Effective connectivity and controllability in brain dynamics

Purpose? In **general**, we want to provide external stimulation to brain regions to achieve desired state - “control the system”

Purpose? In **general**, we want to provide external stimulation to brain regions to achieve desired state - “control the system”

How? We use Linear Control theory - Perturb the linear dynamical system with an external input.

Purpose? In **general**, we want to provide external stimulation to brain regions to achieve desired state - "control the system"

How? We use Linear Control theory - Perturb the linear dynamical system with an external input.

Why linear modelling on brain systems?

We use BOLD fMRI time series.

- Spatial and temporal averaging in dynamics,
- observation noise,
- limited data samples to capture non linear effects.

Linear model to capture dynamics of 72 brain regions. Each brain region is a considered a node.

Purpose? In **general**, we want to provide external stimulation to brain regions to achieve desired state - “control the system”

How? We use Linear Control theory - Perturb the linear dynamical system with an external input.

Why linear modelling on brain systems?

We use BOLD fMRI time series.

- Spatial and temporal averaging in dynamics,
- observation noise,
- limited data samples to capture non linear effects.

Linear model to capture dynamics of 72 brain regions. Each brain region is a considered a node.

How to select couplings in Linear model?

- Anatomical connectivity matrix used earlier by researchers.
- An alternative is EC defined as simplest possible circuit diagram that would replicate the observed timing relationships.

Purpose? In **general**, we want to provide external stimulation to brain regions to achieve desired state - “control the system”

How? We use Linear Control theory - Perturb the linear dynamical system with an external input.

Why linear modelling on brain systems?

We use BOLD fMRI time series.

- Spatial and temporal averaging in dynamics,
- observation noise,
- limited data samples to capture non linear effects.

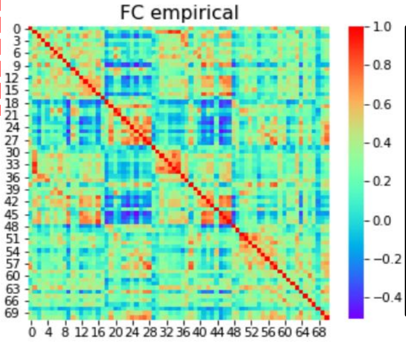
Linear model to capture dynamics of 72 brain regions. Each brain region is a considered a node.

How to select couplings in Linear model?

- Anatomical connectivity matrix used earlier by researchers.
- An alternative is EC defined as simplest possible circuit diagram that would replicate the observed timing relationships.

How EC was obtained? Various linear models were proposed in literature

Effective Connectivity(EC) of brain network

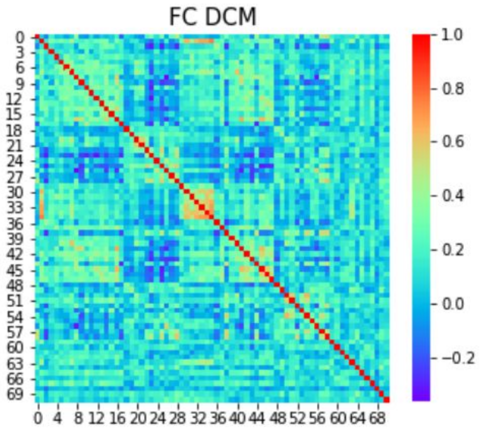


Pearson correlation of observed BOLD fMRI time series of 72 brain regions

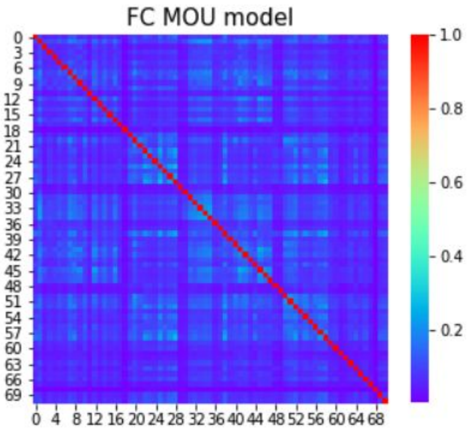
EC extracted from the following models:

- Sparse Dynamic Causal Modelling (spDCM)
- Multivariate Ornstein Uhlenbeck(MOU) model
- Linear model

Prando's Sparse DCM



Gilson's MOU



Nozari's Linear model

- Quality metric of the fit is bad.
- Quality of fit on "our" data is not the same as claimed by the authors.
- Henceforth, EC from LM is not used.

