Sampling algorithms of pure network topologies: Stability and separability of metric embeddings

Edoardo M. Airoldi

May 2005
CMU-ISRI-05-111

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

In a time of information glut, observations about complex systems and phenomena of interest are available in several applications areas, such as biology and text. As a consequence, scientists have started searching for patterns that involve interactions among the objects of analysis, to the effect that research on models and algorithms for network analysis has become a central theme for KDD. The intuitions behind the plethora of approaches rely upon few basic types of networks, identified by specific local and global topological properties, which we term "pure" topology types. In this paper, (1) we survey pure topology types along with existing sampling algorithms that generate them, (2) we introduce novel algorithms that enhance the diversity of samples, and address the case of cellular topologies, (3) we perform statistical studies of the stability of the properties of pure types to alternative generative algorithms, and a joint study of the separability of pure types, in terms of their embedding in a space of metrics for network analysis, widely adopted in the social and physical sciences. We find that the sampling algorithms entail low stability of topological properties entailed by alternative algorithms, and lead to weakly separability topology types. We spell out the implications for the practitioners. We conclude that real world networks hardly present the variability profile of a single pure type, and suggest the assumption of "mixtures of types" as a better starting point for developing models and algorithms for network analysis.

Keywords: Network topology, Pure topology types, Metrics for network analysis, Sampling algorithms, Generating processes, Stability, Separability

Contents

1 Introduction

In recent years, researchers in application areas such as bioinformatics, computational biology, and those that rotate around the processing of electronic texts have made available huge amount of "networked data," to the data mining community at large, to the effect that models and algorithms for network analysis have become a central theme for KDD [29, 15, 18, 24, 28] On the other hand, in the social and mathematical sciences, (social and complex) networks have been an object of research for a few decades now [19, 37, 10, 21, 6, 7, 12, 11]. Over the years, the communication across communities has increased, the major results of each discipline have been shared and assimilated by the others, and, occasionally, old ideas have resurfaced under a different disguise. In particular, the notion of "network topology" has recently gained attractiveness, as several complex phenomena of scientific interest tend to manifest in those networks that are characterized by specific "topological properties" [19, 47, 5, 20, 9]. Thus, it is not surprising to find that a fundamental characteristic shared by recent approaches to network analysis is the central role played by a set of basic types of networks, identified by specific local and global topological properties of interest, which we term "pure" topology types.

In data mining and machine learning, the study of real world networks is essential for the development of sound theoretical models, which ultimately lead to unbiased inferences and robust predictions [35, 40, 44, 25, 26, 30, 36, 2, 45, 31, 3]. Analyses of real networks rely upon two crucial tasks: (1) that of "generating," or "sampling," networks that display realistic properties of interest, and (2) that of "determining" which pure topology type(s) a given network is close to. For example, models that generate networks with realistic properties given few parameters can be used for compression, simulations and testing, models of pure types can be used to compare ideal properties to those of observed networks, and so on. Given an observed network, the ability to discriminate between pure topology types can be used to predict which phenomena the system under scrutiny is expected to display, e.g., in a dynamic setting. Last, in order to apply the large body of type-specific results present in the literature to real world problems, it is crucial to map an observed network to the corresponding pure type(s). In this paper,

1. we survey the pure topology types, along with the existing sampling algorithms for generating each of them;
2. we introduce novel algorithms aimed at enhancing the diversity of sampled networks, and at addressing the case of cellular topology type;
3. we perform statistical studies of the stability of the properties of pure topology types to alternative generative algorithms, and a we perform joint study of the separability of pure topology types, in terms of their embedding in a space of metrics for network analysis, widely adopted in the social and physical sciences.

2 Problems

The utility and appeal of sampling algorithms stems from the following implication. If we can generate a network at random that displays the properties of interest, it is "possible" that the imaginary generative process we posited actually outlines a latent phenomenon that is truly happening in the

data. This implication can be very convincing, depending on the soundness of the semantics that inform the imaginary process, in a specific application, to the effect that the latent phenomenon is perceived as "plausible." For example, the "six degrees of separation" among individuals observed by Milgram (1967) is captured by the "small world" topology of Watts and Strogatz (1998) where the semantic that informs the sampling algorithm is that "individuals form local acquaintances, few of which relocate to places far away." This stylized model of behavior is enough to replicate the phenomenon observed by Milgram, and it "sounds" like a plausible explanation [38, 47]. In section 3.1 we address the following problem.

Problem 1 *(Sampling) How can we generate topologies that have a set of desired properties with high probability?*

Sampling algorithms can be both deterministic and probabilistic, and typically depend on a small set of parameters. To fully exploit their power, it is important to provide ways to estimate such parameters from observed quantities.

