Part I
Normalized clustering coefficients

1 Introduction

1.1 Current measures for clustering

Local clustering coefficient [1]:

$$c_i = \begin{cases} \dfrac{2T_i}{k_i(k_i-1)} & \text{if } k_i > 1 \\ 0 & \text{if } k_i \leq 1, \end{cases} \tag{1}$$

where T_i is the number of triangles passing through vertex i and k_i is its degree.

Average Watts-Strogatz clustering coefficient:

$$\bar{c} = \frac{1}{N}\sum_i c_i. \tag{2}$$

Obs: This average is taken over all the nodes in the network. Some authors use, instead, the only the nodes with degree greater than 1. It is important to be aware of that when interpreting results of other authors.

Degree-dependent clustering coefficient [2]:

$$\bar{c}(k) = \frac{1}{N_k}\sum_{i\in Y(k)} c_i = \frac{1}{k(k-1)N_k}\sum_{i\in Y(k)} 2T_i, \tag{3}$$

where N_k is the number of nodes with degree k and $Y(k)$ is the set of such nodes.

The quantities $\bar{C}$ and $\bar{c}(k)$ are related by

$$\bar{c} = \sum_k p(k)\bar{c}(k), \tag{4}$$

$p(k)$ being the degree distribution.

A different global transitivity measure C was introduced by Newman in [3], and is defined as

$$C = \frac{2T}{\sum_i k_i(k_i-1)}, \tag{5}$$

where $T = \sum_i T_i$ is the number of triangles in the network.

The two global clustering coefficients have similar definitions and, in fact, for some networks both have similar values. In particular, for an uncorrelated network, $\bar{C}$, $\bar{c}(k)$ and C are identical and its value can be computed as [4]

$$C_{\text{unc}} = \frac{1}{N}\frac{\left[\langle k^2\rangle - \langle k\rangle\right]^2}{\langle k\rangle^3}. \tag{6}$$

Nevertheless, these coefficients do differ significantly in some cases, as can be seen in [5, 6].

1.2 Normalized clustering coefficient

We define the normalized clustering coefficients

$$C_{\text{norm}} = \frac{C - C_{\text{rand}}}{C_{\text{max}} - C_{\text{rand}}} \qquad \text{and} \qquad \bar{c}_{\text{norm}} = \frac{\bar{c} - \bar{c}_{\text{rand}}}{\bar{c}_{\text{max}} - \bar{c}_{\text{rand}}} \tag{7}$$

C_{rand} and $\bar{c}_{\text{rand}}$ represent the average of the clustering coefficient computed for an ensemble of random networks with the same degree sequence as the original network, and C_{max} and $\bar{c}_{\text{max}}$ correspond to the

maximum clustering values that can be achieved over the networks in the ensemble. In our work, we will consider the ensamble of all networks having the same degree sequence as the original network.

There are differen posibilities in which C_{rand} can be defined. One way is using the expected value for an uncorrelated network, given by Equation 6, which can be easily computing from the degree sequence. Unfortunately, this measure has some problems for heterogeneous graphs (for which the hypothesis of no degree-degree correlations is not satisfied), as is exemplified in Appendix A. The alternative is to create instances of randomized versions of the network and define C_{rand} as the average of the clustering coefficient for a set of such instances. In this line, we used two different randomization procedures. The first one is the Configuration Model [7], and the second method consists in randomizing the original network by degree-preserving edge rewiring.

The Configuration Model is an algorithm that allows to build a network with a given degree sequence from scratch. It works as follows. We start with an empty graph with N nodes (being N the size of the desired network). We then attach to each node a number of "half-edges" or stubs equal to the node's degree. Then, with uniform probability we pick a pair of stubs and join them, forming an edge. We repeat this step until there is no more free stubs. The algorithm does not impose any restriction on the graph connectivity, and can even create double-edges and self-loops. If one is interested in generating simple graphs, there are two options. The first is to repeatedly apply the algorithm until the resulting graph is simple. It has been proved [TODO: add reference] that by doing that, the sampling is uniform among all the simple graph with the given degree sequence. The downside of this alternative is that it can be computationally prohibitive, in particular for networks with diverging $\langle k^2 \rangle$, where the probability sampling a simple graph is very low [4]. The second alternative is to remove all the self-loops and double-edges from the sampled graph. The expected number of such edges is $\frac{1}{2}\left[(\langle k^2\rangle - \langle k\rangle)/\langle k\rangle\right]^2$ [4], so it vanishes for $N \to \infty$ as long as $\langle k^2\rangle$ remains bounded. This means that this second alternative is practical for networks that are not too heterogeneous. In this work, we chose this second alternative, for which we created 100 random graphs and then took the average of the corresponding clustering coefficients.

