Network Science (VU) (706.703)

Dynamical Systems on Networks

Denis Helic

ISDS, TU Graz

January 7, 2021

Outline

1. Introduction
2. Example: Diffusion
3. Stability and Connection Between Structure and Dynamics

Dynamics on networks

- We model the dynamics on networks with independent dynamical variables $x_i, y_i, \ldots$ on each node i
- The variables are only coupled along links in the network
- In other words, x_i depends only on x_i, other variables on node i and on variables on adjecent nodes
- For simplicity, we will discuss only the case of a single variable x_i on each node i

Dynamics on networks

- For example, the network version of the SI model:

$$\frac{dx_i}{dt} = \dot{x}_i = \beta(1 - x_i) \sum_j A_{ij} x_j$$

- This equation has only terms involving pairs of variables connected by links

General first-order equation

- For a system with a single variable on each node we can write a general equation:

$$\dot{x}_i = f_i(x_i) + \sum_j A_{ij} g_{ij}(x_i, x_j)$$

- We separated the terms that involve adjacent nodes from those that do not
- f_i specifies the intrinsic dynamics of a node - it specifies how x_i would evolve without any connections
- g_{ij} specifies the contribution from the neighbors

General first-order equation

- We specified different functions f_i and g_{ij} for each node and link
- The dynamics can differ from one node to another
- In many cases the dynamics is the same or similar for each node
- In such cases we might ignore the differences and use the same functions for each node:

$$\dot{x}_i = f(x_i) + \sum_j A_{ij} g(x_i, x_j)$$

General first-order equation

- We will also assume the case of an undirected graph
- If x_i is affected by x_j then x_j is similarly affected by x_i
- Although in general case g is not symmetric, i.e. $g(x_i, x_j) \neq g(x_j, x_i)$
- Again, the SI model is an example of a system of this kind:

$$
\begin{aligned}
f(x) &= 0 \\
g(x_i, x_j) &= \beta(1 - x_i)x_j
\end{aligned}
$$

Diffusion

- Diffusion might be defined as a process of moving an entity from the regions of high density to regions of low density
- Simple model of spread across a network
- The spread of an idea
- The spread of an initiative
- In physics the movement of gas from a region of high density to regions of low density

Diffusion

- Suppose we have some quantity on the nodes
- E.g. the content of information in a social network
- E.g. the knowledge level of people in a university social network
- E.g. the number of people at different places in a city
- E.g. the amount of water in water tanks
- Let us denote the amount of this quantity at node i with x_i

Diffusion

- Suppose that this commodity moves from one node to another along the links at a constant rate C
- E.g. if i and j are adjacent to each other then the amount flowing from j to i in some small time interval dt:

$$C(x_j - x_i)$$

- Similarly, the amount flowing from i to j is given by (which is the negative amount from above):

$$C(x_i - x_j)$$

Diffusion

- Then the total amount of commodity flowing to i is given by the same of flows over all of i's neighbors:

$$\dot{x}_i = C \sum_j A_{ij}(x_j - x_i)$$

Diffusion

- By splitting the terms on the right side we obtain:

$$\begin{aligned} \dot{x}_i &= C\sum_j A_{ij}x_j - Cx_i\sum_j A_{ij} = C\sum_j A_{ij}x_j - Cx_ik_i \\ &= C\sum_j (A_{ij} - \delta_{ij}k_i)x_j \end{aligned}$$

- In matrix form we obtain a so-called diffusion equation:

$$\dot{\mathbf{x}} = \mathrm{C}(\mathbf{A} - \mathbf{D})\mathbf{x} = -\mathrm{C}\mathbf{L}\mathbf{x}$$

Python Notebook

- Check diffusion examples from python notebook
- http://kti.tugraz.at/staff/denis/courses/netsci/ndynamics.ipynb

