OPEN ACCESS Freely available online

PLoS one

A Curve Shaped Description of Large Networks, with an Application to the Evaluation of Network Models

Xianchuang Su[1,2], Xiaogang Jin[1,2]*, Yong Min[1,2], Linjian Mo[1], Jiangang Yang[1,2]

1 Institute of Artificial Intelligence, College of Computer Science, Zhejiang University, Hangzhou, Zhejiang, China, **2** Ningbo Institute of Technology, Zhejiang University, Ningbo, Zhejiang, China

Abstract

Background: Understanding the structure of complex networks is a continuing challenge, which calls for novel approaches and models to capture their structure and reveal the mechanisms that shape the networks. Although various topological measures, such as degree distributions or clustering coefficients, have been proposed to characterize network structure from many different angles, a comprehensive and intuitive representation of large networks that allows quantitative analysis is still difficult to achieve.

Methodology/Principal Findings: Here we propose a mesoscopic description of large networks which associates networks of different structures with a set of particular curves, using breadth-first search. After deriving the expressions of the curves of the random graphs and a small-world-like network, we found that the curves possess a number of network properties together, including the size of the giant component and the local clustering. Besides, the curve can also be used to evaluate the fit of network models to real-world networks. We describe a simple evaluation method based on the curve and apply it to the *Drosophila melanogaster* protein interaction network. The evaluation method effectively identifies which model better reproduces the topology of the real network among the given models and help infer the underlying growth mechanisms of the *Drosophila* network.

Conclusions/Significance: This curve-shaped description of large networks offers a wealth of possibilities to develop new approaches and applications including network characterization, comparison, classification, modeling and model evaluation, differing from using a large bag of topological measures.

Citation: Su X, Jin X, Min Y, Mo L, Yang J (2011) A Curve Shaped Description of Large Networks, with an Application to the Evaluation of Network Models. PLoS ONE 6(5): e19784. doi:10.1371/journal.pone.0019784

Editor: Vladimir Brusic, Dana-Farber Cancer Institute, United States of America

Received December 13, 2010; **Accepted** April 14, 2011; **Published** May 17, 2011

Funding: This work was supported by the National Science Foundation of China grants 61070069 and 60803110 (http://www.nsfc.gov.cn/). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

* E-mail: xiaogangj@cise.zju.edu.cn

Introduction

Networks have been widely used as a concise mathematical representation of the structure of systems with interacting objects [1–4]. Protein-protein interaction networks, brain networks, scientific collaboration networks, the Internet and the World Wide Web are a few examples.

Decades ago, the study of graph theory focused on the analysis of small networks, or regular graphs such as a lattice. One could easily lay out the network on a piece of paper and visually investigate its features. However, real-world networks studied in recent years often involve thousands or millions of vertices and edges. Networks on this scale cannot be easily represented in a way that allows quantitative analysis to be conducted by eye [5]. Instead of network drawing, the current understanding of network structure relies mainly on specific properties, measures or statistics, such as degree distributions [6,7], community structure measurements [8–10], or motif counts [11]. But one may note that specific properties characterize the structure of networks point-by-point. We are used to carrying a large bag of measures to describe a network. A good description or representation of network which holds more complete topological information in one bag may provide a clear intuitive understanding of network and reflect some special structural features, such as the curved landscape of the World Wide Web [12], cartographic representation of complex networks [13] and circular perspective drawings of protein interaction networks [14].

With this view in mind, we propose a mesoscopic description of large networks by using breadth-first search. It serves as a bridge linking networks of different structures with a set of particular curves. We use curves of this kind to represent the corresponding networks and refer to them as the *characteristic curves*. Then we apply this curve shaped description to both random graphs and lattice embedded random regular graphs, and derive the expressions of their curves. The curve expression possesses a number of network properties in one bag, such as the size of the giant component and the local clustering. Interestingly, it shows that not only homogeneous random graphs appear to have a power-law degree distribution $P(k) \sim k^{-1}$ under traceroute sampling [15,16], but a small-world-like network also does.

Moreover, characteristic curves or functions shaped by network structures can be used to compare networks compre-

hensively, e.g., the mesoscopic response function [17] resembling fingerprints. The network structural comparison has many applications. A useful one is to evaluate how well a network model fits a real-world network by comparing the network generated by the model with that of the real world. In recent years, network modeling has been attracting tremendous attention. Various models have been proposed to reproduce the topology of the real-world networks to infer their underlying growth mechanisms. Among the notable ones are the preferential attachment model [18,19] and the small-world model [20]. Even a specific real-world network often has a variety of well-fitting models. Take protein-protein interaction (PPI) networks as an example, there are multiple models of widely varying mechanisms (e.g. [21–25],) that perfectly fit the real PPI data in terms of selected network properties, such as the degree distributions or the clustering coefficients. However, questions arise: among so many good models, which one best reproduces the structure of the real data? Which one best reveals the underlying growth mechanisms? It's clear that comparing the well fitted network properties mentioned above is not sufficient to identify the best-fitting model. It needs a discriminative method for network comparison to evaluate the fit of the models to the data.

Recent studies of structural comparison for PPI networks show that the comparison methods based on local structural properties, such as graphlet counts [26–28] or subgraph census [29], have a strong power in discriminating the differences between networks. However, the methods paying too much attention on local network properties may fail to distinguish some obvious global differences between two networks (see section "Evaluation Results" for detailed discussions), and they usually require a large amount of computation time and will be computationally infeasible for large networks with high average degree.

To deal with these issues, we use a fast method to compare large networks that works by comparing their characteristic curves, which are shaped by both the local and global structures of the network. First, we introduce a simple graph distance to evaluate the structural difference between two networks by comparing their curves. The graph distance can then be used to evaluate the fit of a network model to the real data. We apply this evaluation method to the *Drosophila melanogaster* PPI network [30] along with three network models, including linear preferential attachment model [19] and two biologically motivated network models [21,22]. The evaluation results then determine which model better reproduces the topology of *Drosophila's* network. We also compare our results with that achieved by a method using subgraph census and machine learning techniques [29]. And at the same time, we examine the strengths and weaknesses of the two methods.

Methods

In this section, we first describe a network representing method. Then we apply the method to random graphs and lattice embedded random regular graphs, and derive the expressions of their characteristic curves. For the structural comparison between large networks, we introduce a graph distance based on the curve, and apply it to the *Drosophila* PPI network to evaluate the fit of the selected models to it.

Network Representing Method

Consider a network of N vertices and M edges (the terms network/graph, vertex/node and edge/link are interchangeable in this paper). For the convenience of description, we assume that the network is undirected and connected in this section, i.e., every edge in the network is undirected and every pair of distinct vertices can be connected through some path. The proposed representing method is based on the algorithm of breadth-first search (BFS) [31], where the root vertex is selected by taking one end of a randomly chosen edge (different root selection schemes yield different outputs, the affects of root selection are discussed in details in section 3 in Supporting Information S1). One can consider the process of BFS as exploring the graph one vertex at a time in the order of first touch, first explore. At the beginning, the root vertex is labeled pending, and all other vertices are untouched. As an ongoing process (see Figure 1B), a pending vertex will be explored and all its untouched neighbors will be labeled pending and pushed into a queue named *QueueT* in a random order. Each of them is assigned a *position* $x(0/N < x \leq N/N)$ which is the ratio of its sequence in the queue to N, and stores y, the position of its parent who brings it to the queue, i.e., who touches it at first during the process of search. Taking these two sets of positions as the coordinates (x,y) of the vertices, the search tree is mapped into a two-dimensional plane (see Figure 1C) and we refer to it as *BFS-tree*, where each edge is represented by a straight line with one right angle and parallel to each other.

Note that the BFS-tree is not a full representation of the original graph since it has lost too many edges. To get the full linking information, we now record all links of the graph during BFS. Create k copies for each vertex of degree k, and replace each undirected edge with two opposite directed edges connecting two copies owned by the corresponding vertices. Unlike QueueT which only accepts untouched neighbors of the vertex on exploring, another queue named *QueueG* accepts the copies of all its neighbors to preserve full linking information (see Figure 1B). Meanwhile, it is similar to the vertices of QueueT that each copy of QueueG is assigned a position X (the ratio of its order in QueueG to N) and stores Y (the position of its parent copy). Thus the coordinates (X,Y) help to map a network into a two-dimensional plane (see Figure 1D) which is referred to as *BFS-graph*.

Both the BFS-tree and BFS-graph are in the two-dimensional plane, and every vertex or copy can see its neighbors through a mirror placed on the line $y=x$ or $Y=X$. By associating vertex and edge with optical element and light beam, respectively, such a simple layout has potential applications in manufacturing large-scale optical networks. For a large network, as illustrated in Figure 2, the global picture becomes very clear where the vertices or copies line up, and automatically forms a particular curve. Since the BFS-graph holds more linking information than the BFS-tree, we here use the curve of the BFS-graph to represent the corresponding network and refer to it as the *characteristic curve*.