As we discussed above, a related practical problem is that of determining which properties we should expect to observe in a network under analysis. The pure topology types are used by practitioners to this extent, e.g., homeland security officers are interested in determining whether an observed social network is cellular, given partial measurements about it. If so the conclusion will be drawn that destabilization strategies that are successful on pure cellular topologies will be successful in destabilizing the given network. In section 3.2 we address the following problem.

Problem 2 *(Typing) How can we determine which pure topology type a given network is closest to?*

In order for the "homeland security argument" above to be reasonable, it is important for alternative algorithms for the same pure type to be "stable," i.e., to produce networks that close, in some reference space[1], and for algorithms for different pure types to be "separable," i.e., to produce networks that are far apart, in some reference space, see Figure 1. The stability of topological properties, to alternative sampling algorithms for the same topology type, suggests that choosing one specific algorithm over another[2] does not harm the validity of the conclusions. The separability of topological properties, entailed by sampling algorithms for different topology types, implies that any set of observed topological properties uniquely indicates a pure topology type. In other words, separability suggests that it is logically possible to answer questions like "is the given network of type X?" Most of the experiments in section 4 are devoted to assess stability and separability of the sampling algorithms surveyed or introduced in section 3.1.

2.1 Related Work

The pure topology types we consider in the next section have been introduced separately over the years [19, 38, 47, 5, 20, 9, 4, 39, 22]. To the best of our knowledge neither exploratory nor compre-

[1] The reference space used in this paper is defined by 47 metrics widely adopted in the social and physical sciences. We embed all sampled networks in this space.

[2] Note that there are possibly infinitely many sampling algorithms that, although different, produce networks with topological properties typical of the same pure type.

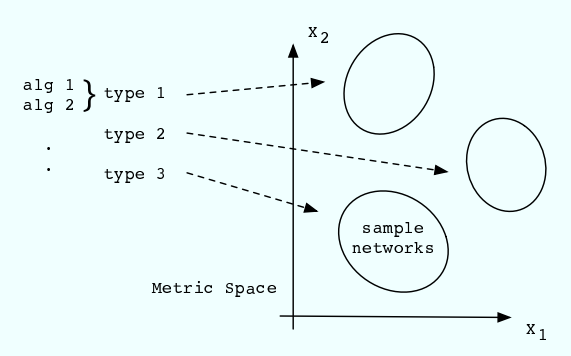

Figure 1: Sampling algorithms for pure topology types 1, 2, and 3 are mapped to the corresponding sets of all possible network samples, in the metric space $\mathcal{X}_1 \times \mathcal{X}_2$. If these sets overlap the pure types are not separable and the logic implication between properties and topologies is broken. That is, topology types still imply observed properties, but observed properties do not imply a specific topology type, rather the lack of properties implies the absence of topology types.

hensive studies exist, which attempt to compare the stability of alternative sampling algorithms, or to assess the separability of the sampled networks, in terms of the collection of metrics commonly used for network analysis.

Typing network topologies form data is a fairly novel area of research. Initial explorations are present in specific application domains such as cover network analysis [17].

Related research efforts aim at providing intuitions and mathematical theory that describe what happens to topological properties when only partial information is available, e.g., sub-samples of scale free networks are not scale free [43], at exploring the effectiveness of search strategies, e.g., greedy search finds short chains of acquaintances in small world networks [33, 34, 32, 1], at developing models of information flow [41, 42] and information exchange [16], or at exploring the robustness of metrics for network analysis to variations in the topological properties [23, 8].

3 Pure Topology Types

We begin with a heuristic description of the pure topology types. The sampling algorithms in the next section provide these intuitive notions with a precise meaning. Without loss of generality we specify the sampling algorithms for the pure topology types on a ring lattice. Note that in a ring lattice there is a natural notion of distance, D_{RL}, that is distinct from the one entailed by shortest paths, D_{SP}. In words, the D_{RL} distance between nodes A and B is proportional to the length of the arc that joins them, along the circle outlined by the ring. Figure 2 shows some examples.

Topology 1 *(Ring Lattice) Each node is connected to its neighbors, according to* D_{RL}.

