Network partitioning towards scale-free structure

Nicolas Martin

CNRS, GIPSA-lab, Grenoble

ERC Workshop 2019

Outline

Scale-FreeBack

Outline

Scale-FreeBack

Scale-free networks history

- Introduced by Price in 1965 for citation networks
- Rediscovered by Barabasi and collaborators around 1999
- "Barabasi's bandwagon": discovery of the scale-freeness of a lot of networks, like the WWW, social networks, biological networks

D.Price

A.-L. Barabasi

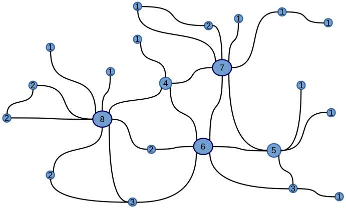
Scale-FreeBack

Definition of scale-free network

- Degree distribution: distribution of the number of connections per node
- Scale-free networks have power-law $P(k) = k$ degree distributions

Definition of scale-free network

- Degree distribution: distribution of the number of connections per node
- Scale-free networks have power-law $P(k) = k$ degree distributions

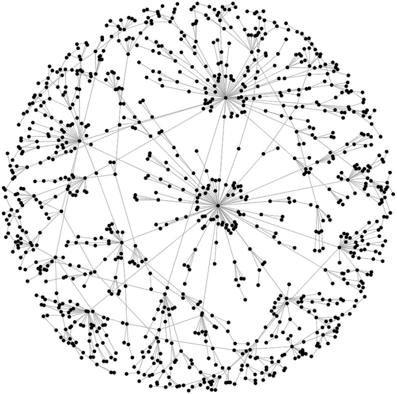


Example of a scale-free graph

Degree distribution of a scale-free graph

Properties of scale-free network

- Presence of *hubs* with large degree
- Small radius/diameter
- Small number of edges
- Robust to random node failures
- Easy to disconnect



Scale-FreeBack

Advantages of scale-freeness for control design

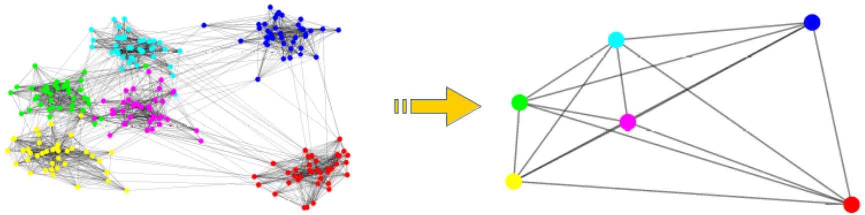
These properties may bring advantages for control design [3]:

- Hubs ! Localised control
- Small distances ! All nodes are easily reachable with few inputs [6]
- Few edges ! Sparsity of the problem [5]

Scale-FreeBack

Network partition

A network partition is a partition of the node set.
From this partition we derive a reduced network



Scale-FreeBack

General problem

Partition a network towards a scale-free structure to take advantage of the properties. We also want to preserve a certain similarity.

Given a graph G_0 find $G^?$ solution of the following minimisation problem:

$$G^? = \arg\min_G \quad J_{SF}(G) + J_{sim}(G; G_0) \quad \text{under constraints on } G \qquad (1)$$

General problem

Partition a network towards a scale-free structure to take advantage of the properties. We also want to preserve a certain similarity.

Given a graph G_0 find $G^?$ solution of the following minimisation problem:

$$G^? = \arg\min_{G} \quad \mathrm{J_{SF}}\ (G) + \mathrm{J_{sim}}(G; G_0) \qquad \text{under constraints on } G \qquad (1)$$

Outline

Main idea

General algorithm which can be used for any particular case of the general problem.

Consists in iteratively merge a pair nodes in the network.
The algorithm does not provide an optimal solution of the problem.

Merging

We merge two nodes into a super-node and we preserve the connections with the other nodes. The weights on the edges are recomputed.

Merging

We merge two nodes into a super-node and we preserve the connections with the other nodes. The weights on the edges are recomputed.

Merging

We merge two nodes into a super-node and we preserve the connections with the other nodes. The weights on the edges are recomputed.

Merging

We merge two nodes into a super-node and we preserve the connections with the other nodes. The weights on the edges are recomputed.