The method of degree-preserving edge rewiring (also known as edge swapping or markov chain approach) has been widely used in the literature [8] [TODO: add more references]. It consists on randomly selecting a pair of non-adjacent edges and, if no double edge is created in the process, swapping them. This procedure preserves the degree sequence, but destroys degree correlations. It has been shown that a number $\gtrsim M$ of swaps is enough to decorrelate the network. Here, we performed 10 independent realizations with a total of $n = 100M$ swaps for networks small networks ($M < 100000$) and $n = 10M$ swaps for big networks ($M < 100000$) [TODO: add references related to mixing time and ergodicity for edge swapping].

1.3 Maximum clustering coefficient

The definitions given by equation 7 require the knowledge of the greatest value for the clustering coefficient that can be achieved by a graph having a specified degree sequence. In this section we will address how to compute that value.

1.3.1 Approximation of $C_{\max}$

Havel-Hakimi algorithm:

This is a known algorithm that is very useful because it allows to determine whether a given degree sequence is graphic or not [9]. It is based on the Havel-Hakimi theorem, which states that the degree sequence $S = [k_1, k_2, ..., k_N]$, where $k_i \geq k_j,\ \forall i \leq j$, is graphic if and only if the sequence $S' = [k_2 - 1, k_3 - 1, ..., k_{k_1+1} - 1, k_{k_1+2}, ..., k_N]$ is also graphic. If the given list S is graphic, then the theorem will be applied at most $N-1$ times setting in each further step $S := S'$. Note that it can be necessary to sort this list again. This process ends when the whole list S' consists of zeros. In each step of the algorithm one constructs the edges of a graph with vertices $v_1, ..., v_N$, i.e. if it is possible to reduce the list S to S', then we add edges $\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_{d_1+1}\}$,. When the list S cannot be reduced to a list S' of non-negative integers in any step of this approach, the theorem proves that the list S from the beginning is not graphic.

The advantage of this algorithm is that it always converges when the original sequence is graphic and that the resulting graph es quite clusterized. The disadvantage is that in general it doesn't give the most clusterized graph. For example, given the degree sequence $S_1 = [3, 3, 2, 2, 2, 1, 1]$, the result of the algorithm

is the graph in left side of figure 1, whilst the most clustered graph for that degree sequence is the graph in the right part of the figure.

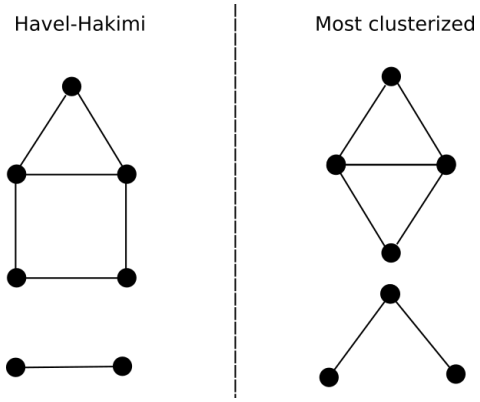

Figure 1: Graphs obtained from the degree sequence $S_1 = [3, 3, 2, 2, 2, 1, 1]$

Greedy algorithm:

This algorithm is similar as the Havel-Hakimi algorithm, with the difference that at each step the list is not sorted. This algorithm gives very good results for most of the real-world networks we studied (the resulting graph is more clusterized than the graph using the Havel-Hakimi algorithm). In particular, in the example in figure 1, it finds the most clusterized graph. Also, comparing with Monte Carlo simulations, it seems that this algorithm finds graphs with a clustering coefficient very close to the maximum.

The main disadvantage of this algorithm is that it doesn't always converge. One counterexample is the degree sequence $S_2 = [3, 3, 2, 2, 2, 2]$. After adding the edges $\{v_1, v_2\}, \{v_1, v_3\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_2, v_4\}, \{v_5, v_6\}$, the algorithm is stuck, as there is no possible pair of nodes where to put the last edge. To overcome this flaw, we perform the following modification. Whenever the algorithm get stuck, lets say after adding the edge $\{v_i, v_j\}$, we remove this edge and try to connect node v_i with v_{j+1}. If the algorithm stuck after adding the edge $\{v_i, v_N\}$, we try with $\{v_{i+1}, v_j\}$. This way, the algorithm always converges.

In most cases, this algorithm seems to converge to graphs very close to the most clustered graph. But if we see the example S_2, we can show that the result of the algorithm is the graph in the right side of figure.

