

Complexity Explorer Networks Series

Max Orhai

October 4, 2012

Contents

1	Introduction	2
2	Small Graphs	3
3	Small World Networks	7
4	Scale-free Networks	12
5	Contagion and Damage Spreading	16
6	Community Structure	16
7	Food Webs	16
8	Genetic Regulatory Networks	16
9	Classroom Network	16

1 Introduction

A mathematical *graph* is, in essence, just some dots with some lines between them. Graph theory is a branch of mathematics which developed out of a variety of different problems (including those in geometry, logistics, transportation, and ecology, to name a few) for which this simple dots-and-lines formalism can provide helpful models.[4] Many of the ‘practical’ techniques in applied graph theory have been developed by sociologists, who have pioneered the empirical study of social relationships represented as graphs. Empirical graphs of any sort, and especially large ones, are often called *networks*.[5] In this document, we’ll use the two terms more or less interchangeably. The emerging field of *network science* is devoted to the study of empirical study of complex ‘real-world’ phenomena modeled as graphs. These include not only natural and social phenomena such as neural populations, patterns of gene expression, food webs, and human collaboration, kinship and community structures, but also technological systems like roads, power grids, or the World Wide Web.

This series of lesson modules, each focused around an interactive NetLogo software model [10], will help students gain familiarity with the basic concepts of network science, and guide them through a few mathematical models and their applications in ecology, genetics, and sociology. The sections are arranged in rough order of difficulty: later sections may depend somewhat on concepts introduced in earlier ones. Exercises, also in increasing order of difficulty, are provided for all lessons.

2 Small Graphs

Quick Overview

Mathematical graphs are a way to describe the discrete structures of everything from molecules to human societies. The small-graph NetLogo model is intended to help you become familiar with some of the most basic terminology and concepts of graph theory, the mathematical foundation of the science of networks. The software model allows you to construct and manipulate concrete visual representations of any directed or undirected graph with exactly ten nodes. Two kinds of representation are explored: a picture of *nodes* and *links*, and an *adjacency matrix*. No mathematical training beyond high school algebra is required, and a sequence of easy exercises helps students learn basic concepts and vocabulary.

Learning Targets

After completing this lesson, students should be able to define in their own words all the italicized terms in this section and correctly answer the questions posed in the exercises.

Essential Concepts

To begin with, a *graph* is a collection $\{a, b, c, \dots\}$ of *nodes* (also called *vertices*), together with a collection of pairs of nodes called *links* (or *edges*) which I'll write here in the form $a \sim b$, where a and b are nodes. It's often convenient to refer to a graph by just its set of links: for example, $\{0 \sim 1, 1 \sim 2\}$. This can be ambiguous, but only if there are nodes having no links. Since the nodes in the model are all in $\{0, 1, 2, \dots, 9\}$, there's no need to mention the unlinked nodes in these graphs. For a *directed graph*, the link $a \sim b$, from a to b , is considered distinct from the 'opposite' link $b \sim a$: the direction is shown in the pictures with an arrow. For an *undirected graph* these two are considered the same.

For our purposes, a *square matrix* can be thought of as a table with the same number of rows and columns. The *adjacency matrix* of a graph is a square matrix, with each column and row corresponding to one node in the graph, and where each element in the table represents a potential link from the column node to the row node. When we write down adjacency matrices or manipulate them mathematically, we typically use '1' or '0' to mean the presence or absence of a link — but since it's much easier to read a matrix of colored squares, the model uses a grey square to mean a link, and a white square to mean no link. For directed graphs, each square corresponds to a different link. But for undirected graphs, links $a \sim b$ between two different nodes (where $a \neq b$) are shown by two entries in the matrix: column a , row b and column b , row a : both will either be 1 (grey) if the link $a \sim b$ is in the graph, or 0 (white) if not. For this reason, we call the adjacency matrix of an undirected graph a *symmetric matrix*: it has a mirror symmetry across one diagonal.