We note $G_{i \leftrightarrow j}$ the network obtained by merging (i, j) in the network G

General algorithm

INPUT : initial network G_0, scale-free coe cient

while : stop

- $(i;j)$ edge maximising $J_{SF}(G_{i\$ j}) + J_{sim}(G_{i\$ j}; G_0)$ under constraints
- G_{k+1} $G_{i\$ j}$

end

OUTPUT : Final network $G_{k\ nal}$

General algorithm

INPUT : initial network G_0, scale-free coe cient

while : stop

- $(i;j)$ edge maximising $J_{SF}(G_{i\$ j}) + J_{sim}(G_{i\$ j}; G_0)$ under constraints
- G_{k+1} $G_{i\$ j}$

end

OUTPUT : Final network $G_{k\ nal}$

Example of algorithm

INPUT : initial network G_0, scale-free coe cient

while : stop

- $(i;j)$ edge maximising $J_{SF}(G_{i \leftrightarrow j})$ under $|x_i - x_j| <$
- G_{k+1} $G_{i \leftrightarrow j}$

end

Outline

Steady-state

$$x^{\star} = P^{\top} x^{\star} \tag{2}$$

with P the row-normalised adjacency matrix:

$$P_{i,j} = \frac{A_{i,j}}{\sum_k A_{i,k}},$$

Steady-state

$$x^\star = P^\top x^\star \tag{2}$$

with P the row-normalised adjacency matrix:

$$P_{i,j} = \frac{A_{i,j}}{\sum_k A_{i,k}},$$

Steady-state

$$x^\star = P^\top x^\star \tag{2}$$

with P the row-normalised adjacency matrix:

$$P_{i,j} = \frac{A_{i,j}}{\sum_k A_{i,k}},$$

Speci c problem for steady-state preservation

Given a graph G_0 nd $G^?$ solution of the following minimisation problem:

$$G^? = \arg\min_{G} \quad J_{SF}(G) + J_{sim}(G; G_0) \qquad \text{under constraints on } G \tag{3}$$

Speci c problem for steady-state preservation

Given a graph G_0 nd $G^?$ solution of the following minimisation problem:

$$G^? = \arg\min_{G} \quad J_{SF}(G) \qquad \text{under } x_G^? \quad x_{G_0}^? \tag{4}$$

Weight computation

Weight computation

Weight computation

Weight computation

Other properties

Within this approach we can also preserve:

- Flow property
  "What goes in goes out"
  $\mathbb{1}A = \mathbb{1}A'$
- Total mass
  $\sum_{i,j} A_0(i,j) = \sum_{i,j} A_{red}(i,j)$

Simulation for steady state preservation

Outline

Model reduction

Model reduction

$$\Sigma : \dot{x} = Ax + Bu \qquad \hat{\Sigma} : \begin{cases} \dot{\hat{x}} = PAP^T \hat{x} + PBu \\ x = P^T \hat{x} \end{cases} \tag{5}$$

x 2 $\mathbb{R}^n$
$\hat{x}$ 2 $\mathbb{R}^m$
$= n \quad m$ is the reduction

Scale-FreeBack

Transfer function minimisation problem

Given a graph G_0 find $G^{?}$ solution of the following minimisation problem:

$$G^{?} = \arg\min_{G} \qquad \mathrm{J}_{\mathrm{SF}} \; (G) + kg(s) \quad \hat{g}(s)k_{H_2} \tag{6}$$

where g and $\hat{g}$ are the transfer functions from u to x and from u to $\hat{x}$ respectively

Algorithm

INPUT : initial network G_0

while $:$ stop

- $(i;j)$ edges minimising $J_{SF}\ (G) + kg(s) \quad \hat{g}(s)k_{H_2}$
- G_{k+1} merge $(i;j)$ in G_k

end

OUTPUT : Final network G_k

Numerical result

TOP: $kg(s)$ $\hat{g}(s)k_{H_2}$ in function of for different value of
BOTTOM : Degree distribution for different value of
In blue : $= 0$ (only similarity cost function)
In yellow : $=$ *1* (only scale-free cost function)

Scale-FreeBack

Outline

Scale-FreeBack

Summary and future work

Summary

We developed a general algorithm able to reduce network into a scale-free network and able to preserve properties and a notion of similarity.
We presented two different implementations of this algorithm.

Future work:

- Different similarity costs and physical properties
- Applications to network control
- Adapting to time-varying networks

Scale-FreeBack

References

- Albert-László Barabási, Réka Albert, and Hawoong Jeong. "Scale-free characteristics of random networks: the topology of the world-wide web". In: *Physica A: statistical mechanics and its applications* 281.1 (2000), pp. 69–77.
- David Gfeller and Paolo De Los Rios. "Spectral coarse graining of complex networks". In: *Physical review letters* 99.3 (2007), p. 038701.
- Gustav Lindmark and Claudio Altafini. "A driver node selection strategy for minimizing the control energy in complex networks". In: *IFAC-PapersOnLine* 50.1 (2017), pp. 8309–8314.
- Nicolas Martin, Paolo Frasca, and Carlos Canudas-De-Wit. "A network reduction method inducing scale-free degree distribution". In: *ECC 2018-European Control Conference*. 2018, pp. 1–6.
- Giulia Menichetti, Luca Dall'Asta, and Ginestra Bianconi. "Network controllability is determined by the density of low in-degree and out-degree nodes". In: *Physical review letters* 113.7 (2014), p. 078701.
- Jose C Nacher and Tatsuya Akutsu. "Dominating scale-free networks with variable scaling exponent: heterogeneous networks are not difficult to control". In: *New Journal of Physics* 14.7 (2012), p. 073005.

Scale-FreeBack