Supplementary Information for "Revealing the Hidden Language of Complex Networks"

Ömer Nebil Yaveroğlu[1], Noël Malod-Dognin[1], Darren Davis[2], Zoran Levnajic[1], Vuk Janjic[1], Rasa Karapandza[3], Aleksandar Stojmirovic[4,5], & Nataša Pržulj[1]

[1] *Department of Computing, Imperial College London, UK*

[2] *Computer Science Department, University of California, Irvine, USA*

[3] *Department of Finance, Accounting & Real Estate EBS Business School, Germany*

[4] *National Center for Biotechnology Information (NCBI), USA*

[5] *Janssen Research and Development, LLC, Spring House, PA, USA*

1 Data

Synthetic networks In order to evaluate how well different network distance measures identify network models, and to characterize the underlying topology of real-world networks, we generated random networks using the following seven random network models:

1. The **Erdös-Rènyi random model (ER)** represents uniformly distributed random interactions [4]. An ER network is generated by fixing the number of nodes in the network, and then by randomly adding edges between uniformly chosen pairs of nodes until a given density is reached.

2. The **Generalized random model (ER-DD)** is an extension of the ER model, where the

degree distribution of the nodes in the generated network is forced to match the one of an input network [4]. An ER-DD network is generated by first randomly assigning stubs (i.e. connection capacities) to the nodes of the network, and then adding edges between nodes that have available stubs uniformly at random while reducing the number of available stubs of the connected nodes after each edge addition.

3. The **Geometric model (GEO)** represents the proximity relationship between uniformly distributed points in a d-dimensional space [16]: two nodes are connected by an edge if the Euclidean distance between the corresponding points is smaller than a distance threshold r. Here, a GEO network is generated by uniformly distributing points in 3-dimensional space, and the distance threshold is chosen so as to obtain a given edge density.

4. The **Geometric models with gene duplication (GEO-GD)** is a geometric model in which the points are distributed according to a duplication rule, mimicking the gene duplication process in biology [20]. A GEO-GD network is generated from a small initial seed network (i.e., a single edge), to which the duplication process is applied: a randomly chosen parent node is duplicated, and the new node is randomly placed within the distance of $2r$ from its originating node (where r is the same distance threshold parameter used in the definition of the GEO model). The duplication process iterates until the required number of nodes is generated, after which the edges are placed according to GEO model rules so as to achieve the requested edge density.

5. The **Barabàsi-Albert Scale-free (SF-BA) model** — also called preferential attachment — generates networks with scale-free topology, which is characterized by power law degree

distributions [15]. A SF-BA network starts from a small seed network and nodes are added iteratively based on the "rich-gets-richer" principle: new nodes are attached existing nodes in accordance with attachment probabilities, which correspond to the degrees of existing nodes within the network.

6. The **Scale-free model with gene duplication and divergence (SF-GD)** is a scale-free model which mimics the gene duplication and the gene divergence processes from biology [19]. A SF-GD model is generated from a small initial seed network (i.e., a single edge) which grows iteratively through the processes of duplication and divergence events: in each iteration, an existing node v is randomly chosen and the new node, v', is connected to all neighbours of v; also, an edge is placed between v and v' with probability p. In a divergence step, we consider all nodes u that are connected to both v and v', and the edge (u, v) or the edge (u, v') is removed with the probability q. Here, $p = 0.5$ and the value of q is set in accordance with the number of edges in the data network.

7. The **Stickiness-index based model (STICKY)** is based on the assumption that the higher the degrees of two nodes, the more likely they are to be neighbours [21]. A STICKY network is generated by randomly assigning stickiness-index values (which are proportional to the node degrees of an input network) to all nodes. Then, the probability of connecting two nodes is defined as the product of their stickiness-indices.

For evaluating the clustering performance of network distance measures, we generated model networks with $1000, 2000, 4000$ and 6000 nodes, and edge-densities of $0.5\%, 0.75\%$ and 1%. These specific values were chosen as they represent the range of sizes and densities observable in real-world networks. We generated 30 random networks for each node count, edge density and network model combination, producing a total of $2,520$ networks: 4 (node counts) $\times 3$ (edge-densities) $\times 7$ (network models) $\times 30$ (network instances) = $2,520$ model networks. For models that required a pre-defined degree distribution, we used SF-BA networks with same number of nodes and edge-densities. For identifying real-world networks, we generated model networks with same node counts and edge densities (and, where relevant, degree distributions) as the real-world networks.

Real-world networks We use five types of real-world networks, which describe interaction data from different fields: economy, social sciences, Internet packet routing and two from biology. Their properties are given in Table **S1**. The real-world networks and their descriptions are as follows:

1. **Autonomous systems networks** describe communications between routers connected to the Internet. Each autonomous system is a subset of these routers and the information exchange between the autonomous systems forms a "who-talks-to-whom" network. The 733 autonomous networks in our study were obtained from SNAP database [*]. Border Gateway Protocol (BGP) was used for logging the traffic as part of the Oregon Route Views project, and each network represents daily communication data between autonomous systems for the time period between 8th November 1997 and 26th May 2001.

2. **Facebook networks** capture friendship relationships between Facebook users that are asso-

[*] Data on router traffic of Oregon University between 09.09.1997 and 02.01.2000. Downloaded on 09.08.2012 from: http://snap.stanford.edu/data/as.html.

ciated to a specific university. The 98 Facebook networks in our study were obtained from the study of Traud et al. [†] and represent data collected from 98 American Universities in Sept. 2005.

3. **Metabolic networks** represent bio-chemical reactions between enzymes and metabolites inside a cell. The $2,301$ metabolic networks used in these studies represent enzyme-enzyme interactions, where two enzymes are connected by an edge if they catalyse reactions that share a common metabolite. We obtained the metabolic networks of all species from the Kyoto Encyclopaedia of Genes and Genomes (KEGG) database in February 2013 [‡]. We filtered out networks which contained less than 100 nodes.

4. **Protein Structure networks** represent interactions between amino acids on a protein. Two amino-acids are said to interact if the Euclidean distance between their alpha-carbons is smaller than 7.5Å. We generated the networks of all the protein structures of the Astral_40 compendium v1.75B (proteins with less than 40% of sequence identity, and at least 100 amino-acids) [§].

5. **World trade networks** represent trading relations between countries. Using commodity trade data from the United Nations Commodity Trade Statistics (UN Comtrade) database [17], we generated 49 trade networks, one for each year between 1962 and 2010. The fact that most countries have both import and export trade makes the trade network inherently directional. However, since we are only interested in the presence or absence of an interaction

[†] A. L. Traud, P. J. Mucha, M. A. Porter, Physica A: Statistical Mechanics and its Applications 391,4165 (2012).

[‡] KEGG database Release 65.0. url: www.genome.jp/kegg/. Downloaded on: 08.02.2013.

[§] Data obtained from Astral 40 compendium v1.75B in January 2011: http://scop.berkeley.edu/astral/

Supplementary Table S1: **Network properties of the real-world networks.**

	Number of	Number of Nodes			Edge Densities (%)		
Network Type	Networks	Min.	Med.	Max.	Min.	Med.	Max.
Autonomous Systems	733	103	4180	6474	0.06	0.09	4.55
Facebook	98	769	9949	41554	0.16	0.78	5.70
Metabolic	2301	100	366	705	0.74	1.17	3.39
Protein Structure	8226	100	178	1419	0.47	3.75	8.31
World Trade	49	86	103	125	8.72	11.64	13.53

between countries, we generated undirected networks and weighted the edges by summing import and export trade volumes. We threshold the network by removing the lowest weighted edges until 90% of the total trade in the network remains. The topology of these thresholded trade networks is non-random (not ER). In addition to these "total" trade networks, we include in our analysis 10 trade networks for specific commodities defined in the Standard International Trade Classification (SITC) Rev.1.

Crude oil prices and economic indicators of countries We obtained the *crude oil prices* for all years between 1962 and 2010 from UNCTADSTAT Reports [¶] (downloaded in November 2012). As the crude oil prices in that data set are given on a monthly basis, we compute the crude oil price of a year as the average price of the corresponding 12 months.

We obtain the *economic indicators* of country wealth from PENN World Table (PENN) [31] (version 7.1; downloaded in November 2011) and International Monetary Fund World Economic Outlook Database (WEO) [32] (downloaded in October 2012). All prices were expressed in 2005 US

[¶] United Nations Conference on Trade and Development (UNCTADSTAT) database, http://unctadstat.unctad.org. Accessed: 03/11/2012

Dollars. The list of used economic indicators and their definitions is as follows:

- **Gross Domestic Product - version 1 (RGDPL):** Purchasing Power Parity converted Gross Domestic Product Per Capita (Laspeyres), derived from the growth rates of consumption share, government consumption share, and investment share. This data is from PENN.
- **Gross Domestic Product - version 2 (RGDPL2):** Purchasing Power Parity converted Gross Domestic Product Per Capita (Laspeyres), derived from growth rates of domestic absorption. This data is from PENN.
- **Gross Domestic Product - version 3 (RGDPCH):** Purchasing Power Parity converted Gross Domestic Product Per Capita (Chain Series). This data is from PENN.
- **Consumption Share (KC):** Consumption Share of Purchasing Power Parity Converted Gross Domestic Product Per Capita at 2005 constant prices (RGDPL). This data is from PENN.
- **Government Consumption Share (KG):** Government Consumption Share of Purchasing Power Parity Converted Gross Domestic Product Per Capita at 2005 constant prices (RGDPL). This data is from PENN.
- **Investment Share (KI):** Investment Share of Purchasing Power Parity Converted Gross Domestic Product Per Capita at 2005 constant prices (RGDPL). This data is from PENN.
- **Openness (OPENK):** Trade openness as a percent of 2005 constant prices. This data is from PENN.
- **Population (POP):** The total population of the country. This data is from WEO.

- **Level of Employment (LE):** The number of people who, during a specified brief period such as one week or one day, (a) performed some work for wage or salary in cash or in kind, (b) had a formal attachment to their job but were temporarily not at work during the reference period, (c) performed some work for profit or family gain in cash or in kind, (d) were with an enterprise such as a business, farm or service but who were temporarily not at work during the reference period for any specific reason. This data is from WEO.

- **Current Account Balance (BCA):** Current account is all transactions other than those in financial and capital items. The major classifications are goods and services, income and current transfers. The focus of the BOP is on transactions (between an economy and the rest of the world) in goods, services, and income. This data is from WEO.

