Q-matrix : an algebraic formulation for the analysis and visual characterization of network graphs

R. Pozo
National Institute of Standards and Technology - Gaithersburg, MD

Abstract

Given an undirected network, we describe a two-dimensional graphical measure based on the connected component distribution of its degree-limited subgraphs. This process yields an unambiguous visual *portrait* which reveals important network properties. It can be used as a classification tool, as graphs from similar application areas have striking similarities. It can also be used as an efficient algorithm to demonstrate graph non-isomorphism for large graphs with identical degree distributions. Finally, it can be used as an analysis tool to help distinguish real-world networks from their synthetic counterparts.

1 Introduction

Attempting to represent a large-scale network as a small picture or a thumbnail image can prove to be a challenging task. Most application networks (e.g. biological, information, social) tend to have large hubs (heavy-tailed degree distributions) [2] and exhibit small-world properties [15], making their layout difficult to embed in two or three-dimensional spaces [6]. Current state-of-the-art algorithms for graph layout and visualization often render such objects as densely colored disks, or entangled "hairballs," making it difficult to extract meaningful information from their appearance. Furthermore, graph layout algorithms do not yield unique images; a single graph may yield many variations, depending on parameter and algorithmic choices. This situation is certainly understandable – it would be rather optimistic to expect graphs containing millions of vertices and edges crammed into a small snapshot (say, a 300x300 pixel image) to yield much insight.

Instead, we offer a different approach based on a simple idea: rather than draw the graph itself, represent the component size distribution of its degree-limited subgraphs. We define the **Q-matrix** of an undirected graph G to be the matrix formulation Q, where Q_{ij} is the number of connected components of size j of the degree-limited subgraph of G consisting of vertices with degree i or lower. The matrix Q, which is typically sparse, can be thought of as a generalization of the graph's degree-distribution, but also reveals such things as the number of connected components, the formation and growth of the giant component, and the effect of node-removal (site percolation) [4] on the connectivity of the remaining subgraphs –useful, for example, in simulations of network reliability[13] and the spread of infectious diseases[5].

Visualizations of the matrix Q can serve as useful network *portraits.* That is, networks from different application areas yield visually distinct portraits (Fig. 3) while networks from the *same* application area bear a strong resemblance. Furthermore, given a network graph, there is only one Q-matrix representation. Visualizations, such as those in Fig. 3, are just a three-dimensional view obtained by mapping the nonzero values of the Q matrix to the z-axis, and can be easily rendered within scientific software packages [1] such as MATLAB [11] or Mathematica [10].

2 Mathematical formulation

Given an undirected graph $G = (V, E)$, its degree distribution can be described as a vector $\vec{d}(G) \equiv \langle d_0, d_1, d_2, \ldots \rangle$, where each d_i is the number of vertices in G with degree equal to i. Note that if D denotes the largest degree of any vertex in G, then $d_i = 0$ for all $i > D$, so $\vec{d}$ is typically truncated to a finite length. We can then define the **degree-limited subgraph** $G_i = G|V_i$ as the induced subgraph created from vertices of G which have degree less than or equal to i. That is, $G_i \equiv (V_i, E_i)$ where

$$V_i = \{v \in V \mid \textbf{degree}(v, G) \leq i\} \qquad (1)$$

[1] Certain commercial hardware and software platforms are identified in this paper in order to specify the experimental procedures adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the products or equipment are necessarily the best available for the purpose.

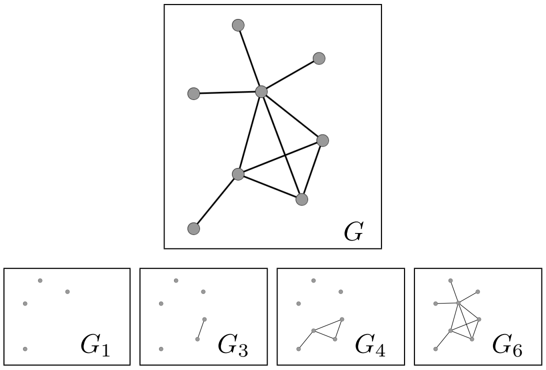


Figure 1: A small graph G and its 4 distinct degree-limited subgraphs.

and

$$E_i \; =: \{(u,v) \in E \,| u, v \in V_i\} \tag{2}$$

where **degree**(v, G) denotes the degree of vertex v in graph G.

Alternately, the subgraphs G_i can be thought of as what remains when every vertex of degree larger than i, together with every edge touching such vertices are removed from the original graph. Viewed either way, these degree-limited subgraphs are often comprised of disconnected components, even if the original graph is completely connected. By analyzing not only the number of components, but also their *size distribution* we can render interesting visualizations that are unique for each network (i.e., invariant under graph isomorphisms) and illustrate fundamental properties of the graph's structure.

Define the **Q-matrix** of a graph G as the two-dimensional component size distribution of its degree-limited subgraphs. Specifically, let $\Pi(j)A$ be the number of connected components of graph A which have j vertices; then

$$Q_{i,j} \;\equiv\; \Pi_j(G_i) \tag{3}$$

In other words, Q_{ij} is the number of connected components of size j in G_i. Note that $G_i = G$ for $i \geq D$. If $M(G)$ denotes the number of vertices in the largest component of G, then Q is a matrix with row indices $[0, 1, \ldots, D]$ and column indices $[1, 2, \ldots, M(G)]$. Although $Q_{i,j}$ is defined for any $i \geq 0$ and $j \geq 1$, it is zero beyond these values, so we typically truncate Q to be of size $(D + 1) \times M(G)$. If the degree distribution is sparse then there will be repeated degree-limited subgraphs, as $(d_i = 0) \Rightarrow (G_i = G_{i-1})$. In such cases, the Q-matrix will therefore have duplicate rows.