Diffusion: analytical solution

$$\dot{\mathbf{x}} = -C\mathbf{L}\mathbf{x}$$

- Linear system, which we know how to solve!
- Let us write $\mathbf{x}$ as the linear combination of the eigenvectors of the Graph Laplacian
- $\mathbf{v}_r$ is the eigenvector with eigenvalue λ_r

$$\mathbf{x}(t) = \sum_{r=1}^{n} a_r(t)\mathbf{v}_r$$

Diffusion: analytical solution

- Now coefficients $a_r(t)$ vary over time
- Substituting this form in the diffusion equation:

$$\sum_{r=1}^{n} \frac{da_r(t)}{dt}\mathbf{v}_r = -C\mathbf{L}\sum_{r=1}^{n} a_r(t)\mathbf{v}_r$$

Diffusion: analytical solution

- We have $\mathbf{L}\mathbf{v}_r = \lambda_r \mathbf{v}_r$:

$$\sum_{r=1}^{n} \frac{da_r(t)}{dt}\mathbf{v}_r = -C\sum_{r=1}^{n} a_r(t)\lambda_r \mathbf{v}_r$$

- Now we multiply both sides with any eigenvector v_s of the Laplacian
- Recollect that since $\mathbf{L}$ is symmetric its eigenvectors are orthogonal to each other

Diffusion: analytical solution

$$\sum_{r=1}^{n} \frac{da_r(t)}{dt} \mathbf{v}_r \mathbf{v}_s = -C \sum_{r=1}^{n} a_r(t) \lambda_r \mathbf{v}_r \mathbf{v}_s$$

- $\mathbf{v}_r \mathbf{v}_s = 0$ if $s \neq r$
- $\mathbf{v}_r \mathbf{v}_s = 1$ if $s = r$

$$\frac{da_r(t)}{dt} = -C a_r(t) \lambda_r$$

Diffusion: analytical solution

- The last equation has the solution (by separating variables and integrating):

$$a_r(t) = a_r(0)e^{-C\lambda_r t}$$

Eigenvalues of the graph Laplacian

- The vector $\mathbf{1}$ is always an eigenvector of $\mathbf{L}$ with eigenvalue 0
- There are no negative eigenvalues, thus this is the lowest eigenvalue
- Convention: $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_n$
- We always have $\lambda_1 = 0$

Diffusion: analytical solution

$$a_0(t) = a_0(0)e^{-C \cdot 0 \cdot t} = a_0(0)$$

- All other coefficients decay exponentially and tend to zero for large t
- Thus, for fixed point:

$$\mathbf{x} = a_0(0)\mathbf{v}_0 = a_0(0)\mathbf{1} = \begin{pmatrix} a_0(0) \\ a_0(0) \\ \vdots \\ a_0(0) \end{pmatrix}$$

Linear stability analysis

- Suppose we are able to find a fixed point $\{x_i^*\}$ by solving the system of equations (for all i):

$$f(x_i^*) + \sum_j A_{ij} g(x_i^*, x_j^*) = 0$$

- In other words, we find the value on each node i for which none of the values at any node changes
- Clearly, the position of the fixed point depends on the dynamics but also on the structure

Linear stability analysis

- Now we linearize around the fixed point by writing $x_i = x_i^* + \epsilon_i$
- We perform multiple Taylor expansion in all variables simultaneously, and drop terms of second and higher orders

$$
\begin{aligned}
\frac{dx_i}{dt} &= \frac{d\epsilon_i}{dt} = f(x_i^* + \epsilon_i) + \sum_j A_{ij} g(x_i^* + \epsilon_i, x_j^* + \epsilon_j) = \\
&= f(x_i^*) + \epsilon_x \left(\frac{\partial f}{\partial x}\right)_{x=x_i^*} + \sum_j A_{ij} g(x_i^*, x_j^*) \\
&+ \epsilon_i \sum_j A_{ij} \left(\frac{\partial g(u,v)}{\partial u}\right)_{\substack{u=x_i^* \\ v=x_j^*}} + \sum_j A_{ij} \epsilon_j \left(\frac{\partial g(u,v)}{\partial v}\right)_{\substack{u=x_i^* \\ v=x_j^*}} \\
&+ O(\epsilon_i^2) + O(\epsilon_j^2) \\
&= \epsilon_x \left(\frac{\partial f}{\partial x}\right)_{x=x_i^*} + \epsilon_i \sum_j A_{ij} \left(\frac{\partial g(u,v)}{\partial u}\right)_{\substack{u=x_i^* \\ v=x_j^*}} + \sum_j A_{ij} \epsilon_j \left(\frac{\partial g(u,v)}{\partial v}\right)_{\substack{u=x_i^* \\ v=x_j^*}} \\
&+ O(\epsilon_i^2) + O(\epsilon_j^2)
\end{aligned}
$$