KC, KI and KG are expressed in percentage of GDP per capita. We included copies of these indicators, converted into constant price per capita, i.e., multiplied by GDP per capita (e.g., KC × RGDPL). We also included copies of the indicators expressed in constant price per capita (also including RGDPL, RGDPL2, RGDPCH) but converted into raw constant price value – these are multiplied by the population (e.g., RGDPL × POP).

2 Methods

Standard node statistics

Degree. The degree of a node is defined as the number of connections it has to other nodes in the network.

Clustering coefficient. The clustering coefficient of a node u, c_u, is the fraction of triangles that touch the node over all possible triplets that can be formed by u and its neighbours.

Betweenness centrality. The betweenness centrality of a node u, bc_u, is the ratio of the number of shortest paths from all vertices to all others that pass through u over all shortest paths:

$$bc_u = \sum_{s \neq u \neq t} \frac{\delta_{st}(u)}{\delta_{st}}, \tag{S.1}$$

where δ_{st} is the number of shortest path from node s to node t, and $\delta_{st}(u)$ is the number of those paths that pass through u.

Closeness centrality. The closeness centrality of a node u, cc_u, is the average of the lengths of the shortest paths from u to all other nodes in the network:

$$cc_u = \frac{1}{\sum_{v=1}^{n} d_{uv}}, \tag{S.2}$$

where d_{uv} is the is the length of the shortest path from node u to node v.

Standard network distance measures We compare the model clustering performance of Graphlet Correlation Distance (GCD) with the model clusterings computed using degree distribution, clustering coefficient, diameter, Relative Graphlet Frequency Distance (RGFD) and Graphlet Degree Distribution Agreement (GDDA). These five network distance measures are defined below.

Degree distribution. The degree distribution of a network is the distribution of the degrees over all nodes. There are many standard ways of comparing two distributions (e.g, the Kolmogorov-Smirnov two-sample test). Here, we use the comparison measure given in [8], which is defined as follows. First, we scale and normalizes the given distributions, in order to reduce the contribution of higher degree nodes. The distance is then computed as the square root of sum of square errors of the two distributions. More specifically, given two degree distributions, d_G and d_H, the distance between these two distributions, $D(d_G, d_H)$, is:

$$S_G(k) = \frac{d_G(k)}{k}, \tag{S.3}$$

$$T_G = \sum_{k=1}^{\infty} S_G(k), \tag{S.4}$$

$$N_G(k) = \frac{S_G(k)}{T_G}, \tag{S.5}$$

$$D(d_G, d_H) = \frac{1}{\sqrt{2}} \sqrt{\sum_{k=1}^{\infty} (N_G(k) - N_H(k))^2}. \tag{S.6}$$

Clustering coefficient. The *clustering coefficient* of a network is the average of the clustering coefficients of all nodes in the network. The clustering coefficient distance of two networks is the absolute difference of their clustering coefficients.

Diameter. The diameter of a network is the maximum shortest path distance that is observed among all node pairs. The diameter distance of two networks is the absolute difference of their diameters.

Spectral Distance. The adjacency matrix, A, of an unweighted network G is an $n \times n$ matrix where $A[u,v]$ is equal to 1 when nodes u and v are connected, and equal to 0 otherwise. A is a symmetric matrix when G is undirected. The diagonal degree matrix, D, of a network is an $n \times n$ matrix whose diagonal elements are equal to the node degrees, $D(u,u) = d_u$ and other elements are all equal to 0. The standard combinatorial Laplacian matrix, L, of a network is computed from the adjacency and diagonal degree matrices as in Equation S.7.

$$L = D - A. \tag{S.7}$$

Spectral network theory explains the topology a network using the eigenvalues and eigenvectors of matrices associated to the network, such as its adjacency matrix or Laplacian matrix [5]. Let X be the matrix associated with the graph. The eigendecomposition $X = \phi\lambda\phi^T$ where $\lambda = diag(\lambda_1, \lambda_2, ..., \lambda_n)$ is the diagonal matrix with the ordered eigenvalues as elements and $\phi = (\phi_1|\phi_2|...|\phi_n)$ is the matrix with the ordered eigenvectors as columns. The spectrum is the set of eigenvalues $s = \{\lambda_1, \lambda_2, ..., \lambda_n\}$, where $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$.

In [5], the spectral distance between two networks G_1 and G_2 is defined as the Euclidean distance between their spectra (Equation S.8).

$$d_s(G_1, G_2) = \sqrt{\sum_i (s_i^{(1)} - s_i^{(2)})^2}. \tag{S.8}$$

When the spectra of two networks are different size, 0 valued eigenvalues are added into the smaller spectrum while preserving the correct magnitude ordering.

Wilson and Zhu [5] compare various spectral distance measures that are defined from different types of matrices, showing that the spectral distance between the Laplacian matrices of two networks is the best measure for classification and clustering purposes. Later on, Thorne and Stumpf [6] also use the spectral distance of Laplacian matrices for the analysis of the evolution in protein interaction networks. In parallel to these studies, we chose spectral distance from Laplacian matrices as the benchmark representing the performance of spectral distance measures against graphlet correlation distance.

Graphlets and graphlet-based network distance measures. *Graphlets* are small, connected, non-isomorphic, induced subgraph of a larger graph $G = (V, E)$ [13]. There are 30 graphlets with 2- to 5- nodes. Each graphlet contains "symmetrical nodes" which are said to belong to the same automorphism orbit (illustrated in the left panel of Supplementary Fig. S1). The automorphism orbits represent topologically different ways in which a graphlet can touch a node (illustrated in the right panel of Supplementary Fig. S1). All 30 graphlets and their 73 automorphism orbits are illustrated in Fig. 1-d. The *Graphlet Degree Vector* (GDV) of a node generalises the notion of a node's degree into a 73-dimensional vector [8] where each of the 73 components of that vector captures the number of times node n is touched by a graphlet at orbit i (where $i \in \{1, 2, \ldots 73\}$).

Relative Graphlet Frequency distance (RGFD) [13] quantifies the topological similarities of two networks based on the frequencies of the appearance of 3- to 5- node graphlets (the only 2-node graphlet, G_0, which captures the degree distribution, is not used). Let $N_i(G)$ represent the number of graphlets of type i in G and $T(G) = \sum_{i=1}^{29} N_i(G)$. Given two networks, G and H, the

RGFD distance between these networks, $D(G, H)$, is defined as:

$$D(G, H) = \sum_{i=1}^{29} \left| \log \frac{N_i(G)}{T(G)} - \log \frac{N_i(H)}{T(H)} \right| . \tag{S.9}$$

Graphlet Degree Distribution Agreement (GDDA) [8] uses the notion of graphlet degree distributions to compare two networks. The graphlet degree distribution of an orbit i is the distribution of the graphlet degrees for all nodes in the network. As the GDV of a node is defined as a 73-dimensional vector, there are 73 different graphlet degree distributions describing the topology of a network. Let $d_G^j(k)$ be the number of nodes in network G that touch k graphlets at orbit j. Given two networks, G and H, the distance between the j^{th} graphlet degree distributions of these networks, $D^j(G, H)$, is computed as follows:

$$S_G^j(k) = \frac{d_G^j(k)}{k} \tag{S.10}$$

$$T_G^j = \sum_k S_G^j(k) \tag{S.11}$$

$$N_G^j(k) = \frac{S_G^j(k)}{T_G^j} \tag{S.12}$$

$$D^j(G, H) = \frac{1}{\sqrt{2}} \sqrt{\sum_k (N_G^j(k) - N_H^j(k))^2} \tag{S.13}$$

The GDDA distance is defined as the arithmetic average of the 73 distances that are computed for each of the 73 graphlet automorphism orbits. The degree distribution distance is computed in a similar way to GDDA with the exception that it is computed only for the graphlet degree distributions of orbit 0.

Redundancies and Dependencies in Graphlet Degree Vectors Graphlet degrees coming from large graphlets are dependent on the graphlet degrees coming from smaller ones. The simplest of these dependencies occurs when two edges (orbit 0) are "combined" (also described in the main paper): given two adjacent edges, (a, b) and (a, c), the orbit touching a from the graphlet induced by $\{a, b, c\}$ is either orbit 3 if b and c are connected by an edge, or orbit 2 otherwise. Therefore, $\binom{C_0}{2}$ is equal to the sum of C_2 and C_3, where C_i represents the graphlet degree for orbit i.

Considering all combinations of 2, 3, and 4 node graphlets that produce graphlets of size ≤ 5, we obtain 17 independent orbit redundancy equations (that cannot be derived from other equations):

1. $\binom{C_0}{2} = C_2 + \mathbf{C_3}$
2. $\binom{C_2}{1}\binom{C_0-2}{1} = 3C_7 + 2C_{11} + \mathbf{C_{13}}$
3. $\binom{C_1}{1}\binom{C_0-1}{1} = C_5 + 2C_8 + C_{10} + 2\mathbf{C_{12}}$
4. $\binom{C_3}{1}\binom{C_0-2}{1} = C_{11} + 2C_{13} + 3\mathbf{C_{14}}$
5. $\binom{C_4}{1}\binom{C_0-1}{1} = \mathbf{C_{16}} + C_{29} + 2C_{34} + 2C_{36} + 2C_{46} + C_{51} + 2C_{52} + C_{59}$
6. $\binom{C_5}{1}\binom{C_0-2}{1} = 2\mathbf{C_{21}} + C_{26} + 2C_{30} + 2C_{38} + 2C_{47} + C_{48} + C_{53} + C_{60}$
7. $\binom{C_6}{1}\binom{C_0-1}{1} = \mathbf{C_{20}} + C_{32} + C_{37} + C_{40} + 2C_{49} + 2C_{54}$
8. $\binom{C_7}{1}\binom{C_0-3}{1} = 4\mathbf{C_{23}} + 2C_{33} + C_{42} + C_{55}$
9. $\binom{C_8}{1}\binom{C_0-2}{1} = \mathbf{C_{38}} + 3C_{50} + C_{53} + 2C_{63} + C_{64} + C_{68}$

10. $\binom{C_9}{1}\binom{C_0-1}{1} = \mathbf{C_{28}} + C_{43} + C_{51} + C_{59} + 2C_{62} + 2C_{65}$

11. $\binom{C_{10}}{1}\binom{C_0-2}{1} = \mathbf{C_{26}} + 2C_{41} + C_{48} + C_{53} + 2C_{57} + C_{60} + 2C_{64} + 2C_{66}$

12. $\binom{C_{11}}{1}\binom{C_0-3}{1} = 2C_{33} + 2C_{42} + 4\mathbf{C_{44}} + 3C_{58} + 2C_{61} + C_{67}$

