Evolving networks, an introduction

Richie Burke

October 16, 2015

Networks

"The important question is to explain how the interaction of a great number of people, each possessing only limited knowledge, will bring about an order that could only be achieved by deliberate direction taken by somebody who has the combined knowledge of all these individuals".

-Friedrich A. Hayek (1979)

Applications

The study of large systems of interacting agents has found application in many diverse fields such as

Applications

The study of large systems of interacting agents has found application in many diverse fields such as

- social networks

Applications

The study of large systems of interacting agents has found application in many diverse fields such as

- social networks
- ecology

Applications

The study of large systems of interacting agents has found application in many diverse fields such as

- social networks
- ecology
- neuroscience

Applications

The study of large systems of interacting agents has found application in many diverse fields such as

- social networks
- ecology
- neuroscience
- unmanned air-vehicle control

Applications

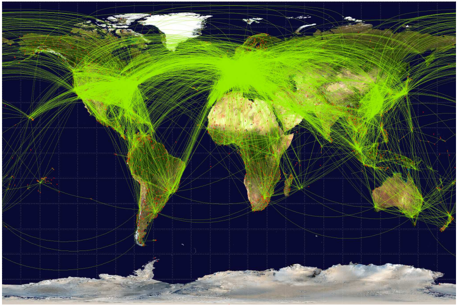
The study of large systems of interacting agents has found application in many diverse fields such as

- social networks
- ecology
- neuroscience
- unmanned air-vehicle control
- consensus problems

Applications

The study of large systems of interacting agents has found application in many diverse fields such as

- social networks
- ecology
- neuroscience
- unmanned air-vehicle control
- consensus problems

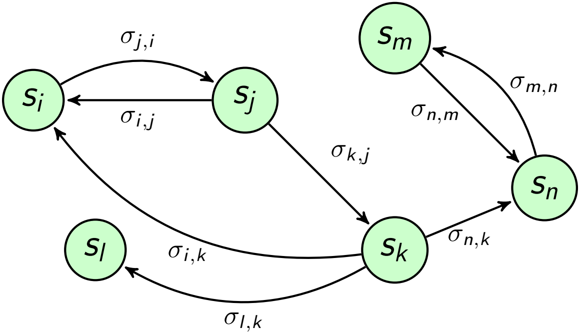


Mathematical networks

A *network* is a weighted graph, that is, a set of elements called *nodes* or *vertices*, which may be connected to one another via relational links (*edges*). To each node we assign a *state* s_i and to each edge a weight (or *gain*), $\sigma_{i,j}$.

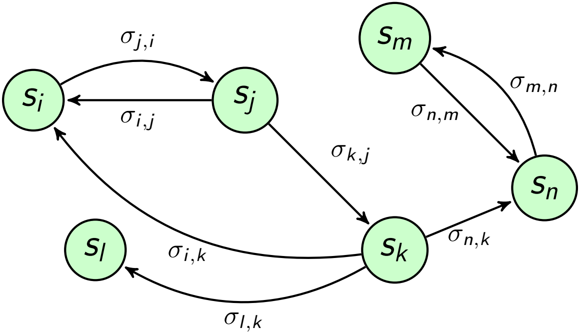
Mathematical networks

A *network* is a weighted graph, that is, a set of elements called *nodes* or *vertices*, which may be connected to one another via relational links (*edges*). To each node we assign a *state* s_i and to each edge a weight (or *gain*), $\sigma_{i,j}$.



Mathematical networks

A *network* is a weighted graph, that is, a set of elements called *nodes* or *vertices*, which may be connected to one another via relational links (*edges*). To each node we assign a *state* s_i and to each edge a weight (or *gain*), $\sigma_{i,j}$.



We want our states and gains to evolve until *consensus* is achieved.

The evolving states and gains could be exemplified by

- hyperlinks between webpages (gains) emerging when websites (nodes) share a common theme (state).

The evolving states and gains could be exemplified by

- hyperlinks between webpages (gains) emerging when websites (nodes) share a common theme (state).
- influence between fish in a shoal (gains) when one fish (node) changes position (state).

