch-Hautus (PBH) Test

The following hold:

\[ v^T A = \lambda v^T, \quad v^T B \neq 0 \]

\[ \lambda : \text{eigenvalue of } A \]

\[ v : \text{nonzero left eigenvector with } \lambda \]
System Observability

Linear Time-Invariant (LTI) system

\[
\dot{x}(t) = Ax(t) + Bu(t)
\]
\[
y(t) = Cx(t)
\]

\(x \in \mathbb{R}^n\): state vector
\(u \in \mathbb{R}^p\): control input
\(A \in \mathbb{R}^{n \times n}\): system matrix
\(B \in \mathbb{R}^{n \times p}\): control matrix

\[
x(t) = x(t_0)e^{(t-t_0)A} + \int_{t_0}^{t} e^{(t-\tau)A}Bu(\tau)d\tau
\]

**Observability**: Input-output pair \((u(t), y(t))\) on \([t_1, t_2]\) uniquely determines the initial state \(x(t_0)\)
What About Directed Networks?

\[
\frac{dx_i}{dt} = f(x_i), \quad x_i \in \mathbb{R}^n
\]

\[
u_i = -\Gamma x_i
\]
In retrospect: large-scale systems theory

Structural Analysis of Dynamical Systems

Q:
Is this kind of structure controllable?

A directed network
Structural Controllability

Corresponding linearized system has the following general form:

\[
\begin{align*}
\dot{x}_1 &= a_{11} x_1 \\
\dot{x}_2 &= a_{21} x_1 + a_{22} x_2 + bu, \quad A = \begin{bmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ 0 & a_{32} & a_{33} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ b \\ 0 \end{bmatrix} \\
\dot{x}_3 &= a_{32} x_2 + a_{33} x_3 \\
\end{align*}
\]

\[
\text{rank}\left[ B, AB, A^2B \right] = \begin{bmatrix} 0 & 0 & 0 \\ b & a_{22}b & a_{22}^2b \\ 0 & a_{32}b & a_{32}(a_{22} + a_{33})b \end{bmatrix} \leq 2
\]

\rightarrow \text{Uncontrollable}
Structural Controllability

In the controllability matrix: \( Q = [B \ AB \ \cdots \ \ A^{n-1}B] \)

All 0 are fixed

There is a realization of independent nonzero parameters such that \( Q \) has full row rank

Example 1:

Realization: All admissible parameters \( a \neq 0, \ d \neq 0 \)

Example 2: Frobenius Canonical Form

\[
Q = \begin{bmatrix}
-a_1 & -a_2 & -a_3 & \cdots & -a_{n-1} & -a_n \\
1 & 0 & 0 & \cdots & 0 & 0 \\
0 & 1 & 0 & \cdots & 0 & 0 \\
0 & 0 & 1 & \cdots & 0 & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
0 & 0 & 0 & \cdots & 1 & 0
\end{bmatrix}
\]
Structural Controllability

A network of single-input/single-output (SISO) node systems
\[ Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & a_{21} & 0 \\ 0 & 0 & a_{32}a_{21} \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & a_{21} & 0 \\ 0 & 0 & a_{31} \end{bmatrix}, \quad \begin{bmatrix} 1 & 0 & 0 \\ 0 & a_{21} & 0 \\ 0 & a_{31} & a_{33}a_{31} \end{bmatrix}. \]

\[ \text{rank } C = 3 = n \quad \text{rank } C = 2 < n = 3 \quad \text{rank } C = 3 = n \]

Controllable \quad Uncontrollable \quad Controllable

Matching in Directed Networks

- **Matching**: a set of directed edges without common heads and tails
- **Unmatched node**: the tail node of a matching edge

**Maximum matching**: Cannot be extended

**Perfect matching**: All nodes are matched nodes

← Maximum but not perfect matching
Solution to Pinning Control:

**Minimum Inputs Theorem**

**Q:** How many?

**A:** The minimum number of inputs $N_D$ needed is:

- **Case 1:** If there is a perfect matching, then $N_D = 1$
- **Case 2:** If there is no perfect matching, then $N_D =$ number of unmatched nodes

**Q:** Where to put them?

**A:** **Case 1:** Anywhere

**Case 2:** At unmatched nodes

This completely answer the pinning control question for SISO networks

Characterization of General Topology with SISO Nodes

\[ \dot{x}_i = Ax_i + \sum_{j=1}^{N} \beta_{ij} H C x_j + \delta_i B u_i, \quad i = 1, 2, \ldots, N \quad x_i \in \mathbb{R}^n \quad y_i \in \mathbb{R}^m \quad u_i \in \mathbb{R}^p \]

\[ L = [\beta_{ij}] \in \mathbb{R}^{N \times N} \quad \Delta = \text{diag}(\delta_1, \ldots, \delta_N) \]

