2
International Trade Networks and World Trade Web

2.1 Introduction

One of the most interesting and cured datasets available in economics is the recorded set of commercial transactions between different countries. This information is available in varying detail, from the large scale of the total yearly trade between one nation and all the others, to the volume of trade of a specific (and this can be very specific) product among the various countries. Trade is another system naturally described by means of a network whenever we define the vertices as the different countries and the directed edges as the trade exchange in a specific year (with a weight given by the volume in dollars). This information is available for a rather long temporal window spanning several decades. The analysis of such a database is particularly problematic for various reasons. First, as the detail increases, the size of the dataset also increases. Second, the number of products produced and traded as well as the number of countries varies year on year. We show in the following how to deal with such problems.

One of the best sources of data is COMTRADE (http://comtrade.un.org/), that is, the United Nations Commodity Trade Statistics Database. These data are collected and organised by International Merchandise Trade Statistics (IMTS), a United Nations Division. Given the immediate applicability of the framework of complex network theory to that system, there are already many papers devoted to the analysis of the topological properties of this trade (Serrano and Boguñá, 2003; Garlaschelli and Loffredo, 2004*a*; Garlaschelli *et al.*, 2007; Tacchella *et al.*, 2012). In particular, various attempts have been made to extract economic information from such data and different approaches have been taken with respect to data (i.e. aggregated (Serrano and Boguñá, 2003) or disaggregated at various levels of refining (Hidalgo *et al.*, 2007; Hidalgo and Hausmann, 2009; Tacchella *et al.*, 2012)), and with respect to the network model used (i.e. simple (Serrano and Boguñá, 2003; Garlaschelli and Loffredo, 2004*a*), bipartite (Hidalgo and Hausmann, 2009; Tacchella *et al.*, 2012), and weighted, multinetwork (Barigozzi *et al.*, 2010)).

Actually, such a system allows us to introduce a first differentiation in the study of complex networks. Indeed, when studying trade, we have the system being composed of two distinct entities: the countries (in the order of hundreds) and the products made by them (in the order of thousands at the first four digit classifications, i.e. at a reasonably detailed specification which distinguishes between apples and bananas in the class of fresh fruits). Various applications are possible and we present them in this chapter.

Data Science and Complex Networks. First Edition. Guido Caldarelli and Alessandro Chessa.

1. The first immediate study is the analysis of relations between different countries as specified by their trade.
2. Alternatively, we can also inspect the products. This can be done by representing the system as a multinetwork (Wasserman and Faust, 1994), where vertices (countries) are connected by edges (products) of different natures (Barigozzi *et al.*, 2010).
3. Finallly, we can decide to represent the same information by means of a bipartite network (i.e. a network that can split into two sets, where the edges connect only elements of the first set with elements of the second) made of countries (whose number is N_c) and products (whose number is N_p). This can be arranged in a non-square matrix of size $N_c \times N_p$ where the entries n_{cp} are the production of product p made by the country c .

The results from these questions shed some light on basic economic quantities such as reciprocity in trade, to more complex intangible quantities such as capabilities of countries (Hidalgo and Hausmann, 2009), which are hidden behind the production and export of goods and ultimately determine the progress of wealth in a nation (at least when measured by its gross domestic product (GDP)). In particular, an indirect knowledge of these capabilities and how they can be inferred from data, is a good predictor of what kind of products that country will produce in the future.

In this chapter we shall start by giving a description of the database, and how to download and organise the data. We then continue by taking the most aggregated version of the WTW at the level of nations and then proceed using successive refinements. Codes, data and/or links for this chapter are available from **http://book.complexnetworks.net**.

2.2 Data from COMTRADE

In the following we shall use United Nations Commodity Trade Statistics Database (UN COMTRADE) data. This database is derived from National Statistical Offices and covers trade from 1962. It is possible to download a series of data where the following are specified: product, producer (reporter), buyer (partner), year. It is impossible to set "all" in all four fields as files are too large to be used in a batch window. Even if, *"Free unlimited access to UN COMTRADE is available for all users on the website of the United Nations Statistics Division (UNSD)"*[1] in practice (for technical reasons) there is a *"download limit of 50,000 data records per query"*. In any case there is no limit on the number of queries a user can make. *"UN COMTRADE offers Premium access, which allows for downloads of more than 50,000 records and for the use of advanced functions of UN COMTRADE. Premium access is payable"*.

This collection of data is used by international organizations such as the World Bank Organization (WBO), International Trade Centre (ITC), World Trade Organization (WTO), and the United Nations Conference on Trade and Development (UNCTAD). Sometimes these organisations complement the original data with different datasets or cure the database, thereby creating other sources of downloads. For example:

[1] Taken from http://comtrade.un.org/db/default.aspx.