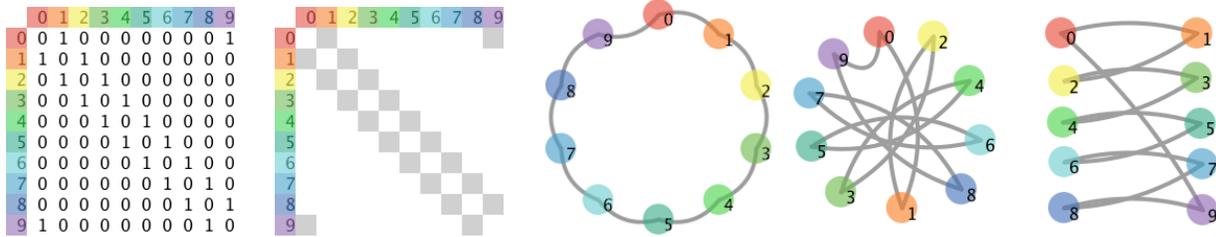


Figure 1: A 10-cycle graph. The adjacency matrix is shown as text and as a colored table, together with three different pictures of the nodes and links.

The model gives you exactly ten nodes, labelled 0 through 9. There are 2^{55} (about 36 quadrillion) distinct undirected graphs on these nodes; see the exercises for more details. Of course, you can ignore ‘extra’ nodes if you want to make a graph with less than ten of them. Although it doesn’t matter to the mathematical structure of a graph and so isn’t shown in the adjacency matrix, you can also move the nodes around in the picture, which can make it look quite different. Figure 1 shows a few different representations of the exact same *cycle* — a ‘circular’ graph of the form $\{a \sim b, b \sim c, \dots, y \sim z, z \sim a\}$. Because this cycle has ten nodes and ten links, we call it a ‘10-cycle’.

Exercises

The following questions are intended to help you get started in thinking about graphs, using the Net-Logo model. This section introduces, informally, some important basic vocabulary, but it does not present any ‘deep’ mathematical content. There are many topics in graph theory, and graphs have a wide variety of applications. To learn more, find a good graph theory textbook! (Chartrand’s[4] is non-technical, inexpensive, and focuses on practical applications. Many others are available, too.)

1. The adjacency matrix of a graph makes it easy to count how many graphs there are with a given number of nodes. For each distinct entry in the matrix, there *may* or *may not* be a link. These two possibilities for each link allow us to count the graphs with a power of 2. For example, an undirected graph without self-links, on two nodes labelled $\{1, 2\}$, has only one possible link $1 \sim 2$. So there are only $2^1 = 2$ possible such graphs. However, if we include self-links, there are three possible links: $1 \sim 1, 1 \sim 2$, and $2 \sim 2$. So, there are $2^3 = 8$ such graphs.
 - (a) List, either as adjacency matrices or as sets of links, all eight undirected graphs on nodes $\{1, 2\}$, allowing self-links.
 - (b) How many directed graphs, *with* and *without* self-links, are there on nodes $\{1, 2\}$?
 - (c) How many undirected graphs with self-links are there on three labeled nodes?
 - (d) How many directed graphs without self-links are there on four labelled nodes?

- (e) Write formulas for the number of undirected graphs on n labelled nodes, with and without self-links.
- (f) Write formulas for the number of directed graphs on n labelled nodes, with and without self-links.
2. An *isomorphism* between two graphs is a way to relabel the nodes of one graph to get the other, without changing any of the links. For example, with nodes $\{a, b, c\}$ any two of the three one-link undirected graphs $\{a \sim b\}$, $\{b \sim c\}$, and $\{a \sim c\}$ are isomorphic. To get $\{b \sim c\}$ from $\{a \sim b\}$, simply relabel a as b and b as c . Visually, two graphs are isomorphic if the picture of one can be made to look just like the picture of the other, ignoring the nodes' labels but without adding or removing any links.
- (a) Use nodes $\{0, 1, 2, 3\}$ in the undirected mode of the NetLogo model to make this 3-cycle on four nodes: $\{0 \sim 2, 0 \sim 3, 2 \sim 3\}$. Which other graphs on the same four nodes are isomorphic to it?
- (b) How many 4-cycles are there on four nodes? (Hint: more than four!)
- (c) Explain why all those 4-cycles are isomorphic.
3. A *subgraph* is a graph which is part of a 'larger' graph, meaning that it can be made from the larger graph by removing some nodes and links. The (undirected) two-node graph $\{1 \sim 2\}$ is a subgraph of the three-node graph $\{0 \sim 1, 0 \sim 2, 1 \sim 2\}$.
- (a) List all subgraphs of $\{0 \sim 1, 0 \sim 2, 1 \sim 2\}$. How many are there?
- (b) Give a relabeling to show that the two-node graph $\{6 \sim 7\}$ is isomorphic to a subgraph of the three-node graph $\{0 \sim 1, 0 \sim 2, 1 \sim 2\}$.
- (c) How many such isomorphisms are there?
4. A *path* in a graph is a sequence of nodes, each linked to the next. For example $(0, 1)$ and $(0, 2, 1)$ are two paths in the 3-cycle on $\{0, 1, 2\}$. A graph is *connected* if there is a path between any two of its nodes. Otherwise it is *disconnected*, having two or more *component* subgraphs.
- (a) How many links would you have to remove in order to disconnect that 3-cycle?
- (b) How many different ways can you divide a 4-cycle on four nodes into two components? How about three components?
5. The *complement* of a graph is another graph on the same nodes, with none of the links, but having instead all other possible links. The four-node graph $\{0 \sim 0, 0 \sim 1, 1 \sim 2, 2 \sim 3, 3 \sim 3\}$ is isomorphic to its complement.
- (a) Construct the complement and give a re-labeling for the isomorphism.
- (b) Explain why are there no three-node self-complementary graphs.

