

Todo list

Write abstract	2
Write introduction	2
Is this section too precise and should the results be given with just a citation ?	3
Introduce 1D variables more clearly	5
Missing reference	6
Missing reference	6
Missing reference	6
Missing reference	6
Find back which one exactly	7
Missing reference	7
Missing reference	8
Choose what multiplex networks should be used and with what parameters and actually produce the results.	8
Write discussion	8
Make the plot more readable and less ugly	9
Find why the two methods seems drift away one from the other far from the center.	9

Boundary condition

Benoît Richard Guiyuan Shi

August 9, 2018

Abstract

Write abstract

1 Introduction

Write introduction

2 Giant viable cluster

Consider a multiplex network with L layers. Let $g_0^{(i)}$ and $g_1^{(i)}$ be the generating functions of respectively the degree and the excess degree in layer i . Moreover define u_i as the probability that a vertex reached after following an edge in layer i is not part of the giant viable cluster. Then if we pick a vertex v at random the probability S that it is part of the giant viable cluster can be written as

$$S = P_0 \left(\bigcap_{i=1}^L \exists w \in N_i(v) \ w \in GVC \right). \quad (1)$$

By requiring that the layers are independent from one others, we can rewrite S as a product

$$S = \prod_{i=1}^L P_0 (\exists w \in N_i(v) \ w \in GVC) \quad (2)$$

$$= \prod_{i=1}^L [1 - P(w \notin GVC \ \forall w \in N_i(v))] \quad (3)$$

$$= \prod_{i=1}^L \left[1 - \sum_{k=0}^{\infty} P((w \notin GVC \ \forall w \in N_i(v) | \deg(v) = k) p_k^{(i)} \right] \quad (4)$$

$$= \prod_{i=1}^L \left[1 - \sum_{k=0}^{\infty} u_i^k p_k^{(i)} \right] \quad (5)$$

$$= \prod_{i=1}^L [1 - g_0^{(i)}(u_i)]. \quad (6)$$

{Multiplex GCC size final}

We can find u_j by a similar reasoning. First note that $1 - u_j$ is the probability that a vertex reached by following an edge in layer j is in the

giant viable cluster. Which as before can be written in the form

$$1 - u_j = P_1^{(j)} \left(\bigcap_{i=1}^L \exists w \in N_i(v) \ w \in GVC \right) \quad (7)$$

$$= \prod_{i=1}^L P_1^{(j)} (\exists w \in N_i(v) \ w \in GVC). \quad (8)$$

Since the layers are independent, the fact that we reached v by following an edge in layer j to reach vertex v is irrelevant in all other layers. However in layer j this means that the degree distribution follows the distribution $q_k^{(j)}$ rather than $p_k^{(j)}$. Putting this together we get

$$1 - u_j = \left[1 - \sum_{k=0}^{\infty} u_j^k q_k^{(j)} \right] \prod_{\substack{i=1 \\ i \neq j}}^L \left[1 - \sum_{k=0}^{\infty} u_i^k p_k^{(i)} \right] \quad (9)$$

$$= \left[1 - g_1^{(j)}(u_j) \right] \prod_{\substack{i=1 \\ i \neq j}}^L \left[1 - g_0^{(i)}(u_i) \right]. \quad (10)$$

{Multiplex u final}

Is this section too precise and should the results be given with just a citation ?

3 Boundary condition

3.1 General case

Up to now, we have considered the multiplex generated to be determined via the degree distributions of each of its layer. However a degree distribution has an infinite number of degrees of freedom, therefore it is more practical to let the degree distributions depend on a finite set of parameters $\lambda_1, \lambda_2, \dots, \lambda_N$ and express the behaviour of the network in term of them. Note that the number of parameters N does not need to match the number of layers L .