Consider for example the graph G in Fig.1, which has 8 vertices and 10 edges. It has a degree distribution of $\vec{d} = \langle 0, 4, 0, 2, 1, 0, 1\rangle$ and gives rise to four distinct degree-limited subgraphs, G_1, G_3, G_4, and G_6. Since the maximum degree of G is 6, and the largest component size is 8, the corresponding Q-matrix of G is given by the 7×8 matrix

$$Q(G) \;=\; \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{4} & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{3} & 0 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 \\ \mathbf{3} & 0 & 0 & \mathbf{1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mathbf{1} \end{bmatrix} \tag{4}$$

The ith row gives the component size distribution for G_i. At $i = 4$, for example, we see that there are three components of size 1 (i.e. isolated vertices) and one component of size 4 in G_4. Thus, $Q_{4,1} = 3$ and $Q_{4,4} = 1$. (The first element in the upper left-hand corner of Q is $Q_{0,1}$, rather than $Q_{1,1}$.) Furthermore, $G_D = G$, so G_6 contains the original graph, consisting of a single connected component of size 8, thus $Q_{6,8} = 1$. We note that Q is sparse and contains redundant rows: $G_2 = G_3$ and $G_5 = G_4$. For practical considerations, we define a compact representation, the **Q^*-matrix**

$$Q^*_{i,j} \;\equiv\; \begin{cases} \Pi_j(G_i) & \text{if } d_i > 0 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

which zeros out these redundant rows of Q:

$$Q^*(G) \;=\; \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{4} & & & & & & & \cdot \\ \cdot & & & & & & & \cdot \\ \mathbf{4} & \mathbf{1} & & & & & & \cdot \\ \mathbf{3} & & & \mathbf{1} & & & & \cdot \\ \cdot & & & & & & & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{1} \end{bmatrix} \tag{6}$$

Here, only the non-zero values are explicitly shown. The matrices Q and Q^* convey the same information –given one, the other can be easily derived. In practice, Q^* provides an economical storage format which more clearly conveys the information content of Q.

The graph characteristics captured by the Q-matrix may not be fully apparent for this small example –it is simple enough to explain the basic ideas, but too coarse to reveal structural patterns. In the next section we examine large real networks in which the usefulness of this representation become apparent.

For **directed graphs**, the Q-matrix can be interpreted as the number of *weakly connected components.* This essentially ignores the direction of edges and allows the same algorithms and analysis to be applied to both directed and undirected graphs. Other extensions to the Q-matrix are described in later sections.

The process of removing or adding specific vertices to a graph, as is done here, is a particular type of site percolation process and arises in several areas of network science, such as modeling the failure of routers in computer networks (information technology) or the spread of infectious diseases in populations (epidemiology). Various mathematical models have been developed to analyze the resilience to targeted attacks.[3] In the Q-matrix formulation, the site percolation process is rather specific (by ordering the removal of nodes by their degree) and occurs in discrete "bulk" steps (i.e., at each percolation step *all* nodes of a given degree are processed simultaneously). This last stipulation differs from conventional approaches in percolation studies, but this slight twist ensures that the process yields consistent results which do not exhibit statistical fluctuations and reduces the overall size of the Q-matrix .

3 Q-matrix visualization

For large networks, it is impractical to display the Q-matrix explicitly, as in Eq. 4 or even Eq. 6. Instead, we lay the matrix down on the x-y plane and plot the nonzero values on the z-axis, creating a three dimensional scatter plot of component size distributions. In this way the degree, component size, and number of components comprise the x, y, and z-axis, respectively. Furthermore, because the values on these axes span several orders of magnitude, it is convenient to render the plot on a log-log-log scale and use the Q^* formulation to provide images which are less cluttered. We refer to this representation as the **Q-matrix plot** to distinguish it from the array representation in Eq. 6. In the sequel we use the term Q-matrix to refer both to the matrix and its plot; the context should make it clear which we mean.

For example, the Q-matrix in Fig. 2 is that of an undirected email communication network [7][8] with 36,692 vertices and 183, 831 edges, where each vertex is an individual email address and two vertices are connected by an edge if there was at least one message sent from one to the other. The original graph is too large to render in its entirety, but its Q-matrix values consist of individual points (non-zeros) which can be effectively plotted. The top-left (0,1) corner of the Q-matrix is now on the floor in the rear corner, with the degree values running along the left rear wall, and component sizes running along the right rear wall.

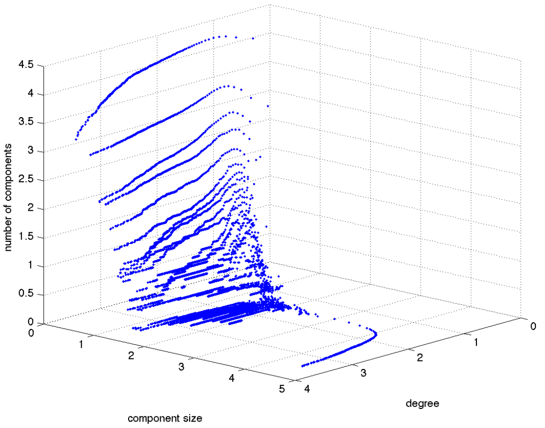


Figure 2: The Q-matrix for an email communication network with 36,692 nodes and 183,831 edges. [7]

The comb-like "lines" appearing in the plot are constant component size contours, for component sizes of $k = 1, 2, 3$, and so on. They are discrete points, but they are so densely represented as to appear as continuous lines when viewed at these scales.