- (c) Make a self-complementary two-node *directed* graph.
 - (d) Why can't there be a self-complementary two-node undirected graph?
6. A connected undirected graph without any cyclic subgraphs is called a *tree*.
- (a) Make a tree graph with all ten nodes in the model. Move the nodes around in the picture so that it looks like a tree.
 - (b) Make a connected directed graph without cycles (no backwards paths allowed!), but having two different paths between two nodes. Explain the similarities and differences between this *directed acyclic graph* and a tree graph.
7. The number of links on the shortest path between two nodes in a graph is their *distance*. The number of links in the longest path in a graph is the graph's *diameter*. The number of links in the largest cycle in a graph is called the *girth*.
- (a) Which undirected graphs on ten nodes have the maximum girth, and which have the minimum?
 - (b) Which have the largest or smallest diameter?

3 Small World Networks

Quick Overview

What do airports, neurons in a brain, collaborating film actors or scientists, social network website users, and yeast genes have in common? They all inhabit *small worlds* in the sense that, although any two of these elements in the same systems are unlikely to be connected to each other directly, there is likely to be a fairly short path between them.[9] Small-world networks (or, as mathematicians say, small-world graphs) can take many different forms. The small-world NetLogo model allows students to visualize, explore and manipulate a simple mathematical model — the Watts–Strogatz ‘beta’ model[8] — which produces a family of small-world graphs.

Learning Targets

After completing this lesson, students should be able to explain what it means for a graph to be a Watts–Strogatz small world. They should also be able to explain (informally) why a social network may be expected to be a small world. In addition to completing the exercises, students can be expected to be able to define all other italicized terms in this section, and to identify or calculate them in the context of a simulated network in the NetLogo model. More advanced students will be prepared for the Barabási – Albert *scale-free* network model, which builds on several of the concepts introduced here.

Essential Concepts

First, students should be familiar with the most basic terminology of graph theory, as introduced by the lesson (in Section 2) accompanying the small-graph NetLogo model, as well as some simple ideas from the study of probability. These are summarized below.

A mathematical graph is a collection of *nodes* (or *vertices*), together with a collection of *links* (or *edges*). A *link* is just a reference to a specific pair of nodes, representing a direct connection between them. A *node* can represent anything that may be thought of as connected to other things of the same kind: a city, a neuron, a person, a gene, and so on. Graphs can represent the potentially complex relationships between any of these. But since we want to be able to talk about and make pictures of all of these graphs without being distracted by other issues besides connectedness, we may as well talk about, and draw, dots connected by lines. Here, as is often the case, we’re only concerned with *simple* graphs: those without self-linked nodes or more than one link between any two nodes.

Each node is linked to some number of other nodes (its *neighbors*); we call that number the node’s *degree*. In the graph shown in Figure 2, only one node has degree two. The entire graph has an

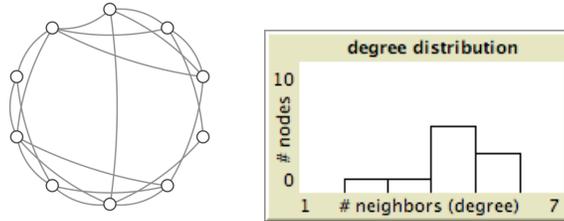


Figure 2: A small graph, and a histogram showing its degree distribution.

average degree which is of course just the sum of all the individual node degrees divided by the number of nodes in the graph. The average degree of the graph in Figure 2 is exactly four.