Topology 2 *(Small World) Each node is connected to several of its neighbors and few distant nodes, according to* D_{RL} *[47].*

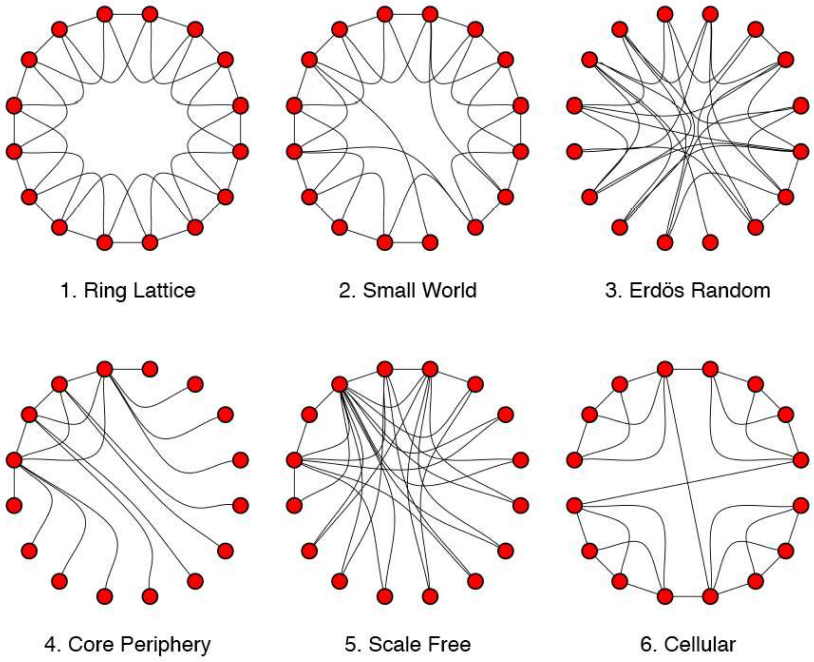

Figure 2: A glance at the relevant topologies on a ring. Note that in a ring there is a natural notion of distance that is distinct from the one entailed by shortest paths, i.e., the distance between nodes A and B is proportional to the arc-length that joins them, along the circle outlined by the ring.

Topology 3 *(Erdös Random) Each node is connected to a random set of the remaining nodes [19].*

Topology 4 *(Core-Periphery) Nodes belong exclusively to either the core or the periphery. Core and periphery nodes are connected to core nodes, while there are no edges among periphery nodes [9].*

Topology 5 *(Scale Free) Most of the nodes are connected to few other nodes, while few nodes are connected to many other nodes. This relation is formally described with a power law, between the number of edges and the number of connections [4].*

Topology 6 *(Cellular) Nodes are divided into cells. Connections are frequent between nodes within each cell, and rare between nodes in different cells [22].*

In the rest of this paper we represent a network in terms of a graph $G = (V, E)$, where V is a set of vertices and E is a set of edges, undirected and of unit weight.

3.1 Sampling Algorithms

We now survey the existing sampling algorithms for each of the pure topology types above. To complement the survey, we introduce novel algorithms aimed at enhancing the diversity of sam-

pled networks, and at addressing the case of cellular topology type. The following algorithms are available in C++ as part of ORA [13].

3.1.1 Ring Lattice

A ring lattice with parameters (n, k) is sampled as follows.

Ring Lattice 1 *Define n as the number of nodes in the graph, and k as the number of neighbors for each node. Given (n, k) do as follows.*

1. FOR: *each node v=1,...,n*

1.1. DO: *add an edge from v to its k closest neighbors.*

Note, this is a deterministic algorithm, and there is no variability in the sampled networks.

3.1.2 Small World

A streamlined definition of a small world topology is one that negatively correlates the probability of two nodes being connected to their distance, for some notion of distance.

Watts and Strogatz (1998) propose a way to generate a small world topology on a ring lattice with undirected edges. In a ring lattice with parameters (n, k) the nodes are placed in a circular fashion, and each node is connected with its $k/2$ closest neighbor clockwise and counter-clockwise by means of undirected edges. In order to generate a small-world topology with parameters (n, k, p) the following algorithm is used.

Small World 1 *Define n as the number of nodes in the graph, k as the number of neighbors for each node, and p as the probability of rewiring. Given (n, k, p) do as follows.*

1. FOR: *k=1,..., k/2*

1.1. FOR: *each node v=1,...,n*

1.1.1. DO: *with probability p, substitute the edge from v to (v+k) with an edge from v to u; where u is selected uniformly at random over the entire ring, with duplicate edges forbidden.*

This algorithm is useful for generating topologies and to study their properties for $p \in [0, 1]$; specifically, for $p = 0$ we retain the original ring lattice topology and for $p = 1$ we generate a random graph, see below. Watts and Strogatz (1998) also define intuitive measures of connectivity. The number of edges in the shortest path between two nodes, averaged over all pairs nodes, is $L(p)$. The fraction of edges that exist among neighbors of v, that is, the number of edges among the kv neighbors of v over $kv(kv-1)/2$, the maximum number of edges, averaged over all nodes, is $C(p)$. They use these two quantities to profile small world topologies by computing the observed values of L and C and comparing them to those of a random graph—see discussion point no.1. In particular, $L_{observed} < L_{random}$ and $C_{observed} > C_{random}$ would constitute evidence to support a small world topology.