Moving from the left wall (x-z plane) towards us, the resulting image appears to resemble a hill, with a hook-like trail appendage closest to us, moving towards the lower right of the matrix, where the degree and component size are greatest. Upon closer inspection we can identify three loosely-defined regions in this type of image: the **wall** occurs near the x-z plane and shows how the small component sizes vary for each G_i; the **hill** middle region shows how small and medium component sizes vary, and the characteristic **hook** on the floor (x-y plane) represents the birth and growth of the largest component. These are not precise mathematical boundaries, but these characteristics do seem prevalent in Q-matrix plots, so the nomenclature is useful in describing these renderings.

4 Application Examples

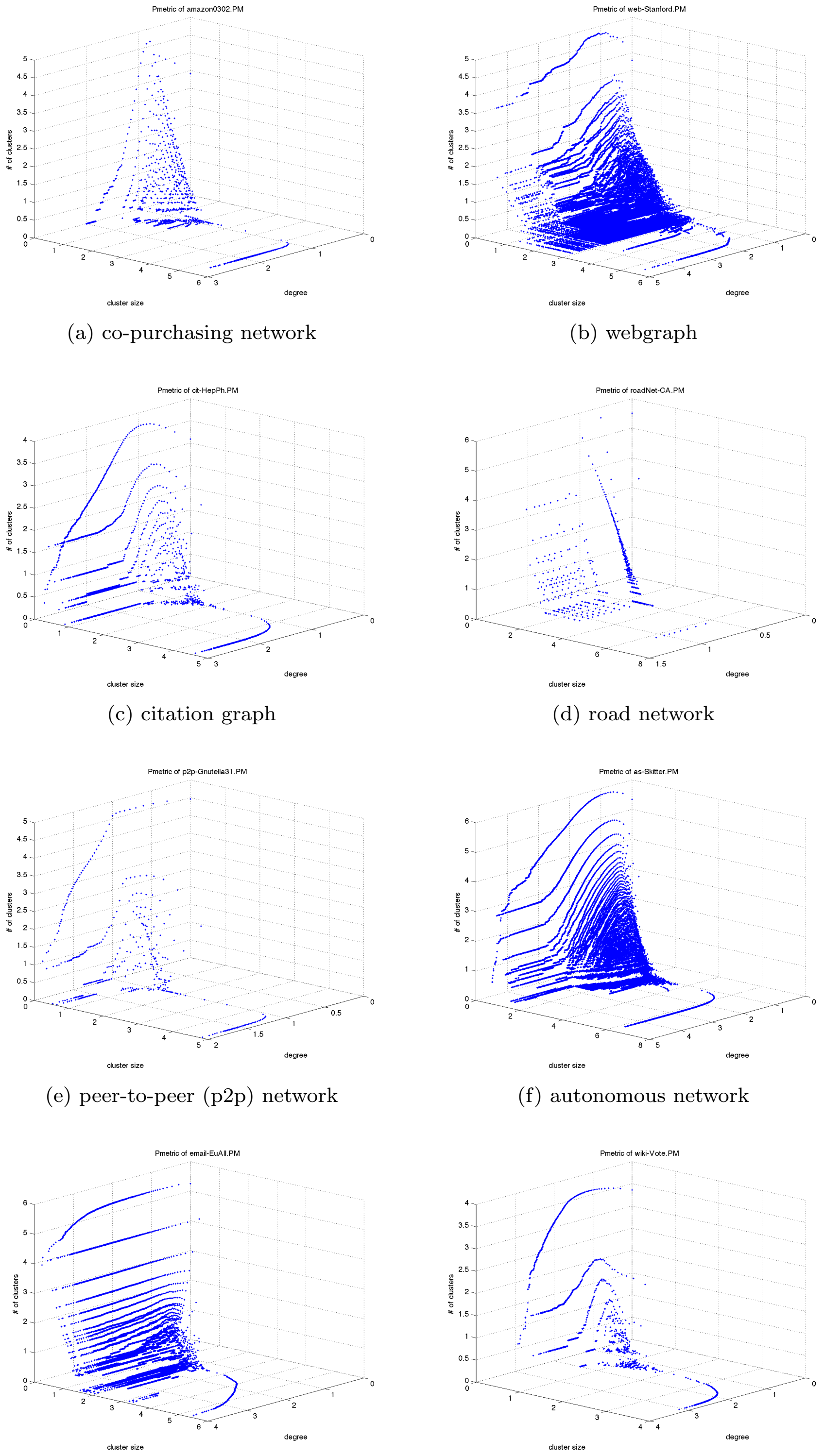
Fig. 3 shows the Q-matrix plots of real-work networks found in the Stanford Large Network Collection[7]. Their detailed descriptions are found in Table 1. In some cases these are directed graphs, and as previously noted, the Q-matrix then refers to the distribution of *weakly connected components*.

First and foremost, the experimental data shows that Q-matrix images of graphs from distinct application areas do, in fact, appear different. In each subfigure of 3 the wall, the hill and hook all have different shapes and aspect ratios.