In order to make our main statement about the critical region for a multiplex network, we first need to introduce several quantities. First, let introduce

$$\mathbf{u} = (u_1, u_2, \dots, u_L) \quad (11)$$

$$\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \dots, \lambda_N) \quad (12)$$

$$f_j(\boldsymbol{\lambda}, \mathbf{u}) = 1 - u_j - \left[1 - g_1^{(j)}(u_j) \right] \prod_{\substack{i=1 \\ i \neq j}}^L \left[1 - g_0^{(i)}(u_i) \right] \quad (13)$$

{Definition fj}

and the function

$$F : \mathbb{R}^N \times \mathcal{I}^L \rightarrow \mathbb{R}^L \quad (14)$$

$$(\boldsymbol{\lambda}, \mathbf{u}) \mapsto F(\boldsymbol{\lambda}, \mathbf{u}) = (f_1(\boldsymbol{\lambda}, \mathbf{u}), f_2(\boldsymbol{\lambda}, \mathbf{u}), \dots, f_L(\boldsymbol{\lambda}, \mathbf{u})), \quad (15)$$

{Definition F}

where $\mathcal{I} = [0, 1]$. The variables \mathbf{u} in which we are interested are always in \mathcal{I}^L , since u_i represents a probability for all i .

Since the functions $g_0^{(i)}$ and $g_1^{(i)}$ are analytic with respect to the u_i , the function

$$F_{\lambda} : \mathcal{I}^L \rightarrow \mathbb{R}^L \quad (16)$$

$$\mathbf{u} \mapsto F_{\lambda}(\mathbf{u}) = F(\lambda, \mathbf{u}), \quad (17)$$

is continuously differentiable for all parameters λ . Therefore we can define Jacobi matrix $J(\lambda, \mathbf{u})$ of F_{λ} as having coefficients

$$[J(\lambda, \mathbf{u})]_{ij} = \frac{\partial f_i(\lambda, \mathbf{u})}{\partial u_j}. \quad (18)$$

With the help of the notation introduced, we can now express solving eq. (10) as being equivalent to finding $\mathbf{u}^* \in \mathcal{I}^L$ such that

$$F(\lambda, \mathbf{u}^*) = 0. \quad (19)$$

{Implicit equation}

If this equation only admits the trivial solution $\mathbf{u}^* = \mathbf{u}_T$, the parameter λ corresponds to a state without GVC. On the other if multiple solutions \mathbf{u}^* exist, a GVC must exist as well. To determine the boundary between these two regions (i.e. the critical region), we use the implicit function theorem.

First, we assume that F (and not only F_{λ}) is continuously differentiable and that we know a solution \mathbf{u}^* of (19) for some parameter vector λ^* . With that assumption the implicit function theorem can be state as follow:

If $\det[J(\lambda^*, \mathbf{u}^*)] \neq 0$ then there is an open neighbourhood $U \subset \mathbb{R}^L$ of λ^* such that there is an unique continuously differentiable function $h : U \rightarrow \mathcal{I}^L$ with

$$h(\lambda^*) = \mathbf{u}^* \quad (20)$$

$$F(\lambda, h(\lambda)) = 0, \quad \forall \lambda \in U. \quad (21)$$

{Implicit solution for F}

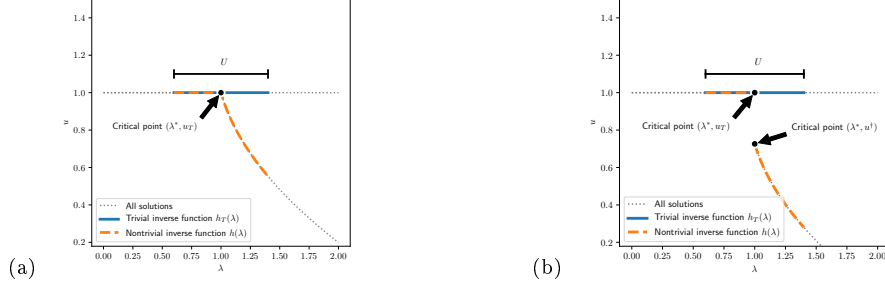
The result in which we are interested here comes from the contrapositive of this statement, namely that if for all neighbourhoods U we can not find a uniquely defined continuous function h , then the determinant of the Jacobi matrix $J(\lambda, \mathbf{u})$ must be zero,

$$\det[J(\lambda^*, \mathbf{u}^*)] = 0. \quad (22)$$

{Boundary condition}

We now prove that such situations arise if λ^* is a critical point of the phase transition between the absence and existence of a GVC, and therefore that eq. (22) is a sufficient condition to find the critical region of such phase transition.