Kleinberg (2001) proposes a model to generate small world topologies on a two-dimensional grid with directed edges. In the two-dimensional grid lattice with parameters (n, k) each one of the n^2 nodes is connected with k of its close neighbors, where the lattice distance between two nodes is defined as the number of lattice steps that separate them, that is, $d((i, j), (k, l)) = |k - i| + |l - j|$. In order to build a small world topology with parameters (n, k, l, r) the following algorithm is used.

Small World 2 *Define n as the number of nodes in the graph, k as the number of neighbors for each node, l as the number of long-range contacts, and r as the exponent of the power law. Given (n, k, l, r) do as follows.*

1. DO: *build a grid lattice (n,k)*

2. FOR: *each node v=1,...,n*

2.1. REPEAT: *l times*

2.1.1. DO: *add a directed edge from v to u, where u is selected with probability proportional to $d(u, v)^{-r}$, with duplicate edges forbidden.*

This algorithm is easier to deal with than the previous one, analytically, in that the parameters (k, l) determine the number of close neighbors and long-range contacts, and in that we do not break the symmetry of the problem by substituting neighbors for long-range contracts but we add the latter on top of the former instead.

We propose a different model to generate small world topologies from a ring-lattice with directed edges. In the ring lattice with parameters (n, k) each one of the n nodes is connected with each of its close neighbors with probability p_1 and to all of its long-range contacts with probability proportional to p_2. In our formulation the control is not on the number of neighbors and long-range contacts, but rather on the probability of having a neighbor and a long-range contacts as in a proper Erdös random graph. In order to build a small world topology with parameters (n, k, p_1, p_2, r) the following algorithm is used.

Small World 3 *Define n as the number of nodes in the graph, k as the number of neighbors for each node, p_1 as the probability of neighbor, p_2 as the probability of a long-range contacts, and r as the exponent of the power law. Given (n, k, p_1, p_2, r) do as follows.*

1. DO: *build a grid lattice (n,k)*

2. FOR: *each node v=1,...,n*

2.1. REPEAT: *k times*

2.1.1. DO: *remove a directed edge from v to u, where u is each of the k close neighbors in turn, with probability $1 - p_1$, with duplicate edges forbidden.*

2.1.2. DO: *with probability p_2 add a directed edge from v to u, where u is selected with probability proportional to d(u,v)-r, with duplicate edges forbidden.*

Remark 1. It is important to note that in the small-world networks the interactions directed out of each node are generated according to the same probability distribution, and independently of other nodes.

3.1.3 Erdös Random

We generate Erdös random graph (Bollobás 2001) using two algorithms. The first algorithm depends on the number of nodes (n) and the probability of a connection (p): it simply scans through the n^2 ordered pairs of nodes and connects each of them with probability p.

Random 1 *Define n as the number of nodes in the graph, and p as the probability of an edge. Given (k, p) do as follows.*

1. FOR: *each node pair $(u, v) \in [1, n]^2$*

1.1. DO: *with probability p, add an edge from u to v.*

The second algorithm depends on the number of nodes (n) and the number of edges (m): it simply samples m ordered pairs of nodes, among the n^2 possibilities, with equal probability and without repetition, and connects them.

Random 2 *Define n as the number of nodes in the graph, and m as the number of edges. Given (k, m) do as follows.*

1. DO: *order all node pairs $(u, v) \in [1, n]^2$ in a vector $\boldsymbol{e}$*

2. DO: *set equal to 1 m components of $\boldsymbol{e}$, uniformly at random with probability $\frac{1}{n^2}$ and without repetition.*

3. DO: *add an edge from u to v if $\mathbb{I}_{(u,v)}(\boldsymbol{e}) = 1$.*

3.1.4 Scale Free

For an undirected network, a scale free topology is one where the degree distribution for all edges is identical and follows a log-normal profile. For a directed network, a scale-free topology entails that the in and out degree distributions of all edges is identical and follows a log-normal profile. In order to build a scale-free topology with parameters (n, n_0, p, p_0) we can use the algorithm by Albert & Barabasi (2001):

Scale Free 1 *Define n as the number of nodes in the graph, n_0 as the number of nodes in the initial graph, p as the probability of an edge, and p as the probability of an edge between initial nodes. Given (n, n_0, p, p_0) do as follows.*

1. DO: *build a random graph (n_0, p_0)*

2. FOR: *each of the remaining nodes $v = n_0 + 1, ..., n$*

2.1. DO: *add node v*

2.2. REPEAT: *v-1 times*

2.2.1. DO: *add a directed edge from v to u with probability p, where u is selected among the nodes in the graph with probability proportional to the total degree, with duplicate edges forbidden.*

Remark 2. This is our version of the algorithm in Albert & Barabasi (2001), which leaves out exact details about how many edges each new node should have.