The evolving states and gains could be exemplified by

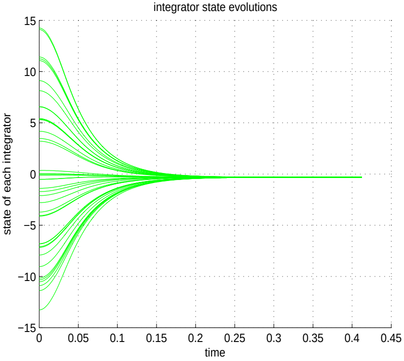
- hyperlinks between webpages (gains) emerging when websites (nodes) share a common theme (state).
- influence between fish in a shoal (gains) when one fish (node) changes position (state).
- friendships (gains) growing or deteriorating as people (nodes) cheer or vex one another.

Consensus

Consensus occurs when our node states evolve to a common value.

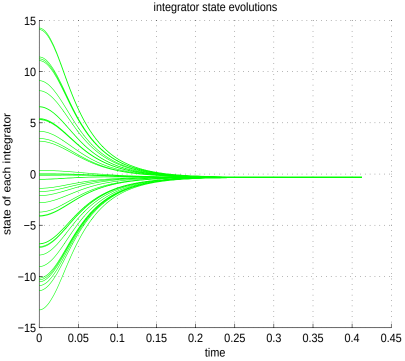
Consensus

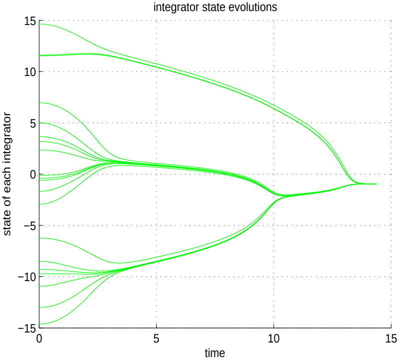
Consensus occurs when our node states evolve to a common value.



Consensus

Consensus occurs when our node states evolve to a common value.





Differential equations

The state and gain evolutions are governed by a system of coupled differential equations. The general form being:

$$\underbrace{\frac{ds_i}{dt}}_{\text{state evolution}} = \underbrace{\xi(\sigma_{i,j}, s_i, s_j)}_{\text{influence of gains and other nodes}} .$$

$$\underbrace{\frac{d\sigma_{i,j}}{dt}}_{\text{gain evolution}} = \underbrace{\psi(s_i, s_j)}_{\text{state dependence}} .$$

Differential equations

The state and gain evolutions are governed by a system of coupled differential equations. The general form being:

$$\underbrace{\frac{ds_i}{dt}}_{\text{state evolution}} = \underbrace{\xi(\sigma_{i,j}, s_i, s_j)}_{\text{influence of gains and other nodes}} .$$

$$\underbrace{\frac{d\sigma_{i,j}}{dt}}_{\text{gain evolution}} = \underbrace{\psi(s_i, s_j)}_{\text{state dependence}} .$$

Let's now consider a particular evolution model.

Switch protocol

The *switch protocol* model allows the gains between nodes to grow until a certain proscribed threshold is reached, whereupon the gains (effectively) lock into that value. The states meanwhile, pull each other (via the gains) until a *consensus* is attained,

Switch protocol

The *switch protocol* model allows the gains between nodes to grow until a certain proscribed threshold is reached, whereupon the gains (effectively) lock into that value. The states meanwhile, pull each other (via the gains) until a *consensus* is attained, i.e. the network nodes are all doing the same thing.

Switch protocol

The *switch protocol* model allows the gains between nodes to grow until a certain proscribed threshold is reached, whereupon the gains (effectively) lock into that value. The states meanwhile, pull each other (via the gains) until a *consensus* is attained, i.e. the network nodes are all doing the same thing.

Focussing on the gain evolutions, the switch protocol gains evolve according to

$$\frac{d\sigma_{i,j}}{dt} =$$

Switch protocol

The *switch protocol* model allows the gains between nodes to grow until a certain proscribed threshold is reached, whereupon the gains (effectively) lock into that value. The states meanwhile, pull each other (via the gains) until a *consensus* is attained, i.e. the network nodes are all doing the same thing.