Surprisingly, Q-matrices of graphs from the *same*

	NETWORK	**NODES**	**EDGES**	**REFERENCE**[7]
Collaboration	Astro Physics	18,772	396,160	ca-AstroPh
Networks	Condensed Matter	23,133	186,936	ca-CondMat
(Fig. 4)	High Energy Physics	12,008	237,010	ca-HepPh
	High Energy Physics Theory	9,877	51,971	ca-HepTh
Web graphs	Google	875,713	5,105,039	web-Google
(Fig. 5)	Notre Dame	325,729	1,497,134	web-NotreDame
	Stanford	281,903	2,312,497	web-Stanford
	Berkeley-Stanford	685,230	7,600,595	web-BerkStan
Road networks	California	1,965,206	5,533,214	roadNet-CA
(Fig. 6)	Pennsylvania	1,088,092	3,083,796	roadNet-PA
	Texas	1,379,917	3,843,320	roadNet-TX
Citation	High Energy Physics	34,546	421,578	cit-HepPh
Networks	High Energy Theoretical Physics	27,770	352,807	cit-HepTh
(Fig. 7)	US Patents	3,774,768	16,518,948	cit-Patents
Co-purchasing	March 2	262,111	1,234,877	amazon0302
networks	March 12	400,727	3,200,440	amazon0312
(Fig. 8)	May 5	410,236	3,356,824	amazon0505
	June 1	403,394	3,387,388	amazon0601
Email networks	Enron	36,692	183, 831	email-Enron
(Fig. 9)	European University	265,214	420,045	email-EuAll
Online social	Epinons	5,879	508,837	soc-Epinions1
(Fig. 10)	LiveJournal	4,847,571	68,993,773	soc-LiveJournal1
	Slashdot (11-2008)	77,360	905,468	soc-Slashdot0811
	Slashdot (02-2009)	82,168	948,464	soc-Slashdot0922

Table 1: Example datasets from the Stanford Large Network Collection [7] used for Q-martrix experiments.



(a) co-purchasing network

(b) webgraph

(c) citation graph

(d) road network

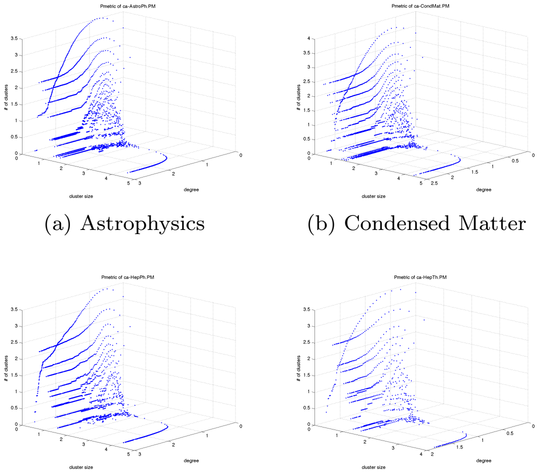
(e) peer-to-peer (p2p) network

(f) autonomous network

(g) email network

(h) Wikipedia network

Figure 3: Q-matrices of networks graphs from the Stanford Large Network Collection[7].



(a) Astrophysics

(b) Condensed Matter

(c) High Energy Physics

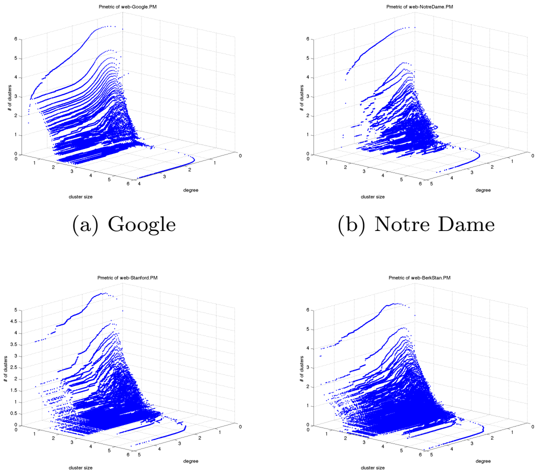
(d) High Energy Physics Theory

Figure 4: Q-matrices of co-authorship networks.

application area appear to have *similar* characteristics, as demonstrated in Fig.s 4-9. In such cases, each group shares similar shape and slope of the wall, hill, and hook regions for every network studied in our experiments. This suggests that the Q-matrix may be used as a crude classification tool to help identify "families" of large network graphs. Indeed, it is a canonical visual representation of the original graph, and unlike matrix structure plots, or graph drawing algorithms, there is *only one representation* for each graph, invariant under graph isomorphisms. This makes it useful for labeling large graphs with a compact image, and using this visual representation to categorize graphs into distinct groups. In particular, it is useful for tagging network graphs in databases with thumbnail images that actually yield distinguishable characteristics.[2] In other words, Q-matrix plots provide a compact data set and a thumbnail image that may serve as a network "identification badge", or a "photo ID," capturing important characteristics beyond its size and degree distribution.

5 Extracting conventional measures

Embedded within the Q-matrix are basic networks measures, which can be inspected visually, or can be computed exactly with simple matrix/vector operations. For example, the nonzeros in the bottom



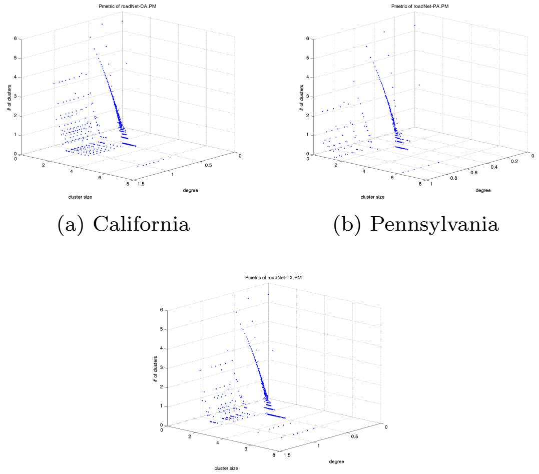
(a) Google

(b) Notre Dame

(c) Stanford

(d) Berkeley-Stanford

Figure 5: Q-matrices of Web graphs.