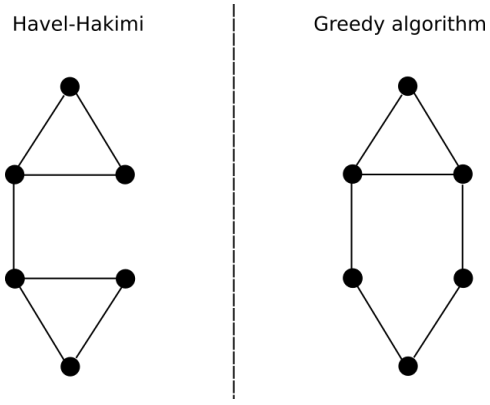

Figure 2: Graphs obtained from the degree sequence $S_1 = [3, 3, 2, 2, 2, 2]$

2 Results

3 Normalized clustering coefficient for real-world networks

The normalized clustering coefficient was computed for several real-world networks (a full list of the networks studied, together with a description of each one) is provided in Appendix D. In Figure 3 we show the normalized version of the Newman clustering coefficient.

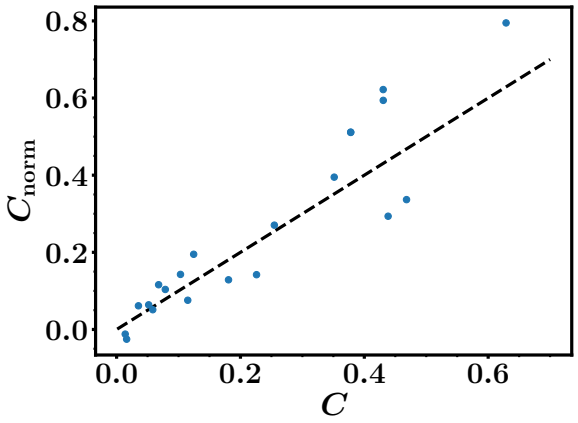

Figure 3: Normalized clustering coefficient compared with its standard version

A Problems with C_{unc}

Equation 6 is valid under the condition that no degree-degree correlation exist. In graphs with an heterogeneous degree distribution, such correlations are unavoidable, and this expresion could give incorrect values. Lets consider a few examples.

Suppose we have a star graph with $N+1$ nodes. This graph has one single node with degree N and N nodes with degree 1. the first and second momenta of the degree distribution are

$$\begin{aligned}\langle k\rangle &= \frac{1}{N+1}\sum_{i=0}^{N} k_i = \frac{1\times N + N\times 1}{N+1} = \frac{2N}{N+1}\\ \langle k^2\rangle &= \frac{1}{N+1}\sum_{i=0}^{N} k_i^2 = \frac{1\times N^2 + N\times 1^2}{N+1} = \frac{N^2+N}{N+1} = N\end{aligned} \tag{8}$$

If we apply equation 6 to this particular case, we obtain

$$C_{\text{unc}} = \frac{1}{N+1}\frac{\left(N^2 - \dfrac{2N}{N+1}\right)^2}{\left(\dfrac{2N}{N+1}\right)^3} = \frac{\left[N^2(N+1)-2N\right]^2}{(2N)^3} = \frac{(N^3+N^2-2N)^2}{8N^3}, \tag{9}$$

which is an increasing function of N and is greater than 1 for $N \geq 3$.

Now suppose we have a sparse network (with mean degree independent of N) and with degree distribution with second momentum scaling as $\langle k^2\rangle \sim N^{\alpha}$, with $\alpha > 0$. In this case, equation 6 gives

$$C_{\text{unc}} \sim N^{2\alpha}, \tag{10}$$

which is unbounded for $\alpha > 0$. That means that the expression for the expected clustering coefficient in uncorrelated networks is not valid for networks that are too heterogeneous.

Now, lets consider the case in which we have a very connected hub with degree $k_{\max} \sim N^{\alpha}$. Suppose also that the degree of the rest of the nodes doesn't scale with N. In this case, we have, for $N \gg 1$,

$$\langle k \rangle \sim N^{\alpha-1}$$
$$\langle k^2 \rangle \sim N^{2\alpha-1}$$

Then, equation 6 gives

$$C_{\text{unc}} \sim \frac{1}{N} \frac{(N^{2\alpha-1} - N^{\alpha-1})^2}{N^{3(\alpha-1)}} \sim \frac{N^{4\alpha-2}}{N^{3\alpha-2}} \sim N^{\alpha}. \tag{11}$$

Thus, if the hub scales with N with any positive exponent, the expected clustering coefficient diverges.

B Computing C_{rand}

In this section, we will discuss the differences betweenn using the Configuration Model and the rewiring procedure to compute C_{rand}.

In Figure 4, we show how the clustering coefficients vary as the network is relaxed to an uncorrelated version using the rewiring procedure. As it can be seen, in most networks a number of iterations equal to M is enough to fully randomize it.