Characteristic Curves

It is desirable to find the exact expressions of the characteristic curves for various networks, and see whether the curves indeed identify networks of different structures. To proceed, let us first track the states of QueueT and QueueG. During the process of BFS, network is explored one vertex at a time (can also be explored one edge at a time, the conclusions are consistent, see section 1 B in Supporting Information S1 for details). Consider a vertex A to be explored at time T has graph degree $G(T)$, and also T/N is A's position in QueueT. After A is explored at time $T+1$, it has one parent and $H(T)-1$ newly touched children, where $H(T)$ is A's degree on the search tree. The states of QueueT and QueueG change as follows, probing the linking information of network:

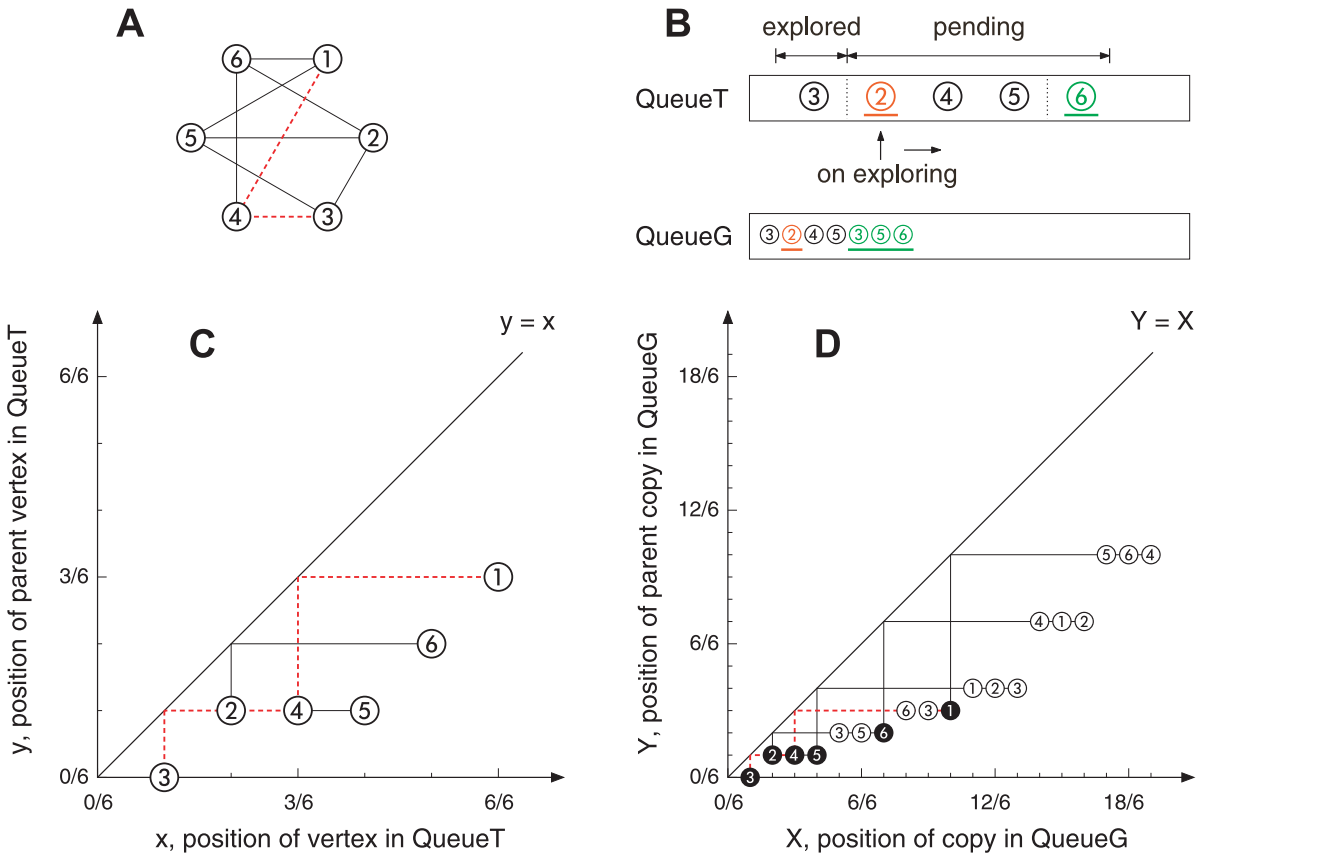

Figure 1. An example of the network representing method. A: A random 3-regular graph of six vertices, where each vertex has three neighbors randomly selected. **B:** A snapshot of the process of BFS: after vertex *3* has been explored, the pointer of QueueT moves to vertex *2*. We explore the neighbors of *2* in a random order *3*, *5*, *6*. Only untouched vertex *6* is pushed into QueueT and assigned coordinates (5/6, 2/6). To preserve all linking information of *2*, we push the copies of *3*, *5* and *6* into QueueG and assign them coordinates (5/6, 2/6), (6/6, 2/6) and (7/6, 2/6), respectively. Then the pointer moves on to *4*. **C:** BFS-tree. **D:** BFS-graph, we highlight the copies in black for their first appearances in QueueG. The line with one right angle represents an edge connecting two vertices or copies. For example, in panel **D**, polylines (2/6, 1/6)-(2/6, 2/6)-(6/6, 2/6) and (4/6, 1/6)-(4/6, 4/6)-(12/6, 4/6) represent an undirected (bidirectional) edge connecting two vertices *2* and *5*. So a vertex can see all its neighbors through a mirror placed on the line Y = X. The dotted polylines (red) represent a pathway *3 - 4 - 1*.
doi:10.1371/journal.pone.0019784.g001

$$\begin{aligned} L_{QT}(T+1)-L_{QT}(T)&=H(T)-1,\\ L_{QG}(T+1)-L_{QG}(T)&=G(T). \end{aligned} \tag{1}$$

where $L_{QT}(T)$ is the number of vertices that QueueT holds and $L_{QG}(T)$ is the number of copies that QueueG holds right before exploring A at time T. In the proposed representing method, each vertex or copy is assigned a coordinates (x,y) or (X,Y) which records the positions of it and its parent. Thus, when the network is explored one vertex at a time, Eq.1 can be written as:

$$\frac{\Delta x}{\Delta y}=H(yN)-1, \quad \frac{\Delta X}{\Delta y}=G(yN). \tag{2}$$

where the initial values of x,y,X and Y are all zeroes, and y increases at a rate of $1/N$ per time step. Hence, knowing the values of every vertex's graph degree $G(yN)$, tree degree $H(yN)$ and its position y in QueueT are crucial for the derivation of the curve expressions.

We then apply this approach to two undirected networks. One is random graphs with arbitrary degree distributions, including random regular graph (RRG), Poisson-distributed random graph (PoissonRG) and power-law distributed random graph (PLRG). The other is lattice embedded random regular graph (LERRG) which is not only similar to many real-world networks, but also has practical applications. We use $y=f(x)$ and $Y=F(X)$ to represent the function of the tree curve and graph curve, respectively, where root vertex is in the giant component of the graph (a giant component is a connected subgraph that contains a majority of the entire graph's vertices). In general, $y=f(x)$ and $Y=F(X)$ are nondecreasing and satisfy: $x,y\in(0,1]$, $f(x)\le x$, $X,Y\in(0,\langle k\rangle]$ and $F(X)\le X$, where $\langle k\rangle$ is the average degree of the graph. The smallest positive root of $x=f(x)$ is just the size of the giant component.

Random Graphs with Arbitrary Degree Distributions. Suppose the degree distribution of a random network is $P(k)=p_k$, defined as the probability that a randomly chosen vertex has k edges. Meanwhile, consider the network is obtained from the configuration model [3]: create k copies for each vertex of degree k, and then choose pairs of these copies uniformly at random and connect them to form the edges. Such network is a multi-graph with self-loops and multiple edges permitted. To derive the curve expressions of BFS-tree and BFS-graph for this network, as Eq. 1 shows, we should at first know the values of $G(T)$ and $H(T)$ varying with T.