13. $\binom{C_{12}}{1}\binom{C_0-2}{1} = \mathbf{C_{47}} + C_{60} + C_{63} + C_{66} + 2C_{68} + 3C_{70}$

14. $\binom{C_{13}}{1}\binom{C_0-3}{1} = C_{42} + 3C_{55} + 2C_{61} + 2C_{67} + 4\mathbf{C_{69}} + 2C_{71}$

15. $\binom{C_1}{2} = C_6 + C_8 + C_9 + C_{12} + \mathbf{C_{17}} + C_{25} + C_{34} + C_{37} + C_{40} + 2C_{49} + C_{51} + C_{52} + 2C_{54} +$
$C_{59} + 2C_{62} + 2C_{65}$

16. $\binom{C_3}{2} = C_{13} + 3C_{14} + C_{44} + C_{61} + C_{67} + 2C_{69} + 2C_{71} + 3\mathbf{C_{72}}$

17. $\binom{C_2}{1}\binom{C_3}{1} = 2C_{11} + 2C_{13} + C_{33} + 2C_{42} + 3C_{55} + 3C_{58} + C_{61} + 2C_{67} + \mathbf{C_{71}}$

Additional 9 equations illustrate redundancies, but they can be derived from the above listed 17 independent equations and are given below for illustrative purposes only:

18. $\binom{C_0}{3} = C_7 + C_{11} + C_{13} + C_{14}$

19. $\binom{C_0}{4} = C_{23} + C_{33} + C_{42} + C_{44} + C_{55} + C_{58} + C_{61} + C_{67} + C_{69} + C_{71} + C_{72}$

20. $\binom{C_1}{1}\binom{C_0-1}{2} = C_{21} + C_{26} + C_{30} + 2C_{38} + C_{41} + 2C_{47} + C_{48} + 3C_{50} + 2C_{53} + C_{57} + 2C_{60} +$
$3C_{63} + 2C_{64} + 2C_{66} + 3C_{68} + 3C_{70}$

21. $\binom{C_2}{1}\binom{C_0-2}{2} = 6C_{23} + 5C_{33} + 4C_{42} + 4C_{44} + 3C_{55} + 3C_{58} + 3C_{61} + 2C_{67} + 2C_{69} + C_{71}$

22. $\binom{C_3}{1}\binom{C_0-2}{2} = C_{33} + 2C_{42} + 2C_{44} + 3C_{55} + 3C_{58} + 3C_{61} + 4C_{67} + 4C_{69} + 5C_{71} + 6C_{72}$

23. $\binom{C_{14}}{1}\binom{C_0-3}{1} = C_{58} + C_{67} + 2C_{71} + 4C_{72}$

24. $\binom{C_2}{2} = 3C_7 + C_{11} + 3C_{23} + 2C_{33} + C_{42} + 2C_{44} + C_{61} + C_{69}$

25. $\binom{C_1}{1}\binom{C_2}{1} = C_5 + 2C_8 + C_{21} + C_{26} + 2C_{38} + C_{41} + 2C_{47} + 3C_{50} + C_{53} + C_{60} + 2C_{63} + C_{68}$

26. $\binom{C_1}{1}\binom{C_3}{1} = C_{10} + 2C_{12} + C_{30} + C_{48} + C_{53} + C_{57} + C_{60} + C_{63} + 2C_{64} + 2C_{66} + 2C_{68} + 3C_{70}$

For example, $Eq.18$ is equivalent to $\frac{Eq.2+Eq.4}{3}$, when C_3 is replaced by using $Eq.1$:

- $\frac{Eq.2+Eq.4}{3}$: $\frac{C_2(C_0-2)+C_3(C_0-2)}{3} = C_7 + C_{11} + C_{13} + C_{14}$
- From $Eq.1$: $C_3 = \binom{C_0}{2} - C_2$
- Replacing C_3 by the term from Eq.1 in $\frac{Eq.2+Eq.4}{3}$:
  $\frac{C_2(C_0-2)+(\binom{C_0}{2}-C_2)(C_0-2)}{3} = C_7 + C_{11} + C_{13} + C_{14}$
- Simplifies to: $\frac{\binom{C_0}{2}(C_0-2)}{3} = C_7 + C_{11} + C_{13} + C_{14}$
- Which is exactly $Eq.18$: $\binom{C_0}{3} = C_7 + C_{11} + C_{13} + C_{14}$

Other equations from the above list, numbered 18-26, can be similarly derived from the 17 independent equations.

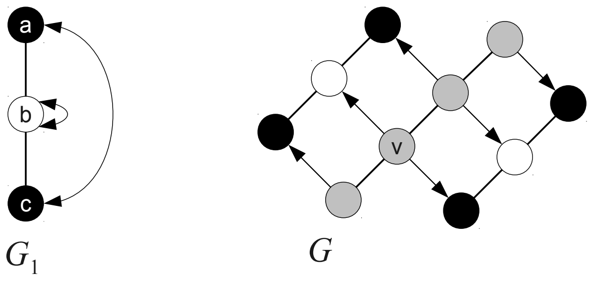
We use these equations to remove redundant orbits from Graphlet Degree Vectors, so they would not contain redundant information. Since there are 17 independent equations, we eliminate

17 orbits as redundant. However, we can make different choices about which orbits to eliminate. One set of redundant orbits that can be eliminated is written in bold in the first 17 equations. For 2- to 4-node graphlets, we can eliminate 4 orbits as redundant. We chose to eliminate orbits 3, 12, 13 and 14 using Equations 1, 2, 3, and 4. The remaining set of non-redundant orbits is illustrated in red in Fig. 1-d of the main paper. We validate that the choice of orbits does not change the results.

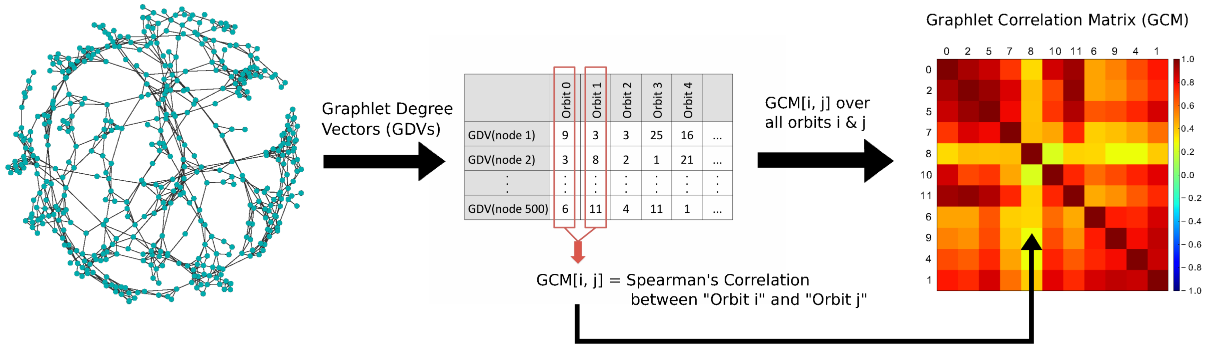
Graphlet Correlation Matrix / Distance As described in the main paper, Graphlet Correlation Matrix encodes the topology of a network based on correlations between various node properties contained in orbit counts, over all nodes. Given network G, we compute graphlet degree vectors of all nodes and construct a matrix where each row represents the graphlet degree vector of a node. We exploit the existence of dependencies between orbits by computing the Spearman's correlation coefficient among all pairs of orbits (i.e., among all columns of the matrix of graphlet degree vectors) and for the graphlet correlation matrix of the network, (GCM_G). Graphlet correlation matrix construction is illustrated in Supplementary Fig. S2, we used a random geometric graph with 500 nodes and 1% edge density in this illustration.

We use Spearman's correlation for measuring monotonic correlations between orbits because the scale in which graphlet degrees evolve are not directly comparable, as graphlet degrees from large graphlets are binomial functions of graphlet degrees from smaller graphlets. Thus, Pearson's correlation, which measures linear correlations, is not suitable.

It is possible that a graphlet does not appear in a network. In that case, graphlet degrees for all of the graphlet's orbits for all nodes are 0. As these orbit values do not change, Spearman's Cor-



Supplementary Figure S1: **Graphlets and automorphism orbits. Left panel:** graphlet G_1 contains three nodes (a, b and c) and three automorphisms represented by arrows (one that maps a to c, and another that maps b onto itself). **Right panel:** graph G (comprised of 4 gray nodes) has two induced graphlets G_1: first one maps node v to orbit $\{a, c\}$ (black nodes); second one maps node v to orbit $\{b\}$ (white node). These two mappings are topologically distinct: in the first case, v plays in G_1 the role of a node having degree equal to 1, while in the second case, it plays the role of a node having degree equal to 2.


Graphlet Degree Vectors (GDVs)
GCM[i, j] over all orbits i & j
Graphlet Correlation Matrix (GCM)
0
2
5
7
8
10
11
6
9
4
1
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
-0.8
-1.0

	Orbit 0	Orbit 1	Orbit 2	Orbit 3	Orbit 4	
GDV(node 1)	9	3	3	25	16	...
GDV(node 2)	3	8	2	1	21	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮
GDV(node 500)	6	11	4	11	1	...

GCM[i, j] = Spearman's Correlation between "Orbit i" and "Orbit j"

Supplementary Figure S2: **Illustration of Graphlet Correlation Matrix computation.** Given a geometric network, G, which has 500 nodes and 1% edge density (illustrated on the left), the graphlet degree vectors of all nodes are computed. The rows of the middle table present graphlet degree vectors of the nodes in the graph (each row contains the 73-dimensional graphlet degree vector of a node), and the distribution of the values in each column defines the graphlet degree distribution for orbit i, d_G^i. The graphlet degree distributions of orbit 0 and 1, d_G^0 and d_G^1 are highlighted in red. The graphlet correlation between orbits i and j, $GCM_G(i, j)$, is defined as the Spearman's correlation coefficient between d_G^i and d_G^j. By computing the $GCM_G(i, j)$ for all pairs of considered orbits, we obtain the symmetric graphlet correlation matrix of G, GCM_G.

relation coefficient cannot be computed for these orbits. To overcome this problem, we introduce a dummy graphlet degree vector, [1, 1, ..., 1] into the matrix of graphlet degree vectors. This small amount of noise resolves the Spearman's correlation coefficient computation problem. As a result, orbits for which all graphlet degrees are all 0 correlate perfectly (having Spearman's correlation coefficients of 1) while these orbits do not correlate with the rest of the non-zero orbits (having Spearman's correlation coefficients close to 0).