Model-data correlation(PCC): **0.789** **0.646**

Controllability

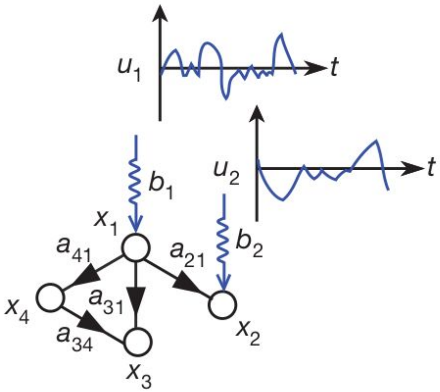
With a suitable choice of input signal vector u(t), the system can be driven from any initial state to any final state in finite steps.

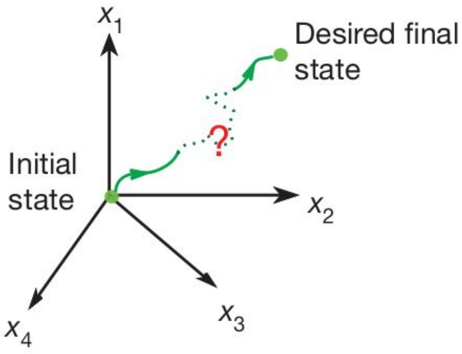
Dynamics is given by Linear Time Invariant(LTI) system:

$$\frac{d\mathbf{x}(t)}{dt} = A\mathbf{x}(t) + B\mathbf{u}(t)$$

- **x**(t) - state vector for N brain regions at time t
- *A* - (N,N) state matrix, in our case will be EC matrix throughout.
- *B* - (N, r) input matrix with “r” being number of control nodes required to control the system.

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ a_{21} & 0 & 0 & 0 \\ a_{31} & 0 & 0 & a_{34} \\ a_{41} & 0 & 0 & 0 \end{pmatrix}; B = \begin{pmatrix} b_1 & 0 \\ 0 & b_2 \\ 0 & 0 \\ 0 & 0 \end{pmatrix};$$





Kalman Controllability

System is controllable

Min. Energy reqd. to steer the system: (controllability metric) $E_{min} = 1/\lambda_{min}(W)$

Algebraic Condition: rank(C) = N | $min(\lambda(W)) > 0$

Computed from:

$C = [B, AB, A^2B, ..., A^{N-1}B]$

Controllability matrix, (N x N.r)

$AW + WA^T = -BB^T$

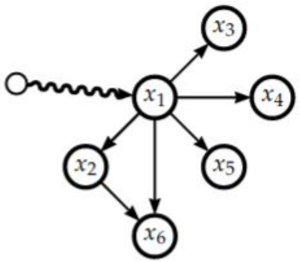
Lyapunov equation

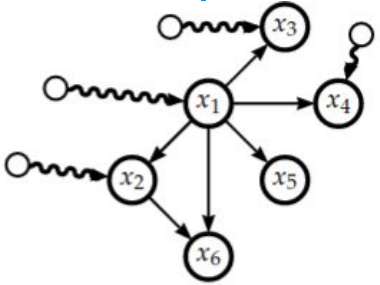
Caveat:

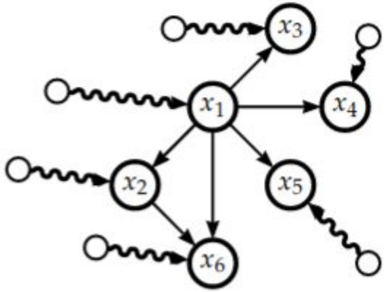
The 2 stability/algebraic conditions are equivalent upto numerical error and only if LTI is stable i.e. $max(Real(\lambda(A))) < 0$

A practical implementation of Kalman controllability

(testing with small matrix 6 x 6)







Tests / Algorithm	Single node controllability	Fraction of nodes controllability (Consider Unmatched nodes)	All nodes controllability
Compute B	$e_i,\ i \in \{1, ..., N\}$	$[e_{k_1}, ..., e_{k_r}]\ dim = (N, r)$ $\{k_1, ..., k_r\}$ are driver nodes	$I_{N \times N}$
Check Kalman rank	False	True	True
$min(\lambda(W))$	----------------	True	True
Compute $E_{min} = 1/\lambda_{min}(W)$	----------------	**1.734×10^{12}**	**1.581**

Practical implementation of Structural controllability

- Struct. Contr. is more general wrt. to Kalman Contr.
- Non zero entries of B are free parameters

Apply **maximum matching** to identify unmatched nodes.