Focussing on the gain evolutions, the switch protocol gains evolve according to

$$\frac{d\sigma_{i,j}}{dt} = \begin{cases} \alpha h(s_i, s_j) e^{-\beta h(s_i, s_j)} & \text{if } \sigma_{i,j} < \tau, \\ 0 & \text{if } \sigma_{i,j} \geq \tau. \end{cases} \quad (1)$$

Switch protocol

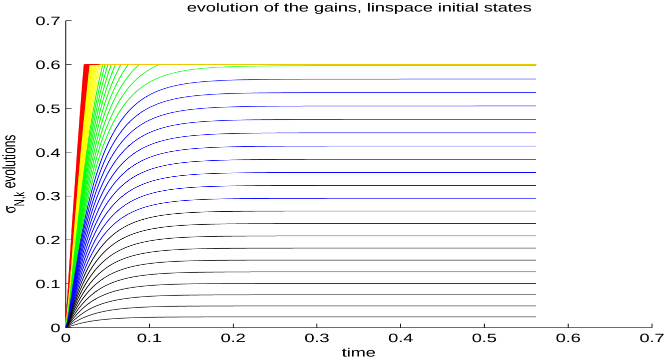
The *switch protocol* model allows the gains between nodes to grow until a certain proscribed threshold is reached, whereupon the gains (effectively) lock into that value. The states meanwhile, pull each other (via the gains) until a *consensus* is attained, i.e. the network nodes are all doing the same thing.

Focussing on the gain evolutions, the switch protocol gains evolve according to

$$\frac{d\sigma_{i,j}}{dt} = \begin{cases} \alpha h(s_i, s_j) e^{-\beta h(s_i, s_j)} & \text{if } \sigma_{i,j} < \tau, \\ 0 & \text{if } \sigma_{i,j} \geq \tau. \end{cases} \tag{1}$$

where α and β are rate parameters, h is some norm of the states and τ is the threshold where we want the gains to cease evolving.

The switch protocol gains grow and level off when the respective states come together. Notice the gains are capped by the threshold parameter τ. Here $\tau = 0.6$

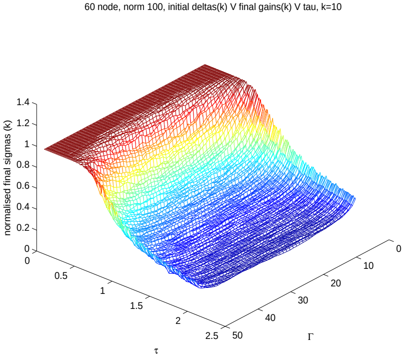


Simulations

We generate networks using **Matlab** and investigate the evolutions of the states, gains and various features of the systems at consensus.

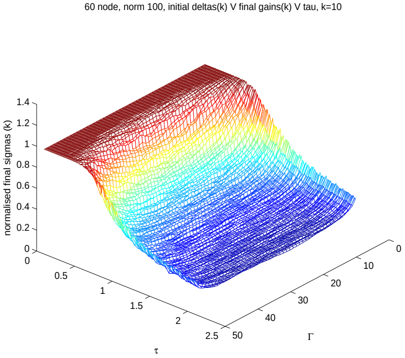
Simulations

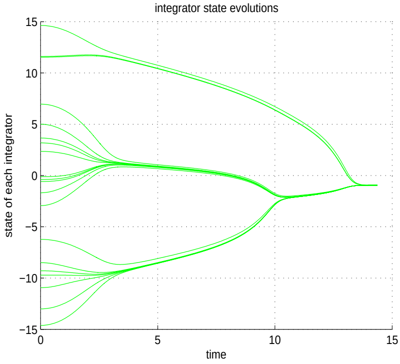
We generate networks using **Matlab** and investigate the evolutions of the states, gains and various features of the systems at consensus.