We define Graphlet Correlation Distance (GCD) between two networks as the Euclidean distance of the upper triangle values of their GCMs. Given GCMs of networks G and H, GCM_G and GCM_H, graphlet correlation distance between G and H is defined as:

$$GCD(G,H) = \sqrt{\sum_{i=1}^{d} \sum_{j=i+1}^{d} (GCM_G(i,j) - GCM_H(i,j))^2}. \tag{S.14}$$

When all orbits from 2- to 5-node graphlets are considered, the GCM of the network is a 73×73 symmetric matrix. We denote the GCD that is computed from this matrix as GCD-73. Similarly, when we compute the GCM of the network using only the non-redundant orbits of 2- to 4-node graphlets (the orbits coloured red in Fig. 1-d, we obtain an 11×11 symmetric matrix, which we denote by GCD-11.

Computational Complexities of Graphlet-based Distance Measures For GCD, RGFD and GDDA, we have to count the number of graphlets/graphlet degrees in the network. Given a network with n nodes, the worst case running time for counting all graphlets and graphlet degrees for 2- to k-node graphlets is O(n^k) and a tighter upper-bound is O(nd^{k-1}), where $d \leq n$ is the maximum degree over all nodes in the network.

For GCD, computing the Spearman's correlation coefficients between the orbits over n nodes is done in O($n \ln(n)$) time, and the Euclidean distance between two GCMs is computed in O(1) time. In RGFD, computing the differences between the number of graphlets is done in O(1) time. In GDDA, computing the differences between the normalized distributions of graphlet degrees is done in O(n) time, since each graphlet degree distribution contains up to n distinct values. The arithmetic average of these differences is then computed in O(1) time.

Hence, the time complexities are dominated by the complexity of counting graphlets. However, since GCD performs better when it uses up to 4-node graphlets rather than up to 5-node graphlets, it reduces the time complexity of GCD-based network comparison from O(nd^4) to O(nd^3). This is a big improvement for large networks. For example, for Facebook network of Berkeley University (which contains 22,937 nodes and 852,444 edges), counting all graphlets/graphlet degrees for up to 5-node graphlets takes ~ 4 days, while it takes only ~ 5 hours to count all of its up to 4-node node graphlets/graphlet degrees. This performance improvement makes GCD-based analyses feasible even for large networks.

3D Embedding of Networks using Multi-dimensional Scaling Multidimensional scaling (MDS) is a set of statistical techniques that assigns n-dimensional coordinate values to a set of data points while trying to preserve the given pairwise distances between the data points [22]. To visualize GCDs between networks, we use MDS-based embedding into 3-dimensional space, using metric squared stress criterion as the fit-criterion. We use Matlab's *mdscale* function for performing multi-dimensional scaling.

Precision-Recall Curves For evaluating the performance of network distance measures, we use the Precision-Recall (PR) curves [23]. Network pairs that are generated from the same model define the *True* set of the evaluation, while networks that are generated from different models define the *False* set. For a given distance threshold ϵ for a network distance measure, four values are computed: the true positives, TP, is the number of True pairs having pairwise distances smaller than ϵ; the true negatives, TN, is the number of False pairs having pairwise distances greater than or equal to ϵ; the false negatives, FN, is the number of True pairs having pairwise distances greater than or equal to ϵ; and the false positives, FP, is the number of False pairs having pairwise distances smaller than ϵ.

The precision and recall are defined as follows:

$$\text{Precision} = \frac{TP}{TP + FP}. \tag{S.15}$$

$$\text{Recall} = \frac{TP}{TP + FN}. \tag{S.16}$$

PR curve plots the precision versus the recall for varying values of ϵ. The Area Under the Precision-Recall curve (AUPR) is equal to the average precision of the distance measure. Thus, the closer the AUPR is to 1, the better the considered distance for clustering/separating network models.

Generating Noisy Networks For noise-tolerance and sampling experiments, we generate noisy networks by randomly rewiring a percentage of edges. For a network that has $|E|$ edges, a "k% noisy network" is generated as follows: at each step, three nodes, a, b, c, are chosen such that there is an edge (a, b) but there is no edge (a, c). Edge (a, b) is removed and edge (a, c) is added. This process is repeated $(|E| \times k)/100$ times.

To reduce the computational requirements of the noise tolerance experiments, we use model networks having {1000, 2000} nodes and {0.5%, 1%} edge-densities, so since we have 7 network models, this results in $2 \times 2 \times 7 \times 10 = 280$ model networks in total that we randomize. We choose to perform these experiments with smaller size networks since it is harder to distinguish between network models at lower network sizes. We randomize each of the 280 networks by rewiring $k\%$ of edges as described above. This results in 280 noisy model networks. We evaluate the clustering performance of a network distance measure on this set of noisy networks. We repeat this 30 times and report the average and standard deviation of the 30 experiments.

Note that this amounts to a large number of computations, since for each noise level (and we had 9 of them, in increments of 10%), we have $30 \times 280 = 8,400$ networks to count graphlets for. That is, we count graphlets for $9 \times 8,400 = 75,600$ networks, which takes a long time even if done in parallel on a decent compute cluster.

Edge Sampling Experiments Many real-world networks are incomplete, i.e., they have missing edges. For evaluating the performance of network distance measures on incomplete networks, we sample $k\%$ of edges from a model network and make a subgraph induced on the sampled edges.

We do this sampling for each of the 280 above described model networks.

We sample $\{10\%, 20\%, 30\%, \ldots, 90\%\}$ of edges for each of the 280 model networks 30 times. This results in $280 \times 9 \times 30 = 75,600$ networks to count graphlets for.

In addition, to test the clustering performance of the distance measures for both noisy and incomplete data, we sample $\{10\%, 20\%, 30\%, \ldots, 90\%\}$ edges from 280 networks with 40% of edges rewired as described in the previous section. (Doing this for the full set of $75,600$ noisy networks described in the previous section is computationally prohibitive.) As before, we repeat this 30 times, so the number of networks that we count graphlets for in this experiment is again $75,600$.

So in total, we count graphlets for $2 \times 75,600 = 151,200$ networks.

Node Sampling Experiments We test if the clustering of distance measures is still robust even if we use the properties of only $k\%$ of nodes of a network to compose a distance measure. We compute distance measures by sampling $k\%$ of nodes as follows:

- **Spectral Distance:** We compute the Laplacian matrix of the complete network, randomly choose k% of the nodes, and compute the spectrum from the submatrix formed by the rows and columns of the Laplacian matrix corresponding to these nodes.
- **Graphlet Correlation Distance (GCD):** We randomly choose $k\%$ of the nodes of a network and compute GDVs for each of the nodes (we compute GVDs using the entire network).

Then, GCM is computed over GDVs of the $k\%$ of the selected nodes and GCDs are computed from these GCMs.

- **Relative Graphlet Frequency Distance (RGFD):** For graphlet $i \in \{1, 2, \ldots 29\}$ of network G, we make $N_i(G)$ by counting only graphlets i that touch each of the selected $k\%$ nodes. We compute RGFD from these $N_i(G)$ as described in section 2.

- **Graphlet Degree Distribution Agreement (GDDA):** As for GCD and RGFD, we randomly chose $k\%$ of the nodes, for which GDVs are computed by using the entire network. Then, Graphlet Degree Distributions (GDDs) are computed over these GDVs, and GDDA is computed using these distributions.

- **Clustering Coefficient:** We randomly choose k% of the nodes, and compute their clustering coefficients using the entire network. We then average these clustering coefficients to obtain the clustering coefficient of the network.

- **Diameter:** We randomly choose k% of the nodes of a network and compute their eccentricities in the entire network (eccentricity of a node is the maximal shortest path distance of the node to all other nodes in the network). We choose the largest eccentricity over the $k\%$ sampled nodes and that is the diameter of the network.

We sample $\{10\%, 20\%, 30\%, \ldots, 90\%\}$ of nodes from each of the 280 model networks 30 times to compute the average and standard deviation of clustering performances of sampled distance measures.

In addition, to test the clustering performance of the sampled distance measures for noisy networks, we sample $\{10\%, 20\%, 30\%, \ldots, 90\%\}$ nodes from 280 networks with 40% of edges rewired as described above. As before, we repeat this 30 times to find averages and standard deviations of clustering performance of such sampled measures.

Canonical Correlation Analysis To relate a country's economic wealth (as per economic indicators listed above) and its position in the world trade network, we apply Canonical Correlation Analysis. Canonical Correlation Analysis [30] finds the weights for two sets of variables that maximize the correlation between linear combinations of the two sets of variables. The resulting weight values indicate which variables (i.e., orbits and economic indicators) are correlated with each other. For increased robustness to numerical artefacts, we use canonical cross-loadings instead of directly interpreting the weights. The value of a canonical cross-loading for a variable in one set is the correlation of that variable with the weighted sum (using the weights from Canonical Correlation Analysis) of all variables in the other set.

In our analysis, the first set of variables is composed of economic indicators of a country: RGDPL, RGDPL2, RGDPCH, KC, KG, KI, OPENK, POP, LE, BCA. The second set of variables is composed of the graphlet degrees (corresponding to orbits) of a country in the trade network: $C_0, C_1, \ldots, C_{72}$.