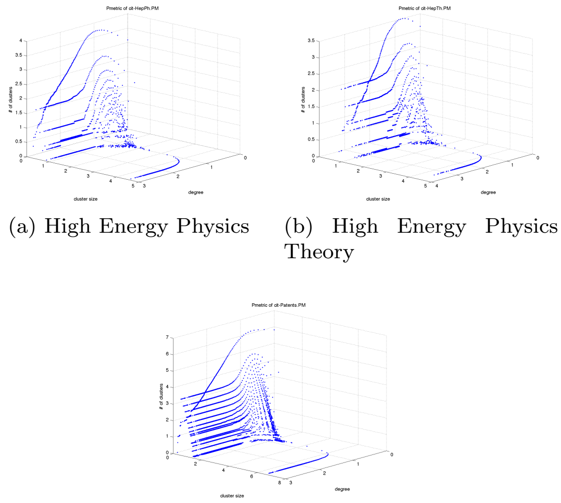
(a) California

(b) Pennsylvania

(c) Texas

Figure 6: Q-matrices of U.S. road networks.

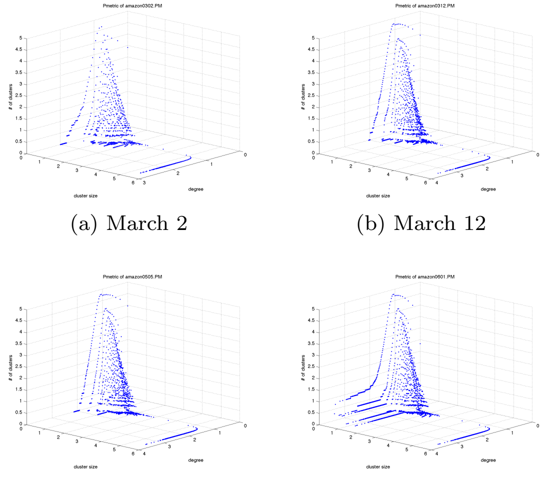
[2] Current graph-drawing techniques have difficulty rendering meaningful visualizations for large graphs with heavy-tail degree distributions.



(a) High Energy Physics (b) High Energy Physics Theory

(c) US Patents

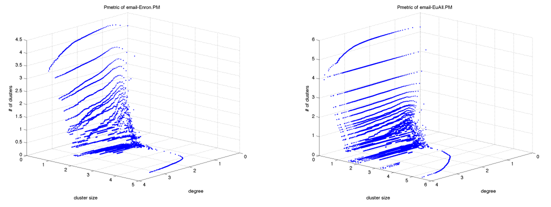
Figure 7: Q-matrices of citation networks.



(a) March 2 (b) March 12

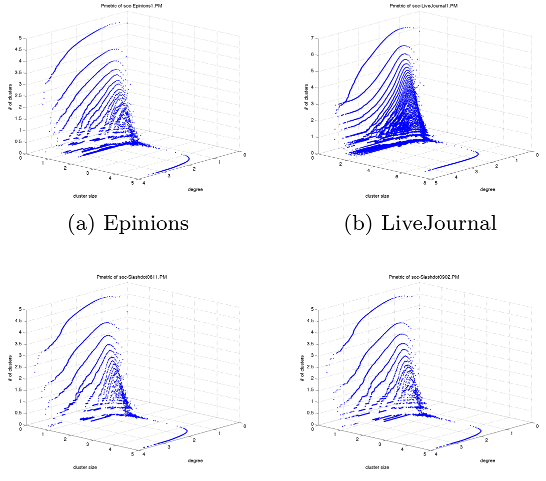
(c) May 5 (d) June 1

Figure 8: Q-matrices of Amazon co-purchasing networks (2003).



(a) Enron (b) European University

Figure 9: Q-matrices of email networks.



(a) Epinions (b) LiveJournal

(c) Slashdot (Nov. 2008) (d) Slashdot (Feb. 2009)

Figure 10: Q-matrices of online social networks.

row ($Q_{D+1,*}$) enumerate the connected components of each size in the original network; the first element of Q, $Q_{0,1}$ tells us how many isolated vertices, if any, are present in G; the height and extension of the left *wall* capture leaf and low-degree vertex behavior as one increases the degree i for G_i.

Using $|x|_1$ to denote the vector 1-norm, and $[0 : N]$ to denote the vector of $N + 1$ nonnegative integers $< \ 0, 1, 2, \ldots, N \ >$, we can derive the following quantities:

- **number of components in** G_i is the row sum of $Q_{i,*}$

$$\begin{aligned} \Pi(()G_i) &= \sum_j Q_{i,j} \\ &= |Q_{(i,*)}|_1 \end{aligned} \tag{7}$$

  In particular, $G_D = G$, so $\Pi(G) = |Q_{(D,*)}|_1$

- **size of largest component in** G_i, denoted $M(G_i|)M((,$ is the index of the last non-zero in the i-th row:

$$M(G_i) \ = \ \max_j \{ j \mid Q_{i,j} > 0 \} \tag{8}$$

- **number of vertices in** G_i is the number of vertices with degree less than or equal to i, which is the number of components in the i-th. row of Q multiplied by their respective sizes:

$$\begin{aligned} |V_i| &= \sum_{j=0}^{M(G)} (Q_{(i,j)} \times j) \\ &= Q_{(i,*)} \cdot [0 : M(G)] \end{aligned} \tag{9}$$

  In particular, $|V| = |V_D| = Q_{(i,*)} \cdot [0 : M(G)]$.