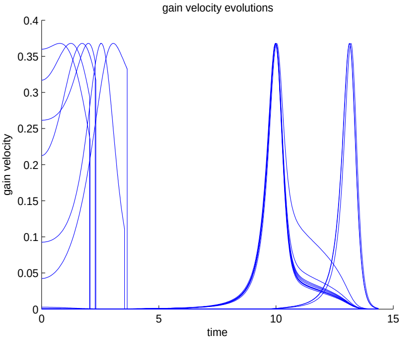
Simulations

We generate networks using **Matlab** and investigate the evolutions of the states, gains and various features of the systems at consensus.



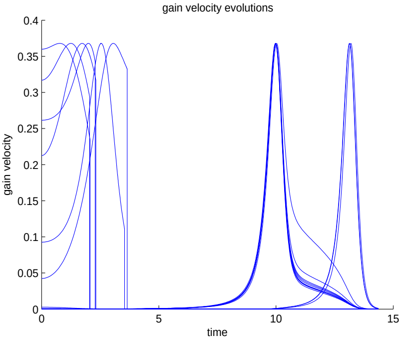


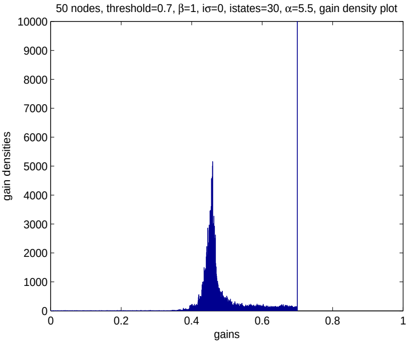
Gain evolutions



The maximum gain velocity for the *switch protocol* is capped at $\frac{\alpha}{\beta e}$.

Gain evolutions





The maximum gain velocity for the *switch protocol* is capped at $\frac{\alpha}{\beta e}$.

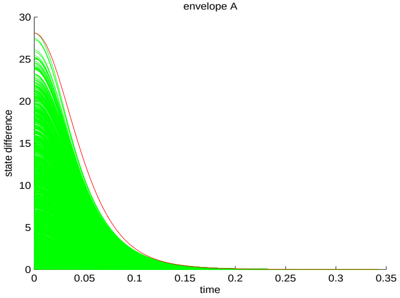
Bimodal gain distributions are often observed for localised systems.

Reduced order approximation

We have built a qualitative *envelope* to track the convergence of our large systems.

Reduced order approximation

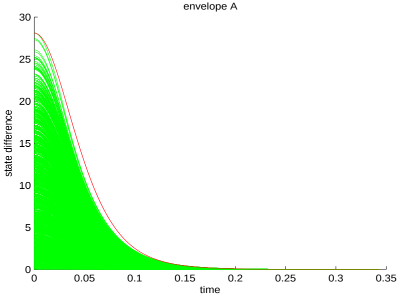
We have built a qualitative *envelope* to track the convergence of our large systems.



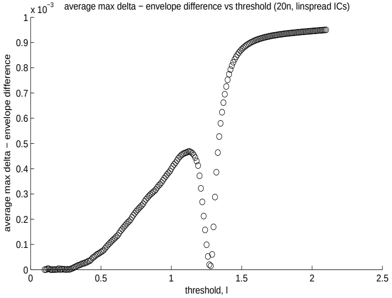
Envelope A (red), all other errors (green).

Reduced order approximation

We have built a qualitative *envelope* to track the convergence of our large systems.



Envelope A (red), all other errors (green).



Absolute difference between (actual) max error and envelope.

References

- "Evolution of Complex Networks via Edge Snapping", P. DeLellis, M. diBernardo, F. Garofalo and M. Porfiri, IEEE Transactions on circuits and systems-1: regular papers, vol. 57, pp. 2132-2143 (2010).
- "Complex Networks: Structure and Dynamics", S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, D. U. Hwang, Physics Reports 424 pp. 175308 (2006).
- "The structure and function of complex networks", M. E. J. Newman, SIAM Rev.45, pp. 167-256 (2003).
- "Controllability of complex networks", Y-Y. Liu, J.J. Slotine and A-L. Barabsi, Nature 473, pp. 167173 (2011).