First notice that in the context of multiplex network a phase transition appears between the trivial solution $\mathbf{u}_T = (1, 1, \dots, 1)$ (where $S = 0$) and non trivial solutions ($S > 0$). However, the trivial solution \mathbf{u}_T solves eq. (10) for any generating function and thus for any parameter vector λ . For a continuous phase transition this immediately gives us $\mathbf{u}^* = \mathbf{u}_T$. Moreover, on one side of the phase transition occurring at λ^* one solution exists, while on the other at least two do. Therefore for any U open containing λ^* we can define two distinct functions on U that fulfil eq. (21), the trivial $h_T(\lambda) = \mathbf{u}_T$ and another function h corresponding to the non trivial solutions, with $h(\lambda^*) = h_T(\lambda^*) = \mathbf{u}_T$. So the function h of the implicit function theorem is not uniquely defined and thus $\det[J(\lambda^*, \mathbf{u}_T)] = 0$.



{Figure: Scheme of continuous and di

Figure 1: (a) Scheme of a continuous phase transition. (b) Scheme of a discontinuous phase transition.

On the other hand, let consider a discontinuous phase transition at λ^* . For any neighbourhood U of λ^* there are two sequences $(\lambda_n, \mathbf{u}_n)$ and (η_m, \mathbf{v}_m) with $\lambda_n, \eta_m \in U$ such that

$$\lim_{n \rightarrow \infty} (\lambda_n, \mathbf{u}_n) = (\lambda^*, \mathbf{u}^\dagger) \quad \text{avec } \mathbf{u}^\dagger \neq \mathbf{u}_T \quad (23)$$

$$\lim_{m \rightarrow \infty} (\eta_m, \mathbf{v}_m) = (\lambda^*, \mathbf{u}_T) \quad (24)$$

$$F(\lambda_n, \mathbf{u}_n) = 0 \quad \forall n \quad (25)$$

$$F(\eta_m, \mathbf{v}_m) = 0 \quad \forall m. \quad (26)$$

If we assume that an unique continuous function h solving eq. (21) exists, we would have

$$h(\lambda_n) = \mathbf{u}_n \quad \forall n \quad (27)$$

$$h(\eta_m) = \mathbf{v}_m \quad \forall m. \quad (28)$$

The continuity of h would furthermore imply

$$h(\lambda^*) = \lim_{n \rightarrow \infty} h(\lambda_n) = \lim_{n \rightarrow \infty} \mathbf{u}_n = \mathbf{u}^\dagger, \quad (29)$$

but also

$$h(\lambda^*) = \lim_{m \rightarrow \infty} h(\eta_m) = \lim_{m \rightarrow \infty} \mathbf{v}_m = \mathbf{u}_T. \quad (30)$$

Since $\mathbf{u}_T \neq \mathbf{u}^\dagger$, this gives raise to the contradiction $h(\lambda^*) \neq h(\lambda^*)$. Therefore our assumption must be false and no continuous function h can be defined to solve eq. (21). So finally, we have $\det[J(\lambda^*, \mathbf{u}^*)] = 0$, \mathbf{u}^* being either \mathbf{u}_T or \mathbf{u}^\dagger .

3.2 One dimensional case

Introduce 1D variables more clearly

If $L = N = 1$, the problem is the classical problem of a one layer network which degree distribution is determined by a single parameter λ . In that case the Jacobi matrix J reduces to the scalar quantity

$$J(\lambda, u) = \frac{\partial}{\partial u} (g_1(u) - u) = \frac{\partial g_1(u)}{\partial u} - 1. \quad (31)$$

Therefore the condition for the boundary $\det J(\lambda, u) = 0$ becomes

$$\frac{\partial g_1(u)}{\partial u} = 1. \quad (32)$$

This condition was already introduced and verified previously by[?].