An average doesn't really tell us that much, though; after all, it's just one number. We might want to ask how many different degrees there are, and how many nodes have each given degree. Then we can make a kind of bar chart (a *histogram*) showing all the node degrees; this is the *degree distribution*. The degree distribution of the pictured graph is also shown in Figure 2. In addition to the one node with two links, there is one node having three links, five nodes with four links, and three with five links.

There are two more concepts which are a bit more involved; the *distance* between any two nodes, and the *clustering coefficient* of a single node. Distance is pretty intuitive: it's the number of links on the shortest path between two nodes. The only tricky part is finding that shortest path, since there may be many different paths between the two nodes. In the graph pictured in Figure 2, no two links have distance greater than two.

The *clustering coefficient* C of a node is a measure of how connected the node's neighborhood is. In other words, how many of the node's neighbors are neighbors of each other? If none are, then $C = 0$ and the neighborhood is called a *star* subgraph. If, in the other extreme, all of the node's neighbors are linked, then $C = 1$ and the neighborhood subgraph is called a *complete* graph or a *clique*. In general, C is defined for a node as the number of actual links between pairs of the node's neighbors, divided by the number of possible such links. One way we can define C for the entire graph is simply as the average C of all nodes.

Given a collection of n nodes, one way we can construct a 'classical' *random graph* of the form $G(n, m)$ having exactly m links, is simply by adding a link between m pairs of nodes, chosen with uniform probability out of all possible unlinked pairs. There's another kind of random graph, of the form $G(n, p)$ where p is a real number between 0 and 1, representing the *probability* of any possible link. When $p = 0$, $G(n, p)$ is the 'empty' graph, with n nodes but without any links. When $p = 1$, $G(n, p)$ is the complete graph on n nodes, where every node is linked with all others. But in general, $G(n, p)$ doesn't represent any single particular graph, but rather a whole collection of possible graphs. The $G(n, p)$ model is easier to work with mathematically than the $G(n, m)$ model, and it is often called the Erdős–Rényi (or just 'ER') random graph model, after the authors of an influential series of papers about it in the 1950s and 1960s.[5]

With all these concepts, we're ready to make the definition a little more precise: a Watts–Strogatz

small-world network is a graph with average distance between pairs of nodes proportional to the logarithm of the number of nodes, like a random graph, but also having a significantly higher clustering coefficient than that of a ER random graph with the same numbers of nodes and links. It's named for Duncan Watts and Steven Strogatz, who invented the model in 1998.[8]

Model User Interface

The small-world NetLogo model displays a network with all the nodes evenly spaced around a circle. There are three parameters for the Watts–Strogatz small world network mathematical model, and each is represented with a slider in the NetLogo model. The node-count slider determines the number of nodes, and the neighbor-count slider determines the number of neighbors (that is, the degree) of each node when $\beta = 0$. In this case, the neighbors are always those closest on the circle in the picture, so we have a regular *ring graph*. Set $\beta = 0$ and try a few different values of neighbor-count to get the idea.

When $\beta > 0$, some of the links may be ‘rewired’. That is, every link has (independent) probability β of having one of its nodes replaced by a randomly chosen other node. When $\beta = 1$, all the links are rewired, and we have a classical random graph of the $G(n, m)$ sort, where $n = \text{node-count}$ and $m = \text{node-count} \cdot \text{neighbor-count} / 2$.

The button marked *setup* makes a new network with the parameter values shown on the sliders. The other three buttons display extra information about the network. The *show-neighborhood* highlights the neighborhood of a selected node, and the *show-link-distances* buttons displays the graph distance of all other nodes to a selected node. To use either of these, you must first click the button and then click a node in the picture to select it. Clicking on the background of the picture instead of a node will reset the display.

The *show-clustering-coefficients* button displays the clustering coefficients of all the nodes. The link distances and clustering coefficients of the nodes can be shown either as decimal fraction labels or as shades of grey, depending on how the *use-grey?* switch is set. For link distances, darker grey means nearer (*i.e.*, lower numbers). For clustering coefficients, darker means more clustered, *i.e.*, a more interconnected neighborhood (and thus higher numbers). The entire graph's average clustering coefficient, once calculated, is shown in a little box under the *show-clustering-coefficients* button.