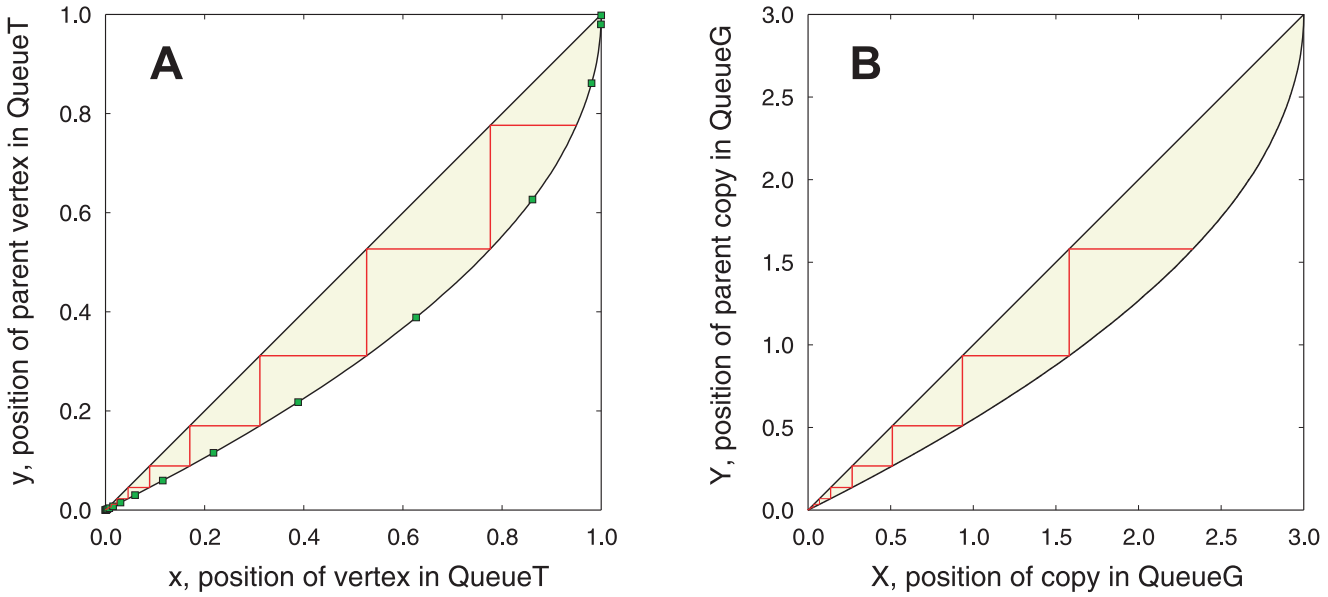

Figure 2. Diagrams of a random r-regular graph of size $N=10^5$ and $r=3$. A: BFS-tree, where vertices are closely located around the curve $(1-x)=(1-y)^2$. Each small square (green) represents the last vertex of its tree level of the BFS tree. **B:** BFS-graph, where copies of vertices are closely located around the curve $(1-X/3)=(1-Y/3)^2$. In the two diagrams, the shaded areas (yellow) represent the edges, and the polylines with right angles (red) represent a same shortest path between the root and a destination node.
doi:10.1371/journal.pone.0019784.g002

During the process of BFS, QueueT accepts newly touched vertices one by one and assigns them positions. The term $G(T)$ stands for the number of edges possessed by a vertex with position T/N. To trace the value of $G(T)$ varying with T, consider a situation when QueueT has accepted $tN-1 (0/N<t\leq N/N)$ vertices and is going to accept a new one A. The new vertex A will be pushed into QueueT and assigned position t, our goal is to find A's degree $G(tN)$.

Vertex A is selected from the $(1-t)N+1$ untouched vertices. Because in a random network, the copies of vertices are coupled uniformly at random, the probability of vertex A having degree k is proportional to $kp'(k)$, where $p'(k)$ is the degree distribution of the $(1-t)N+1$ untouched vertices. The distribution $p'(k)$ varies with $(1-t)N+1$ when QueueT obtains untouched vertex one by one. For the technical convenience to describe the relationship between $p'(k)$ and t, we use $p_k e^{-zk}/\sum_{k'=0}^{\infty} p_{k'} e^{-zk'}$ to represent $p'(k)$, where z is a variable changes as a function of t: $\sum_{k=0}^{\infty} p_k e^{-zk}=1-t+1/N$. Let

$$S_0(z)=\sum_{k=0}^{\infty} p_k e^{-zk}, S_1(z)=\sum\nolimits_{k=0}^{\infty} k p_k e^{-zk}, \quad S_2(z)=\sum\nolimits_{k=0}^{\infty} k^2 p_k e^{-zk}. \tag{3}$$

where $z\geq 0$ (note that $S_0(0)=1$ and $S_1(0)=\langle k\rangle$, which is the average degree of the graph). Then we arrive at the distribution $p'(k)=p_k e^{-zk}/S_0(z)$, where z changes as a function of t in the limit of large N (the term $1/N$ is omitted):

$$S_0(z)=1-t \tag{4}$$

Let $g(t)=E[G(tN)]$ be the expected graph degree of the newly touched vertex A. Since the probability of vertex A having degree k is proportional to $kp'(k)=kp_k e^{-zk}/S_0(z)$, we can write:

$$g(t)=\sum_{k=0}^{\infty} k\frac{kp_k e^{-zk}}{S_1(z)}=\frac{S_2(z)}{S_1(z)} \tag{5}$$

Next, we trace the value of the tree degree $H(T)$. Suppose xN vertices have been touched before exploring a vertex A with position y. In the limit of large N, the expected number of untouched vertices that A will meet through its $(G(yN)-1)$ edges (except one edge connecting its parent) is:

$$E[H(yN)]-1=\frac{2M-\sum_{t=0}^{x} G(tN)}{2M-\sum_{t=0}^{y} G(tN)}(G(yN)-1) \tag{6}$$

where M is the total number of edges, see section 1 A in Supporting Information S1 for the detailed explanation of this equation. This equation is also valid for random graphs with extremely dense edges ($\langle k\rangle\sim N$), which have numerous self-loops and multi-edges (see section 1 B in Supporting Information S1 for details).

In the limit of large N, we use a mean-field approximation where $G(tN)$ and $H(tN)$ are represented by their expectations $g(t)$ and $h(t)$, respectively. Substituting Eqs. 2 and 5 into Eq. 6 and associating it with Eqs.3 and 4, the curve function $y=f(x)$ of BFS-tree satisfies (see section 1 C in Supporting Information S1 for the detailed derivation):

$$\begin{aligned} x&=1-S_0(z(x)),\\ y&=1-S_0(z(y)),\\ z(x)&=\ln\frac{\langle k\rangle}{S_1(z(y))}-z(y). \end{aligned} \tag{7}$$

where $0\leq y\leq x\leq t_{end}\leq 1$, $t_{end}=1-S_0(z(t_{end}))$. $z(t_{end})$ is the smallest positive root of $2z=\ln\langle k\rangle-\ln S_1(z)$. Note that t_{end} is simply the size of the giant component of the graph, which is

consistent with the size derived in different forms by Molloy and Reed [32] and Newman *et al.* [33] for random graphs with arbitrary degree distributions.

From Eqs.3 and 4, we get $d\,z = d\,t/S_1(z)$, substituting this into Eqs. 2 and 5, the curve function $Y = F(X)$ of the BFS-graph satisfies:

$$\begin{aligned} X &= \langle k\rangle - S_1(z(y)),\\ Y &= \langle k\rangle - S_1(z(f(y))). \end{aligned} \tag{8}$$

where $0 \le Y \le X \le T_{end} \le \langle k\rangle$, $T_{end} = \langle k\rangle - S_1(z(t_{end}))$. As mentioned above, t_{end} is the size of the giant component and $z(t_{end})$ is the smallest positive root of $2z = \ln\langle k\rangle - \ln S_1(z)$. When x reaches t_{end}, the BFS explored all vertices in the giant component and the mapping comes to the end (we here only consider the curves of the giant component since it retains the significant structural features of the graph).

As examples, we now introduce three commonly studied graphs.

(1) *Random r-regular graphs*. In a graph of this kind, each vertex has a fixed degree r, $G(T) \equiv r$. The curve functions are:

$$\begin{aligned} 1 - x &= (1-y)^{r-1},\\ 1 - X/r &= (1 - Y/r)^{r-1}. \end{aligned} \tag{9}$$

where $0 \le y \le x \le 1$, $0 \le Y \le X \le r$, and $r \ge 3$ which implies that the graph is connected with high probability [34,35].

(2) *Poisson-distributed random graphs*. This is one of the best studied graph models [34], and is also known as Erdös-Rényi random graph that has a Poisson degree distribution in the limit of large graph size, as given by $p_k = \langle k\rangle^k e^{-\langle k\rangle}/k!$. The curve functions are (see section 1 D in Supporting Information S1 for the detailed derivation):

$$\begin{aligned} y &= -\frac{\ln(1-x)}{\langle k\rangle},\\ X &= \langle k\rangle y - (1-y)\ln(1-y),\\ Y &= \ln\frac{1}{1-y} - \left(\frac{\ln(1-y)}{\langle k\rangle} + 1\right)\ln\left(\frac{\ln(1-y)}{\langle k\rangle} + 1\right). \end{aligned} \tag{10}$$

where $0 \le y \le x \le t_{end} < 1$, and t_{end} is the smallest positive root of $t = 1 + LambertW(-\langle k\rangle e^{-\langle k\rangle})/\langle k\rangle$. $LambertW$ is Lambert's function, defined as $LambertW(u) = w$ where $we^w = u$.

(3) *Power-law distributed random graphs*. It was found that a wide range of real networks, such as the Internet and science collaboration graph, display power-law degree distributions, also known as scale-free networks [1]. In Figure 3, we only consider a random graph possessing a power-law degree distribution given by

$$\begin{aligned} p_k &= Ck^{-\alpha} \quad for\ 1 \le k_{min} \le k \le k_{max}\\ &\equiv 0 \qquad otherwise \end{aligned}$$

where α is a constant and $C = 1/\sum_{k=k_{min}}^{k_{max}} k^{-\alpha}$. k_{min} and k_{max} are the minimal and maximal degree of the graph, respectively. The curve expressions are the same as Eqs.7 and 8.

Lattice Embedded Random Regular Graphs. A graph of this type is formed from a superposition of an r-RRG and a d-dimensional finite lattice with periodic boundary conditions, i.e., each vertex has $2d$ nearest lattice neighbors and r long-range neighbors chosen uniformly at random from the lattice. This is similar to the small-world model proposed by Watts and Strogatz [20], in which there are many local links and a few long-range links connecting local clusters together. These links lead to both small path lengths and high clustering called small-world property and have been observed in a wide range of real-world networks, such as the collaboration graph of film actors and the power grid. Moreover, the LERRG is not only similar to a number of real-world networks, it also has practical applications. For example, Korniss *et al.* [36] and Guclu *et al.* [37] found that two typical graphs of LERRGs have remarkable advantages in constructing a parallel discrete-event simulation scheme since the processing elements can carry out the tasks distributed on them at a nonzero, near-uniform rate without requiring global synchronization.