Unmatched nodes = driver nodes

Construct B with non zero entries as free parameters

Minimize cost function i.e. energy

$$\min_{\Theta} f_{\Theta}(A,B) = \min_{\Theta} E_{min}(W(A,B))$$

Optimize the free parameters of B

Constraints

$$rank(C) = N$$

$$\lambda_{min}(W(A,B)) > 0$$

Unmatched nodes = Driver nodes identified = { 2, 5 }

$$B_{kalman} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \quad B_{struct} = \begin{bmatrix} 0 & 0 \\ b_1 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & b_2 \\ 0 & 0 \end{bmatrix} \quad B_A = \begin{bmatrix} 0 & 0 \\ 0.543 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 1.531 \\ 0 & 0 \end{bmatrix} \quad B_{A*5} = \begin{bmatrix} 0 & 0 \\ 1.8860 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 5.30205e+03 \\ 0 & 0 \end{bmatrix}$$

Results comparison of rescaling A(via scalar product) and B (via Optimization):

EC matrix	$max(Real(\lambda(A)))$	Kalman min. Energy (Unmatched nodes)	Kalman min. Energy (All nodes)	Structural min. Energy
A	**-0.5289**	**1.734×10^{12}**	**1.581**	**7.392×10^{11}**
A x 5	**-0.4272**	**1.25×10^{7}**	**1.877**	**0.445**

Factors influencing controllability

Possibilities that can affect controllability

No. of entries of EC matrix
Density

Position of the entries
Topology

Choice of model to extract EC

Sparsifying it.

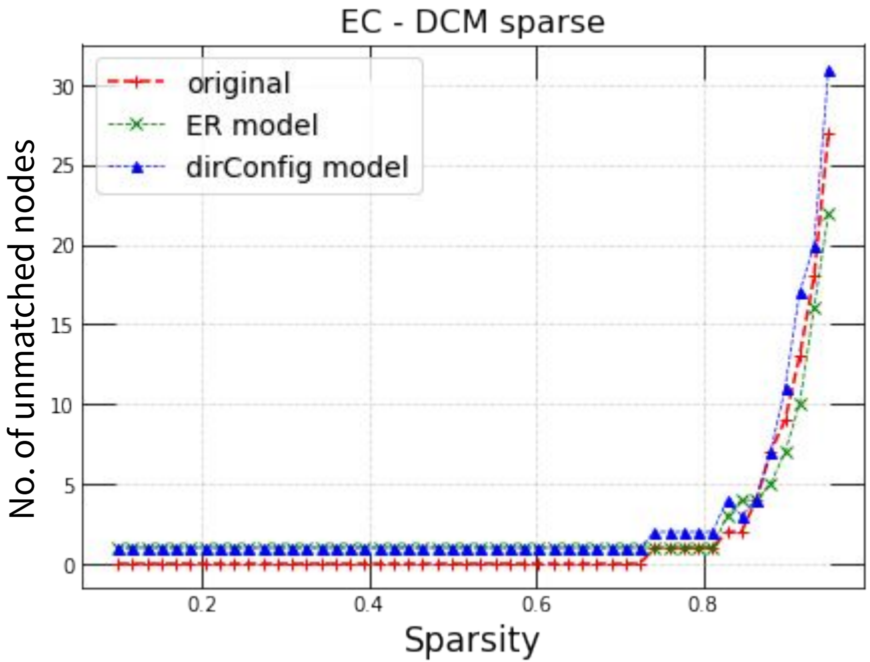
Setting the absolute of max weighted entries to zero in fractions 0.1 to 0.95