Missing ref

4 Interval estimation of the critical region

4.1 Theoretical foundation

Let $C \subset \mathcal{R}^L$ be the set of all parameters λ corresponding to a critical point. This set correspond to the parameters that solve simultaneously eq. (10) and eq. (22) for some $\mathbf{u} \in \mathcal{I}^L$. In this section we present an alternative and independent method to estimate C in order to verify that eq. (10) and (22) yield the expected result.

To do so we introduce concepts from *interval arithmetic*[?]. First of all an interval I is defined a set of the form

Missing ref

$$I = [a, b] = \{x \in \mathbb{R} | a \leq x \leq b\}. \quad (33)$$

The set of all intervals is denoted as \mathbb{IR} . The N -dimensional equivalent of an interval is an interval box B , defined as the Cartesian product of N intervals,

$$B = I_1 \times I_2 \times \cdots \times I_N, \quad I_k \in \mathbb{IR} \quad \forall k = 1, \dots, N \quad (34)$$

The set of all N -dimensional interval boxes is denoted \mathbb{IR}^N .

Given a function $\phi : \mathbb{R}^M \rightarrow \mathbb{R}^N$ it is possible[?] to determine a new interval valued function $\Phi : \mathbb{IR}^M \rightarrow \mathbb{IR}^N$ such that

Missing ref

$$x \in B \quad \Rightarrow \quad \phi(x) \in \Phi(B). \quad (35)$$

{Definition interval extension}

A function Φ with this property is called an interval extension of ϕ . To determine were the critical region is, we would like, in a sense, to solve eq. (10) and (22). Several general schemes exist to solve equations in a guaranteed way using intervals[?], but here we use a simpler algorithm inspired by them and more suited to our present needs.

Missing ref

We define

$$\psi(\lambda, \mathbf{u}) = F(\lambda, \mathbf{u}) + \mathbf{u}, \quad (36)$$

with components

$$\psi_j(\lambda, \mathbf{u}) = 1 - \left[1 - g_1^{(j)}(u_j)\right] \prod_{\substack{i=1 \\ i \neq j}}^L \left[1 - g_0^{(i)}(u_i)\right]. \quad (37)$$

Also we define Ψ as an interval extension of ψ .

Now, let $\mathbf{u}^* \in U_0$ be a solution of (19) for some $\lambda \in \Lambda$. By the definition of ψ and eq. (35),

$$F(\lambda^*, \mathbf{u}^*) = 0 \quad \Rightarrow \quad \mathbf{u}^* = \psi(\lambda^*, \mathbf{u}^*) \in \Psi(\Lambda, U_0). \quad (38)$$

Therefore if we apply $\Psi(\Lambda, \cdot)$ to an interval box U_0 containing a solution, the resulting interval box contains the solution as well. We know that all solutions \mathbf{u}^* are contained in \mathcal{I} by definition of \mathbf{u} and thus by iterating the

previous argument from $U_0 = \mathcal{I}$, all solutions are elements of the interval boxes $U_k(\Lambda)$ recursively defined by

$$U_{k+1}(\Lambda) = \Psi(\Lambda, U_k(\Lambda)). \quad (39)$$

{Recursion relation for U_k }

Therefore if we find k such that $U_k = \{\mathbf{u}_T\}$, we know that the system for any $\lambda \in \Lambda$ only admits the trivial solution.

In praxis however, the sequence U_k never converges to exactly the set $\{\mathbf{u}_T\}$, we therefore consider the criterion to be met if

$$U_k \subset [1 - \varepsilon, 1]^L, \quad (40)$$

{Criterion for trivial region}

for some small tolerance ε .

Furthermore it is possible in some cases to guarantee the presence of non trivial solutions, allowing to conclude that a GVC emerges. Indeed, if we can find some interval boxes Λ and U such that

$$\Psi(\Lambda, U) \subset U \quad \text{and} \quad \mathbf{u}_T \notin U, \quad (41)$$

{Criterion for non trivial solution}

then by definition of the interval extension (eq. (35)), we have

$$\psi(\lambda, U) \subset U, \quad \forall \lambda \in \Lambda. \quad (42)$$

Since U is closed and simply connected, the fixpoint theorem [?] applies, implying that for each $\lambda \in \Lambda$ there must be at least one \mathbf{u}^* in U such that \mathbf{u}^* is a fixpoint, or in other words such that $\mathbf{u}^* = \psi(\lambda, \mathbf{u}^*)$. Therefore eq. (41) is a sufficient condition to prove the existence of at least one solution. Moreover since we imposed $\mathbf{u}_T \notin U$, the solution present can not be the trivial one.