In addition to the degree distribution histogram mentioned above in the Concepts section, which is drawn for every new graph displayed, there is another histogram, labeled “link distance distribution”, which shows the link distances between all nodes and the selected target node. This chart may be different for each target node picked. It should help a student understand how much variation there is in the the distribution of distances for the graph displayed. The exercises give some suggestions for how to interpret and extend these measurements.

Exercises

1. This short sequence of questions will help you use the small-world Netlogo model to discover a basic but important property of WS small-world networks.
 - (a) With $\text{node-count} = 10$, $\text{neighbor-count} = 2$, and $\text{beta} = 0.10$, click `setup` and look at the graph displayed. Count the number of links. How is it related to the number of nodes? Try it again with a new graph with the same parameters. Does the same relationship hold?
 - (b) Now try changing beta . Does this affect the relationship? Why or why not? (Hint: how does rewiring work?)
 - (c) Try a couple of graphs with $\text{neighbor-count} = 4$. Can you predict the total number of links?
 - (d) Write the number of links for any network as an equation, with two of the slider parameters as independent variables. Test your predictions with larger networks and more neighbors! You can have NetLogo do the counting for you by typing `count links` into the observer line of the Command Center.
2. These questions will help you understand the clustering coefficient. First, make a graph with $\text{node-count} = 20$, $\text{neighbor-count} = 4$, and $\text{beta} = 0.20$. Click `show-neighborhood`, and then click on a node.
 - (a) How many nodes are in this neighborhood? How many links?
 - (b) Explain why the greatest possible number of links between n nodes in a simple graph is $n(n - 1)/2$.
 - (c) Calculate the clustering coefficient for the node you selected, using the formula $C = 2l/n(n - 1)$, where l is the number of links in the node's neighborhood. (The Command Center makes a pretty good calculator, if you need one.) Do this calculation for three different nodes, and then check your answers by clicking `show-clustering-coefficients`.
3. These questions are about some of the applications of the model.
 - (a) Explain, in a brief sentence or two, what each of the following terms mean in the context of a social network: *node*, *link*, *neighborhood*, *degree*, *link distance*, *clustering coefficient*.
 - (b) In your own words, what would it mean for a group of people to have a friendship graph with a *star* shape? A *clique* shape?
 - (c) Imagine a graph representing the city or town you live in, where the nodes are street intersections with stop signs or stop lights. In other words, the graph's link distance between two places is the number of times a car would have to stop when driving the

- shortest route between them. Is this graph a small world? Why or why not? (Hint: does your town have any highways? What's the fastest way to drive across town?)
4. Which graphs produced by this model have the highest global clustering coefficients? Which have the lowest?
 5. For a particular WS graph, which node would you expect to have the lowest average link distance to all other nodes, and why?
 6. These questions are about the distances between two nodes in a BA network.
 - (a) How would you calculate the average distance from a selected node to every other node, given the information from the show-link-distances button?
 - (b) How would you use this number, assuming you could calculate it for each node, to find a 'characteristic' link distance for the entire graph?
 - (c) Try calculating this characteristic path length for a small graph with node-count = 10, neighbor-count = 2. Then use the global-average-distance command, from the Command Center, to automatically find the characteristic path length for the same small graph. Do the numbers match?
 7. This (more advanced) exercise guides you through reproducing the 1998 results of Watts and Strogatz.
 - (a) Find the global-average-distance of the graph with node-count = 200, neighbor-count = 10, beta = 0. Call this number $L(0)$, and call the global clustering coefficient for the same graph $C(0)$. Now use your favorite spreadsheet software to record the global-average-distance and global clustering coefficient for each of three different graphs with the same parameters, except for beta = 0.01. Call the average characteristic path length of these three trials $L(0.01)$, and the average global clustering coefficient $C(0.01)$. Now find L and C for beta (β) values of 0.1 and 1. Make a chart plotting $C(\beta)/C(0)$ and $L(\beta)/L(0)$ against β , for all the values of β you found. (You'll want to use a logarithmic scale for the β axis.)
 - (b) Use your results to suggest which values of β will produce graphs having both small characteristic path lengths and large global clustering coefficients. Test your hypothesis for graphs with other node-count and neighbor-count values, and compare your results with those published by Watts and Strogatz!
 8. There is another way to define a clustering coefficient for a graph: as the number of triangles divided by the number of connected triples in the graph, where a connected triple is any three nodes a, b, c such that $a \sim b$ and $b \sim c$ (\sim meaning "is directly linked to"). If also $a \sim c$, then the connected triple is a 'closed triple' or a 'triangle'. Write a NetLogo procedure to calculate this other kind of global clustering coefficient, and compare these numbers with the average local clustering coefficients.