Therefore, the obtained weight values highlight the positively and negatively correlated economic indicators and graphlet orbits. We restrict the canonical correlation analysis with the trade networks of 1980 to 2010 because of the availability of economic indicator values (specifically

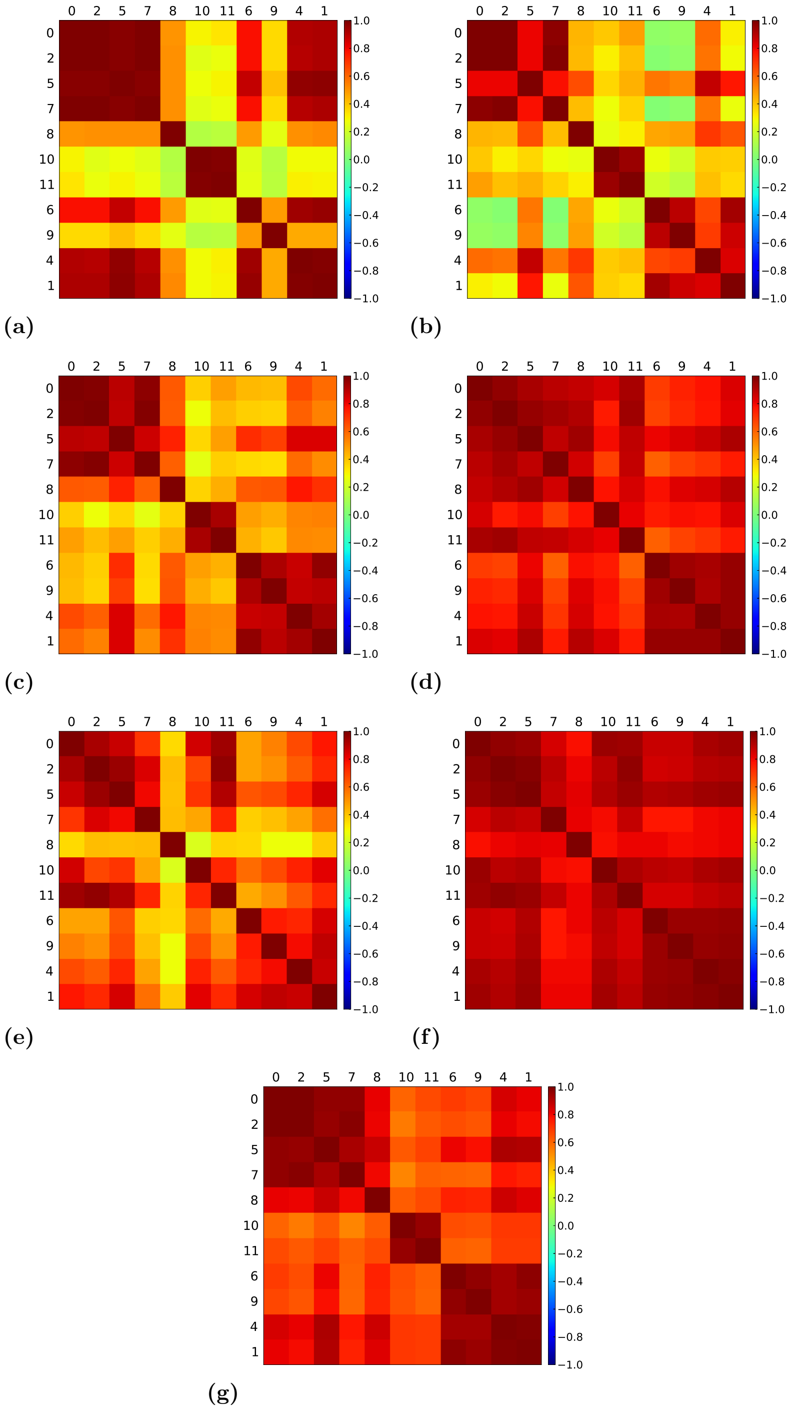
BCA and LE) before 1980 is scarce.

We define the *brokerage score* as the weighted linear combination of the broker graphlet degrees (i.e., C_{23}, C_{33}, C_{44}, and C_{58}), using the coefficients obtained from the canonical correlation analysis. This score measures how broker-like position a country is in within the world trade network of a particular year. Similarly, we define the *peripheral score* as the weighted linear combination of the graphlet degrees for peripheral orbits (C_{15}, C_{18}, and C_{27}), using the coefficients obtained from the canonical correlation analysis. This score measures how peripheral is the position of a country within the world trade network of a particular year.

3 Supplementary Results

Graphlet Correlation Matrices for Model Networks Supplementary Fig. S3 presents the graphlet correlation matrices (GCMs) of the 7 network models that are generated with 500 nodes and 1% density. The GCMs of different models differ from each other especially by the observed Spearman's correlation coefficients between the orbit sets {0, 2, 5, 7}, {8}, {10, 11}, {6, 9}, and {1, 4}.

Evaluation of GCDs that use different orbits We evaluate several GCD variants: (1) GCD-11, computed by using non-redundant 2- to 4-node graphlet orbits (i.e., orbits 0, 1, 2, 4, 5, 6, 7, 8, 9, 10, and 11 in Fig. 1-d of the main paper), (2) GCD-15, computed by using all 2- to 4-node graphlet orbits (i.e., orbits 0-14 in Fig. 1-d of the main paper), (3) GCD-56, computed by using non-redundant 2- to 5-node graphlet orbits (i.e., orbits other than 3, 5, 7, 14, 16, 17, 20, 21, 23, 26,



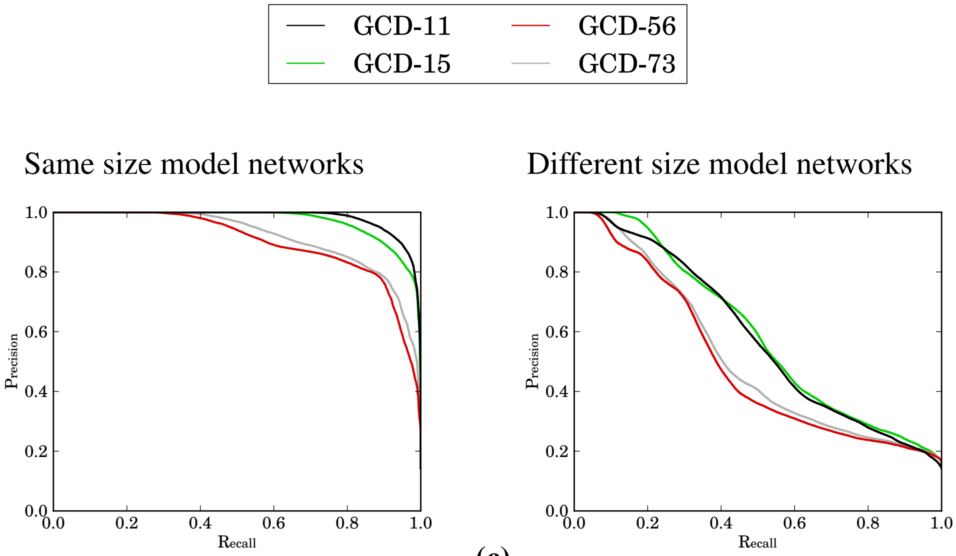
(a) (b) (c) (d) (e) (f) (g)

Supplementary Figure S3: **The graphlet correlation matrices of different network models.** Networks are of sizes 500 nodes and 1% density. The network models that the matrices represent are: (a) ER, (b) ER-DD, (c) SF-BA, (d) SF-GD, (e) GEO, (f) GEO-GD, (g) STICKY.

28, 38, 44, 47, 69, 71, and 72 in Fig. 1-d of the main paper), and (4) GCD-73, computed using all 2- to 5-node graphlet orbits (i.e., orbits 0-72 in Fig. 1-d of the main paper). We formally compare the performances of these four GCD versions against each other in the same way in which we compared GCD-11 and GCD-73 in the main paper. We use the same 2,520 model networks that we used in the experiments of Fig. 3. The results are presented in Supplementary Fig. S4.

For evaluating the performance of the four GCDs (GCD-73, GCD-56, GCD-15, and GCD-11) for grouping networks of the same size and density, the number of network pairs that we compare is $\binom{210}{2} \times 12 = 263,340$, because for one network size and density (and there are 12 different network sizes and densities that we use, described in the main paper), we have 210 networks, since there are 7 network models that we use and 30 networks from each model. In these experiments, GCD-11 outperforms all other GCD versions in terms of AUPR (panels a, b, and c in Supplementary Fig. S4). For evaluating the performance of these GCDs for grouping networks of different size and density, the number of network pairs that we compare is $\binom{2,520}{2} = 3,173,940$ (this is because we have 7 models, 12 node size and edge densities, and 30 network instances for each model and node size and endge density). GCD-15 performs the best for grouping networks of different size and density.

In the main paper, we use GCD-11, since it is the best for determining the fit of models to data networks (which are all of the same size) and hence also for tracking changes in the world trade network topology over consecutive years.



(a) **(c)**

Distance	AUPR
GCD-11	**0.982**
GCD-15	0.971
GCD-56	0.898
GCD-73	0.917

(b)

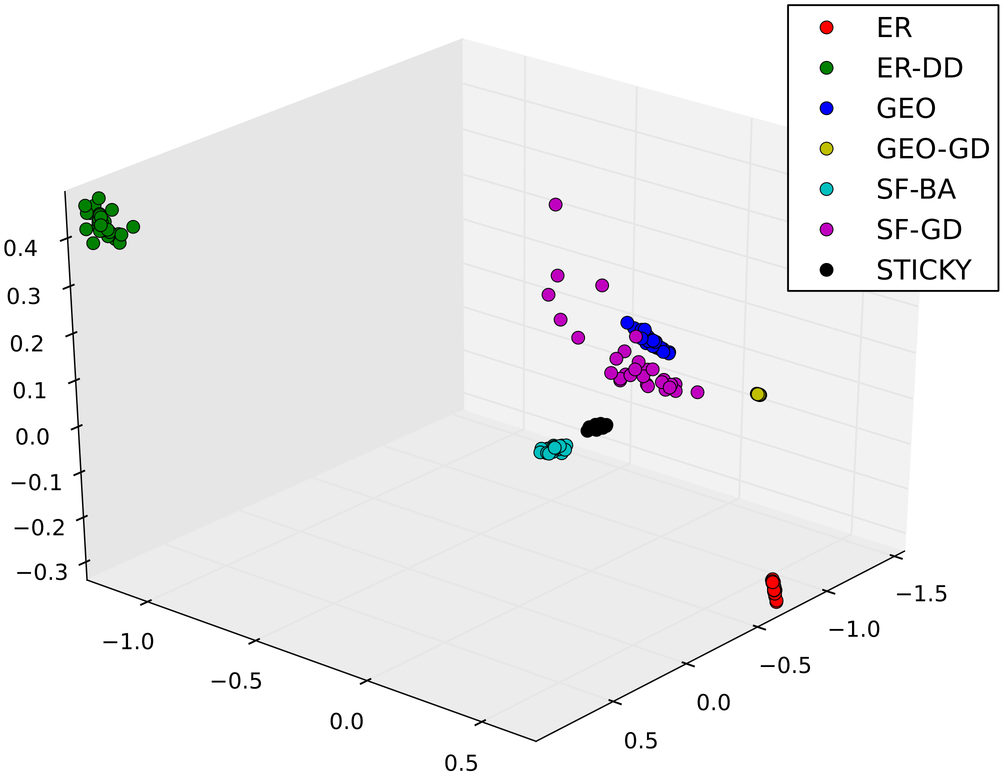
Distance	AUPR
GCD-11	0.583
GCD-15	**0.595**
GCD-56	0.492
GCD-73	0.511

(d)

Supplementary Figure S4: **Quality of clustering networks by using different GCD versions: GCD-11, GCD-15, GCD-56, and GCD-73**. The 2,520 model networks are used as in Fig. 3. Panels (a) and (b) present the quality of clustering of the same size and density networks, while (c) and (d) present quality of clustering of networks of different sizes and densities. Panels (a) and (c) are the Precision-Recall curves for clustering of model networks by using each of the four GCD versions. Panels (b) and (d) are tables summarizing the Area Under the Precision-Recall curve (AUPR) achieved by each GCD version.