- **degree distribution of G**, $\vec{d} =< d_i >$: The number of vertices in G with exactly degree i can be seen as the difference between the number of those with degree i or less, and those with degree $i-1$ or less:

  $$\begin{aligned} d_i &= |V_i| - |V_{i-1}| \qquad (10) \\ &= \left[Q_{(i,*)} - Q_{(i-1,*)}\right] \cdot [0 : M(G)] \end{aligned}$$

  For $i = 0$, we just have $d_0 = |V_0| = Q_{(0,*)} \cdot [0 : M(G)]$ as the number of isolated nodes in the original graph.

- **number of edges in G** for an undirected graph is the sum of the degrees of each vertex divided by two:

  $$\begin{aligned} |E| &= \frac{1}{2} \sum_i i \times d_i \qquad (11) \\ &= \frac{1}{2} [0 : D] \cdot \vec{d} \end{aligned}$$

  where $\vec{d} = \{d_0, d_1, \ldots, d_D\}$ is given by Eq.(11).

6 Practical considerations

The Q-matrix plot works best for large, complex networks with non-trivial degree distributions, where the Q-matrix contains sufficient non-zeros to yield a visually interesting image. For small graphs, like our toy example (Fig. 1) it is difficult to identify the wall, the hill and the hook. In fact, the Q-matrix plot works best precisely where other approaches, such as conventional graph drawing layouts fail, thus creating a useful complement to conventional methods for annotating network graphs.

In practice, the Q-matrix is sparse for large network graphs: there are relatively few distinct degrees (nonzeros in $\vec{d}$) and it is unlikely to find a component of particular size j in G_i. Hence, although the dimensions of Q are $(D+1)$ by $M(G)$, the actual number of nonzeros is quite small. Fig. 11 illustrates the ratio between the number of nonzeros in the Q-matrix and the number of edges in original graph from a sample of 44 applications , ranging in size from several hundred to several million. The results are plotted on a log-log scale, and we see that the number of nonzeros in Q grows roughly as $O(n^{0.4})$ where n is the number of edges in G. Networks with millions of edges are often represented by Q-matrices with just a few thousand numbers.

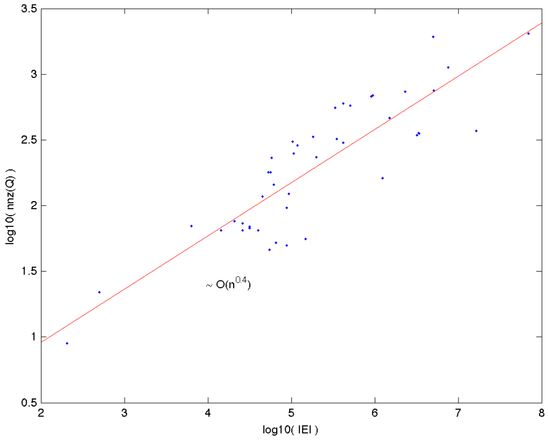
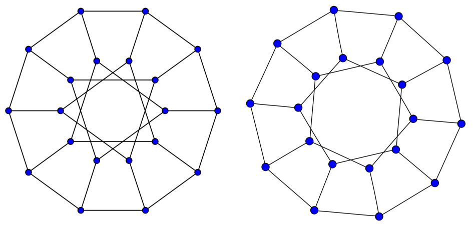


Figure 11: The size of the Q-matrix (number of nonzeros) grows at about $O(n^{0.4})$ compared to the size of graph (number of edges).

Computing the Q-matrix is pratical for large networks. In another paper [12], we describe an efficient algorithm which builds the Q-matrix incrementally, without calculating G_i explicitly in the intermediate steps. It begins by sorting and partitioning the vertices by their percolation order (degree) and growing equivalence classes corresponding to the intermediate subgraph components. Large network graphs with millions of edges can be processed in a few seconds on a desktop computer.

Furthermore, because a small perturbation to the graph structure (e.g., edge swap) could have a cascading effect to the resulting Q-matrix , provides a fast method for identifying **graph non-isomorphism** of two large networks with identical size and degree distributions. On the other hand, proving the converse remains challenging –different graphs may yield the same Q-matrix . For example, any k-regular graph (i.e., where every vertex has the same degree k) will yield a Q-matrix that has exactly one nonzero: $Q_{k,|V|} = 1$. In particular, two non-isomporphic 3-regular (cubic) graphs, cited in [1]: the Desargue graph, and the Dodecahedral graph (Fig. 12) have 20 nodes and 30 edges each, and identical degree distributions. Both yield identical Q-matrices . Thus, the Q-matrix for a graph is not an invertible representation, and there are specific counter-examples where comparing two Q-matrix pilots may yield little insight. Nevertheless, the interesting idea here is that the Q-matrix works best precisely when there is diversity in the degree-distribution, and when large hubs are present: two key characteristics that separate real-world networks from structured and uniformly random graphs.