- The international Trade Centre has import/export databases publicly available at http://www.intracen.org/ByCountry.aspx, where, by selecting a single country (e.g. Algeria) we have access to the relative web page: http://www.intracen.org/country/algeria/
- The French research center in international economics (CEPII) published the BACI International Trade Database at the product level (Gaulier and Zignago, 2010). From the site description, *"BACI is constructed using an original procedure that reconciles the declarations of the exporter and the importer"*. In particular, *"First, as import values are reported CIF (cost, insurance and freight) while exports are reported FOB (free on board), CIF costs are estimated and removed from imports values to compute FOB import values. Second, the reliability of country reporting is assessed based on the reporting distances among partners. We decompose the absolute value of the ratios of mirror flows using a (weighted) variance analysis, and an index is built for each country. These reporting qualities are used as weights in the reconciliation of each bilateral trade flow twice reported."*
- The World Trade Organization (WTO) is developing a software to access COMTRADE and other data. This software, called World Integrated Trade Solutions (WITS), allows aggregation of information from other dataset, as such, for example, tariff or other legal data, and it also allows one to realise simulations on the effect of change, such as tariff cuts etc. From the web site[2] it is possible to download data and software following a (free of charge) user registration.
- From the *Journal of Conflict Resolution* (Gleditsch, 2002) where a list of the data is available in printed form.

2.2.1 Product classification

The classification of products is a rather complicated issue. Indeed one has to define the rules of the taxonomy when aggregating them. For example, we may consider origin, form, production method, or use of any particular good as different classification criteria. To add difficulty, levels of detail on the above criteria for a particular good can be very different from country to country. Finally, technological progress constantly challenges any classification applied (e.g. smartphones are phones, music players, or mini computers?). It is therefore not surprising that various initiatives have been attempted and an international team is required in order to find a proper standard. Below we report some information taken from the web sites of the various datasets.

- From the historical background reported in United Nations publications[3] we know that the first tentative international classification was made by the League of Nations, which produced the Minimum List of Commodities for International Trade Statistics (League of Nations, 1938 (II.A.14; and corrigendum, 1939)). With the new United Nations organisation, this work was continued and expanded in the first United Nations Standard International Trade Classification (SITC) in 1950. In the first instance such classification was based on the material from which the products were made. Successive classifications rearranged products also according

[2]http://wits.worldbank.org/wits/

[3]http://unstats.un.org/unsd/trade/sitcrev4.htm

to the stage of fabrication or their industrial origin. With time different classification procedures started to become more effective: they took into consideration things like the processing stage, market practices, material used in production etc. This originated a series of revisions up to the current (4^{th}) revision that was presented in 2006.

- HS, The Harmonized Commodity Description and Coding System has been developed and maintained by the World Customs Organization (WCO). The classification can be bought from the WCO web site and it is made from about 5000 commodity groups. In this list every good is identified by a six digit code. These codes are arranged in a hierarchical structure where the first few digits correspond to a broad series of goods and extra digits are then added to account for legal information (on customs tariffs increasing the digits from 6 to 8) and for statistical purposes (increasing from 8 to 10).

These two basic classifications have many correlations of structure. For example, by employing the headings of HS as building blocks, the United Nations Statistical Office, in consultation with experts from other governments, interested international organisations, and expert groups, produced a third and a fourth revision of SITC, while taking into account the need for continuity with the previous versions. Very good mapping between the various nomenclatures for the same product is available from the WITS web page[4].

UN COMTRADE, the most important source of trade information, has used SITC since 1962 and HS since 1988. As an example of SITC classification we list here the sections corresponding to the first digit (indicated as "sections" in the web site)

0. Food and live animals
1. Beverages and tobacco
2. Crude materials, inedible, except fuels
3. Mineral fuels, lubricants, and related materials
4. Animal and vegetable oils, fats, and waxes
5. Chemicals and related products, n.e.s.
6. Manufactured goods classified chiefly by material
7. Machinery and transport equipment
8. Miscellaneous manufactured articles
9. Commodities and transactions not classified elsewhere in the SITC

In this classification, we have section 0 with nine divisions (01= *"Meat and meat preparations"*), a total of 36 groups (every division has some: 01 has for instance four different groups), 132 subgroups (01 has 17), and finally 335 basic headings. For example "0161" is the subgroup of *"Bacon, ham and other salted, dried or smoked meat of swine"* with basic headings as 016.11 *"Hams, shoulders and cuts thereof, with bone in"*. Therefore in this classification, the lower is the number of digits used, and the broader is the category of products.

[4]http://wits.worldbank.org/wits/product_concordance.html

2.2.2 Country Classification

The situation is somewhat easier (but not too much so) for the classification of countries. Among the various sources of ambiguity that could affect time evaluation of trades we have independence of former colonies, splitting and/or unification of different countries, wars, international recognition by other countries, or simply a change of the codes adopted to describe the countries worldwide. Facing only this last issue, the standard suggested by the International Organization for Standardization is what is called the ISO 3166-1.[5] Not all databases described follow the same standard, so we list three possible situations:

- Two-letter code (ISO3166-1 alpha-2). The same for the internet domains. Mostly used for practical reasons, it is less immediate to associate a country with its code (Italy → IT, France → FR, Gibraltar → GI).
- Three-letter code (ISO3166-1 alpha-3). Three-letter country codes which allow a better visual association between the codes (Italy → ITA, France → FRA, Gibraltar → GIB).
- Three-digit numeric (ISO3166-1 numeric). The most practical for countries not using the latin alphabet. No clue on what is what (Italy → 380, France → 250, Gibraltar → 292).