In the LERRG, $G(T) \equiv 2d + r$, and in the limit of large network size $N \to \infty$, $E[H(T)] - 1 = \lambda_1(1-x)/(1-y)$, where λ_1 is the largest real root of $(\lambda-1)^d = r(\lambda+1)^{d-1}$ (see section 2 in Supporting Information S1 for the detailed derivation). In association with Eq. 2, the curve functions are:

$$\begin{aligned} 1 - x &= (1-y)^{\lambda_1},\\ 1 - \frac{X}{2d+r} &= (1 - \frac{Y}{2d+r})^{\lambda_1}. \end{aligned} \tag{11}$$

Interestingly, they have a similar form as that of RRGs (Eq.9), and are consistent with Eq.9 when $d = 0$.

Graph Distance

Each of the example networks studied above corresponds to a particular curve. We here use the curve as a discriminating feature for network comparison. To evaluate the structural difference between two networks, we describe a simple graph distance $D_{\mathcal{G}}$ by comparing their curves

$$\begin{aligned} D_{\mathcal{G}}(\mathcal{G}_1, \mathcal{G}_2) &= \sum_{X=0}^{\langle k\rangle} |\mathcal{G}_1(X) - \mathcal{G}_2(X)|\,\frac{1}{2M}\\ \mathcal{G}(X) &= \begin{cases} Y/\langle k\rangle, & 0 \le X \le T_{end}\\ X/\langle k\rangle, & T_{end} < X \le \langle k\rangle \end{cases} \end{aligned} \tag{12}$$

where $\mathcal{G}(X)$ represents the characteristic curve. $\mathcal{G}_1(X)$ and $\mathcal{G}_2(X)$ stand for the curves of a pair of graphs to be compared. The distance $D_{\mathcal{G}}$ is simply the area between the two curves. Note that Onnela et al. define a graph distance based on mesoscopic response function in a similar fashion and performs well for network taxonomy [17]. Because the BFS-graph holds more linking information than the BFS-tree, we chose the curve of the BFS-graph to calculate the difference. The two-tuple (X, Y) is the coordinates of vertex's copy in the BFS-graph, and X increases at a rate of $1/2M$, where M is the total number of edges in the graph. To align two graphs with different average degrees $\langle k\rangle$, we assign $Y/\langle k\rangle$ to $\mathcal{G}(X)$ until X reaches T_{end}, that is, until the BFS has explored all vertices in the giant component. To ease the calculation of the distance $D_{\mathcal{G}}$ between two graphs with different sizes of the giant components, we assign $X/\langle k\rangle$ to $\mathcal{G}(X)$ when the value of X exceeds T_{end}. We only consider the giant component since it retains the significant structural features of the graph. For graphs which consist of small isolated groups of connected vertices, that is, whose giant components are too small (e.g., $T_{end} < 0.1$) to represent the significant structural features of the entire graphs, the

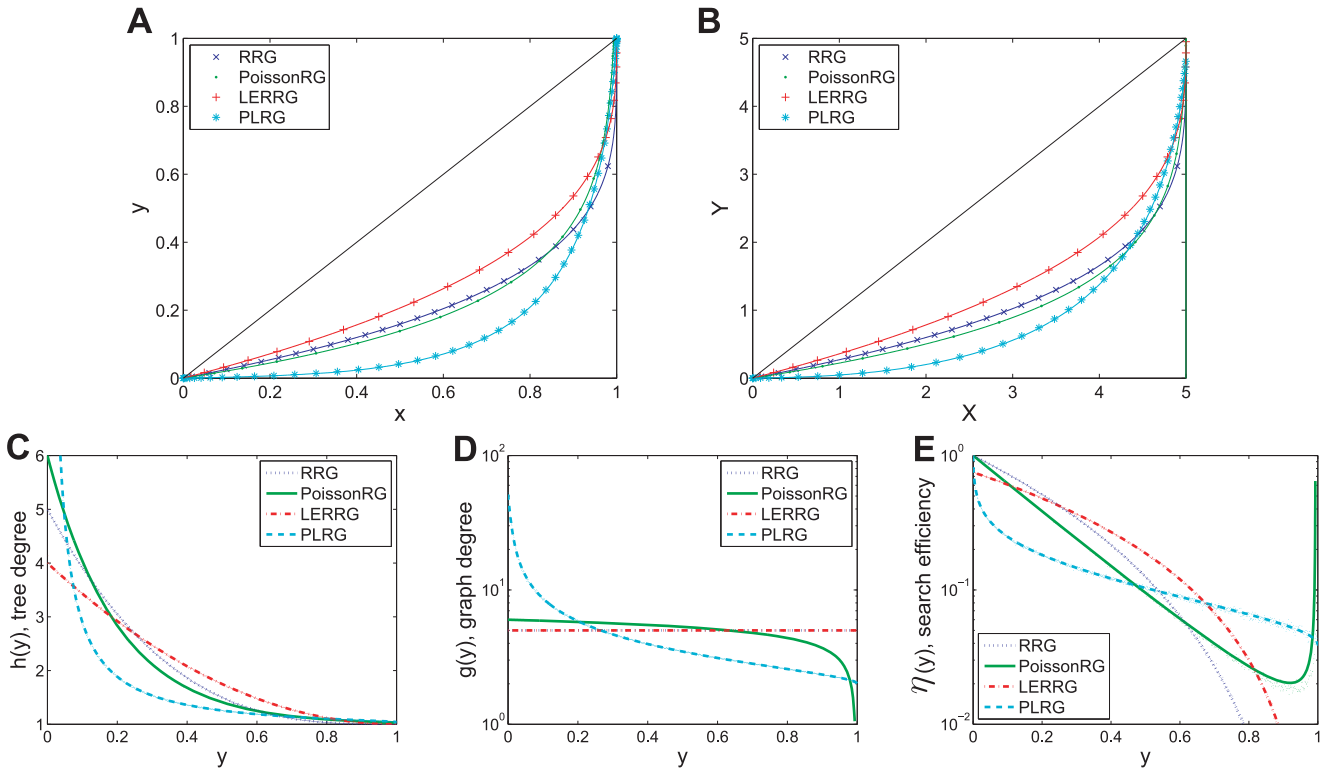

Figure 3. BFS-trees, BFS-graphs and auxiliary views of four example networks. Random regular graph (RRG, $r=5$), Poisson-distributed random graph (PoissonRG, average degree $\langle k\rangle=5$), LERRG ($d=2$, $r=1$, $\lambda_1=3$) and power-law distributed random graph (PLRG, $\alpha=2.41$, $k_{min}=2$, $k_{max}=1{,}000$, $\langle k\rangle\approx 5.02$) with edges not shown. In BFS-trees (panel **A**) and BFS-graphs (panel **B**), each solid line represents the vertices or copies resulted from one run of BFS on the associated network of size $N=10^6$, and the dots are the theoretical values. **C:** $h(y)=dx/dy+1$, the expected tree degree of a vertex on BFS tree varies with its position y in QueueT. **D:** $g(y)=dX/dy$, the expected graph degree. **E:** Here the expected search efficiency $\eta(y)$ is defined as $E[(H(yN)-1)/(G(yN)-1)]$ measuring the efficiency of a vertex exploring new ones through its edges (the η of vertices with one degree are set to zero). In panels **C–E**, the tiny dots are sampled uniformly from simulated results averaged over 10^4 runs of BFS on the associated networks, and the lines are the analytic results.
doi:10.1371/journal.pone.0019784.g003

distance $D_{\mathcal{G}}$ is not suitable to measure the structural difference for them.

As an example, we use $D_{\mathcal{G}}$ to evaluate the differences between the four example graphs in Figure 3. If we take RRG as the center graph, PoissonRG is the most similar graph with $D_{\mathcal{G}}\approx 0.019$ from the RRG. The LERRG is the second similar with $D_{\mathcal{G}}=0.05$ and PLRG is the most different with $D_{\mathcal{G}}\approx 0.063$. The results agree with the common understanding of the four types of graphs.

Data Set

We use a protein-protein interaction data derived from *Drosophila melanogaster* based on yeast two-hybrid screening [30]. A PPI network can be constructed from the data by taking proteins as vertices, and observed interactions between proteins as undirected edges. The degree or connectivity of a protein is defined as the number of its interaction partners. Because the data has numerous false positives, Giot *et al.* [30] assign each interaction a confidence score $P_c\in[0,1]$, measuring how likely the interaction occurs *in vivo*. To exclude unlikely interactions, they suggest a confidence threshold $P_c^*=0.5$. An edge appears only if its confidence score $P_c>P_c^*$. We also present results for a higher threshold $P_c^*=0.65$ which is suggested by Middendorf *et al.* in ref. [29], and $P_c^*=0.0$ which includes all interactions observed. After removing the multiple edges and self-loops from the network [38] and eliminating isolated vertices, the resulting networks consist of 3,279/4,508/6,823 vertices and 2,728/4,569/19,630 edges for $P_c^*=0.65/0.5/0.0$, respectively.