Evaluation of GCD Against Other Measures Fig. 2-e illustrates that the model networks of different sizes and densities cluster together. Here, we do the same for networks of the same size and density: Supplementary Fig. S5 shows a 3-dimensional MDS embedding by using GCD-11 distances among model networks of size 6000 and density 1%. It illustrates that GCD-11 clusters even better networks of the same size and density than those of different sizes and densities.

We formally evaluate the performance of GCD against those of other network distance measures as described in the main paper (Fig. 3), but only for networks of the same size and edge density. The expectation is that the performance will be even better in this case. The results are presented in Supplementary Fig. S6. For the experiments that are shown in Panels a-c of the Figure, we use the same set of 2,520 networks that are generated for the experiments in Fig. 3, but rather than computing the distances among all pairs of networks, we compare network pairs which are of the same size and edge-density. The number of those pairs is $\binom{210}{2} \times 12 = 263,340$, because for one network size and density setting (and there are 12 of them as described in the main paper), we have 210 networks since there are 7 models and 30 networks from each model. For the experiments in panels d-f of Supplementary Fig. S6, we randomize each network 30 times when we simulate each type and level of noise (as above), which if performed on the entire set of 2,520 networks would be computationally prohibitive. Hence, we use the same subset of 280 out of the 2,520 networks (having {1000, 2000} nodes and {0.5%, 1%} edge-densities) that we have used for the experiments in Fig. 3. We use these node sizes and edge densities because these networks are more difficult to cluster than larger networks (as explained in the main paper). Since we only consider network pairs of the same size and edge-density, we evaluate the clustering performance



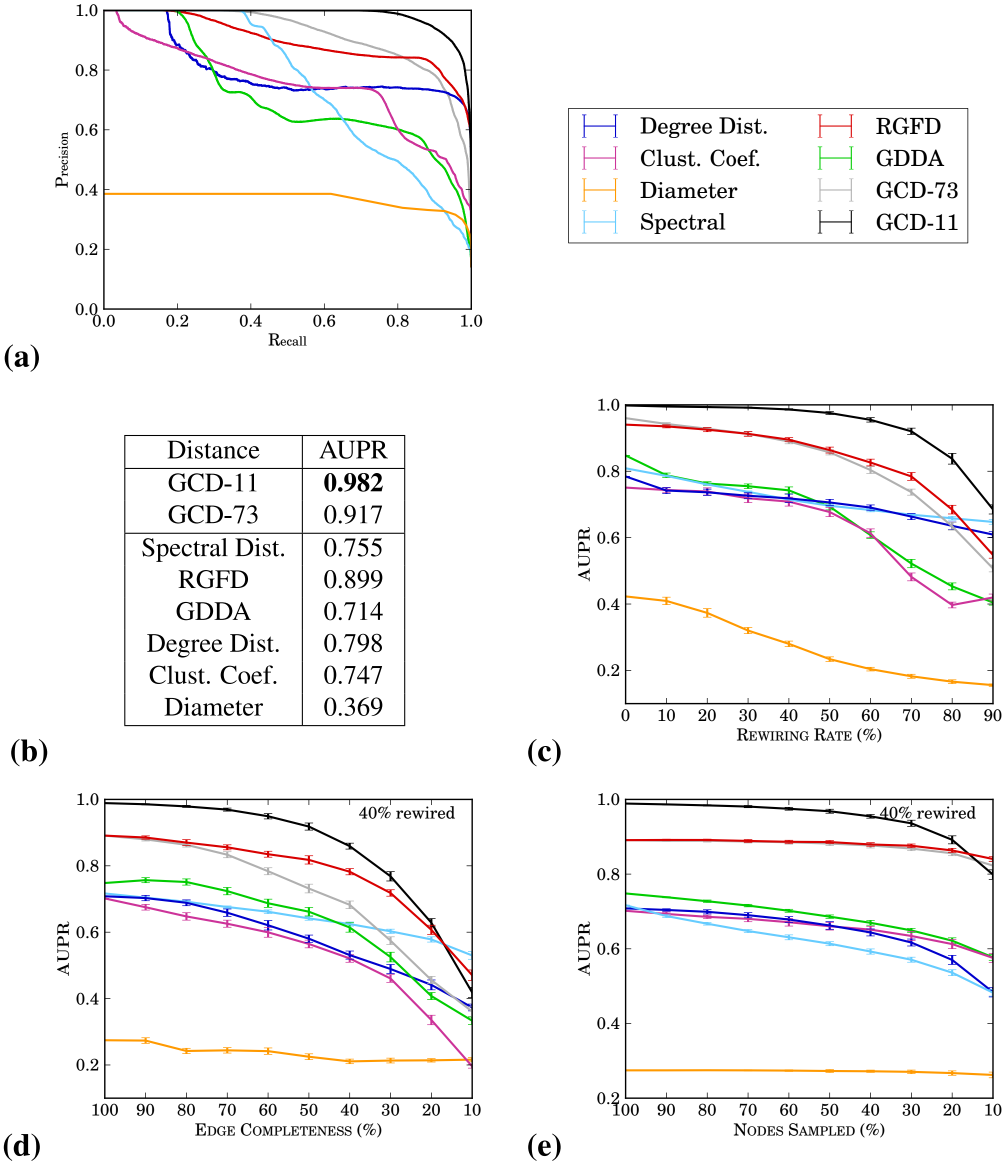
Supplementary Figure S5: **3-dimensional MDS embedding using GCD-11 distances** for model networks with 6000 nodes and 1% density.

using the distances among $\binom{70}{2} \times 4 = 9,660$ network pairs (for each network size and edge density that we use, there are 70 networks and there are 4 network size and density settings). Here, GCD-11 outperforms all other distance measures: it achieves the highest AUPR and it is the most robust to noisy and missing data.

Supplementary Fig. S7 shows the results of edge sampling and of node sampling experiments (described above) on noiseless model networks. These results are indeed very similar to those obtained on noisy networks (40% of rewired edges) presented in the main paper Fig. 3 (when comparing all networks) and in the Supplementary Fig. S6 (when only comparing networks having same size and edge density), with GCD-11 being the most robust.

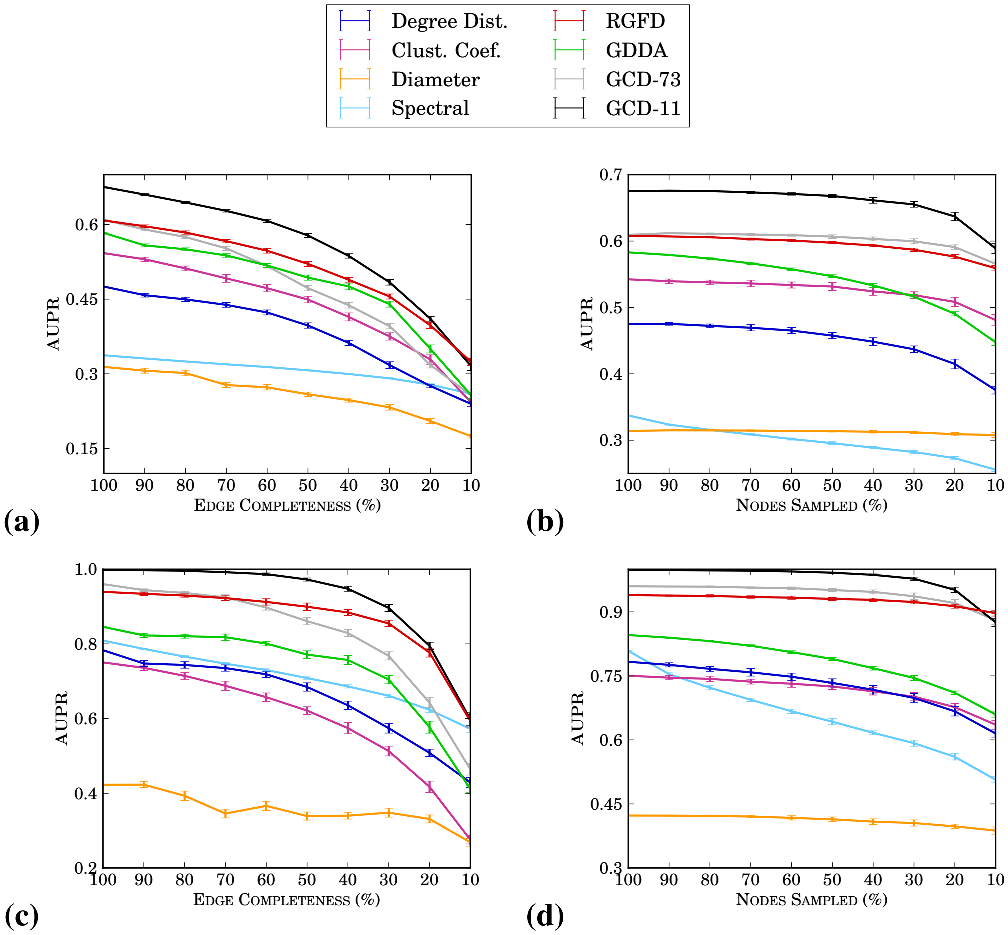
Model-fitting for Real-world Networks The graphlet correlation distance can be used for evaluating which network model best fits the structure of a real network. In order to test this, for each real-world network, we generate 30 networks from each of the 7 network models that are of the same size and density as the real-world network. We compute the GCD-11 between the real-world (data) networks and the 30 model networks (per model) along with the GCD-11 among the model networks of the same type. Supplementary Fig. S8 illustrates the distributions of all these data-vs-model and model-vs-model GCD-11 distances.