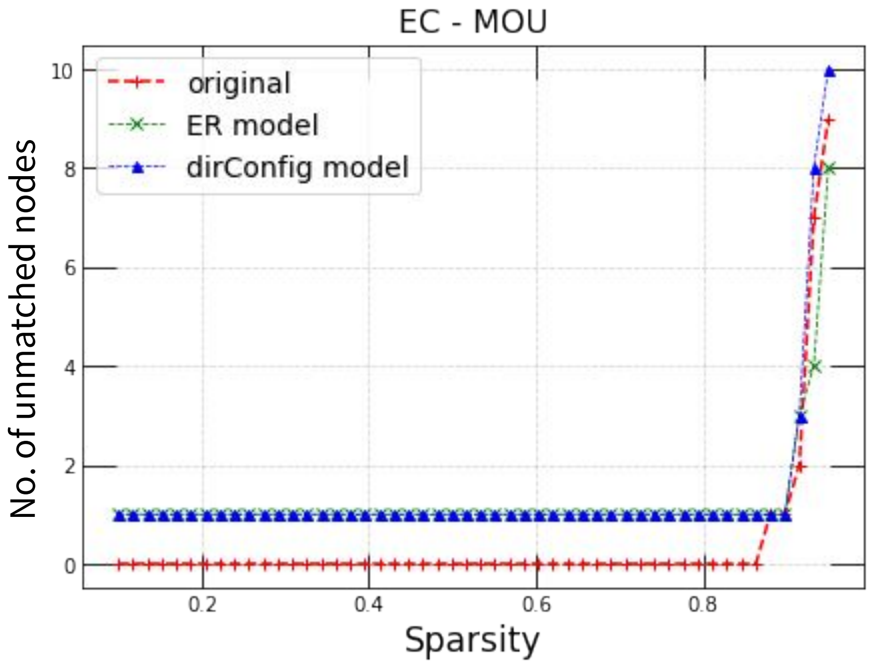
Randomizing the connections.