Network Models

We select three network models and compare their generated networks with that of the *Drosophila* to determine which model better describes the evolutionary processes of the *Drosophila*. The first two models are biologically motivated, and have been argued as the best two models to reproduce the *Drosophila* network among seven candidate models [29] including the linear preferential attachment model and the small-world model. The last one is the linear preferential attachment model. All the three models start with a small seed graph and grow the network one vertex at a time following these steps:

Duplication-mutation-complementation model (DMC). The model proposes a gene duplication followed by mutations (divergence) which preserve functional complementarity [22]. At each time step, a new vertex v_{new} is added. It then chooses an existing vertex v_{old} at random, and copies all links of v_{old}, i.e., places edges between v_{new} and all neighbors of v_{old}. For each pair of their links connected to a same neighbor u, one randomly selects one of the two links (v_{new}, u) or (v_{old}, u) and deletes it with a probability q_{del}. It ensures that if one of the duplicate genes loses one of its functions (links), the other preserves the same function (the link to the same neighbor). The duplicate pair v_{old} and v_{new} are themselves connected with a probability q_{con}, representing an interaction of a protein with its own copy. The parameters q_{del} and q_{con} are sampled uniformly in [0,1].

Duplication-mutation using random mutations model (DMR). The model has a different duplication algorithm from

that of the DMC. It emphasizes the creation of new advantageous functions by random mutations in gene and neglects possible interactions between duplicate pairs [21]. At each time step, a newly added vertex v_{new} chooses an existing vertex v_{old} at random, and copies all links of v_{old}. For each link of v_{new} inherited from v_{old}, one deletes it with a probability q_{del}. New links can be created between v_{new} and any other existing vertices with a probability q_{new}/N_t, where N_t is the total number of existing vertices, introducing new viable interactions between proteins. The parameters q_{del} and q_{new} are sampled uniformly in [0,1].

Linear preferential attachment model (LPA). At each time step, a newly added vertex preferentially attaches to existing vertices with probabilities proportional to their degrees [19]. This simple probabilistic model can give rise to scale-free degree distribution which is one of the most important features that many real-world networks exhibit, including the PPI networks.

Network Classification Method

Given a network G and a set of network classes, a network classifier should find which class the network G belongs to. The graph distance (Eq.12) can be used to design a simple and efficient network classifier. Consider a given set of network classes which is composed by network instances, that is, each class possesses a number of networks. If the given set of network classes is composed by network models, we generate a certain amount of network instances for each class. For each of the network classes, the classifier calculates the graph distance $D_{\mathcal{G}}$ between G and every network in this class. We simply use the median graph distance $\tilde{D}_{\mathcal{G}}$ to represent the distance between G and the class, where the median graph distance is a value separating the closer half from the farther half. Finally, the classifier obtains all the median graph distances and classifies G as the class which has the minimal $\tilde{D}_{\mathcal{G}}$ from G.

To validate the proposed classification method, we use four network models, including the DMC, DMR, LPA and PoissonRG, by following steps. First, generate 1,000 instances for each of the four models and obtain their graph curves. Second, build a network G by using one of the four models, and calculate the graph distance between G and the $4\times1{,}000$ graphs generated in the first step. Classify G as the class which has the minimal $\tilde{D}_{\mathcal{G}}$ from G. Repeat the second step 1,000 times for each of the four models, and we obtain a classification accuracy table at last (see Table 1).

The overall classification accuracy is high, around 98.7%, and most of these networks can find back their generative models. Classification errors among DMC, DMR, and PoissonRG networks are due to equivalence of the models in specific parameter regimes and correspondingly show overlaps. For example, when the growth parameter q_{new} of a DMR network approximates to zero, the growth of such a network is dominated by the duplication mechanisms, which is similar to that of the DMC model. Therefore, a small fraction of DMR networks are classified as DMC.

Table 1. Classification accuracy (%) for four network models.

	Classification			
Original	**DMC**	**DMR**	**LPA**	**PoissonRG**
DMC	99.0	1.0	0.0	0.0
DMR	3.4	95.7	0.0	0.9
LPA	0.0	0.0	100.0	0.0
PoissonRG	0.0	0.0	0.0	100.0

The (i, j) entry is the probability of classifying class j given that the original class is i. The networks built by models are based on the size of the *Drosophila* protein network with a confidence threshold of $P_c^*=0.5$.
doi:10.1371/journal.pone.0019784.t001

To test the robustness of our classification method against noise, we carried out a sensitivity analysis by perturbing the structure of the original networks by using two kinds of edge random mechanisms [17,29]. The first is to replace some percentage of original edges in the network by random ones (noise1), and the second is to randomly rewire some percentage of edges while maintaining the degree distribution of the original network (noise2). The numerical results show that the classification performs well for small and intermediate amounts of the noises on the DMC, DMR and LPA networks (see section 4 in Supporting Information S1 for details). Meanwhile, the robustness again the second noise is better than the first one since the second noise maintains the degree distribution of the original network.

Results and Discussion

Properties of Graph Curves

As an example shown in Figure 3, the characteristic curves coupled with auxiliary views identify networks of different topologies and reflect several local and global structural features. Among the four example networks with close average degree $\langle k\rangle$, PLRG is the most special because it has an inhomogeneous degree distribution, where a small fraction of vertices (hubs) are richly connected while many other vertices are not. At an early stage of BFS on PLRG, a small fraction of vertices with high degree are firstly touched. They explore the majority of vertices and leave few opportunities for latter vertices to touch new ones. The $h(y)$, $g(y)$ and $\eta(y)$ of PLRG decline with y much faster than those of the other three homogeneous networks, in which the vertices have approximately the same number of edges. Such decline of LERRG is the slowest due to its high local clustering, where $h(0)<\langle k\rangle$ and $\eta(0)<1$. The two homogeneous random graphs RRG and PoissonRG are the most similar.

Now we turn to characterize the structure of local clustering by the use of search efficiency η. It is known that a highly clustered group of vertices has more links between them than expected by chance. A simple effect of such a structure related to BFS is that, the search explores many links but harvests less new vertices (see an example in Figure 4 A). In contrast, the search on a random graph gets more new vertices with the number close to the links explored (see Figure 4 B). Therefore, the search efficiency η of a vertex in a lattice is smaller than that in a random graph at the first stage of search process.

Furthermore, observe that the search efficiency η of a lattice or an LERRG is lower than that of its random counterpart (a random network with the same degree distribution allowing self-loops and multiple edges, here it is an RRG) at the early stage of the search process, but becomes larger than its counterpart later (see Figures 3 E and 4 D). That is, although the search process catches less new vertices in a clustered network than its random counterpart at first, it still has chance to meet new ones much later for its local clustering structure.

Guided by these observations, we conjecture that the larger the difference of η between a network G with its random counterpart G', the higher the degree of local clustering of G is. We then use a relative difference of η between G and G' to measure the degree of local clustering of G:

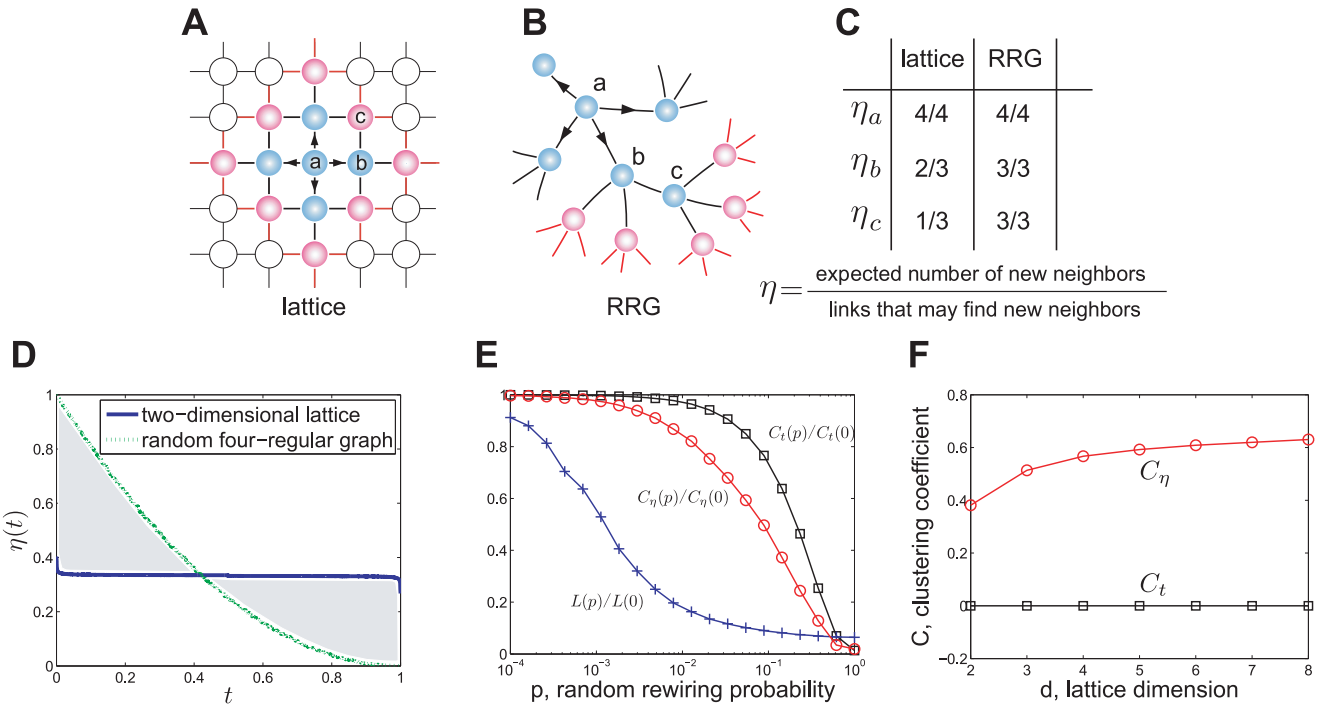

Figure 4. Measure the degree of local clustering. A: The first few steps of a BFS on a two-dimensional lattice. The blue, pink and white vertices stand for the explored, pending and untouched vertices, respectively. **B:** A BFS on a 4-RRG. **C:** Search efficiency. **D:** Search efficiencies of a network (lattice) with its random counterpart (RRG) vary with the vertices' position t in QueueT. **E:** Average shortest path length $L(p)$, average clustering coefficient $C_t(p)$ and $C_\eta(p)$ vary with the random rewiring probability p for a family of small-world networks, where the p transforms a regular ring lattice ($N=1{,}000$ and each vertex has 10 nearest neighbors) to random graphs from 0.0 to 1.0. Each data point is averaged over 100 random realizations of the rewiring process, and have been normalized by the values $L(0)$, $C_t(0)$ and $C_\eta(0)$ of the regular lattice. **F:** Clustering coefficient C_t and C_η vary with d, the dimension of regular lattice. Each data point is averaged over 5 realizations of lattices with network size $N\sim 250{,}000$.
doi:10.1371/journal.pone.0019784.g004

$$C_\eta = \frac{\sum_{t=0}^{1} |\eta_G(t) - \eta_{G'}(t)|}{\sum_{t'=0}^{1} \eta_G(t') + \eta_{G'}(t')} \tag{13}$$

which is simply the area between two η variation curves of G and G' (see Figure 4 D for an example, corresponds to the shaded area between two curves), normalized by the sum of their areas.

To validate this measure, we apply it to small-world networks [20] and regular lattices which are known to have high local clustering, finding that the measure performs similar to the average clustering coefficient for small-world networks (see Figure 4 E), and can catch the local clustering of lattice network where its average clustering coefficient equals to zero (see Figure 4 F).

Next, we consider a series of characteristic curves:

$$\begin{aligned} 1-x &= (1-y)^a, \\ 1-X/b &= (1-Y/b)^a. \end{aligned} \tag{14}$$

where $a \geq 2, b \geq a+1$ and $\langle k \rangle = b$. Both RRGs and LERRGs fall into this category (Eqs.9 and 11) though they are very different since the latter have high local clustering ($b \geq a+1$) but the former have none ($b = a+1$).

Eq. 14 can illustrate what the associated networks look like under a single-source, all-destination traceroute sampling. In the study of Internet mapping, traceroute sampling is widely used to infer the topology of the Internet, typically by collecting paths from a small number of sources to a large number of destinations through the network. However, Lakhina *et al.* [39], Petermann and De Los Rios [40] independently showed that traceroute sampling can significantly bias the observed degree distribution since it only samples a fraction of links. In particular, they found that the sampled subgraphs have power-law degree distributions while the substrate networks are Poisson distributed. Later, Clauset and Moore [15] presented an analytical approach to derive the power law observed in ref. [39]. Achlioptas *et al.* [16] and Dall'Asta *et al.* [41] studied the bias of traceroute sampling analytically and systematically for random graphs. Interestingly, Achlioptas *et al.* found that RRGs also have apparent power laws under traceroute sampling.

Here we find that even a LERRG, which is small-world-like and homogeneous, appears to have a power-law degree distribution $P(k) \sim k^{-1}$ under traceroute sampling (see Figure 5). Since this sampling essentially generates a BFS tree under the common assumption that Internet routing protocols approximate shortest paths [15,16], we turn to calculate the degree distribution of the BFS-tree. Use $h(t) = E[H(tN)]$ to represent the expected tree degree of a vertex with position t in QueueT. Eqs. 2 and 14 give

$$h(t) = a(1-t)^{a-1} + 1 \tag{15}$$

Since $h(t)$ is a monotonic decreasing function, a rough estimate of the tree degree's density $\tilde{P}(h(t))$ can be given by only considering the expected tree degrees during the search process

$$\tilde{P}(h(t)) \sim -\frac{d\,t}{d\,h(t)} \tag{16}$$

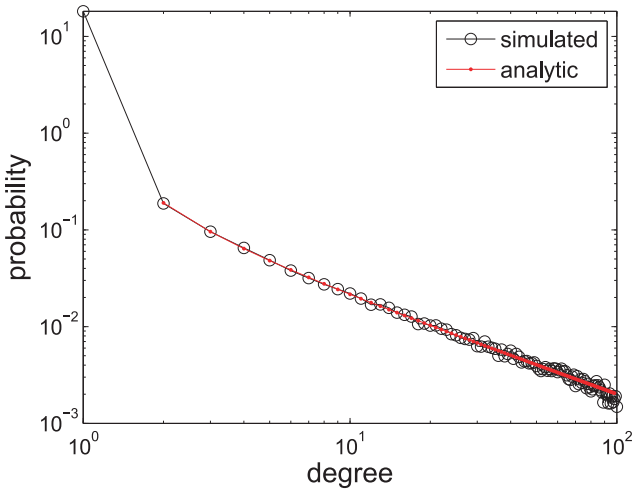

Figure 5. The degree distribution of a BFS tree in a LERRG (d = 10, r = 80). The power-law behavior $P(k) \sim k^{-1}$ extends up to a cutoff at degree $k = 2d + r$. The hollow dots are results from one numerical simulation on a network of size $N = 4^{10}$, and the solid dots are our analytic results.
doi:10.1371/journal.pone.0019784.g005

substituting from Eq. 15 and letting $\tilde{k}$ be the tree degree give

$$\tilde{P}(\tilde{k}) \sim \frac{(\tilde{k}-1)^{-1+1/(a-1)}}{a^{1/(a-1)}(a-1)} \qquad (17)$$

where $1 < \tilde{k} \le a+1$. In the limit of large a, $\tilde{P}(\tilde{k}) \sim (\tilde{k}-1)^{-1}$. For RRGs with $\tilde{k} \ge 3$, this approximate result agrees with a more rigorous one derived by different means by Achlioptas *et al.* [16]. Furthermore, our result is valid not only for RRGs but also for other networks described in Eq.14, including LERRGs. Therefore, even a LERRG, which is small-world-like and homogeneous, displays a power-law degree distribution $P(k) \sim k^{-1}$ (in the limit of large a) under traceroute sampling.

Evaluation Results

Comparing large networks by their graph curves gives an intuitive understanding of the topological differences between the networks of *Drosophila* and each of the three models (see Figure 6, Table 2, and section 5 in Supporting Information S1 for more details). The results suggest that the DMR model better reproduces the topology of *Drosophila*'s network than the DMC and LPA for high confidence thresholds $P_c^* = 0.65/0.5$. To test the robustness of this result, we artificially introduce two kinds of noises into the original *Drosophila* network ($P_c^* = 0.5$), finding that the result still holds for small and intermediate amounts of the noises (see Figure 7).

For the DMC networks, their characteristic curves are far from that of the *Drosophila*, a result which indicates that the structures of the DMC networks are very different from that of the *Drosophila*. This result is completely opposite to the result achieved by a method based on subgraph census [29], which suggests that the DMC best reproduces *Drosophila*'s network among seven candidate models, including the DMR and LPA (see Table 3).

These contradictory results are due to the different angles from which subgraph census and BFS-graph characterize the structure of a network, where the former focuses on the substructures of the network, while the latter cares about a