A network model fits a real-world network if there is an intersection between the data-vs-model and model-vs-model histograms [28]. The size of the intersection defines the goodness-of-fit. Hence, autonomous networks are best fit by the ER-DD model, however ER-DD network model is a very weak fit. Facebook networks are best fit by SF-GD networks, while GEO and GEO-GD

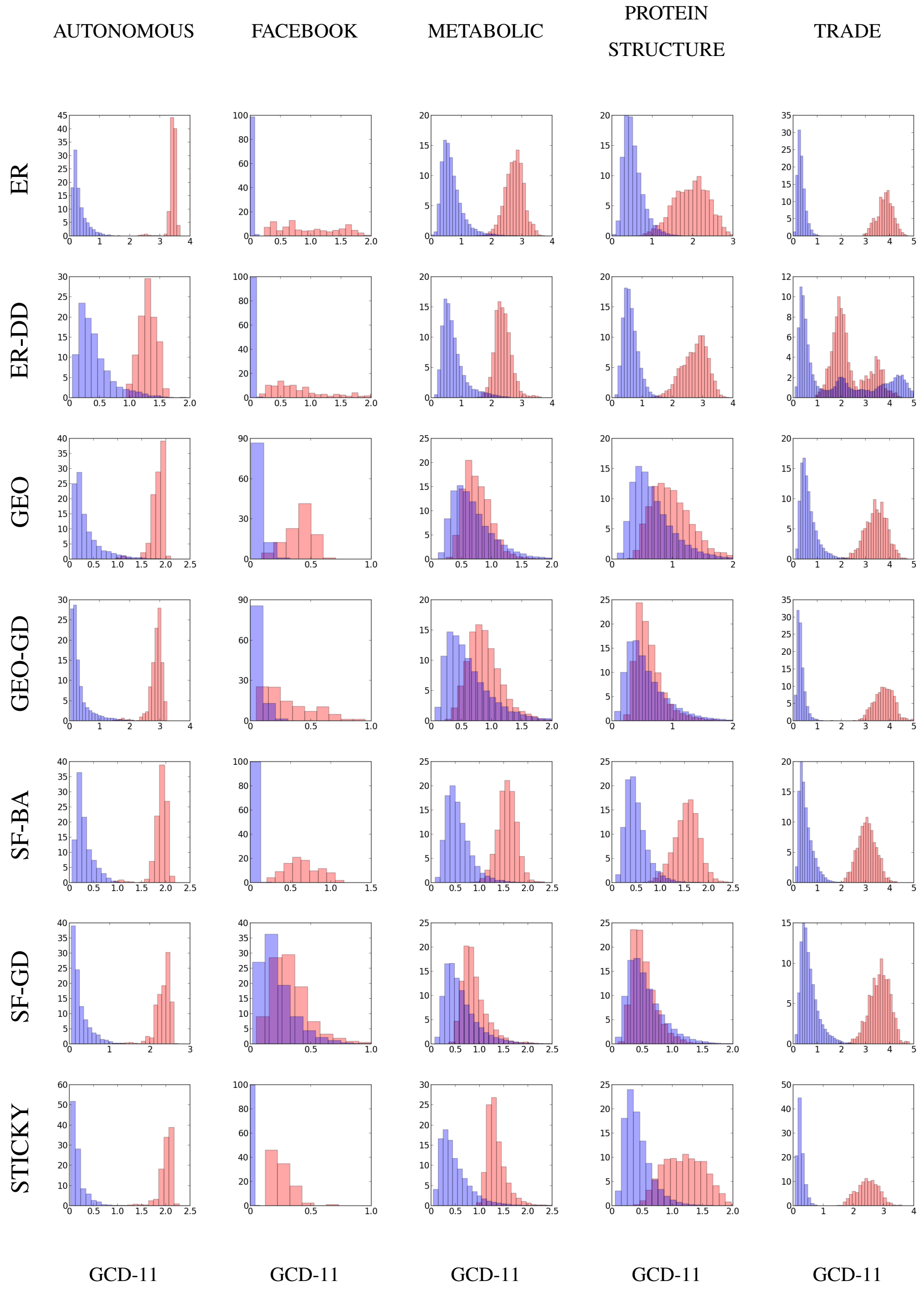


Distance	AUPR
GCD-11	**0.982**
GCD-73	0.917
Spectral Dist.	0.755
RGFD	0.899
GDDA	0.714
Degree Dist.	0.798
Clust. Coef.	0.747
Diameter	0.369

Supplementary Figure S6: **Clustering of the model networks of the same size and edge density by using the eight network distance measures (color coded).** Error bars are one standard deviation above and below the mean. **(a)** Precision-Recall curves for clustering the model networks with each of the eight distance measures. **(b)** Table summarizing the Area Under the Precision-Recall curve (AUPR) achieved by each distance measure. **(c)** AUPR for different levels of noise ("Rewiring Rate") added to the model networks in 10% increments. **(d)** On the model networks with 40% of noise, i.e., randomly rewired edges ("40% rewired"), AUPR for incomplete such networks: $x\%$ of edges were randomly removed from them in increments of 10% (so "Edge Completeness" of 100% means that no edges were removed). **(e)** On the model networks with 40% of noise (as described for panel (d)), AUPR when only a percentage of a measure was taken to comprise the distance measure (e.g., $x\%$ of Graphlet Degree Vectors of a network (denoted by "Nodes Sampled") were randomly chosen to make up its GCM-11 and subsequently its GCD-11).



Supplementary Figure S7: **Effects of missing data and Graphlet Degree Vector sampling on distance measures (color coded) applied to model networks.** **(a)** Considering all pairs of model networks of different sizes and edge densities, AUPR for model networks with $x\%$ of edges randomly removed from model networks to simulate missing data, where x is in increments of 10% (denoted by "Edge Completeness"). **(b)** Considering all pairs of model networks of different sizes and edge densities, AUPR for model networks when only a percentage of a measure (denoted by "Nodes Sampled") is taken to comprise the entire distance measure. **(c)** is the same as (a), but when we consider pairs of model networks of the same size and edge density. **(d)** is the same as (b), but when we consider pairs of model networks of the same size and edge density.



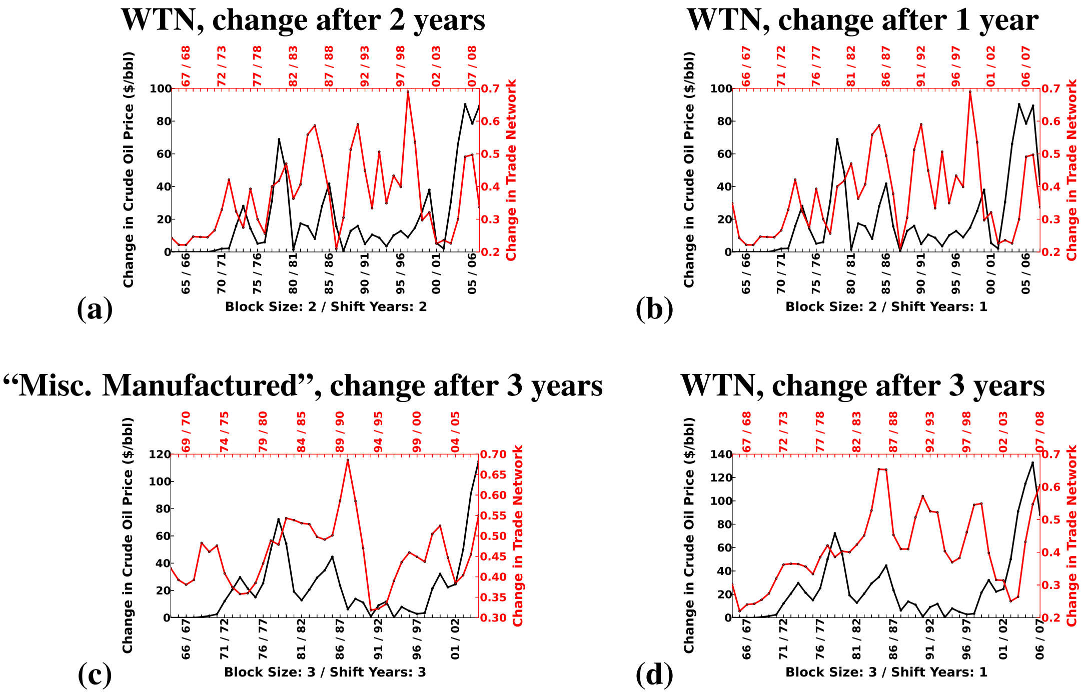
Supplementary Figure S8: **Modelling the autonomous, Facebook, metabolic, protein structure, and world trade networks with seven random network models** by using Rito *et al.*'s non-parametric test [27]. Each row represents one theoretical network model and each column represents one real network domain. On each panel, the horizontal axis is GCD-11, the vertical axis is measured probability density; the blue bars represent a histogram of GCD-11 distances across all pairs of randomly generated model networks with the same size and density as the corresponding real-world network (this gives us an expectation of how well the models compare with each other); the red bars represent a histogram of the GCD-11 distances of the real-world network compared to the 30 corresponding model networks. The quality of the fit is measured by the amount of overlap (i.e., shared area) under the distributions.

network models also fit. For metabolic and protein structure networks, all of the GEO, GEO-GD, and SF-GD network models fit well. For trade networks, none of the seven network models fit well. There exists a small intersection of the data-vs-model and model-vs-model distances for the ER-DD model. However, because of the small size of the trade networks, the ER-DD model is unstable (see [28]) as can be understood from the wide-spread model-vs-model distances. Therefore, it does not fit the structure of trade networks.

Effects of Crude Oil Price Changes on Trade Network Topology Table **S2** lists all the significant (p-values < 0.05) positive correlations between the change distributions of crude oil price and network topology. Supplementary Fig. S9 illustrates the distributions of the changes in crude oil price and network topology that have significant Spearman's correlations. Similarly, Supplementary Fig. S10 illustrates the distributions of the changes in crude oil price and network topology that have significant Phi correlations.

The changes in crude oil price are correlated with the changes in "TOTAL" trade network topology that occur one and two years later (the strongest correlation is observed two years later, with a Spearman's correlation coefficient of 0.414 and p-value of 0.005). These correlations are expected, since petroleum is critical for moving goods. Freight transportation consumes about 35% of all transport energy that is used worldwide, which is virtually based only on petroleum. The increases in crude oil price raise the transportation costs, and thus erodes the advantages of the long-distance supply chains.