Find back
which one
exactly

Missing ref

4.2 Algorithm

Equations (40) and (41) give guaranteed criteria for respectively the absence and presence of a non trivial solution in the region Λ considered. This is sufficient to propose an algorithm to estimate the critical region C .

1. Choose a starting parameter region Λ_0 and store it in the set S_{work} of regions yet to be processed.
2. If S_{work} is empty terminate, otherwise retrieve the next parameter region from S_{work} , and call it Λ .
3. If the radius of Λ is smaller than some tolerance δ , store it in the set $S_{unknown}$ of regions for which the algorithm is unable to conclude using the tolerance δ .
4. Compute $U_k(\Lambda)$ for k big, using eq. (39).
5. If $U_k(\Lambda)$ fulfil eq. (40), store Λ in the set $S_{trivial}$ of trivial regions and go to 2.
6. Take a subset V of $U_k(\Lambda)$ such that $\mathbf{u}_T \notin V$.
7. If V fulfil eq. (41), store Λ in the set S_{GVC} of non trivial regions and go to 2.
8. Bisect Λ in two sub regions and add both to S_{work} . Go to 2.

By construction, when the algorithm terminates, we have for the critical region C

$$C \subset \bigcup_{U \in S_{unknown}} U, \quad (43)$$

thus providing an estimation of C . For the estimation of C to be faithful, it is crucial for the interval extension Ψ to be the tightest possible. To get good result in that regard, we used the interval arithmetic implementation of the `IntervalArithmetic.jl` Julia package[?].

Missing ref

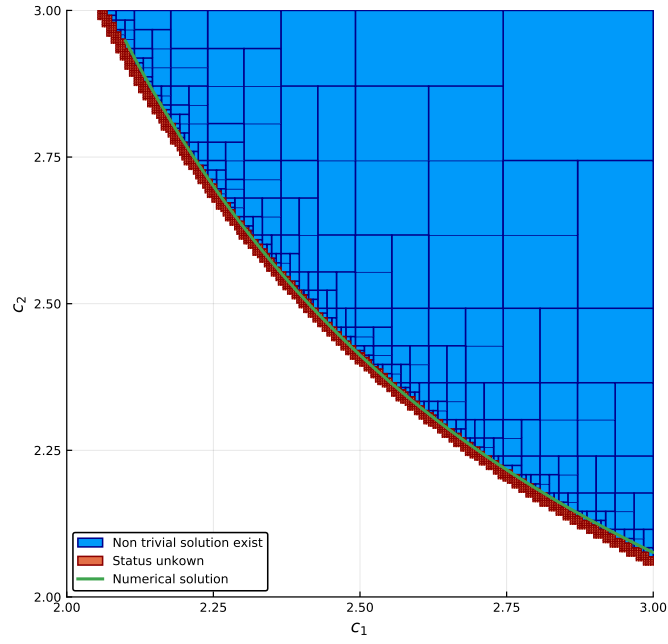
5 Results

We apply the two methods presented to different cases of multiplex network in order to test it. The algorithm using interval arithmetic is compared to the numerical solution of the system composed by eq. (10) and (22), computed using the `NLsolve.jl` Julia package.

Choose what multiplex networks should be used and with what parameters and actually produce the results.

6 Discussion

Write discussion



{Figure: Regions and boundary}

- Make the plot more readable and less ugly
- Find why the two methods seems drift away one from the other far from the center.

Figure 2: Phase diagram for a multiplex network composed of two Erdos-Renyi layer with mean degree c_1 and c_2 . In the blue region a non trivial solution for \mathbf{u} has been found, in the uncolored region only the trivial solution \mathbf{u}_T exists and in the red region the algorithm was unable to conclude in favor of either case. The solid green line is the numerical solution to eq. (10) and (22).