Countries may also change name, and it is for this reason (as well as the foregoing) that several revisions have been made.

2.3 Projecting and symmetrising a bipartite network

The first analysis that can be done is related to the total production and export of a single country against all the others. In this system, the countries are the vertices of the graph and the total export from country i to country j is a weighted edge. This graph can be indicated by the adjacency matrices A^I and A^E for imports and exports. The first problem is that (probably because of different accounting procedures) the export data $i \to j$ of the total export from i to j does not exactly match the import data $j \leftarrow i$ of the import of j from i. In other words $A^E_{ij} \neq A^I_{ji}$. Even by restricting to an unweighted version of the network, an edge is drawn only if the import/export is relevant for the country of origin, notwithstanding the fact that the relevance could be different for the counterpart. At the coarser level of aggregation only the most important products are indicated for a particular country and in the first analysis of the WTW (Serrano and Boguñá, 2003) this problem has been solved by symmetrisation of the two datasets.

In particular if both A^I_{ij} and A^E_{ji} are different from zero,

$$\begin{cases} A^I_{ij} = \frac{1}{2}(A^I_{ij} + A^E_{ji}) \\ \\ A^E_{ij} = \frac{1}{2}(A^E_{ij} + A^I_{ji}) \end{cases} \tag{2.1}$$

and

[5] http://www.iso.org/iso/country_codes

$$\begin{cases} A^I_{ij} = A^E_{ji} \\ \\ A^E_{ij} = A^I_{ji} \end{cases} \tag{2.2}$$

otherwise.

Network symmetrisation

```
def net_symmetrisation(wtn_file, exclude_countries):
    DG=nx.DiGraph()

    Reporter_pos=1
    Partner_pos=3
    Flow_code_pos=2
    Value_pos=9

    dic_trade_flows={}
    hfile=open(wtn_file,'r')

    header=hfile.readline()
    lines=hfile.readlines()
    for l in lines:
        l_split=l.split(',')
        #the following is to prevent parsing lines without data
        if len(l_split)<2: continue
        reporter=int(l_split[Reporter_pos])
        partner=int(l_split[Partner_pos])
        flow_code=int(l_split[Flow_code_pos])
        value=float(l_split[Value_pos])

        if ( (reporter in exclude_countries) or \
            (partner in exclude_countries) or (reporter==partner) ):
            continue

        if flow_code==1 and value>0.0:
            #1=Import, 2=Export
            if dic_trade_flows.has_key((partner,reporter,2)):
                DG[partner][reporter]['weight']= \
                (DG[partner][reporter]['weight']+value)/2.0
            else:
                DG.add_edge(partner, reporter, weight=value)
                dic_trade_flows[(partner,reporter,1)]= \
                value #this is to mark the exixtence of the link

        elif flow_code==2 and value>0.0:
```

```
            #1=Import, 2=Export
            if dic_trade_flows.has_key((reporter,partner,1)):
                DG[reporter][partner]['weight']= \
                (DG[reporter][partner]['weight']+value)/2.0
            else:
                DG.add_edge(reporter, partner, weight=value)
                #this is to mark the exixtence of the link
                dic_trade_flows[(reporter,partner,2)]=value
        else:
            print "trade flow not present\n"

    hfile.close()

    return DG
```

In a more recent approach (Barigozzi *et al.*, 2010), it has been found that in the case of doubt it is better to use the value of the trade flow as reported by the importer.

A second problem arises when we want to consider the time evolution of this system. Especially after the Second World War and in the 1990s, new countries appeared on the scene inheriting production and export from parent countries. In order to perform a temporal analysis and to allow comparisons across different years, the solution adopted in the literature (Squartini *et al.*, 2011*a*) is to consider only a stable panel of N = 162 countries that have been present in the COMTRADE data throughout the period from 1992 to 2002. The database used at this level of aggregation is freely available (Gleditsch, 2002), with various levels of detail. In spite of the fact that various way of aggregating are possible, happily enough, the various datasets show similar properties (Fagiolo *et al.*, 2009; Garlaschelli and Loffredo, 2004*a*; Squartini *et al.*, 2011*a*).

Given these caveats, the network constructed in such a way presents a series of interesting properties.

- The structure of trade channels reveals a complex organisation similar to other networks with a skewed distribution of number of economic partners and total (in US dollars) trade.
- The degree distribution $P(k)$ of the total degree $k = k_{in} + k_{out}$ for the various countries shows a fat-tailed distribution, compatible with a power-law fit of the kind $P(k) \simeq k^{-\gamma}$, with $\gamma = 2.6$.
- The degree (number of trade partners for unweighted networks) of a country is correlated with its GDP *per capita*.
- As we shall see in the following, using the above properties we can define a model based on the GDP distribution to model the evolution of trade.
- For every vertex there is a strong correlation between the in and the out degree, even if not necessarily towards the same countries (see Section 2.4.1).
- The clustering coefficient per vertex whose degree is k, i.e. the $c(k)$ defined in (1.4) of the various countries decreases with their degree k. This behaviour is well

fitted by a function of the kind $c(k) \propto k^{\omega}$, with $\omega \propto 0.7 \pm 0.05$. The clustering coefficient averaged over the whole network is $c = 0.65$, larger than the value corresponding to a random network with similar edges and vertices (Serrano and Boguñá, 2003).