4 Scale-free Networks

Quick Overview

The Barabási–Albert (BA) *scale-free network* model is a method for generating a family of random mathematical graphs which, no matter their size, always include some nodes which have neighborhoods much larger than average.[3] Because of this property, these nodes in these networks have no ‘characteristic’ (*i.e.*, ‘typical’, ‘natural’, or ‘expected’) degree. The degree distribution of such a graph does not have the familiar bell-curve shape, but rather approximates a *power law* of the form $P(k) = k^{-a}$ where P is the probability of a randomly selected node having degree k (that is, k neighbors), and a is a positive constant. (Figure 3 shows an example.) The name ‘*scale-free*’ refers to independence of the shape of this distribution from the scale at which one views it. The approximation has to do with the finite size of real networks. Scale-free networks have especially small average distances between any two nodes: when $2 < a < 3$, the greatest such distance (the *diameter* of the graph) is proportional to $\ln \ln N$, where N is the total number of nodes. In that sense, they are ‘smaller worlds’ than Watts–Strogatz small-world networks or Erdős–Rényi random graphs.[2] The BA model produces graphs which show more clustering than ER random graphs, but less than those of the Watts-Strogatz model.

In 1999 at the University of Notre Dame, Albert-László Barabási and Réka Albert found that the connectivity of at least some part of the World Wide Web is scale free, and suggested this *preferential attachment* model to explain why this might be the case.[1] They were unknowingly reproducing the work of historian of science Derek de Solla Price, who had found (in 1965)[6] that the citation graph of published scientific papers had a degree distribution which followed a power law, and (in 1976)[7] suggested essentially the same developmental explanation, which he termed “cumulative advantage”. Many social, biological, and computer network systems appear to be scale-free networks. However, the BA model is only one of many possible ways to produce these kinds of graphs.

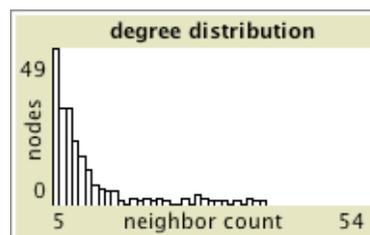


Figure 3: The degree distribution histogram of a BA scale-free network.

Essential Concepts

Students should already be familiar with the fundamental concepts of graph theory and probability which are introduced in the Watts–Strogatz small-world network lesson in Section 3. Some programming experience, and familiarity with the NetLogo programming language, will be very helpful for the more advanced exercises.

Learning Targets

Upon gaining familiarity with this model, students should be able to identify graphs that have *scale-free* (power-law) degree distributions, and explain the consequences of this property for the average distance between any two nodes. They should also be able to explain how the way a network is formed over time can cause its degree distribution to be scale free. Students can be expected to be able to explain the significance of the terms *scale free* and *preferential attachment*, and to compare graphs produced by the Barabási–Albert method with those of the Watts–Strogatz small-world model. More advanced students may be expected to find and implement other methods of generating scale-free graphs.

Model Mechanism and User Interface

There are two important parameters to the BA graph model. These are node-count, the number of nodes in the finished graph, and new-link-count, which is the number of links added with each newly-connected node. When these parameters are set, clicking setup will show you the ‘seed graph’ from which the finished graph will be grown by the BA algorithm. This seed graph consists of a *clique* (fully connected graph) of the first new-link-count+1 nodes, counted clockwise from the top of the picture. If you want to make a different seed graph, the manual-link button will allow you to connect any two nodes by clicking on them. (You may want to set new-link-count to 1 if you want a smaller seed graph than the default.) The step button connects the next unconnected node to exactly new-link-count existing nodes, chosen randomly but according to a biased or *weighted* distribution: an existing node is likely to be picked as a new neighbor in direct proportion to the number of links it already has. For example, as shown, if between three nodes 1, 2, 3 there are two links 1~2 and 2~3, and a new node 4 is going to join the graph by choosing a new neighbor, the link 2~4 is twice as likely (with probability 1/2) to result as is 1~4 or 3~4 (each with probability 1/4). If new-link-count > 1, the same neighbor can’t be chosen twice, so exactly new-link-count distinct new links result. The go button repeats this process for each of the unconnected nodes around the circle and plots the resulting degree distribution histogram.