Linear stability analysis

- If we know the position of the fixed point then the derivatives are just numbers
- Let us write:

$$
\begin{aligned}
\alpha_i &= \left(\frac{\partial f}{\partial x}\right)_{x=x_i^*} \\
\beta_{ij} &= \left(\frac{\partial g(u,v)}{\partial u}\right)_{\substack{u=x_i^* \\ v=x_j^*}} \\
\gamma_{ij} &= \left(\frac{\partial g(u,v)}{\partial v}\right)_{\substack{u=x_i^* \\ v=x_j^*}}
\end{aligned}
$$

Linear stability analysis

- We obtain then:

$$\dot{\epsilon}_i = \left[\alpha_i + \sum_j \beta_{ij} A_{ij}\right] \epsilon_i + \sum_j \gamma_{ij} A_{ij} \epsilon_j$$

$$\begin{aligned} \dot{\boldsymbol{\epsilon}}_i &= \mathbf{M}\boldsymbol{\epsilon} \\ M_{ij} &= \delta_{ij}\left[\alpha_i + \sum_j \beta_{ij} A_{ij}\right] + \sum_j \gamma_{ij} A_{ij} \end{aligned}$$

Linear stability analysis

- Linear system, which we know how to solve!
- Again, we can solve the last equation by writing ϵ as a linear combination of the eigenvectors of $\mathbf{M}$
- $\mathbf{v}_r$ is the eigenvector with eigenvalue λ_r

$$\epsilon(t) = \sum_{r=1}^{n} a_r(t)\mathbf{v}_r$$

Linear stability analysis

- Now coefficients $a_r(t)$ vary over time
- Substituting this form in the diffusion equation:

$$\sum_{r=1}^{n} \frac{da_r(t)}{dt}\mathbf{v}_r = \mathbf{M}\sum_{r=1}^{n} a_r(t)\mathbf{v}_r = \sum_{r=1}^{n} a_r(t)\lambda_r\mathbf{v}_r$$

Linear stability analysis

- The last equation has the solution (by comparing terms in each eigenvector and separating variables and integrating):

$$a_r(t) = a_r(0)e^{\lambda_r t}$$

- The same discussion applies as in a non-network case
- However, eigenvalues of $\mathbf{M}$ depend on both: dynamics and the structure

Special case

- A simple case arises when the fixed point is symmetric, i.e. $x_i^* = x^*$ for all nodes i
- The derivatives then evaluate to the same value for each node i:

$$
\begin{aligned}
\alpha_i &= \left(\frac{\partial f}{\partial x}\right)_{x=x^*} = \alpha \\
\beta_{ij} &= \left(\frac{\partial g(u,v)}{\partial u}\right)_{u,v=x^*} = \beta \\
\gamma_{ij} &= \left(\frac{\partial g(u,v)}{\partial v}\right)_{u,v=x^*} = \gamma
\end{aligned}
$$

Special case

- Then the linear expansion around the fixed point is given by:

$$\dot{\epsilon}_i = (\alpha + \beta k_i)\epsilon_i + \gamma \sum_j A_{ij}\epsilon_j$$

- Suppose also that the coupling function g is given by $g(x_i, x_j) = g(x_i) - g(x_j)$