A network with SISO nodes is \textbf{controllable if and only if}

\( (A, H) \) is controllable
\( (A, C) \) is observable
For any eigenvalue \( s \) of \( A \) and \( \alpha = Re(s) \), \( \alpha L \neq 0 \) for \( \alpha \neq 0 \)
For any eigenvalue \( s \) of \( A \), \( \text{rank}(I - L \Gamma_1, \Delta \Gamma_2) = N \),
with \( \Gamma_1 = C[sl - A]^{-1}H \), \( \Gamma_2 = C[sl - A]^{-1}B \)

State Controllability

A network of multi-input/multi-output (MIMO) node systems, where the node systems are of higher-dimensional
A Network of Multi-Input/Multi-Output LTI Systems

Node system
\[
\dot{x}_i = Ax_i + Bu_i \quad y_i = Cx_i \quad x_i \in \mathbb{R}^n \quad y_i \in \mathbb{R}^m \quad u_i \in \mathbb{R}^p
\]

Networked system
\[
\dot{x}_i = Ax_i + \sum_{j=1}^{N} \beta_{ij} Hy_j, \quad y_i = Cx_i, \quad i = 1, 2, \ldots, N
\]

Networked system with external control
\[
\dot{x}_i = Ax_i + \sum_{j=1}^{N} \beta_{ij} HCx_j + \delta_i Bu_i, \quad i = 1, 2, \ldots, N
\]

\[\delta_i = 1: \text{with external control} \quad \delta_i = 0: \text{without external control}\]

Some notations

Node system \((A,B,C)\)  
Network structure \(L = [\beta_{ij}] \in \mathbb{R}^{N \times N}\)
Coupling matrix \(H\)  
External control inputs \(\Delta = diag(\delta_1, \ldots, \delta_N)\)

Counter-intuitive example 1

Network structure

Node system

Networked MIMO system

\[ L = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \]

\[ A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \]

\[ C = \begin{bmatrix} 0 & 1 \end{bmatrix} \]

\((A, B)\) is controllable

\((A, C)\) is observable

\(H = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \)

\((A, B)\) is controllable

state uncontrollable
Counter-intuitive example 2

Network structure

Node system

Networked MIMO system

\[ L = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \]

\[ A = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \]

\[ B = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \]

\[ C = [0 \ 1] \]

\[ H = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \]

\[ (A, B) \text{ is uncontrollable} \]

\[ (A, C) \text{ is observable} \]

\[ (A, C) \text{ is observable} \]

\[ \text{state controllable} \]
A Network of Multi-Input/Multi-Output LTI Systems

A necessary and sufficient condition

\[
\dot{x}_i = Ax_i + \sum_{j=1}^{N} \beta_{ij} H C_j + \sum_{k=1}^{s} \delta_{ik} B u_k , \quad x_i \in \mathbb{R}^n , \quad i = 1, \ldots N
\]

\[
y_l = \sum_{j=1}^{N} m_{lj} D x_j , \quad u_k \in \mathbb{R}^p , \quad k = 1, \ldots s
\]

\[
L = [\beta_{ij}] \in \mathbb{R}^{N \times N} \quad \Delta = [\delta_{ij}] \in \mathbb{R}^{N \times s}
\]

If and only if

State Controllable

Matrix equations

\[
\Delta^T X B = 0, \quad L^T X H C = X (\lambda I - A) \quad \forall \lambda \in \mathbb{C}
\]

have a unique solution

\[
X = 0
\]

Pinning Control of MIMO Networks

Solution to Pinning Control: How many? Where to pin?

\[
\Delta = \text{diag}[\delta_i] \quad \text{such that the above algebraic matrix equations has a unique zero solution } X
\]

\[
\text{How many } \delta_i = 1 \text{ and which } \delta_i = 1
\]

This completely answer the pinning control question for MIMO networks
Robustness of Network Controllability

Robustness of Controllability

Against Destructive Attacks

(Node-Removals / Edge-Removals)
Measure for Controllability Robustness

Let $N_D$ be the minimum number of external control input needed to maintain the network controllability.