We can also use the Q-matrix to investigate how



(a) Desargues graph (b) Dodecahedral graph

Figure 12: Two non-isomorphic cubic graphs of same size (20 nodes and 30 edges) and degree distributions, which yield the same Q-matrix , i.e. $G \to Q(G)$ is not one-to-one.

various graph properties compare to those of random graphs with similar degree sequences. For example, how does the growth of the giant component compare between real-world graphs and randomized versions with the same degree distribution? While formulations exist for calculating expected size at a given degree point [9], it is insightful to see how the variations behave over the complete degree spectrum. Fig. 13, for example, shows the giant component growing much faster for the randomized graphs, in some regions by almost three orders of magnitude. In this experiment, we computed random variations of the original network by accumulative edge swaps that preserved the degree distribution. (That is, each edge in the graph was randomly swapped with another edge in such a way to preserve the original degree distribution.) We then computed the Q-matrix for these randomized versions, and compared the largest component size growth. The results demonstrate that original and randomized graphs have a completely different signature, and that the Q-matrix can be used as a validation tool to help separate real-world graphs from their synthetic counterparts.

7 Comparing graphs

Given two graphs, A and B, and their respective Q-matrices , $Q(A)$ and $Q(B)$, we may define a distance function $\Delta(A, B)$ between these two graphs as the **Q-metric** :

$$\begin{aligned} \Delta(A, B) &\equiv ||Q(A) - Q(B)|| \qquad (12) \\ &= \sum_i \sum_j |Q(A)_{i,j} - Q(B)_{i,j}| \end{aligned}$$

In cases where the matrices $Q(A)$ and $Q(B)$ are of different sizes, the smaller one can be padded with zeros so they are conformant. This formulation is essentially the vector 1-norm, interpreting the elements $Q(A)$ and $Q(B)$ as a long vector. This definition is chosen over the more common Frobenius matrix norm to keep all computation in integer arithmetic, and hence its numerical value exact.

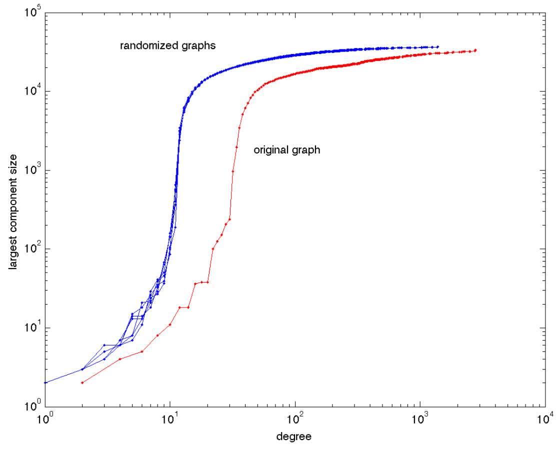


Figure 13: The growth of the giant component for the email network (Fig. 2) shown in red, compared to random graphs of same size and degree distribution, all computed directly from their corresponding Q-matrices . The original graph behaves significantly different, and this technique can be used to identify real networks from their synthetic counterparts.

Note that Δ does satisfy the requirement for a **pseudometric space**. Namely, for any graph A, B, C

$$\begin{aligned} \Delta(A, A) &= 0 \qquad (13) \\ \Delta(A, B) &= \Delta(B, A) \qquad (14) \\ \Delta(A, C) &\leq \Delta(A, B) + \Delta(B, C) \qquad (15) \end{aligned}$$

(Since $\Delta(A, B) = 0$ does not imply that $A = B$, the requirements for conventional metric space are not met.) We can use this distance function as a way to measure how different two graphs are in respective Q-matrix formulation. For example, Fig. 14 shows this metric applied to the email communication network (Fig. 2) and 100 random graphs generated as before with identical degree distribution. Here, a distribution of the $\binom{101}{2} = 5,050$ pairwise Q-metrics are plotted on a logarithmic x-axis. The result is a bimodal distribution illustrating the difference between random graphs (left mode) and the original graph. That is, the pairwise Δ for each random graph is over 40 times smaller than the Δ between the original graph and its random counterparts. If we normalize this difference by the number of vertices in the graph, the mean difference between random matrices is 0.7964

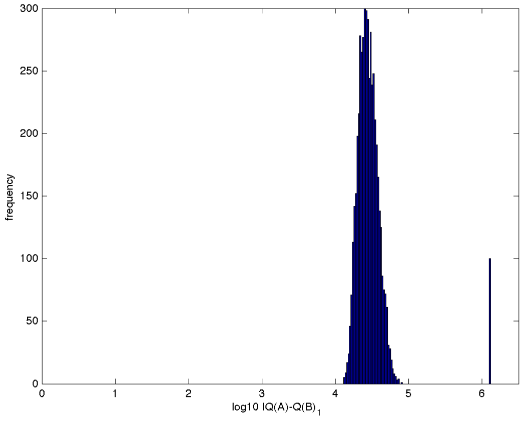


Figure 14: Comparison between the email network (Fig. 2) and 100 random graphs with identical degree distribution. Here, a distribution of the 5,050 pairwise Q-metrics are plotted on a logarithmic x-axis. showing that the original graph (right mode) is quite different than its random counterparts (left mode).

with a standard deviation of 0.2494, while the mean difference between the original graph and all 100 random graphs is 34.9570, with a standard deviation of 0.2941. Once again, the original graph behaves significantly different than its random counterparts and such a test can help identify real networks from those generated synthetically.

8 Generalizations and extensions

The Q-matrix has been defined for directed and undirected graphs, but further refinements could be made for the **directed graph** case by distinguishing between weakly-connected and **strongly-connected components**. One possible generalization of the Q-matrix formulation would be to define a version that creates two Q-matrices for directed graphs: one each for in-degree and out-degree distributions, and measure strongly-connected components for each.