Use Erdos Renyi(ER) and Directed Config.(DC) Random models

spDCM and **MOU** models

Henceforth, Whole brain network is considered.
I.e. all 72 brain regions



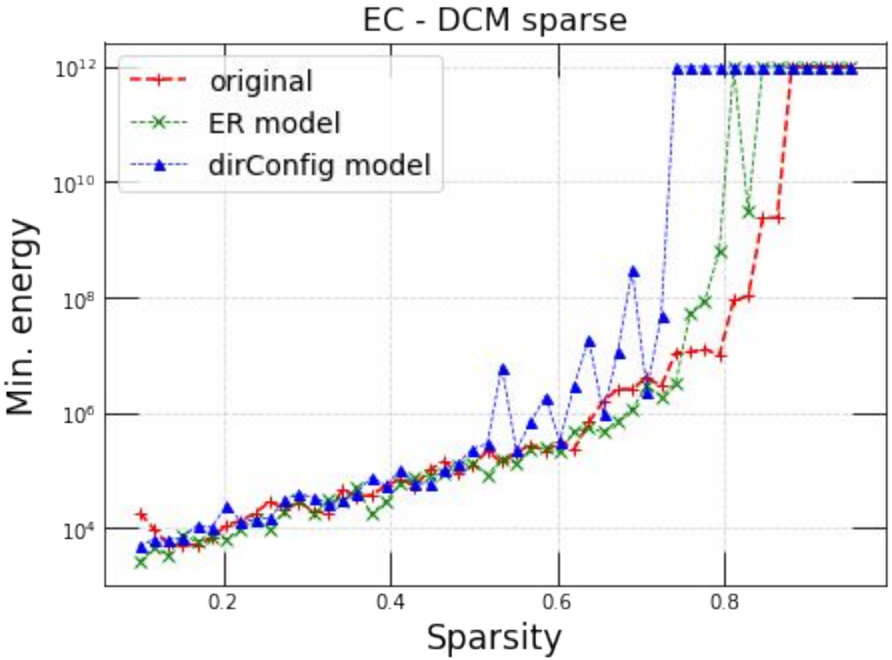


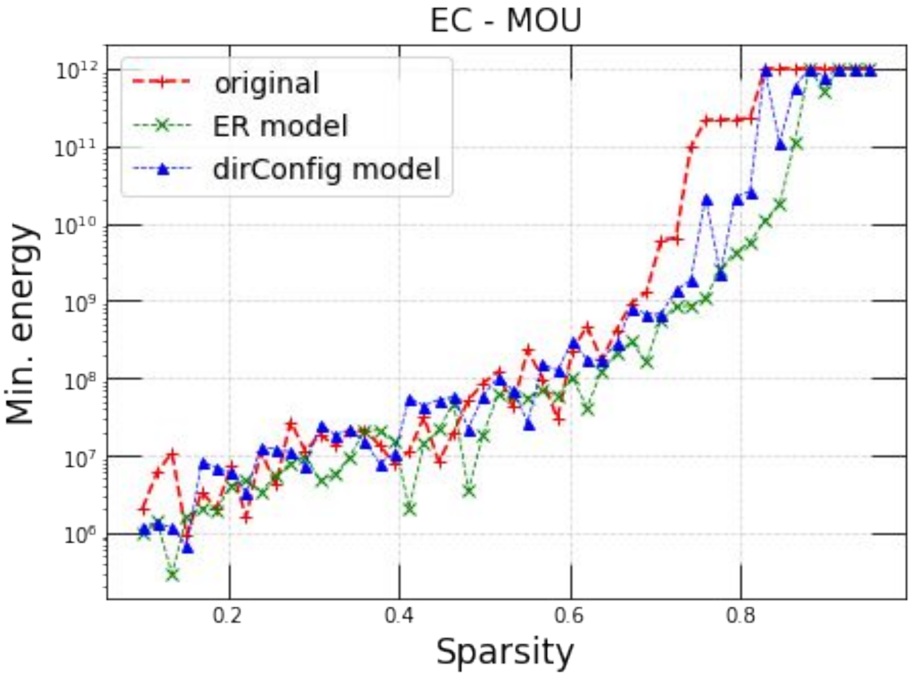
Greater the sparsity - lower the density - higher is the number of unmatched nodes obtained.

Irrespective of:

- model to extract EC
- Random models

Same trend is observed.





- EC spDCM
- sparsity = 0.95
- N_D = 27
- $E_{min} \sim 10^{22}$

10^{12} is considered upperbound.
Values > 10^{12} set to the same value

50% of total nodes are **randomly selected** and fixed as Driver nodes.

Conclusion

- Control energy is sensitive to scaling of A and B.
- Single node controllability is not feasible.
- Controllability with unmatched nodes gives extremely large energy. In theory yes, but in practice it is not feasible.
- Greater the sparsity, higher is number of unmatched nodes obtained.
- Controllability of a network is weakly affected by topology and choice of model used to extract EC.
- Controllability is sensitive to degree of sparsity.

End of presentation

Proof-of-concept Complete State Controllability

Fix initial state **[1, 1, 1, 1, 1, 1]** final state **[2, 2, 2, 2, 2, 2]**.

A: (6 x 6) EC matrix from spDCM

B: Optimal B from structural controllability

Solve for **u**

$$C^{N\times N.r}\begin{bmatrix}\mathbf{u}(N-1)\\ \mathbf{u}(N-2)\\ \vdots \\ \mathbf{u}(1)\\ \mathbf{u}(0)\end{bmatrix}^{(N.r)\times 1} = \mathbf{x}(N) - A^N\mathbf{x}(0)^T = \mathbf{x}_{final} - A^N\mathbf{x}_{initial}^T$$

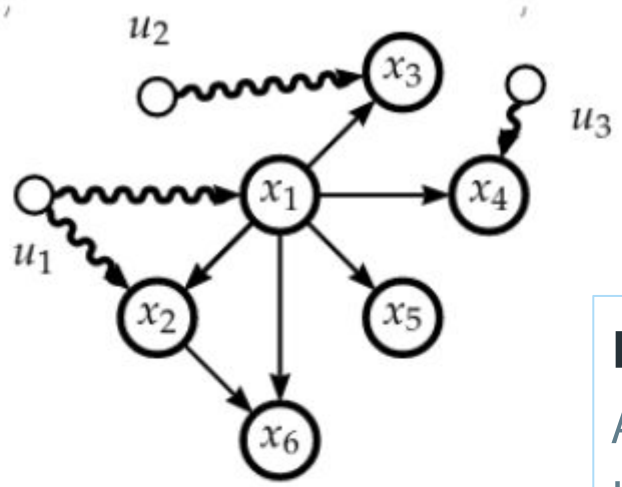
$$\begin{array}{ccc} & \downarrow 2 & \downarrow 5 \\ \mathbf{u} = & \begin{bmatrix} 1.2878 & 6.4182e+04 \\ -1.0183 & 3.37287e+05 \\ 0.8051 & 5.46988e+05 \\ -0.6366 & 8.1585e+04 \\ 0.5034 & -5.21282e+05 \\ -0.3980 & -3.39333e+05 \end{bmatrix} & \downarrow \text{Time steps} \end{array}$$