Define

Controllability index:

$$n_D = \frac{N_D}{N}$$

Controllability Robustness:

The smaller the value of $n_D$, the better the robustness against (node-removal or edge-removal) attacks.
Complex Network Models

- Random-Graph (RG) Network
- Scale-Free (SF) Network
- Multiplex Congruence Network (MCN)
- $q$-Snapback Network (QSN)
- Random Triangle Network (RTN)
- Random Rectangle Network (RRN)
Comparison of Controllability Robustness

- Attack Methods
- Simulation Results
- Comparisons

## Attack Methods

<table>
<thead>
<tr>
<th>Targeted</th>
<th>Betweenness</th>
<th>Node-removal</th>
<th>Edge-removal</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Remove the node with the largest betweenness</td>
<td>Remove the edge with the largest betweenness</td>
</tr>
<tr>
<td>Degree</td>
<td></td>
<td>Remove the node with the largest out-degree</td>
<td>Remove the edge with the largest edge degree</td>
</tr>
<tr>
<td>Random</td>
<td></td>
<td>Remove a node randomly</td>
<td>Remove an edge randomly</td>
</tr>
</tbody>
</table>

Edge degree for an edge $A_{ij}$ is $\sqrt{k_i \times k_j}$, where $k_i$ and $k_j$ are the out-degrees of nodes $i$ and $j$, respectively.
Simulation Results (Comparison)

Random Node-Removal

RRN outperforms the other networks.

RRN, RG, and RTN performs similarly.

SF performs the worst.

Observation:
RRN, RTN have many loops
RG is homogeneous
Random Edge-Removal

RRN outperforms the other networks.

RRN, RG, and RTN performs similarly.

SF performs the worst.

Observation:
RRN, RTN have many loops
RG is homogeneous
Motivation of applying Machine Learning:

There is no clear correlation between the topological features and the controllability robustness of a general (directed or undirected) network.

Machine Learning using Convolutionary Neural Network (CNN)

CNN architecture used for controllability robustness prediction
FM – feature map
FC – fully connected
data size $N_i = \lceil N/(i + 1) \rceil$, for $i = 1, 2, \ldots, 7$.
$N_{FC1} = N_7 \times N_7 \times 512$, $N_{FC2} \in (N_{FC1}, N - 1)$ is a hyperparameter
$N_{FC2} = 4096$ for $N = \{800,1000,1200\}$
Networks and Image Representation

**Topology:**

```
A -- B -- E
|    |    |
C   D
```

**Adjacency Matrix:**

```
<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
```

**Image:**

![Image representation](image.png)
Erdos-Renyi Random Graph (ER)

**ER:** uniformly randomly connect any two nodes by $M$ edges; the directions are evenly-randomly assigned

**ER-image:** uniformly randomly distribute the $M$ light pixels into an $N \times N$ matrix
**Barabasi-Albert Scale-Free Network (SF)**

**SF**: nodes $i$ and $j$ ($i \neq j$, $i, j = 1, 2, \ldots, N$) are randomly picked with a probability proportional to their weights $w_i$ and $w_j$, respectively. Then, an edge $A_{ij}$ from $i$ to $j$ is added only if they are not connected.

**SF-image**: a heterogeneous network and thus a heterogeneous image; very strong structural characteristics.
Simulations

(There are many simulation results, but only one is shown for illustration)

input: image

output: CR performance prediction

CR = Controllability Robustness

RA – Random Attack
ER – Erdos-Renyi Random Network
Blue/Red – True/Prediction
Black/Green – Errors/Deviations
Knowledge-Based Learning

Sufficiently utilize the prior knowledge (network types) in pre-processing for improving predictions

Simulation

BA = BA scale-free network
ER = ER random-graph network
QSN = q-snapback network
SW = Small-world network

PCR = Predicted controllability robustness
iPCR = improved Predicted controllability robustness

Training size = 4000
Testing size = 1000
Network size = 200

$P_N$ = Attack probability
**Significant Finding:**

*Cycles and Homogeneity are good for both Controllability and Robustness*

An empirical necessary (homogeneity) condition:

\[ \frac{M}{N} \leq k_i^{in, out} \leq \left\lfloor \frac{M}{N} \right\rfloor \quad (i = 1, 2, ..., N) \]

\(M\) - number of edges, \(N\) – number of nodes, \(k\) - degree

Research Outlook

General Theory

Higher-order Topology

Cycle, Clique, Cavity

Betti Number, Euler Characteristic Number
References


Acknowledgement:

Xiaofan WANG 汪小帆, Shanghai University
Lin WANG 王琳, Shanghai Jiao Tong University
Yang LOU 楼洋, City University of Hong Kong