Alternatively we propose a different algorithm, that controls the variability of the final degree distribution directly, rather than adding nodes aiming at obtaining a degree distribution with the desired parameters in the infinite limit. Our approach is more effective for modest network sizes, and allows for control on the parameter values "exactly" rather than depending on an infinite limit that is never reached. The algorithm depends on parameters (n, m, r) and works as follows.

Scale Free 2 *Define n as the number of nodes in the graph, m as the number of edges, and r as the exponent of the power law. Given (n, m, r) do as follows.*

1. FOR: *each pair of nodes* $(u, v) \in [i, n]^2$

1.1. DO: *set* $p_{(u,v)} = d(u, v)^{-r}$

2. REPEAT: *m times*

2.1. DO: *sample (u,v) with probability* $p_{(u,v)}$

2.2. DO: *add an edge from u to v*

Remark 3. An analytic result that describes sets of parameter values for these two algorithms that lead to the same degree distribution is not presented here.

3.1.5 Core-Periphery

We give two algorithms to generate this type of networks that differ in the way periphery nodes are connected to core nodes. Both algorithms depend on the number of nodes (n), the proportion of core nodes (p) and the connectivity among core nodes (p_{in}): we scan through each node and assign it to the core with probability p or to the periphery with probability $1 - p$, and we connect the core nodes with a random topology. At this point the random attachment algorithm connects each of the periphery nodes to a core node chosen at random with equal probability—this is the algorithm implicit in Borgatti and Everett (1999)—whereas the preferential attachment algorithm connects each of the periphery nodes to a core node chosen according to a probability proportional its total degree—in the same fashion of our version of the scale-free topology.

3.1.6 Cellular

We give two algorithms to generate cellular networks [22] that differ in the way the nodes are distributed among cells. Both algorithms depend on the number of nodes (n), the number of cells (k), the average connectivity among nodes within the same cell (p_{in}) and the average connectivity among cells (p_{out}): briefly the algorithms create a random graph (k, p_{out}) that represents the interconnections among cells and then for each cell they create a random graph (n_k, p_{in}). Any connection between two cells in the random graph (k, p_{out}) is specified to the node level by choosing to random pair (i, j) of nodes, where node i and node j belong to the interconnected cells. In the first algorithm the size of the cell is uniform, that is, each node is assigned to one of k cells with probability $1/k$. In the second algorithm an extra parameter controls the distribution of the size

Type	Proposed by	Parameters
Ring Lattice		n (number of nodes), k (number of neighbors)
Random 1. (prob.)	Erdös & Renyi	n (number of nodes), p (prob. of an edge)
Random 2. (number)	Erdös & Renyi	n (number of nodes), m (number of edges)
Small World 1. (rewire)	Watts & Strogatz	n (number of nodes), k (number of neighbors), p (prob. of rewire)
Small World 2. (number)	Kleinberg	n (number of nodes), k (number of neighbors), l (number of distant contacts), r (exponent of power law)
Small World 3. (prob.)	Airoldi	n (number of nodes), k (init. number of neighbors), p (prob. of neighbor), q (prob. of distant contact), r (exponent of power law)
Scale Free 1. (pref.)	Albert & Barabasi	n (number of nodes), n_0 (init. number of nodes), p (prob. of edge), p_0 (prob. of edge between init. nodes)
Scale Free 2. (power)	Airoldi	n (number of nodes), m (number of edges), r (exponent of power law)
Cellular 1. (uniform)	Airoldi & Carley	n (number of nodes), k (number of cells), p (prob. of edge within), q (prob. of edge between)
Cellular 2. (power)	Airoldi & Carley	n (number of nodes), k (number of cells), p (prob. of edge within), q (prob. of edge between), r (exponent of power law)
Core-Periphery 1. (uniform)	Borgatti & Everett	n (number of nodes), p_0 (proportion of core nodes), p (porb. of edge)
Core-Periphery 2. (pref.)	Airoldi	n (number of nodes), p_0 (proportion of core nodes), p (porb. of edge)

Table 1: Summary of generative algorithms.

of the cells (r), nodes are assigned to cell i with probability $p_i = 1/i^r$, which entails a power-law distribution for the cell size.

In Table 1 we summarize the 13 algorithms, their inputs and their author.