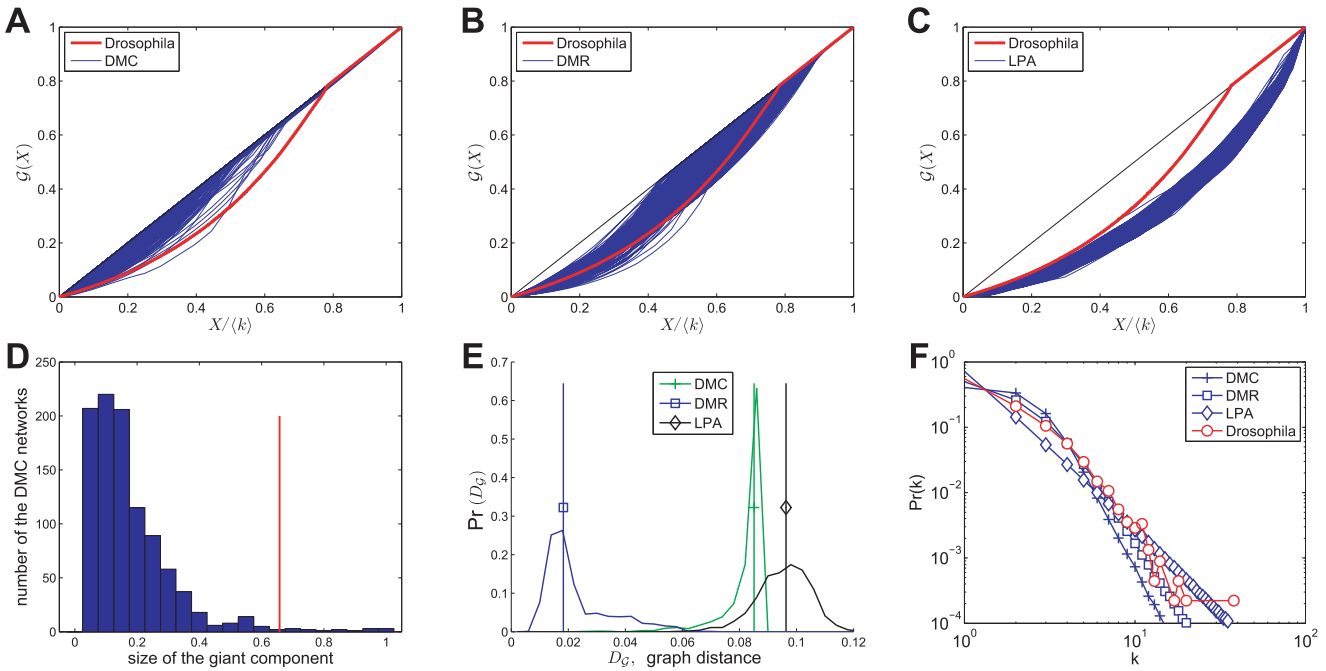

Figure 6. Comparisons between the model networks and the *Drosophila* PPI network for $P_c^* = 0.5$. A–C: BFS-graphs. In each diagram, the thick red curve represents *Drosophila's* network, and the thin blue curves represent the 1,000 generated model networks. **D:** The size distribution of the giant components of the 1,000 DMC networks. In *Drosophila's* network, 66% of the vertices are in the giant component (red vertical bar). **E:** Graph distance distributions. Each vertical bar represents a median graph distance $\tilde{D}_{\mathcal{G}}$ which is a value separating the closer half from the farther half to the center graph, i.e., the *Drosophila* network. **F:** Degree distributions. Each distribution of the three models is averaged over the 1,000 generated networks. Although their degree distributions are similar to that of the *Drosophila*, their curves vary widely.
doi:10.1371/journal.pone.0019784.g006

Table 2. Median graph distance $\tilde{D}_{\mathcal{G}}$ between the model networks and the *Drosophila* PPI network for different confidence thresholds P_c^*, and the model with the minimal distance wins.

	$P_c^*=0.65$		$P_c^*=0.5$		$P_c^*=0.0$	
Rank	**Model**	$\tilde{D}_{\mathcal{G}}$	**Model**	$\tilde{D}_{\mathcal{G}}$	**Model**	$\tilde{D}_{\mathcal{G}}$
1	DMR	0.0162	DMR	0.0183	LPA	0.0351
2	LPA	0.0230	DMC	0.0851	DMR	0.0651
3	DMC	0.0310[a]	LPA	0.0963	DMC	0.2181

[a]For $P_c^*=0.65$, the DMC network consists of small isolated groups of connected vertices. Its giant component is too small (only around 4.5% of the vertices are in the giant component, nine times smaller than that of *Drosophila*, 44%) to represent the significant structural features of the entire graph. Though we give the distance value, the graph distance is not suitable for this case.

doi:10.1371/journal.pone.0019784.t002

global view of the network. Because subgraph census counts every occurrence of a set of small subgraphs in the network, it's clear that the census can reveal more local network properties than the BFS-graph. However, subgraph census so deeply concerns the local network properties that it may fail to distinguish some obvious structural differences between two networks. The most obvious difference between the DMC and *Drosophila* is that the size of the giant component of the DMC network is much smaller than that of the *Drosophila* for high confidence thresholds $P_c^*=0.65/0.5$, where the former is around $0.045/0.18$ (i.e., $4.5\%/18\%$ of the nodes are in the giant component), while the latter is more than nine/three times larger, $0.44/0.66$ (see Figure 6 D, and section 5 in Supporting Information S1). For the higher confidence threshold $P_c^*=0.65$, the DMC network consists of small isolated groups of connected vertices, a structure which is very different from that of the *Drosophila*. This failure of subgraph census implies that although the census knows every occurrence of the particular subgraphs in the network, it lacks a general assembly drawing of how these amounts of subgraphs are assembled into the original large network. The same amount of subgraphs may form a network different from the original network, resembling using the same building blocks to construct different buildings.

On the other hand, the BFS-graph presents a global view of the network by assembling the vertices one by one, which reflects a complementary aspect of the network to that reflected by the degree distribution and subgraph census. The degree distribution counts the degrees of all vertices and shows their distribution. Similarly, subgraph census counts the occurrences of a set of small subgraphs and shows their distribution. The two clearly know the amounts of the building blocks, but lack a general assembly drawing of how to assemble them into the original network. In contrast, the BFS-graph possesses the assembling information of the network through BFS, which strings up the vertices one by one from the bottom up, and at last, gives a global view of the network. Thus, the structural information reflected by BFS-graph and subgraph census complement each other. Applying both of them can provide a more comprehensive understanding of the network structure, which will improve the accuracy of the structural comparison.

Except for the DMC, the two methods based on BFS-graph and subgraph census agree well on the DMR and LPA, that is, the DMR better reproduces the topology of the *Drosophila* network than the LPA for $P_c^*=0.65/0.5$. It is worth noting that the method based on BFS-graph (with time complexity $O(N+M)$) is fast for large networks with high average degree $\langle k\rangle$, for which the subgraph census (with time complexity at least $O(M\langle k\rangle^7)$) may be computationally infeasible. For example, subgraph census will cost a great deal of time for the *Drosophila* network when it includes all interactions observed ($P_c^*=0.0$), which has many more vertices and edges than that for $P_c^*=0.65/0.5$. But the BFS-graph can quickly figure out the differences between the *Drosophila* network and the networks generated by the models (see section 5 in Supporting Information S1 for details). It shows that the fits of the three models to the data are relatively poor for $P_c^*=0.0$ (see Table 2), a result probably due to the presence of strong additional noise in the data when including low confidence value interactions.

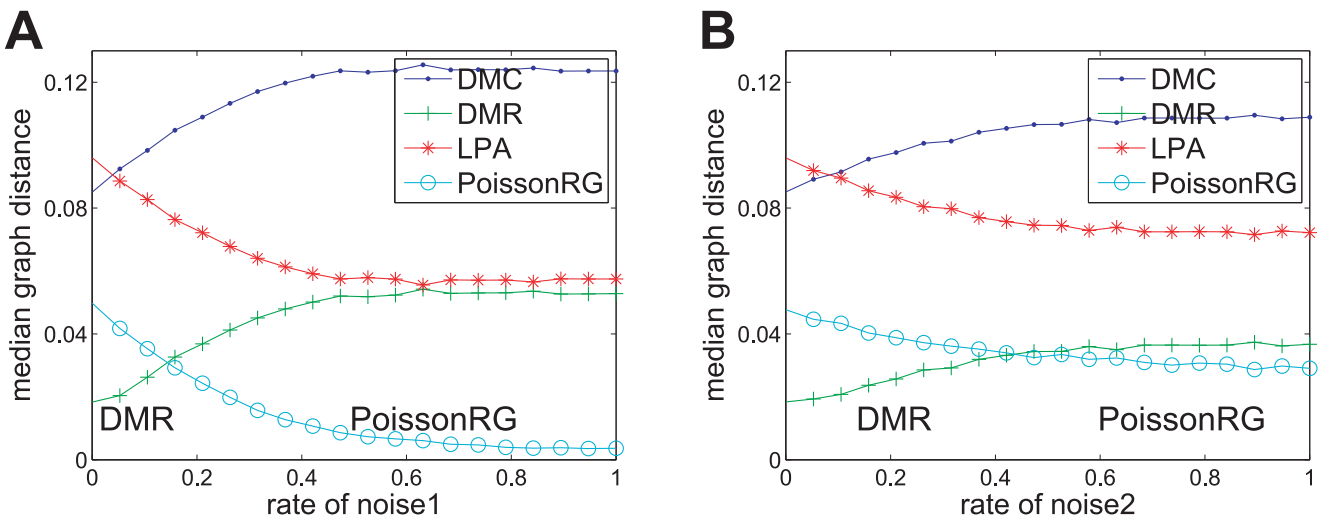

Figure 7. Robustness test against noises for *Drosophila* PPI network ($P_c^*=0.5$). A fraction of edges in *Drosophila* network are replaced by random ones (noise1, panel **A**) or randomly rewired while maintaining the degree distribution of the original network (noise2, panel **B**). Classify the noised network as one of the four classes which has the closest median graph distance. Each data point is averaged over 100 different realizations of the randomization procedure. As validation, the networks are confidently classified as a PoissonRG with the increasing of noise1.

doi:10.1371/journal.pone.0019784.g007

Table 3. The *Drosophila* network is classified as a DMC network over DMR and LPA by a network classification method based on subgraph census [29].