Special case

- The derivatives then evaluate to:

$$
\begin{aligned}
\alpha_i &= \left(\frac{\partial f}{\partial x}\right)_{x=x^*} = \alpha \\
\beta_{ij} &= \left(\frac{\partial g(u,v)}{\partial u}\right)_{u,v=x^*} = \left(\frac{\partial g}{\partial x}\right)_{x=x^*} = \beta \\
\gamma_{ij} &= \left(\frac{\partial g(u,v)}{\partial v}\right)_{u,v=x^*} = \left(\frac{\partial g}{\partial x}\right)_{x=x^*} = -\beta
\end{aligned}
$$

Special case

- Then the linear expansion around the fixed point is given by:

$$\dot{\epsilon}_i = \alpha\epsilon_i + \beta\sum_j (k_i\delta_{ij} - A_{ij})\epsilon_j$$

- In matrix form we have:

$$\dot{\boldsymbol{\epsilon}} = (\alpha\mathbf{I} + \beta\mathbf{L})\boldsymbol{\epsilon}$$

Special case

- The exponent depends on the dynamics and the eigenvalues of the graph Laplacian
- The eigenvalues must satisfy:

$$\alpha + \beta\lambda_r < 0$$

- $\lambda_1 = 0$, all other are greater than zero
- Thus, it is necessary that $\alpha < 0$
- Additionally:

$$\frac{1}{\lambda_n} > -\frac{\beta}{\alpha}$$

Example

- Population growth ($r > 0$) with migration ($a > 0$)

$$
\begin{aligned}
f(x_i) &= rx_i(1 - x_i) \\
g(x_i, x_j) &= g(x_i) - g(x_j) \\
g(x_i) &= ax_i^2
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial f}{\partial x} &= r - 2rx \\
\frac{\partial g}{\partial x} &= 2ax
\end{aligned}
$$

- Two symmetric fixed points $x_1^* = 0, x_2^* = 1$

Example

- $x_1^* = 0$
- The derivatives then evaluate to:

$$\begin{aligned} \alpha &= r \\ \beta &= 0 \end{aligned}$$

- Always unstable (population starts to reproduce)

Example

- $x_1^* = 1$
- The derivatives then evaluate to:

$$
\begin{aligned}
\alpha &= -r \\
\beta &= 2a
\end{aligned}
$$

Example

- The stability equation becomes:

$$
\begin{aligned}
\frac{1}{\lambda_n} &> -\frac{\beta}{\alpha} = \frac{2a}{r} \\
\lambda_n &< \frac{r}{2a}
\end{aligned}
$$

Python Notebook

- Check population growth examples from python notebook
- http://kti.tugraz.at/staff/denis/courses/netsci/ndynamics.ipynb

Example

- Gossip, or diffusion of an idea
- Some idea is circulating around a social network
- x_i represent the amount person i is talking about the idea

$$
\begin{aligned}
f(x) &= a(1-x) \\
g(x_i, x_j) &= g(x_i) - g(x_j) \\
g(x) &= \frac{bx}{1+x}
\end{aligned}
$$

- Symmetric fixed point $x^* = 1$

Example

- $f(x)$ is the intrinsic tendency to talk
- Regardless of their friends
- $g(x)$ represents copying of the friends
- If their friends talk more than people tend to talk more

Example

- The derivatives then evaluate to:

$$
\begin{aligned}
\alpha &= \left(\frac{\partial f}{\partial x}\right)_{x=x^*} = -a \\
\beta &= \left(\frac{\partial g(u,v)}{\partial u}\right)_{u,v=x^*} = \left(\frac{\partial g}{\partial x}\right)_{x=x^*} = \frac{b}{4}
\end{aligned}
$$

Example

- The stability equation becomes:

$$
\begin{aligned}
\frac{1}{\lambda_n} &> -\frac{\beta}{\alpha} = \frac{b}{4a} \\
\lambda_n &< \frac{4a}{b}
\end{aligned}
$$

Python Notebook

- Check gossip examples from python notebook
- http://kti.tugraz.at/staff/denis/courses/netsci/ndynamics.ipynb

Example

- Thus, when we compromise the stability by increasing the influence of the friends
- We arrive at another fixed point, which is not symmetric
- When the influence of friends becomes to strong everybody is doing the things her own way
- Contrary, to what we expect that everybody behaves in the same way