Among the commodity based networks, "FOOD and LIVE ANIMALS" show the strongest

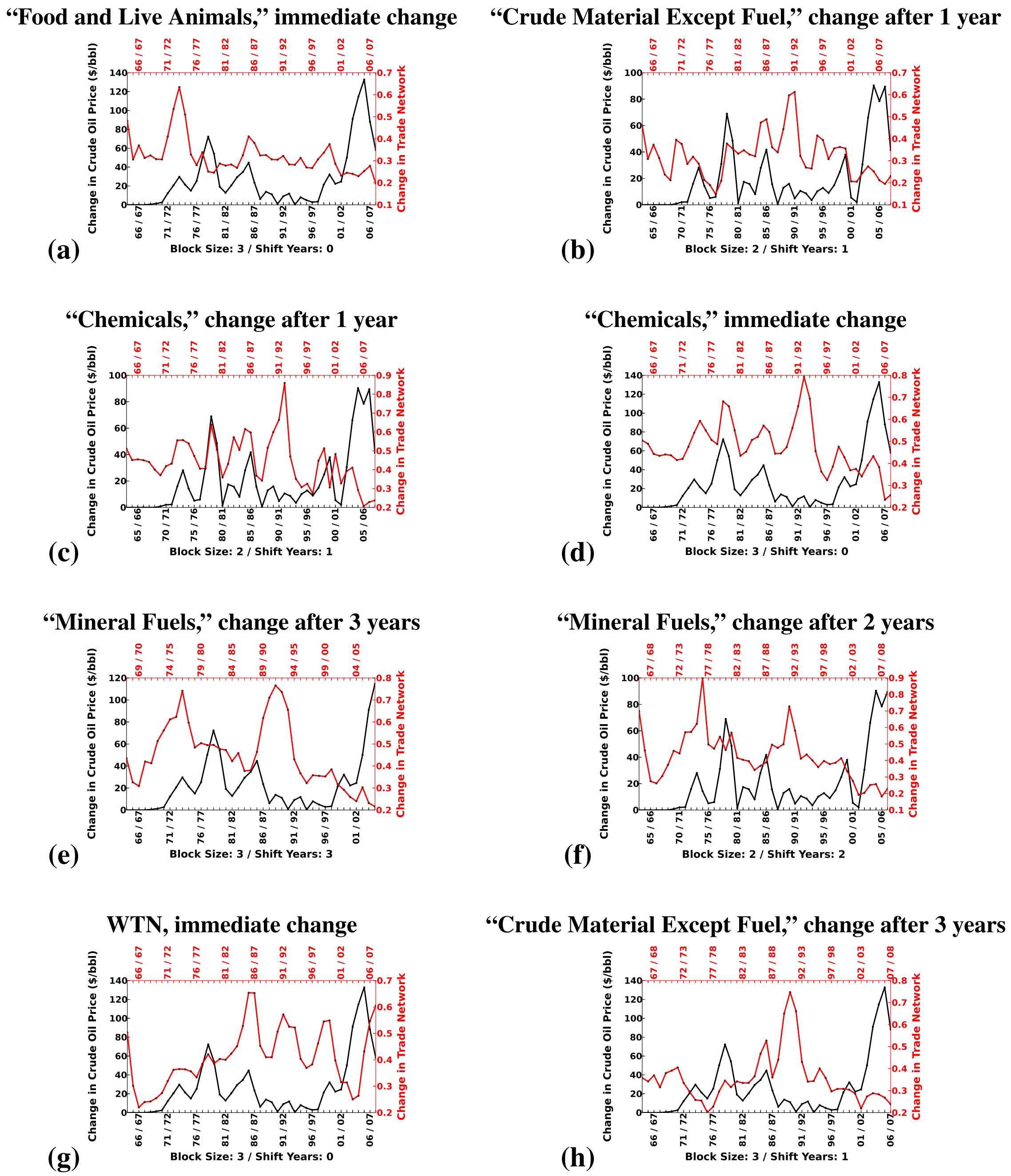

WTN, change after 2 years

WTN, change after 1 year

“Misc. Manufactured”, change after 3 years

WTN, change after 3 years

Supplementary Figure S9: **Statistically significant correlations of the crude oil price and WTN topology changes (p-value < 0.05) using Spearman’s Correlation.** WTN change patterns that are presented in the figures are: **(a)** WTN with the block size of 2 years and 2 year shift (WTN topology changes 2 years after oil price changes) (Sp. Corr. = 0.414; p-value = 0.005), **(b)** WTN with block size of 2 years and 1 year shift (Sp. Corr. = 0.356; p-value = 0.016), **(c)** The trade network of “Misc. Manufactured” commodity, with block-size of 3 years and 3 year shift (Sp. Corr. = 0.347; p-value = 0.026), **(d)** WTN with block size of 3 years and 1 year shift (Sp. Corr. = 0.316; p-value = 0.039).

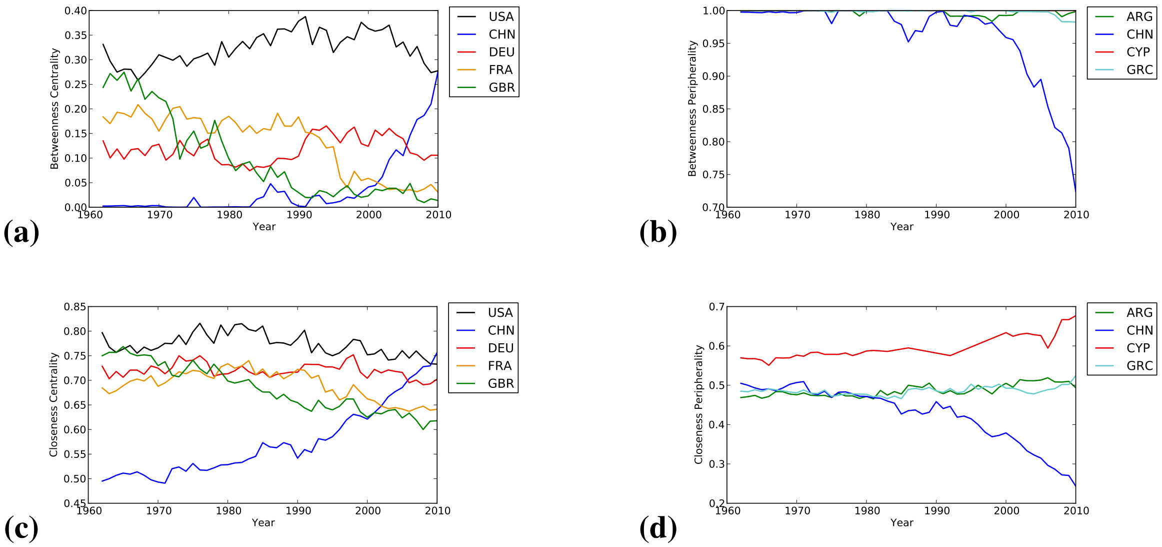


Supplementary Figure S10: **Statistically significant correlations of the crude oil price and WTN topology changes (p-value < 0.05) using Phi Correlation:** **(a)** "Food and Live Animals" commodity network with year-shift of 0 years (immediate change) and block-size of 3 years (Phi Corr. = 0.479; p-value = 0.001), **(b)** "Crude Material (except Fuel)" commodity network with year-shift of 1 year (network topology changes 1 year after oil price changes) and block-size of 2 years (Phi Corr. = 0.468; p-value = 0.001), **(c)** "Chemicals" commodity network with year-shift of 1 year and block-size of 2 years (Phi Corr. = 0.465; p-value = 0.001), **(d)** "Chemicals" commodity network with year-shift of 0 years and block-size of 3 years (Phi Corr. = 0.403; p-value = 0.007), **(e)** "Mineral Fuels" commodity network with year-shift of 3 years and block-size of 3 years (Phi Corr. = 0.402; p-value = 0.001), **(f)** "Mineral Fuels" commodity network with year-shift of 2 years and block-size of 2 years (Phi Corr. = 0.399; p-value = 0.001), **(g)** WTN with year-shift of 1 year and block-size of 2 years (Phi Corr. = 0.371; p-value = 0.001), **(h)** "Crude Material (except Fuel)" commodity network with year-shift of 1 year and block-size of 2 years (Phi Corr. = 0.334; p-value = 0.001).

correlation (with Phi coefficient of 0.479 and p-value of 0.001) on the same year as the crude oil price changes. This correlation is also expected, since (i) oil is needed for agriculture and (ii) oil price increase leads to increase in demand for corn, soy and other corns that are used for production of bio-ethanol and bio-diesel. We also find that this correlation increases over time: the Phi correlation coefficient rises from 0.31 for years in between 1962 and 1986, to 0.51 for years in between 1986 to 2007.

Supplementary Table S2: **All significantly correlated changes in Crude Oil Price and Trade Network Topology** ($p - value < 0.05$), when using block sizes of [1, 3] and shift years of [-3 , 3]).

Commodity	Block Size	Shift Years	Corr. / p-value (Spearman)	Corr. / p-value (Phi Coef.)
TOTAL	2	2	**0.414 / 0.005**	-0.055 / 0.725
TOTAL	2	1	0.356 / 0.016	-0.025 / 0.875
MISC. MANUFACTURED	3	3	0.347 / 0.026	0.012 / 0.940
TOTAL	3	1	0.316 / 0.039	0.089 / 0.575
FOOD and LIVE ANIMALS	3	0	-0.321 / 0.033	**0.479 / 0.001**
CRUDE MATERIAL (except FUEL)	2	1	-0.022 / 0.885	0.468 / 0.001
CHEMICALS	2	1	-0.021 / 0.893	0.465 / 0.001
CHEMICALS	3	0	-0.084 / 0.589	0.403 / 0.007
MINERAL FUELS	3	3	-0.087 / 0.588	0.402 / 0.010
MINERAL FUELS	2	2	-0.114 / 0.461	0.399 / 0.008
TOTAL	3	0	0.212 / 0.166	0.371 / 0.014
CRUDE MATERIAL (except FUEL)	3	1	-0.469 / 0.001	0.334 / 0.031



Supplementary Figure S11: **Betweenness and closeness centralities of countries in WTN over years: (a)** Betweenness centrality of the United States (USA), China (CHN), Germany (DEU), France (FRA), and the United Kingdom (GBR) from 1962 to 2010. **(b)** Betweenness peripherality (that we define as 1-[betweenness centrality]) of Argentina (ARG), China (CHN), Cyprus (CYP), and Greece (GRC) from 1962 to 2010. **(c)** Closeness centrality of the United States (USA), China (CHN), Germany (DEU), France (FRA), and the United Kingdom (GBR) from 1962 to 2010. **(d)** Closeness peripherality (that we define as 1-[closeness centrality]) of Argentina (ARG), China (CHN), Cyprus (CYP), and Greece (GRC) from 1962 to 2010.