Cross Check.
With **u** computed.
Fix **A**, **B**, **x**(t=0)

$$\mathbf{x}(N) = A^N\mathbf{x}(0)^T + A^{N-1}B\mathbf{u}(0)^T + ... + B\mathbf{u}(N-1)^T$$

$$\mathbf{x}_n = \mathbf{x}(t=T-1)$$
$$= [\, 1.97618996,\ 1.75563045,\ 1.84599689,\ 2.00013076,\ 2.08584186,\ 1.92522838 \,]$$

MSE: 0.016



- $\{u_1, u_2, u_3\}$ input signal vertices
- $\{x_1, x_2, x_3, x_4\}$ control nodes
- Either $\{x_1, x_3, x_4\}$ or $\{x_2, x_3, x_4\}$ are driver nodes

Driver nodes

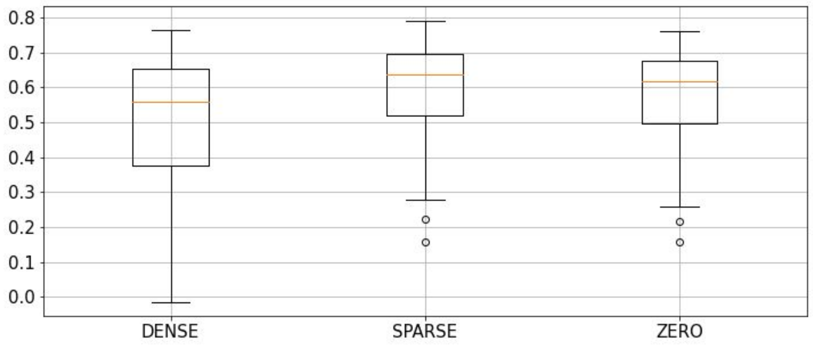
Apply maximum matching to find unmatched nodes. Unmatched nodes if >1 becomes driver nodes to control the system.

If unmatched nodes = 0, it is perfect matching. Driver nodes = 1 in this case.

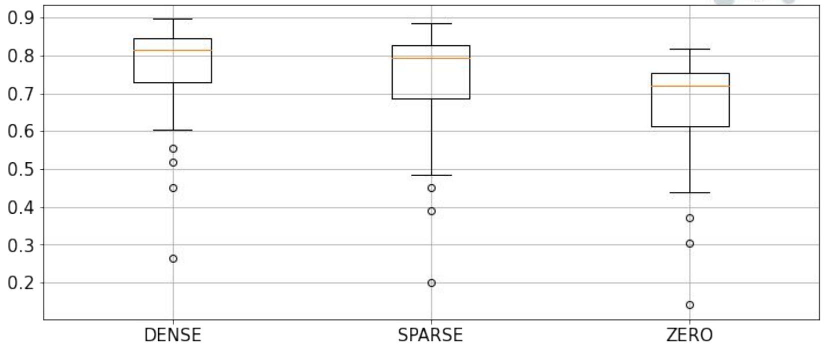
$$
A = \begin{bmatrix}
-0.61502668 & 0 & 0 & 0 & 0 & 0.04969445 \\
0 & -0.79069292 & 0 & 0 & 0 & 0 \\
0 & 0 & -0.69419440 & 0 & -0.04564398 & -0.03556111 \\
0 & 0 & 0.04076849 & -0.53235401 & -0.00303871 & -0.01251824 \\
0 & 0 & 0.16217933 & 0 & -0.67205625 & 0 \\
0.02078494 & 0 & 0 & -0.02069000 & 0 & -0.65260954
\end{bmatrix}
$$