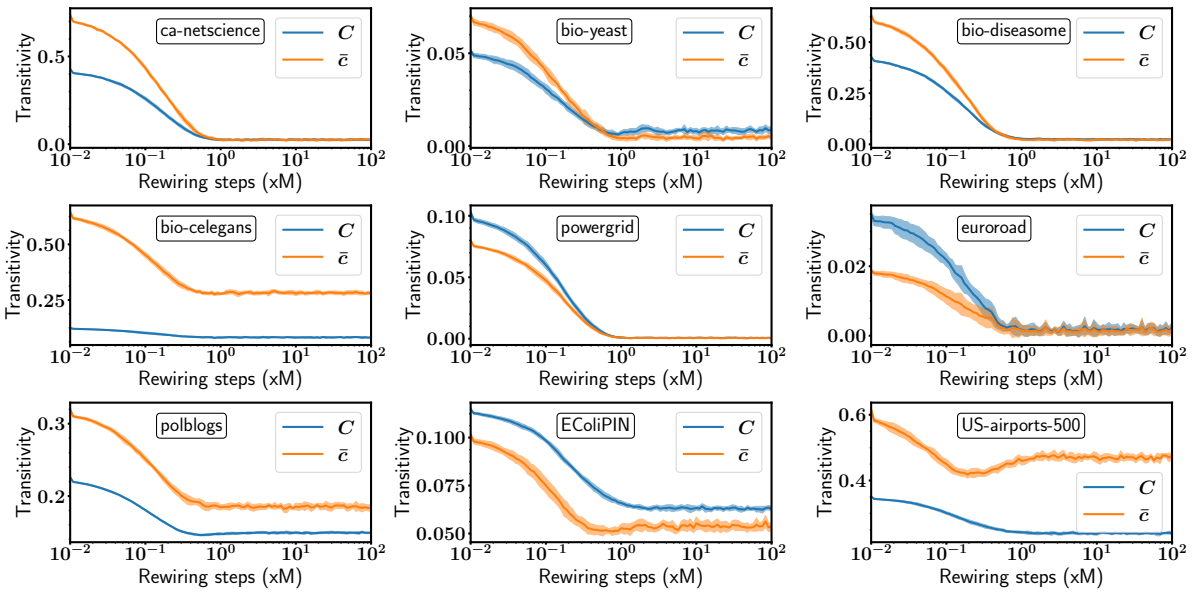

Figure 4: Relaxation of different real-world networks using the rewiring procedure. After $\sim M$ rewiring steps, the network converges to its random version. Each curve correspond to an average taken over 10 realizations, and shade regions correspond to a standard deviation from the mean.

Whilst the rewiring procedure keeps the degree sequence fixed, the configuration model can in principle produce multiple edges, as well as self loops. Given that the clustering coefficients are defined for simple graphs, if that happens, it is necesary to simplify the graph before computing the values. Once simplified, the resulting graph will have a different degree sequence than the original, so the values of the clustering coefficients computing using this null model could potentially differ from the values obtained by the rewiring process. As it is pointed out in [CITE], the number of multiple edges and self loops for a sparse graph scales as $\sim N^{-1}$, so for big networks there should be no significant differences. In Figure 5 we show that indeed, the differences between these two values are not significant.

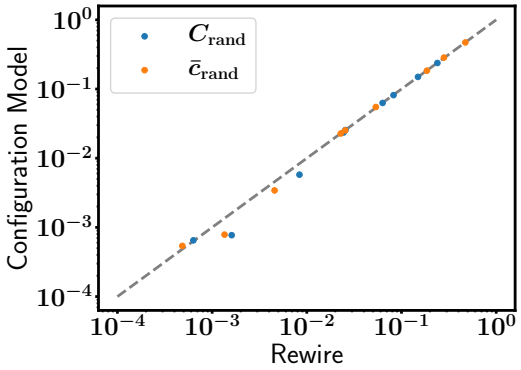

Figure 5: Comparison between the transitivity values obtained in the two null models proposed: configuration model and rewiring process.

C Analytic Results

In this Appendix we will discuss the problem of finding the maximum value of the clustering coefficient for a given degree sequence.

In [10], Rivin obtains sharp bounds for the number of n-cycles in a simple, undirected graph as a function of the number of edges. In particular, the bound for the number T of triangles of a network with N nodes and M edges is found. Using Lagrange multipliers to maximize

$$\sum_i \lambda_i^3 \tag{12}$$

under the constraints

$$\sum_i \lambda_i = 0 \qquad \text{and} \qquad \sum_i \lambda_i^2, \tag{13}$$

he obtains that the number of triangles is bounded by

$$T \leq \frac{N-2}{\sqrt{N(N-1)}} \frac{2^{1/2}}{3} M^{3/2}. \tag{14}$$

Also, taking the complete graph as an example, he proves that the bound is sharp.

Now, inequality 14 could be improved by imposing the condition that the network has a specified degree sequence.

Let G be a simple undirected graph with N nodes and M links and let A be its adjacency matrix. The elements $a_{ij} = (A)_{ij}$ then satisfy

$$a_{ij} = a_{ji}, \tag{15}$$

$$a_{ii} = 0, \tag{16}$$

$$\sum_j a_{ij} = k_i, \tag{17}$$

$$\sum_{i,j} a_{ij} = 2M, \tag{18}$$

where k_i is the degree of the node i.