- The average degree of the neighbours of one county decreases with the degree of the country (see below assortativity) with a power-law decay $knn(k) \propto k_k^{\nu}$, with $\nu_k \simeq 0.5 \pm 0.05$ (Serrano and Boguñá, 2003).

Generate the aggregate network

```
#importing the main modules

import networkx as nx

#countries to be excluded
exclude_countries=[472,899,471,129,221,97,697,492,838,473,536,\
637,290,527,577,490,568,636,839,879,0]

#this is the magic command to have the graphic embedded
#in the notebook
%pylab inline

DG=net_symmetrisation("data/comtrade_trade_data_total_2003.csv", \
                      exclude_countries)
print "number of nodes", DG.number_of_nodes()
print "number of edges", DG.number_of_edges()

#OUTPUT
number of nodes 232
number of edges 27901
```

2.4 Neighbour quantities: reciprocity and assortativity

2.4.1 Reciprocity

The reciprocity between two vertices in a directed graph is a quantity measuring the probability of having edges in both directions between two vertices. In a complete reciprocal case if vertex A has an edge towards B, then B must reciprocate with an edge towards A. Of course we cannot define this quantity in undirected graphs. From a different point of view, we can say that this is because any undirected link establishes a reciprocal connection between the vertices involved. In the case of directed graphs the situation is different: having an edge from A to B does not ensure that also the opposite edge is present (and actually this is seldom the case in real networks). The economic meaning of such a quantity can be expressed as a measure of how much the economies of the two countries are interconnected, or rather it measures how much one depends on the other to fulfil its needs. If the graph is also weighted, reciprocity is no more the simple exchange of an edge; but there is another quantity to be reciprocated, that is, the weight of the edge. In the case considered in this chapter we can think of one edge as the export from A to B. Since the export is measured in dollars a complete reciprocity would be obtained if there is also a comparable (similar in amount) export from B to A.

Intuitively, the reciprocity in a network should take into account the likelihood with which if we have an edge from one vertex i to another vertex j we also have its counterpart going from j to i. The most obvious way to measure this probability is to check the frequency with which we have edges pointing in both directions. This is done by defining the ratio r between the number of reciprocal links $L^{\leftrightarrow}$ and the total number of links L,

$$r = \frac{L^{\leftrightarrow}}{L}. \tag{2.3}$$

When no reciprocity is present we have $r = 0$, and when every link is reciprocated we have $r = 1$. Apart from these limit cases the value of r is between 0 and 1 ($0 < r < 1$). Note that r also counts loops (self-links) as reciprocal edges; in this case, the correct normalisation would be L minus the number of loops.

As in other cases of network theory, we are typically interested not in the reciprocity itself, but in the possible deviations from an "expected" or "typical" reciprocity (the one we can measure in a directed random graph). On this point we have to bear in mind that as the density increases, also the reciprocity increases, because it becomes more probable to have reciprocal links (Garlaschelli and Loffredo, 2004b).

Another measure of reciprocity, ρ, can be defined based on statistical considerations. In order to do so we first start from the quantity

$$\bar{a} \equiv \frac{\sum_{i \neq j} a_{ij}}{N(N-1)} = \frac{L}{N(N-1)} \tag{2.4}$$

which measures the ratio of observed to possible directed links (link density). In this way the self-linking loops are now excluded from the normalisation L.

The new reciprocity measure ρ can now be written in the following form:

$$\rho = \frac{r - \bar{a}}{1 - \bar{a}}. \tag{2.5}$$

The new definition of reciprocity gives an absolute quantity which allows one to directly distinguish between reciprocal ($\rho > 0$) and antireciprocal ($\rho < 0$) networks, with mutual links occurring more and less often than randomly respectively.

- If all the links occur in reciprocal pairs $r = 1$ and $\rho = 1$.
- If instead $r = 0$ then $\rho = \rho_{min}$, where $\rho_{min} \equiv \frac{-\bar{a}}{1-\bar{a}}$.