Nozari's Linear Model

Test set - 25%



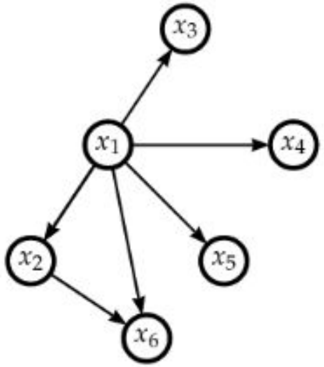
Training set - 25%

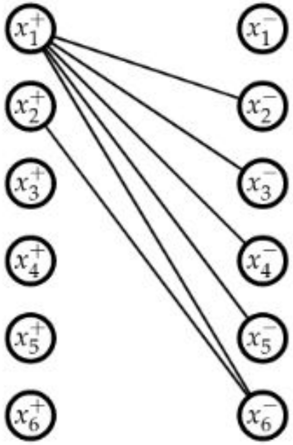


$$R_i^2 = 1 - \frac{\sum_t E_i^{test}(t)^2}{\sum_t (y_i^{test} - \overline{y}_i^{test}(t))^2}$$

Maximum matching examples

Example 1:

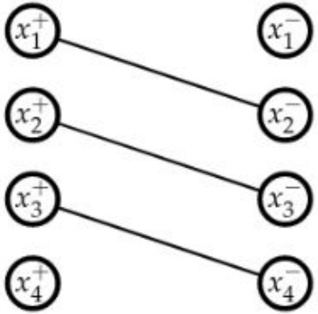




Matched nodes: { 6, 5 } or { 6, 4 } or { 6, 3 } or { 6, 2 }

Unmatched nodes: { 1, 2, 3, 4 } or { 1, 2, 3, 5 } or { 1, 2, 4, 5 } or { 1, 3, 4, 5 }

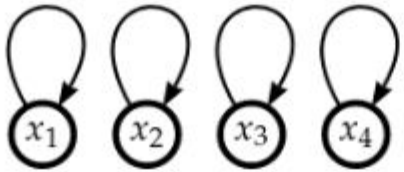
Example 2:

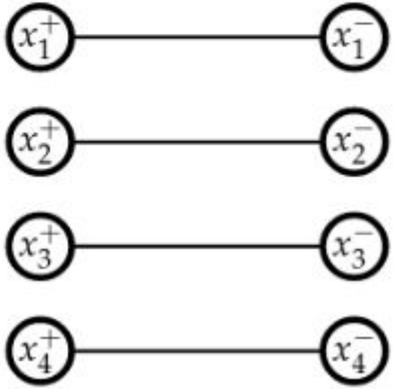


Matched nodes: { 2, 3, 4 }

Unmatched nodes: { 1 }

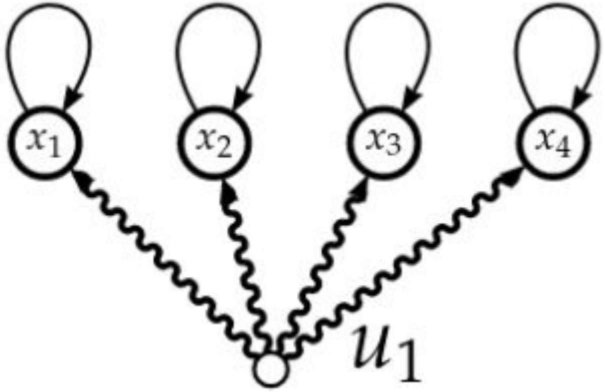
Example 3:





Matched nodes: { 1, 2, 3, 4 }
Unmatched nodes: { }

By Minimum input theorem, Number of driver nodes = 1



Computational complexity

- Compute C: $O(\,N.N^2 r\,)$
- Computing rank(C): $O(\,N^2.Nr\,)$
- Total complexity of Kalman controllability: $O(N^3 r)$
- Solve Lyapunov equation: $O(N^3)$

References:
Kalman rank: https://math.stackexchange.com/questions/3322915/what-is-the-complexity-order-of-kalman-rank-condition
Lyapunov complexity: https://people.kth.se/~eliasj/NLA/matrixeqs.pdf