The Newman's clustering coefficient is given by

$$C = \frac{6 \times N_T}{|P_2|} \tag{19}$$

where N_T is the number of triangles in the network and $|P_2|$ the number of path of length 2.

To compute the number of path of length two, lets start by noting that, for a given node i, the number of path of length 2 that start in i are

$$|P_2|(i) = \sum_l (A^2)_{il} - (A^2)_{ii} = \sum_{lj} a_{il}a_{lj} - k_i = \sum_l a_{il}k_l - k_i. \tag{20}$$

Then, the total number of paths of length two is simple

$$|P_2| = \sum_i |P_2|(i) = \sum_i \left(\sum_l a_{il}k_l - k_i \right) = \sum_l k_l^2 - 2M. \tag{21}$$

On the other side, the number of triangles in the graph is given by

$$N_T = \frac{1}{6} Tr(A^3) = \frac{1}{6} \sum_{ijl} a_{ij}a_{jl}a_{li}. \tag{22}$$

Thus, the Newman's clustering coefficient is given by

$$C = \frac{Tr(A^3)}{|P_2|} = \frac{Tr(A^3)}{\sum_l k_l^2 - 2M}, \tag{23}$$

If we consider the ensamble of graphs having a given degree sequence $\{k_1, k_2, \ldots, k_N\}$, the clustering coefficient depends only on the numerator of equation 23, as the denominator is determined by the degree sequence.

The Watts-Strogatz clustering coefficient is defined by

$$C_{\mathrm{WS}} = \frac{1}{N} \sum_i c(i), \tag{24}$$

where $c(i)$ is the local clustering of node i, defined as

$$c(i) = \begin{cases} \dfrac{2N_T(i)}{k_i(k_i-1)} & \text{if } k_i > 1 \\ 0 & \text{if } k_i \leq 1 \end{cases} \tag{25}$$

$N_T(i)$ being the number of triangles that have i as one of its vertices, which can be computed as

$$N_T(i) = \sum_{jl} a_{ij}a_{jl}a_{li} = (A^3)_{ii}. \tag{26}$$

Thus, equation 24 can be written as

$$C_{\mathrm{WS}} = \frac{1}{N} \sum_{i:k_i>1} \frac{2N_T(i)}{k_i(k_i-1)} = \frac{1}{N} \sum_{i:k_i>1} \frac{2(A^3)_{ii}}{k_i(k_i-1)} \tag{27}$$

C.1 Lagrange multipliers

In order to get $C_{\max}$, we need to maximize the number of triangles in the network (the number of triads is fixed by the degree sequence, see Eq. 21). According to Eq. 22, the number of triangles is determined by the trace of the adjacency matrix A. Considering the degree sequence as constrains, together with the conditions imposed in considering the graph as simple and undirected, the function we have to maximize is

$$f(\{a_{ij}\}) = Tr(A^3) + \sum_i \theta_i \left(d_i - \sum_j a_{ij} \right) - \sum_i \omega_i a_{ii} + \sum_{i \neq j} \beta_{ij}(a_{ij} - a_{ji}) + \gamma \left(2M - \sum_{ij} a_{ij} \right). \quad (28)$$

Differentiating with respect to a_{ij} and equaling to 0,

$$\frac{\partial f(\{a_{pq}\})}{\partial a_{pq}} = 3(A^2)_{pq} - \theta_p - \omega_p \delta_{pq} + (\beta_{pq} - \beta_{qp})(1 - \delta_{pq}) - \gamma = 0, \quad (29)$$

where δ_{pq} is the Kronecker delta. For $p = q$, we obtain

$$3(A^2)_{pp} = \theta_p + \omega_p + \gamma \quad (30)$$
$$3d_p = \theta_p + \omega_p + \gamma. \quad (31)$$

Summing over p,

$$6M = \Theta + \Omega + N\gamma, \quad (32)$$

where we defined $\Theta = \sum_p \theta_p$ and $\Omega = \sum_p \omega_p$.

On the other side, for $p \neq q$, we have

$$3(A^2)_{pq} = \theta_p + (\beta_{qp} - \beta_{pq}) + \gamma. \quad (33)$$

Analogously, differentiating with respect to a_{qp}, $q \neq p$, we have

$$3(A^2)_{qp} = \theta_q - (\beta_{qp} - \beta_{pq}) + \gamma. \quad (34)$$

Taking into account that A^2 is symmetric and adding Eq. 34 to Eq. 33,

$$6(A^2)_{pq} = \theta_p + \theta_q + 2\gamma. \quad (35)$$

Summing 35 over $p \neq q$,

$$\begin{aligned} \sum_{p \neq q} 6(A^2)_{pq} &= \sum_{p \neq q} (\theta_p + \theta_q) + 2N(N-1)\gamma \\ 6|P_2| &= 2(N-1)\Theta + 2N(N-1)\gamma \\ \frac{3|P_2|}{N-1} &= \Theta + N\gamma. \end{aligned} \quad (36)$$