	$P_c^*=0.65$	$P_c^*=0.5$
Rank	**Model**	**Model**
1	DMC	DMC
2	DMR	DMR
3	LPA	LPA

doi:10.1371/journal.pone.0019784.t003

In summary, none of the three models is simultaneously ranked as the best by both the methods based on BFS-graph and subgraph census, implying that there is still room for improvement for these models. The DMC gets a higher rank than the DMR and LPA when using subgraph census, a result indicating that the gene duplications that preserve functional complementarity and facilitate the connections between duplicate pairs are good at reproducing the substructures of *Drosophila*'s network. When using the BFS-graph, the DMR has a closer graph distance to *Drosophila*'s network than those of the other two for high confidence thresholds $P_c^*=0.65/0.5$, a result showing that the gene mutations that create new interactions between proteins are important for keeping the global connectivity of the PPI networks. These results suggest that a model integrating several mechanisms might be able to fit the *Drosophila* PPI network more accurately.

Conclusions

We have presented a mesoscopic description of large networks which associates networks with a set of curves. Specific examples show that the curves can reflect a number of structural features commonly shared by a series of networks. Moreover, the curve can be used to classify networks and evaluate the fit of network models to real-world networks. After evaluating the fit of three network models to the *Drosophila* protein interaction network, we found that the model DMR better reproduces the topology of the *Drosophila* network than the DMC and LPA, although there is still room to improve the three models. We also compared our evaluation method and results with that of Middendorf *et al.*'s in ref. [29], where they identify the best-fitting model based on subgraph census, and found that the structural information reflected by characteristic curve and subgraph census complement each other. Applying the two together can provide a more comprehensive understanding of the network structure, which will improve the accuracy of the structural comparisons and model evaluations.

Using the characteristic curve, we preliminarily investigated the network properties and the fit of network models. Our further work will include the relationship study between the network structure and the curves, conditions that make this relationship one-to-one and the general algorithm (if there is one) that could recover the networks from the characteristic curves. With this algorithm and the well designed curves or functions one could generate networks with required topological features. The network describing method, in essence, utilized the process of BFS and depicted its trace on the network to capture network structure. Other processes such as random walks may also be useful to develop new approaches and applications including network characterization, comparison, classification, modeling and model evaluation.

Supporting Information

Supporting Information S1 Supporting Information S1 includes the detailed derivations of characteristic curves of random graphs and LERRGs, discussion on the effects of root selection, robustness test of the network classification method and more network comparison results.
(PDF)

Acknowledgments

We thank Jianjun Ma, Aaron Clauset, Haijun Zhou, Yingxian Zhao, Xiaolin Shi, Chengbin Peng, Yujie Wan and Hongying Gu for their helpful corrections, comments and suggestions on the manuscript, and Shifa Su, Yixiao Li, Jie Chang, Bo Peng, Lifeng An and Agata Fronczak for useful discussions. Special acknowledgment is due to the anonymous reviewer's valuable comments and suggestions, and Yue Chen for her excellent lectures on Numerical Analysis. We benefited from the KITPC 2008 program "Collective Dynamics in Information Systems".

Author Contributions

Conceived and designed the experiments: XS XJ JY. Performed the experiments: XS XJ YM LM. Analyzed the data: XS XJ YM. Contributed reagents/materials/analysis tools: XS XJ LM. Wrote the paper: XS XJ YM LM JY.

References

1. Albert R, Barabási AL (2002) Statistical mechanics of complex networks. Rev Mod Phys 74: 47–97.
2. Dorogovtsev SN, Mendes JFF (2002) Evolution of networks. Adv Phys 51: 1079–1187.
3. Newman MEJ (2003) The structure and function of complex networks. SIAM Rev 45: 167–256.
4. Boccaletti S, Latora V, Moreno Y, Chavez M, Hwang DU (2006) Complex networks: Structure and dynamics. Phys Rep 424: 175–308.
5. Newman MEJ, Leicht EA (2007) Mixture models and exploratory analysis in networks. Proc Natl Acad Sci U S A 104: 9564–9569.
6. Albert R, Jeong H, Barabási AL (1999) Internet: Diameter of the world-wide web. Nature (London) 401: 130–131.
7. Faloutsos M, Faloutsos P, Faloutsos C (1999) On power-law relationships of the internet topology. ACM SIGCOMM Comput Commun Rev 29: 251–262.
8. Girvan M, Newman MEJ (2002) Community structure in social and biological networks. Proc Natl Acad Sci U S A 99: 7821–7826.
9. Zhou H (2003) Distance, dissimilarity index, and network community structure. Phys Rev E 67: 061901.
10. Fortunato S (2010) Community detection in graphs. Phys Rep 486: 75–174.
11. Alon U (2007) Network motifs: Theory and experimental approaches. Nat Rev Genet 8: 450–461.
12. Eckmann JP, Moses E (2002) Curvature of co-links uncovers hidden thematic layers in the World Wide Web. Proc Natl Acad Sci U S A 99: 5825–5829.
13. Guimerμa R, Amaral LAN (2005) Cartography of complex networks: modules and universal roles. J Stat Mech: Theor Exp 2005: P02001.
14. Li W, Kurata H (2008) Visualizing global properties of large complex networks. PLoS ONE 3: e2541.
15. Clauset A, Moore C (2005) Accuracy and scaling phenomena in internet mapping. Phys Rev Lett 94: 018701.
16. Achlioptas D, Clauset A, Kempe D, Moore C (2005) On the bias of traceroute sampling: or, power-law degree distributions in regular graphs. In: STOC '05: Proceedings of the 37th ACM Symposium on Theory of Computing. New YorkNY, , USA: ACM Press. pp 694–703.
17. Onnela JP, Fenn DJ, Reid S, Porter MA, Mucha PJ, et al. (2010) A taxonomy of networks Available: http://arxiv.org/abs/1006.5731v1.
18. de Solla Price DJ (1965) Networks of Scientific Papers. Science 149: 510–515.
19. Barabási AL, Albert R (1999) Emergence of scaling in random networks. Science 286: 509–512.
20. Watts DJ, Strogatz SH (1998) Collective dynamics of "small-world" networks. Nature (London) 393: 440–442.
21. Solé RV, Pastor-Satorras R, Smith E, Kepler TB (2002) A model of large-scale proteome evolution. Adv Complex Syst 5: 43–54.

22. Vázquez A, Flammini A, Maritan A, Vespignani A (2003) Modeling of protein interaction networks. Com Plex Us 1: 38–44.
23. Wagner A (2003) How the global structure of protein interaction networks evolves. Proc R Soc Lond B Biol Sci 270: 457–466.
24. Berg J, Lässig M, Wagner A (2004) Structure and evolution of protein interaction networks: a statistical model for link dynamics and gene duplications. BMC Evol Biol 4: 51.
25. Ispolatov I, Krapivsky PL, Yuryev A (2005) Duplication-divergence model of protein interaction network. Phys Rev E 71: 061911.
26. Pržulj N, Corneil DG, Jurisica I (2004) Modeling interactome: scale-free or geometric? Bioinformatics 20: 3508–3515.
27. Pržulj N (2007) Biological network comparison using graphlet degree distribution. Bioinformatics 23: e177–e183.
28. Memisevic V, Milenkovic T, Pržulj N (2010) An integrative approach to modelling biological networks. J Integr Bioinform 7: 120.
29. Middendorf M, Ziv E, Wiggins CH (2005) Inferring network mechanisms: The Drosophila melanogaster protein interaction network. Proc Natl Acad Sci U S A 102: 3192–3197.
30. Giot L, Bader JS, Brouwer C, Chaudhuri A, Kuang B, et al. (2003) A Protein Interaction Map of Drosophila melanogaster. Science 302: 1727–1736.
31. Knuth DE (1997) The Art Of Computer Programming, volume 1. Boston: Addison-Wesley, third edition.
32. Molloy M, Reed B (1998) The size of the giant component of a random graph with a given degree sequence. Combin Probab Comput 7: 295–305.
33. Newman MEJ, Strogatz SH, Watts DJ (2001) Random graphs with arbitrary degree distributions and their applications. Phys Rev E 64: 026118.
34. Bollobás B (2001) Random Graphs. Cambridge, UK: Cambridge University Press, second edition.
35. Wormald NC (1999) Models of random regular graphs. In: Lamb JD, Preece DA, eds. Surveys in Combinatorics, 1999. Cambridge, UK: Cambridge University Press, volume 267 of London Mathematical Society Lecture Note Series. *pp 239–298.*
36. Korniss G, Novotny MA, Guclu H, Toroczkai Z, Rikvold PA (2003) Suppressing Roughness of Virtual Times in Parallel Discrete-Event Simulations. Science 299: 677–679.
37. Guclu H, Korniss G, Novotny MA, Toroczkai Z, Rácz Z (2006) Synchronization landscapes in small-world-connected computer networks. Phys Rev E 73: 066115.
38. Yu J, Pacifico S, Liu G, Finley R (2008) Droid: the drosophila interactions database, a comprehensive resource for annotated gene and protein interactions. BMC Genomics 9: 461.
39. Lakhina A, Byers JW, Crovella M, Xie P (2003) Sampling biases in ip topology measurements. In: INFOCOM 2003: Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. PiscatawayNJ, , USA: IEEE Press, volume 1. pp 332–341.
40. Petermann T, De Los Rios P (2004) Exploration of scale-free networks: Do we measure the real exponents? Eur Phys J B 38: 201–204.
41. Dall'Asta L, Alvarez-Hamelin I, Barrat A, Vázquez A, Vespignani A (2006) Exploring networks with traceroute-like probes: Theory and simulations. Theor Comput Sci 355: 6–24.