This is another advantage of using ρ, because it incorporates the idea that complete antireciprocal is more statistically significant in networks with larger density, while it has to be regarded as a less pronounced effect in sparser networks.

```
#unweighted case
N=DG.number_of_nodes()
L=DG.number_of_edges()

r=float((2*L-N*(N-1)))/L

print r

#OUTPUT
0.079208630515

#weighted case
W=0
W_rep=0
for n in DG.nodes():
    for e in DG.out_edges(n,data=True):
        W+=e[2]['weight']
        if DG.has_edge(e[1],e[0]):
            W_rep+=min(DG[e[0]][e[1]]['weight'],DG[e[1]][e[0]] \
                       ['weight'])

print W,W_rep,W_rep/W

#OUTPUT
7.17766475925e+12 5.19627606057e+12 0.723950788293
```

2.4.2 Assortativity

Another two-vertices property of a network is given by its assortativity. The assortativity coefficient measures the tendency of a vertex to be connected to others with a similar/dissimilar values of degree. In the former case the network is said to be "assortative". If instead in the network (on average) the hubs are connected with vertices of low degree, the whole network is said to be"disassortative". This quantity is measured practically by computing the average degrees $K_{nn}(k)$ of the neighbours of a vertex whose degree is k. To compute this quantity let's first compute the average degree $K_{nn}(i)$ of the neighbours of a vertex i,

$$K_n n(i) = \frac{\sum_{\langle ji \rangle} k_j}{n_j}, \tag{2.6}$$

where j is a neighbour of i. Then let's average again on all the vertices i whose degree is k,

$$K_n n(k) = \frac{\sum_{i:k_i=k} K_{nn}(i)}{n_k}, \tag{2.7}$$

where n_k is the number of the vertices with degree k. In assortative networks this is an increasing function of k, while in disassortative ones it decreases with k. Another measure of assortativity is given by the assortativity coefficient that measures the correlation coefficient of the degrees of neighbour sites, normalised with the variance of the degree distribution. To compute the correlation between the value of degrees k_i and k_j of two neighbour vertices i and j we need to introduce the joined probability $P(k_i, k_j)$, related to the frequency with which we measured these two values of degrees together in a graph. At this point we can write the correlation function as

$$\langle k_i k_j \rangle - \langle k_i \rangle \langle k_j \rangle = \sum_{k_i, k_j} k_i k_j (P(k_i, k_j) - P(k_i) P(k_j)), \tag{2.8}$$

since the variance of the $P(k)$ is

$$\sigma^2 = \sum_k k^2 P(k) - (\sum_k k P(k))^2, \tag{2.9}$$

and we can write the assortative coefficient as

$$r = \frac{1}{\sigma^2} \sum_{k_1, k_2} k_1 k_2 (P(k_1, k_2) - P(k_1) P(k_2)). \tag{2.10}$$

In a network with no assortativity we can factorise the joint probability, so that

$$P(k_1, k_2) = P(k_1) P(k_2) \tag{2.11}$$

and the coefficient is zero. Positive values of r signal assortative mixing. Disassortativity corresponds to negative values of r.

Assortativity

```
#K_nn distribution
list_Knn=[]
for n in DG.nodes():
    degree=0.0
    for nn in DG.neighbors(n):
        degree=degree+DG.degree(nn)
    list_Knn.append(degree/len(DG.neighbors(n)))

#plot the histogram
hist(list_Knn,bins=12)

#basic Pearson correlation coefficient for the
r=nx.degree_assortativity_coefficient(DG)
print r
```

HS Code	Commodity	w_{ij}	Density	NS_{in}/ND_{in}	NS_{out}/ND_{out}
9	Coffee	0.309	3.3811	2.553	2.3906
10	Cereals	0.1961	5.5195	5.9919	2.5718
27	Min. Fuels	0.3057	3.3575	2.6786	3.2979
29	Org. Chem.	0.3103	3.3664	2.3579	1.6286
30	Pharmaceutical	0.3662	2.803	2.3308	1.267
39	Plastics	0.4926	2.0478	1.753	1.1385
52	Cotton	0.2864	3.5839	2.7572	2.1254
71	Prec. Metals	0.2843	3.6746	1.9479	2.6704
72	Iron	0.3081	3.3315	2.5847	1.8484
84	Nuclear Machin.	0.6195	1.6281	1.3359	1.0259
85	Electric Machin.	0.5963	1.6917	1.3518	1.0692
87	Vehicles	0.4465	2.259	1.7488	1.1105
90	Optical Instr.	0.4734	2.1492	1.5879	1.0993
93	Arms	0.1415	8.4677	6.0618	4.0279

Table 2.1 Density and node-average of topological properties of commodity-specific networks vs. aggregate trade network for the 14 most relevant commodity classes in year 2003. Ratios of the statistic value in the commodity-specific network to aggregate network are showed. Values larger (smaller) than 1.0 mean that average of commodity-specific networks is larger (smaller) than its counterpart in the aggregate network.

```
#weighted version
r=nx.degree_pearson_correlation_coefficient(DG,weight='weight', \
                                            x='out',y='out')
print r

#OUTPUT
-0.335002643638
-0.0696781960521
```

2.5 Multigraphs

Multigraphs turns out to be useful in describing trade. In practice, we can imagine having different layers of products where on every layer the vertices are connected by different kinds of edges (product). For example, the UK and the USA will then be connected by more than one edge, where every edge accounts for a different commodity traded. The aggregate weighted, directed Word Trade Web or International Trade Network is obtained by simply summing the all-commodity-specific layers. A topological analysis of the multinetwork structure allows us to assess the commodity heterogeneity in commodity-specific networks, as compared to those of the aggregate network. Following previous analysis (Barigozzi *et al.*, 2010) we can conclude that WTW (ITN) is composed of layers with rather heterogeneous properties. By considering ten of the most relevant commodities exchanged, as shown in Table 2.1, we can find a rather different behaviour with respect to degree, density, weight and strength.