Summing 35 over $q : q \neq p$,

$$\begin{aligned} \sum_{q:q \neq p} (A^2)_{pq} &= (N-1)\theta_p + \sum_{q:q \neq p} \theta_q + 2(N-1)\gamma \\ |P_2(p)| &= (N-1)\theta_p + \Theta - \theta_p + 2(N-1)\gamma \\ |P_2(p)| &= (N-2)\theta_p + \Theta + 2(N-1)\gamma \\ |P_2(p)| &= (N-2)\theta_p + (N-2)\gamma + \frac{3|P_2|}{N-1} \\ |P_2(p)| &= (N-2)(\theta_p + \gamma) + \frac{3|P_2|}{N-1} \end{aligned} \quad (37)$$

From 32 and 36, we have

$$\Omega = 6M - \frac{3|P_2|}{N-1}. \tag{38}$$

Summing over $p \neq q$,
Now, subtracting Eq. 34 to Eq. 33, we have

$$2(\beta_{qp} - \beta_{pq}) = \theta_q - \theta_p. \tag{39}$$

Defining $\beta_{pp} = 0$, $\forall p$ and summing over p,

$$2\sum_p (\beta_{qp} - \beta_{pq}) = N\theta_q - \Theta. \tag{40}$$

Then,
Summing over $p \neq q$,

$$2\sum_{p\neq q} (\beta_{qp} - \beta_{pq}) = \sum_{p\neq q} (\theta_q - \theta_p) = 0. \tag{41}$$

Based on the previous results, we have that

$$\begin{aligned}
6Tr(A^3) &= 6\sum_{ij} a_{ij}(A^2)_{ij} \\
6Tr(A^3) &= \sum_{ij} a_{ij}(\theta_i + \theta_j + 2\gamma) \\
6Tr(A^3) &= 2\sum_{ij} a_{ij}(\theta_i + \gamma) \\
3Tr(A^3) &= \sum_{ij} a_{ij}\theta_i + 2M\gamma \\
3Tr(A^3) &= \sum_{i} d_i\theta_i + 2M\gamma \\
3Tr(A^3) &= \sum_{i} d_i\theta_i + 2M\left[\frac{3|P_2|}{N(N-1)} - \frac{1}{N}\Theta\right] \\
3Tr(A^3) &= \frac{2M}{N}\frac{3|P_2|}{N-1} + \sum_i \left(d_i\theta_i - \frac{2M}{N}\theta_i\right) \\
3Tr(A^3) &= \langle k\rangle \frac{3|P_2|}{N-1} + \sum_i \left(d_i - \langle k\rangle\right)\theta_i
\end{aligned} \tag{42}$$

Now let's do the same but trying to maximize $\bar{c}$. The function with the constrains is

$$g(\{a_{ij}\}) = \sum_i \frac{2(A^2)_{ii}}{d_i(d_i-1)} + \sum_i \mu_i \left(d_i - \sum_j a_{ij}\right) + \sum_i \omega_i a_{ii} + \sum_{i\neq j} \beta_{ij}(a_{ij} - a_{ji}) + \gamma\left(2M - \sum_{ij} a_{ij}\right). \tag{43}$$

Differentiating with respect to a_{pq},

$$\frac{\partial g(\{a_{pq}\})}{\partial a_{pq}} = \frac{2a_{qp}}{d_p(d_p-1)} + \frac{2a_{pq}}{d_q(d_q-1)} - \mu_p + \omega_p\delta_{pq} + (\beta_{pq} - \beta_{qp})(1-\delta_{pq}) - \gamma = 0. \tag{44}$$

For $p = q$,

$$\omega_p = \mu_p. \tag{45}$$

For $p \neq q$,

$$2a_{pq}\left[\frac{1}{d_p(d_p-1)}+\frac{1}{d_q(d_q-1)}\right]-\mu_p+(\beta_{pq}-\beta_{qp})-\gamma=0. \tag{46}$$

Differentiating with respect to a_{qp}, $q \neq p$,

$$2a_{pq}\left[\frac{1}{d_p(d_p-1)}+\frac{1}{d_q(d_q-1)}\right]-\mu_q-(\beta_{pq}-\beta_{qp})-\gamma=0. \tag{47}$$

Substracting:

$$\begin{aligned}\mu_q-\mu_p+2(\beta_{pq}-\beta_{qp})&=0\\ \sum_{p\neq q}(\beta_{pq}-\beta_{qp})&=0.\end{aligned} \tag{48}$$

D Networks studied

Facebook-combined: Facebook user-user friendship network. Nodes represent users and links represent friendship. Data correspond to 10 egocentric networks combined. First used in [11]. Available at `http://snap.stanford.edu/data/egonets-Facebook.html`.