The show-degrees and show-clustering-coefficients buttons display the degree and clustering coefficient of each node, as either a printed number or a shade of grey depending on the position of the use-grey? switch. The average clustering coefficient of the entire graph is shown in the box below the button. There are also two buttons, show-neighborhood and show-link-distances, which display

information about a particular ‘target’ node selected with a click: the distances are also plotted in a histogram.

Exercises

1. Using the default seed subgraphs, how many links are in a graph with a given node-count and new-link-count, and why? Write a formula for this number.
2. Set node-count to 100 and new-link-count to 1; click setup and go. Now type layout-radial nodes links node 0 into the Command Center. What kind of graph is this, and why? What will the clustering coefficients be? You may also want to try the layout-radial nodes links command with node 50, or some other node. This is another way to see the link distance distribution for a specific node.
3. Make a large network, and right-click on the histogram to export the chart data as a CSV file. Open that file in your favorite spreadsheet, and reproduce the chart as a scatter plot (a logarithmic scale might be helpful). Now use the spreadsheet to fit a power-law trend (or ‘regression’) line to the data. You might want to try linear, polynomial, or logarithmic regressions as well, and compare R^2 values to convince yourself that the power law is the best fit! What are the values of k and a in the $P(k) = k^{-a}$ equation of the trend line? Try this with a few networks having different parameters. Do you find similar values for k and a ?
4. Compare this model with the Watts–Strogatz small world network model. What kinds of real-world systems would each be more appropriate for?
5. Find a ‘real-world’ phenomenon that you think may involve preferential attachment or ‘cumulative advantage.’ Can you identify a network structure? How does it develop?
6. This model only ‘grows’ by one node at a time. What would happen if instead, at each step, a variable number of new nodes were added simultaneously? How would the change in this ‘growth rate’ over time affect the structure of the network? Modify the the NetLogo code and find out!
7. There are other ways of generating networks which have power-law degree distributions. For example, let the graph grow by introducing new nodes – but instead of connecting them directly to existing nodes, let the new node create new links with the *neighbors* of a randomly chosen old node. Implement this model by modifying the NetLogo code. What differences do you see between these graphs and the BA networks?

Advanced Topics

- Adapt a dynamical system, such as a Kauffman random boolean network or an iterated n -player Prisoners Dilemma game, to use an BA network for interconnection. How does this

affect the dynamics or player strategies?

- Collect data on your own social network and measure the degree distribution. Can you use the BA network growth model to produce a similar distribution?

5 Contagion and Damage Spreading

- to be done -

6 Community Structure

- to be done -

7 Food Webs

- Juliana's writeup goes here? -

8 Genetic Regulatory Networks

- ??? -

9 Classroom Network

- Deadletter's writeup goes here? (Or elsewhere?) -

References

- [1] Barabási, A-L; Albert, R. (October 15, 1999). “Emergence of scaling in random networks”. *Science* 286 (5439): 509–512. arXiv:cond-mat/9910332.
- [2] Cohen, R; Havlin, S. (2003). “Scale-Free Networks are Ultrasmall”. *Physical Review Letters* 90 (5): 058701. arXiv:cond-mat/0205476.
- [3] Caldarelli, G. (2007) *Scale-Free Networks: Complex Webs in Nature and Technology*. Oxford University Press. ISBN: 0-19-921151-5.
- [4] Chartrand, G. (1977) *Introductory Graph Theory*. Dover. ISBN 978-0-486-24775-5.
- [5] Newman, M. E. J. (2010) *Networks: An Introduction*. Oxford University Press. ISBN 978-0-19-920655-0.
- [6] Price, D. (1965). “Networks of Scientific Papers”. *Science* 149 (3683):510-515.
- [7] Price, D. (1976). ”A general theory of bibliometric and other cumulative advantage processes”. *Journal of the American Society for Information Science* 27:292-306.
- [8] Watts, D.J.; Strogatz, S.H. (1998). ”Collective dynamics of ’small-world’ networks.” *Nature* 393 (6684): 409–10. doi:10.1038/30918.
- [9] Watts, D. J. (1999). *Small Worlds: The Dynamics of Networks Between Order and Randomness*. Princeton University Press. ISBN 0-691-00541-9.
- [10] Wilensky, U. (1999). *NetLogo*. <http://ccl.northwestern.edu/netlogo/> . Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.