3.2 Determining Topology Types

In order to determine the type of topology of a network or a sub-network there are two main approaches: generative and discriminative. According to the generative approach, given an observed network we use its adjacency matrix to estimate the parameters underlying the sampling algorithms associated with the pure types. We then compare the estimates; the pure type associated with the "best" estimates is chosen as the pure topology type for the given network. The notion of "good" estimate can be made precise in both a probabilistic and a deterministic fashion, in terms of likelihood or distance, respectively. According to the discriminative approach, given an observed network we disregard the possible ways it may be sampled and we focus on the topological properties instead, as captured by a set of metrics for network analysis, widely adopted in the social and physical sciences. In particular, we sample a large quantity of networks, with different parameter values, for each pure type. We then compute the corresponding metrics for each of them, and we train Bayesian classifiers that are good at discriminating between the types. Given an observed network we classify it into a type according to the posterior probability of types given its adjacency matrix. In this paper we follow the discriminative approach.

The generative approach is more desirable, in principle, because it allows for a clean inter-

pretation of the type assignments in terms of the parameters underlying sampling algorithms. Unfortunately, it is hard to establish a comprehensive framework for all pure types. For example, multiple algorithms exist that generate the same topology type. These algorithms involve different parameters that can be difficult to estimate from the adjacency matrix in a consistent fashion, without bias. Different algorithms differ in crucial dimensions, e.g., the a small-world topology requires the existence of two metrics, as we noted above. It is not possible to posit a simple generative model that is able to generate all topologies as a smooth function of its underlying parameters. Further obstacles exist.

The discriminative approach leads to less interpretable result, as it disregards the way a given network topology arises and focuses on its measurable properties instead. This approach is very useful in practice, though. We can sample a large quantity of networks, as we explore the full parameter space for each one of the sampling algorithms, in order to obtain a representative sample of instances of pure topology types. We then compute the metrics on the networks in the sample to obtain profiles for each topology type in terms of the metrics of interest. At this point, we can learn the mapping from metrics of interest to pure topology types using our favorite classification method. Classification errors indicate the degree to which pairs of pure topology types overlap in the reference space of metrics, see Figure 1.

4 Experiments

Here we present the experiments to assess stability and separability of pure topology types. The classification methods we used are off-the-shelf classifiers, such as naïve Bayes (based on Multinomial and Poisson distributions), logistic regression, maximum entropy, SVM (with a linear kernel), voted perceptron, decision trees and k-nearest neighbor [14].

The results below correspond to the Poisosn flavor of naïve Bayes classifier that turned out to be more accurate in predicting the topology type of a given network [3]. In order to estimate the prediction errors we used a stratified five-fold cross validation scheme. The stratification controls that in every one of the five folds the proportions of networks by type are the same as the proportions of networks by type in the overall sample. The stratification aims at balancing the bias in those experiments where the simpler topologies are under-represented [27].

4.1 Network Metrics

We focused our analysis on a set of metrics widely adopted in the social and physical sciences. The metrics we computed for each of the instances of the pure network topology types were the following.

1-4. Degree centrality (min, max, average, standardized).

5-8. Betweenness centrality (min, max, avg, std).

9-12. Closeness centrality (min, max, avg, std).

13-16. Inverse closeness centrality (min, max, avg, std).

17-20. Eigenvector centrality (min, max, avg, std).

21-24. Clustering coefficient (min, max, avg, std).

25-28. Effective network size (min, max, avg, std).

29-32. Network constraint (min, max, avg, std).

33-36. Node levels (min, max, avg, std).

37-40. Triad count (min, max, avg, std).

41. Global efficiency.

42. Local efficiency.

43. Efficiency.

44. Connectedness.

45. Hierarchy.

46. Upper boundedness.

47. Average distance.

48-51. All pairs shortest path (min, max, avg, std).

Formal definitions are available in Wasserman and Faust (1994) [46]. The metrics above are available in ORA [13].

4.2 Design of Experiments

Overall, the hypotheses we wish to test are: (1) stability, i.e., to what extent different sampling algorithms for the same pure topology type lead to consistent topological properties, as captured by the set of metrics of interest, and (2) separability, i.e., to what extent the embedding of ideal networks into the reference space of metrics of interest can uniquely determine the pure topology types.

In order to control for possible sources of variations we were not interested in, such as size of the network and density, we devised a design of experiments structured as follows. There are six topology types. For each topology we explore the parameter space using an evenly spaced grid; we sampled at least ten topologies for each parameter configuration, which resulted in more example networks the more complex generating algorithms. We attempted to control density and size of networks, across topologies, to make the discrimination as hard as possible, and ultimately get estimates of separability as low as possible. Further, we attempted to control for other relevant parameters when generating the same topology type using different algorithms, with the goal of making sampled topologies of a same type very consistent across the various generating algorithms, and ultimately get estimates of stability as high as possible.