Twitter: This dataset consists of 'circles' (or 'lists') from Twitter. Twitter data was crawled from public sources. The dataset includes node features (profiles), circles, and ego networks. First used in [11]. Available at `http://snap.stanford.edu/data/egonets-Twitter.html`.

Google+: This dataset consists of 'circles' from Google+. Google+ data was collected from users who had manually shared their circles using the 'share circle' feature. The dataset includes node features (profiles), circles, and ego networks. First used in [11]. Available at `http://snap.stanford.edu/data/egonets-Gplus.html`.

Euro Road: Nodes represent European cities and edges represent roads. Available at `http://konect.uni-koblenz.de/networks/subelj_euroroad`.

Power grid: This is the network is the high-voltage power grid in the Western States of the United States of America. The nodes are transformers, substations, and generators, and the ties are high-voltage transmission lines. Although the transmission lines can be directed and differentiated based on their capacity, this information is not available. First used in [1]. Available at `https://toreopsahl.com/datasets`.

Internet: Contains a symmetrized snapshot of the structure of the Internet at the level of autonomous systems, reconstructed from BGP tables posted at archive.routeviews.org (University of Oregon). This snapshot was created by Mark Newman from data for July 22, 2006.

Internet-Oregon: Internet as autonomous systems as for My 26 2001. Data collected by University of Oregon using route-views. Data first used in [12]. Available at `http://snap.stanford.edu/data/Oregon-1.html`

Email-Tarragona: List of edges of the network of e-mail interchanges between members of the Univeristy Rovira i Virgili (Tarragona). Data compiled by members of our group. First used in [13]. Available at `http://deim.urv.cat/~alexandre.arenas/data/welcome.htm`.

Email-Enron: The Enron email network consists of 1,148,072 emails sent between employees of Enron between 1999 and 2003. Nodes in the network are individual employees and edges are individual emails. It is possible to send an email to oneself, and thus this network contains loops. First used in [14]. Available at

http://snap.stanford.edu/data/email-Enron.html. See also [15, 16].

Open Flights: A tie exists between two airports if a flight was scheduled between them in 2002. The weights corresponds to the number of seats available on the scheduled flights. Even thought this type of networks is directed by nature as a flight is scheduled from one airport and to another, the networks are highly symmetric (Barrat et al., 2004). Therefore, the version of this network is undirected (i.e., the weight of the tie from one airport towards another is equal to the weight of the reciprocal tie). This network was obtained from the Complex Networks Collaboratorys website. Data first used in [17]. Available at https://toreopsahl.com/datasets/#usairports.

US-airports-500: This is the directed network of flights between the 500 mos busiest commercial airports in the US in 2010. Each edge represents a connection from one airport to another, and the weight of an edge shows the number of flights on that connection in the given direction, in 2010.

PGP: List of edges of the giant component of the network of users of the Pretty-Good-Privacy algorithm for secure information interchange. Data compiled by members of our group. First used in [18]. Available at http://deim.urv.cat/~alexandre.arenas/data/welcome.htm.

bio-celegans: metabolic network of the roundworm *Caenorhabditis elegans* nodes are metabolites (e.g., proteins) and edges are interactions between them. First used in [19]. Available in http://networkrepository.com/bio-celegans.php. Obs: in [19], the authors study many more networks, but I couldn't find the data.

bio-diseasome:

Coautorship networks: Nodes are authors and links are present when two authors published together. (ca-netscience) (ca-HepTh) (ca-HepPh) (ca-GrQc) (ca-CondMat) (ca-AstroPh)

Air traffic: Air traffic control network. This network was constructed from the FAA (Federal Aviation Administration) National Flight Data Center (NFDC), Preferred Routes Database. Nodes in this network represent airports or service centers and links are created from strings of preferred routes recommended by NFDC; downloaded from: www.fly.faa.gov. Available at http://research.mssm.edu/maayan/datasets/qualitative_networks.shtml

Terrorist: 9/11 terrorist communication network. Nodes represent people involved in the 9/11 attack and edges represent known conections between them. First used in [20]. See also [21].

Jazz: Collaboration network among jazz musicians. Nodes represent musicians and links are present between pairs of musicians that play in the same band. First used in [22].

Protein interaction networks: Networks of experimentally determined interactions between proteins [23, 24]. We used data from the following species: *C. Elegans*, *E. Coli*, *S. Cerevisiae* [25], *M. Musculus*, *H. Pylori* and *H. Sapiens* [26]. Available at [27].

Chess: Networks of chess players. Nodes represent players and conections are established between nodes if the players played at least once. Three different networks were built, using games played on physical boards (chess-OTB), internet portals (chess-Portal) and by correspondence (chess-Corr). The data was provided by www.openingmaster.com. Reference: [28]

References

[1] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, pp. 440–442, 6 1998.