Weighted networks, the strength

```
dic_product_networks={}
commodity_codes=['09','10','27','29','30','39','52','71','72','84', \
'85','87','90','93']
for c in commodity_codes:
    dic_product_networks[c]=net_symmetrisation( \
    "data/comtrade_trade_data_2003_product_"+c+".csv", \
    exclude_countries)

DG_aggregate=net_symmetrisation( \
"data/comtrade_trade_data_total_2003.csv",exclude_countries)

#rescale the weighted adjacency matrices
#aggregate
w_tot=0.0
for u,v,d  in DG_aggregate.edges(data=True):
    w_tot+=d['weight']
for u,v,d  in DG_aggregate.edges(data=True):
    d['weight']=d['weight']/w_tot
#products
for c in commodity_codes:
    l_p=[]
    w_tot=0.0
    for u,v,d  in dic_product_networks[c].edges(data=True):
        w_tot+=d['weight']
    for u,v,d  in dic_product_networks[c].edges(data=True):
        d['weight']=d['weight']/w_tot

density_aggregate=DG_aggregate.number_of_edges() / \
(DG_aggregate.number_of_nodes()*(DG_aggregate.number_of_nodes()-1.0))

w_agg=[]
NS_in=[]
NS_out=[]
for u,v,d in DG_aggregate.edges(data=True):
    w_agg.append(d['weight'])
for n in DG_aggregate.nodes():
    if DG_aggregate.in_degree(n)>0:
        NS_in.append(DG_aggregate.in_degree(n,weight='weight')/ \
                     DG_aggregate.in_degree(n))
    if DG_aggregate.out_degree(n)>0:
        NS_out.append(DG_aggregate.out_degree(n,weight='weight')/ \
```

```
                        DG_aggregate.out_degree(n))

for c in commodity_codes:
    density_commodity=dic_product_networks[c].number_of_edges() / \
    (dic_product_networks[c].number_of_nodes()* \
    (dic_product_networks[c].number_of_nodes()-1.0))
    w_c=[]
    NS_c_in=[]
    NS_c_out=[]
    for u,v,d  in dic_product_networks[c].edges(data=True):
        w_c.append(d['weight'])
    for n in dic_product_networks[c].nodes():
        if dic_product_networks[c].in_degree(n)>0:
            NS_c_in.append(dic_product_networks[c].in_degree (n, \
            weight='weight')/dic_product_networks[c].in_degree(n))
        if dic_product_networks[c].out_degree(n)>0:
            NS_c_out.append(dic_product_networks[c].out_degree(n, \
            weight='weight')/dic_product_networks[c].out_degree(n))

    print c,str(round(density_commodity/density_aggregate,4))+ \
    " & "+str(round(mean(w_c)/mean(w_agg),4))+" & "+ \
    str(round(mean(NS_c_in)/mean(NS_in),4))+" & "+ \
    str(round(mean(NS_c_out)/mean(NS_out),4))

#OUTPUT
09 0.309 & 3.3811 & 2.553 & 2.3906
10 0.1961 & 5.5195 & 5.9919 & 2.5718
27 0.3057 & 3.3575 & 2.6786 & 3.2979
29 0.3103 & 3.3664 & 2.3579 & 1.6286
30 0.3662 & 2.803 & 2.3308 & 1.267
39 0.4926 & 2.0478 & 1.753 & 1.1385
52 0.2864 & 3.5839 & 2.7572 & 2.1254
71 0.2843 & 3.6746 & 1.9479 & 2.6704
72 0.3081 & 3.3315 & 2.5847 & 1.8484
84 0.6195 & 1.6281 & 1.3359 & 1.0259
85 0.5963 & 1.6917 & 1.3518 & 1.0692
87 0.4465 & 2.259 & 1.7488 & 1.1105
90 0.4734 & 2.1492 & 1.5879 & 1.0993
93 0.1415 & 8.4677 & 6.0618 & 4.0279
```

Starting from this static analysis of every layer that can be extended to various network quantities we can now move to the analysis of cross-product correlations.

2.6 The bipartite network of products and countries

A simple way to look at the matrix M_{cp} of countries and products is to disregard for a moment information on volume of production and to transform the weighted elements M_{cp} giving the flow of US dollars in c for the trade of product p into a binary variable, specifying only whether the country is an effective producer of the product p. The criterion adopted in order to understand whether a country can be considered, or not, a producer of a particular product is the so-called Balassa's revealed comparative advantage (RCA) (Balassa, 1965). Indeed, an export relevant for a country is not necessarily relevant also for the counterpart and vice versa. Therefore, it is necessary to weigh export of a good in relation to how much of the same product is produced worldwide, i.e. $\sum_{c'} M_{c'p}$.