Algorithm	Samples	Parameters
Ring Lattice	25	$n = 250$, $k = 2, 4, .., 50$
Random 1. (prob.)	17	$n = 250$, $p = 0.10, 0.15, ..0.90$
Random 2. (number)	17	$n = 250$, $m = 311, 622, .., 28012$
Small World 1. (rewire)	484	$n = 250$, $k = 2, 4, .., 50$, $p = 0.10, 0.15, .., 0.90$
Small World 2. (number)	1250	$n = 250$, $k = 2, 4, .., 50$, $l = 1, 2, .., 10$, $r = 1, 2, .., 5$
Small World 3. (prob.)	2670	$n = 250$, $k = 2, 4, .., 50$, $p = 0.20, 0.30, .., 0.80$, $q = 0.20, 0.30, .., 0.80$, $r = 1, 2, .., 5$
Scale Free 1. (pref.)	729	$n = 250$, $n_0 = 10, 15, .., 50$, $p = 0.10, 0.20, .., 0.90$, $p_0 = 0.10, 0.20, .., 0.90$
Scale Free 2. (power)	45	$n = 250$, $m = 311, 622, .., 28012$, $r = 1, 2, .., 5$
Cellular 1. (uniform)	360	$n = 250$, $k = 2, 4, .., 20$, $p = 0.25, 0.35, .., 0.75$, $q = 0.25, 0.35, .., 0.75$
Cellular 2. (power)	360	$n = 250$, $k = 2, 4, .., 20$, $p = 0.25, 0.35, .., 0.75$, $q = 0.25, 0.35, .., 0.75$, $r = 1$
Core-Periphery 1. (uniform)	54	$n = 250$, $p_0 = 0.10, 0.20, .., 0.90$, $p = 0.25, 0.35, .., 0.75$
Core-Periphery 2. (pref.)	54	$n = 250$, $p_0 = 0.10, 0.20, .., 0.90$, $p = 0.25, 0.35, .., 0.75$

Table 2: Design of experiments.

4.3 Results: Stability and Separability

We start by reporting the stability of topological properties corresponding to single pure topology types generated with different algorithms. The figures quoted are five-fold cross-validated errors in a classification task, the lower the error is, the less stable topological properties are, since a slight variation in the sampling algorithm leads to distinguishable sets of measurements.

Random Graphs. Using the set of metrics we can distinguish almost exactly which topology was generated by which algorithm. The extremal statistics (min, max) are very powerful discriminators in this case. The area under the Receiver Operating Characteristic (ROC) curve is about 1 and the classification error about is 0.00%.

Core-Periphery. Using the set of metrics we cannot discriminate which topology was generated from which algorithm. The classification error is about 50% and the area under the ROC curve is 0.501.

Cellular. Using the set of metrics we can discriminate fairly well which topology was generated from which algorithm. The area under the ROC curve is 0.928 and the classification error is 17.64%.

Scale-Free. Using the set of metrics we can discriminate almost exactly which topology was generated from which algorithm. The area under the ROC curve is about 1 and the classification error is 0.07%.

Small-World. Using the set of measures we can poorly discriminate which topology was generated from which algorithm. The area under the ROC curve is not available (this is a three-way classification problem) and the classification error is 24.78% (base error is at 33.33%). Pairwise comparisons suggest that our classifiers may be too simple, in fact, they can not distinguish the three algorithms at the same time, even as pairs of them are fairly distinguishable.

We now report the overall stability and separability of sampling algorithms for different pure topology types. Table 4 below summarizes the five-fold cross-validated errors in the corresponding classification tasks. Diagonal cells replicate the stability results discussed above. Off-diagonal cells quote separability results. The lower the error is, the more separable topological properties are,

	SW 1.	SW 2.	SW 3.
SW 1.		16.04%	21.12%
SW 2.			13.31%
SW 3.			

Table 3: Stability of small world topology types.

since the instances of different pure types entail distinguishable sets of metrics.

	RL	Rnd	SW	SF	Cel	CP
RL	*N/A*	27.00%	7.45%	0.00%	0.00%	0.00%
Rnd		0.00%	41.22%	27.94%	32.55%	25.00%
SW			24.78%	8.66%	13.12%	5.31%
SF				0.07%	26.45%	33.33%
Cel					17.64%	37.15%
CP						50.00%

Table 4: Joint study of stability and separability of pure topology types. The column labels are: RL for ring lattice, Rnd for Erdös random, SW for small world, SF for scale free, Cel for cellular and CP for core-periphery.

5 Discussion

Our experiments point out few limitations of the sampling algorithms in section 3.1, which we discuss below.