[2] A. Vázquez, R. Pastor-Satorras, and A. Vespignani, "Large-scale topological and dynamical properties of the Internet," *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, vol. 65, no. 6, pp. 1–12, 2002.

[3] M. E. J. Newman, "Mixing patterns in networks," *Physical Review E*, vol. 67, 2 2002.

[4] M. E. J. Newman, *Networks*. Oxford University Press, 2 ed., 2018.

[5] B. Bollobás and O. M. Riordan, "Mathematical results on scale-free random graphs," in *Handbook of Graphs and Networks*, pp. 1–34, Wiley-VCH Verlag GmbH & Co. KGaA, dec 2004.

[6] E. Estrada, *The Structure of Complex Networks*. Oxford University Press, oct 2011.

[7] M. Molloy and B. Reed, "A critical point for random graphs with a given degree sequence," *Random Structures & Algorithms*, vol. 6, no. 2-3, p. 161180, 1995.

[8] C. Orsini, M. M. Dankulov, P. Colomer-De-Simon, A. Jamakovic, P. Mahadevan, A. Vahdat, K. E. Bassler, Z. Toroczkai, M. Boguñã, G. Caldarelli, S. Fortunato, and D. Krioukov, "Quantifying randomness in real networks," *Nature Communications*, vol. 6, no. May, 2015.

[9] H. S., "On realizability of a set of integers as degrees of the vertices of a linear graph. i," *Journal of SIAM*, vol. 10, pp. 496–506, 1962.

[10] I. Rivin, "Counting cycles and finite dimensional lp norms," *Advances in Applied Mathematics*, vol. 29, pp. 647–662, nov 2002.

[11] J. Leskovec and J. Mcauley, "Learning to discover social circles in ego networks," *Advances in neural information processing ...*, pp. 1–9, 2012.

[12] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graphs over time: densification laws, shrinking diameters and possible explanations," *KDD '05: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, p. 177, 2005.

[13] R. Guimerà, L. Danon, A. Díaz-Guilera, F. Giralt, and A. Arenas, "Self-similar community structure in a network of human interactions," *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 2003.

[14] J. Leskovec, K. J. Lang, A. Dasgupta, and M. W. Mahoney, "Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters," *Internet Math.*, vol. 6, no. 1, pp. 29–123, 2009.

[15] "Enron network dataset – KONECT," Apr. 2017.

[16] B. Klimt and Y. Yang, "The Enron corpus: A new dataset for email classification research," in *Proc. European Conf. on Machine Learning*, pp. 217–226, 2004.

[17] V. Colizza, R. Pastor-Satorras, and A. Vespignani, "Reaction-diffusion processes and metapopulation models in heterogeneous networks," *Nature Physics*, vol. 3, no. 4, pp. 276–282, 2007.

[18] M. Boguñá, R. Pastor-Satorras, A. Díaz-Guilera, and A. Arenas, "Models of social networks based on social distance attachment," *Physical Review E - Statistical Physics, Plasmas, Fluids, and Related Interdisciplinary Topics*, 2004.

[19] H. Jeong, R. Albert, Z. N. Ottval, and A. L. Barabási, "The large scale organization of metabolic networks," *Nature*, vol. 407, no. 6804, pp. 651–654, 2000.

[20] V. E. Krebs, "Mapping Networks of Terrorist Cells," *Connections*, vol. 24, no. 3, pp. 43–52, 2002.

[21] L. Tian, A. Bashan, D. N. Shi, and Y. Y. Liu, "Articulation points in complex networks," *Nature Communications*, vol. 8, pp. 1–9, 2017.

[22] P. Gleiser and L. Danon, "Community Structure in Jazz," 2003.

[23] I. Xenarios, D. W. Rice, L. Salwinski, M. K. Baron, E. M. Marcotte, and D. Eisenberg, "DIP: the Database of Interacting Proteins," Tech. Rep. 1, 2000.

[24] I. Xenarios, "DIP, the Database of Interacting Proteins: a research tool for studying cellular networks of protein interactions," *Nucleic Acids Research*, 2002.

[25] V. Colizza, A. Flammini, M. A. Serrano, and A. Vespignani, "Detecting rich-club ordering in complex networks," *Nature*, vol. 2, no. February, pp. 110–115, 2006.

[26] K. I. Goh, G. Salvi, B. Kahng, and D. Kim, "Skeleton and fractal scaling in complex networks," *Physical Review Letters*, vol. 96, no. 1, pp. 1–4, 2006.

[27] "Database of Interacting Proteins." `http://dip.doe-mbi.ucla.edu/dip/`.

[28] N. Almeira, A. Schaigorodsky, J. Perotti, and O. Billoni, "Structure constrained by metadata in networks of chess players," *Scientific Reports*, vol. 7, no. 1, 2017.