This must be compared with the importance of the export of single country, which is again a ratio between the total export of c (i.e. $\sum_{p'} M_{cp'}$) with respect to the global value of the exports for every country (i.e. $\sum_{c',p'} M_{c'p'}$). In formulas, we get

$$RCA_{cp} = \frac{M_{cp}}{\sum\limits_{p'} M_{cp'}} \Big/ \frac{\sum\limits_{c'} M_{c'p}}{\sum\limits_{c',p'} M_{c'p'}}. \tag{2.12}$$

We consider country c to be a competitive exporter of product p if its RCA is greater than some threshold value. In the standard economics literature this value is taken as one and small variations around such a threshold do not qualitatively change the results (Hidalgo and Hausmann, 2009).

Revealed comparative advantage

```
def RCA(c,p):
    X_cp=dic_product_networks[p].out_degree(c,weight='weight')
    X_c=DG_aggregate.out_degree(c,weight='weight')

    X_p=0.0
    for n in dic_product_networks[p].nodes():
        X_p+=dic_product_networks[p].out_degree(n,weight='weight')

    X_tot=0.0
    for n in DG_aggregate.nodes():
        X_tot+=DG_aggregate.out_degree(n,weight='weight')

    RCA_cp=(X_cp/X_c)/(X_p/X_tot)

    return RCA_cp

p='93'
```

```
c=381
print RCA(c,p)

#OUTPUT
2.10470555164
```

Once we have the data in the form of a binary matrix, we can extract features by means of spectral theory. In order to do so, we define two complementary graphs corresponding to projection of the original bipartite network on the country nodes and on the products nodes. In that way the projected graphs with respectively N_c and N_p nodes are homogeneous with respect to the two different types of nodes. The easiest way to perform this projection is to consider the following two matrix products:

$$\begin{aligned} C &= MM^T \\ P &= M^T M, \end{aligned} \tag{2.13}$$

where M^T is the transposed matrix and the square matrices C and P define the country–country network and the product–product network. The element $C_{cc'}$ defines the weight associated to the link between countries c and c' in the country–country network. Analogously $P_{pp'}$ gives the weight of the link between products p and p' in the product product network. These weights have an interesting interpretation: if we write explicitly the expression of a generic element of the C matrix according to (2.13), we have that $C_{cc'} = \sum_p M_{cp} M_{c'p}$. Therefore the element $C_{cc'}$ (since M_{cp} is a binary unweighted matrix) counts the number of products exported by both countries c and c'. In a similar way the the element $P_{pp'}$ counts the number of countries which export both products p and p'. The diagonal elements C_{cc} and P_{pp} are respectively the number of products exported by the country c and the number of exporters of the product p.

Bipartite Networks. As shown in Fig. 2.1 bipartite networks are networks composed of two distinct sets, where links connect only elements of one set with elements of the other. Typically, bipartite networks arise in social systems where one set is formed by people and the second set by the object of their collaboration. Among the many examples we have:

- the actor–movie network, where actors are the nodes in one set, and they are connected to the films in which they played, forming the nodes in the second set (Albert and Barabási, 2002);
- scientist networks that can be defined as having in one set scientists and in the other their papers (Newman, 2003);
- directors of companies and the boards in which they sit (Caldarelli *et al.*, 2004);
- and finally countries of the world and the products they produce (Caldarelli *et al.*, 2012; Tacchella *et al.*, 2012).

Any two columns of such a dataset can be adopted to define a bipartite network, the properties of which could reveal patterns and regularities worthy of attention. More generally bipartite graphs can be defined whenever it is possible to partition the graph