Likewise, the Q-matrix formulation could also be extended to **weighted graphs**, where each edge has a weight, ω_e for $e = \{1, 2, \ldots, |E|\}$, by extending the notion of degree of a vertex to the sum of its edge weights.

Also, we may create alternate versions of the Q-matrix using other node orderings (centralities) in place of degree, e.g. between-ness, eigenvalue, or Pagerank centralities. A similar framework can be developed for **edge centralities**, in which edges, rather than vertices are removed (sometimes referred to as **bond-percolation**).

Finally, we note that the Q-matrix of G can *itself* be interpreted as a weighted graph, written in adjacency format. That is, $Q(G)$ is itself a graph. In this case one could apply this formulation twice, $Q(Q(G))$, to create a Q^2-matrix, or any number of times to create the Q^n-matrix. Such an approach would produce a family of graph reductions that could collapse a large network graph into a single number. We are just beginning to investigate the implications of these extended interpretations.

9 Conclusion

The Q-matrix is a condensed representation of a network graph, which provides a meaningful visualization and encodes several measures of the graph's underlying topological structure. It is small, relatively easy to compute, and provides a convenient identification of the original network graph. (The Q^* formulation is used in practice, but both are mathematically equivalent.)

We have illustrated Q-matrix identities for computing the degree distribution, giant component growth, and basic parameters of undirected graphs (Eqs. 8, 8, 10, 11, and 12.)

Computing the Q-matrix is computationally efficient. Optimized algorithms allow networks with millions of edges to be processed in a few seconds on a laptop. Furthermore, the resulting Q-matrix is compact. The size of a Q-matrix grows around $O(n^{0.4})$ as the number of edges in the original graph, thus the file size ratio approaches zero for large networks. For example, the LiveJournal network[7] has nearly 69 million edges, yet its Q-matrix requires less than 67 thousand values – a reduction ratio about 1,000:1.

Experimental data indicates that the visualization provided by the Q-matrix distinguishes between graphs from different applications areas (Fig. 3) and that graphs from the same application area share visual similarities (Fig. 4-9). This includes examples from citations graphs, web graphs, road networks, peer-to-peer networks, autonomous networks, email networks, and Wikipedia networks, ranging from sizes of just a few thousand to nearly 70 million edges[14] [7]. While these experiments are not exhaustive of all network data available, they do suggest that the approach appears promising in practice.

The Q-matrix approach can also reveal differences between an organic (real-world) graph and randomized variations from its corresponding configuration model (ensemble of random graphs with identical degree distribution). We have shown example cases where the giant component grows much slower, by as much as three orders of magnitude, and such difference can be computed exactly from their corresponding Q-matrices .

Furthermore, the difference between Q-matrices of different graphs may be quantified by the induced Q-metric $\Delta(A, B)$, as given by Eq.13. This defines an exact, reproducible measure for network graphs which can be also be useful in identifying application graphs from their randomized counterparts (Fig.14).

Finally, we have outlined extensions to this approach that for directed and weighted graphs, as well as generalized percolation orderings, like eigenvalue or between centrality. We have also proposed a recursive Q-matrix formulation approach that can reduce a large network graph to a single number.

The understanding of large-scale networks remains a challenging problem, and hopefully such approaches may shed light on our comprehension of systems. There is still much work to be done, and we hope that these formulations can help further that understanding.

References

[1] J.P. Bagrow, E.M. Bolt, J.D. Skufca, and D. Ben-Avraham. Portraits of complex networks. *EPL (Europhysics Letters)*, 81, 2008.

[2] A. Barabási. Scale-free networks: A decade and beyond. *Science*, 325:412, 2009.

[3] D. S. Callaway, S. H. Strogtaz M. E. J. Newman and, and D. J. F. Watts. xxx. *Phys. Rev. Letters*, 85:5468–5471, 2000.

[4] A. Aharony D. Stauffer. *Introduction to Percolation Theory.* Taylor and Francis, London, 1992.

[5] H. W. Hethcote. The mathematics of infectious diseases. *Siam Rev.*, 42:599–563, 2000.

[6] Y. Hu. Algorithms for visualizing large networks. In Uwe Naumann and Olaf Schenk, editors, *Combinatorial Scientific Computing*, pages 525–549. Chapman & Hall/CRC Computational Science Series, CRC Press, 2012.

[7] J J. Leskovec. Stanford Large Network Dataset Collection. `http://snap.stanford.edu/data`.

[8] B. Klimt and Y. Yang. Introducing the enron corpus. In *CEAS*, 2004.

[9] B. Reed M. Molloy. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probability and Computing*, 7:295–305, 2008.

[10] Mathematica. *version 7.0.* Wolfram Research, Inc., Champaign, Illinois, 2008.

[11] MATLAB. *version 7.10.0 (R2010a).* The MathWorks Inc., Natick, Massachusetts, 2010.

[12] R. Pozo. Efficient Q-matrix computation for the visualization of complex networks. In *Proceedings of Complex Networks, SITIS*, 2012.

[13] A. Barabási R. Albert, H. Jeong. Attack and error tolerance in complex networks. *Nature.*, 406:378–382, 1999.

[14] Y. Hu T. Davis. University of Florida Sparse Matrix Collection. *ACM TOMS*, 38, 2011.

[15] D. Watts and S. Strogatz. Collective dynamics of ‘small-world’ networks. *Nature*, 393:440–442, 1998.