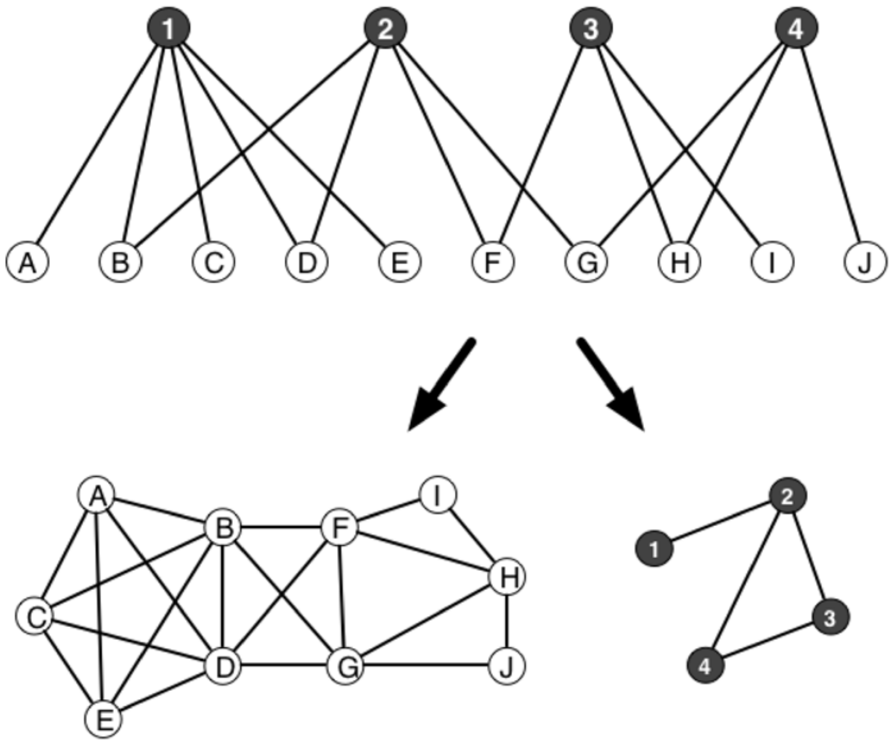

Fig. 2.1 An example of a bipartite network with the two possible projections. Note that an edge between vertex i and j in the projection can be weighted. The weight is given by the number of vertices of the other set connected to both i and j.

into two sets. This is always possible whenever the graph has no odd cycles, as shown in Fig. 2.2. The standard way to study such structures is by restricting to one of the two sets. In this way we can build a network of actors where the connection is given by having played in one or more films, or a network of films if they share one or more actors. We can keep track of the number of connections by considering a weighted graph, so that one connection between two actors is an integer number representing the total of the films they have in common. Some of the total information is then "projected" in the subspace of one of the two sets. Reasons for doing this are, for example, the need to assess the centrality (how crucial it is for the structure of the whole network) of one node with respect to the others or to determine clusters of vertices that share similar properties (community detection).

Computing the bipartite network and projections

```
import numpy as np

num_countries=DG_aggregate.number_of_nodes()
num_products=len(commodity_codes)
```

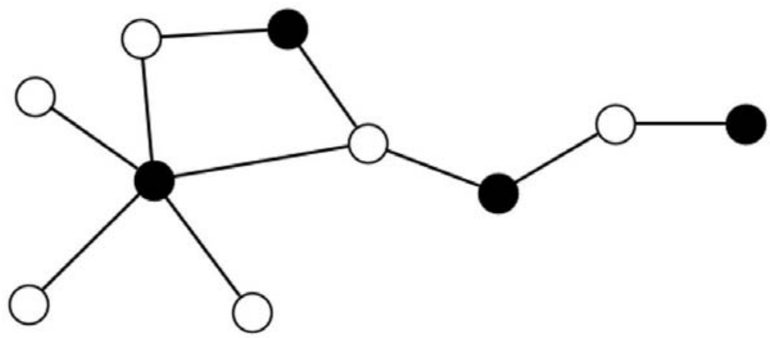

Fig. 2.2 An example of a graph that can be partitioned into two sets of white and black vertices.

```
#generate array indices
country_index={}
i=0
for c in DG_aggregate.nodes():
    country_index[c]=i
    i+=1

M=np.zeros((num_countries,num_products))

for pos_p,p in enumerate(commodity_codes):
    for c in dic_product_networks[p].nodes():
        if RCA(c,p)>1.0:
            M[country_index[c]][pos_p]=1.0
    print "\r"

C=np.dot(M,M.transpose())
P=np.dot(M.transpose(),M)

print C
print P

#OUTPUT
[[ 2.  1.  1. ...,  0.  1.  1.]
 [ 1.  4.  2. ...,  0.  3.  1.]
 [ 1.  2.  3. ...,  0.  2.  1.]
 ...,
 [ 0.  0.  0. ...,  1.  0.  0.]
 [ 1.  3.  2. ...,  0.  4.  1.]
 [ 1.  1.  1. ...,  0.  1.  2.]]
[[ 83.  27.  28.   4.   6.   6.  29.  31.  20.   1.  3.  3.  5. 12.]
```

```
[ 27.  59.  19.   4.   4.   8.  27.  18.  19.   5.  3.  7.  3. 12.]
[ 28.  19.  71.   4.   2.   7.  20.  16.  14.   3.  4.  4.  1.  9.]
[  4.   4.   4.  20.   9.   9.   2.   6.   5.   5.  4.  3.  7.  7.]
[  6.   4.   2.   9.  27.  15.   7.   6.  10.   9.  3.  8.  9. 10.]
[  6.   8.   7.   9.  15.  37.  10.   7.  15.  10. 10.  8.  9. 11.]
[ 29.  27.  20.   2.   7.  10.  69.  19.  18.   4.  5.  7.  5. 14.]
[ 31.  18.  16.   6.   6.   7.  19.  57.  10.   4.  3.  4.  6.  9.]
[ 20.  19.  14.   5.  10.  15.  18.  10.  56.   7.  7. 12.  2. 15.]
[  1.   5.   3.   5.   9.  10.   4.   4.   7.  26. 12.  9.  7.  6.]
[  3.   3.   4.   4.   3.  10.   5.   3.   7.  12. 26.  5.  8.  6.]
[  3.   7.   4.   3.   8.   8.   7.   4.  12.   9.  5. 27.  5. 11.]
[  5.   3.   1.   7.   9.   9.   5.   6.   2.   7.  8.  5. 20.  5.]
[ 12.  12.   9.   7.  10.  11.  14.   9.  15.   6.  6. 11.  5. 38.]]
```