1. *Unrealistic variability profiles.*

Both the generative algorithms we surveyed and those we introduced are very simple. Algorithms may entail "no variability" for a specific metric over a fairly large range of parameter values, or by construction, e.g., all instances of an Erdös random (n, m) have the same number of edges, i.e., m. While these algorithms are of theoretical value and help us grasp insights about phenomena of interest, it is very dangerous to employ them for statistical testing purposes, e.g., to compute p-values, as it is often done in practice.

This is because rich variability profiles are crucial in determining the stability of topological properties of a pure type to alternative sampling algorithms that generate it. In other words, low variability profiles lead to high sensitivity of topological properties, as captured by the metrics of interest, and ultimately to high sensitivity of relevant statistics to the specific version of the algorithms adopted. For example, the variability profile of the clustering coefficient is extremely sensitive to the specific algorithm used to sample both random and scale free types. As a consequence the p-value, e.g., of small-world-ness, will vary.

A simple suggestion to overcome this problem is to sample topology types according to different algorithms, and then to mix the networks, somehow. This directly aims at increasing the variability profiles of the metrics of interest, and possibly leads to more robust, e.g., p-values.

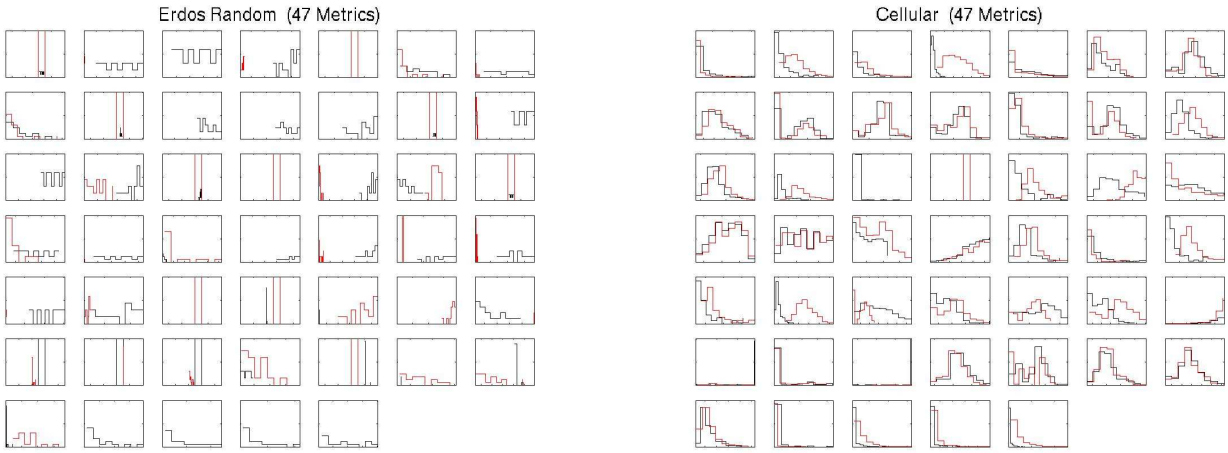

Figure 3: Profiles of 47 metrics of interest, excluding shortest path 48-51, as measured over the sampled networks. The left panel refers to Erdös random topology, whereas the right panel refers to cellular topology. Within each panel, each small plot shows two histograms in different colors, which summarize the metric values of the two different sampling algorithms for each topology type. Metrics are numbered left to right, top to bottom.

2. *Two main flavors of topological properties.*

Table 4 suggests that cellular, core-periphery and scale free types are weakly separable, and share common topological properties with random types. These types are separable from small world topologies that, in turn, shares a set of different topological properties with random types. Note that, key differences between cellular, core-periphery, scale free and random are that (a) more apparent at moderate density (apx .25 range) and (b) certain metrics can be used to separate these four types of networks.

3. *Low stability and low separability.*

Overall, alternative sampling algorithms we considered for the same type appear very similar. Yet topological properties are neither stable to alternative algorithms that are meant to generate the same topology type, nor separable across different topology types.

The low stability (not desirable) is likely to be a consequence of the fact that the algorithms are too simple and do not lead to rich enough variability profiles for the metrics of interest. In fact, we find that the extremal statistics (min and max) have high information gain with respect to the topology type categories, and drive the classification in several cases.

The low separability (not desirable) it means that pure types are stylized models of behavior at the sampling level, which lead to networks that share topological properties, as captured by the network metrics of interest. Aside from the simplicity of the algorithms, this is consistent with what we would expect to see in the real world, i.e., observed networks display multiple stylized behaviors to different degrees. This translates into the more realistic hypothesis of "mixtures of

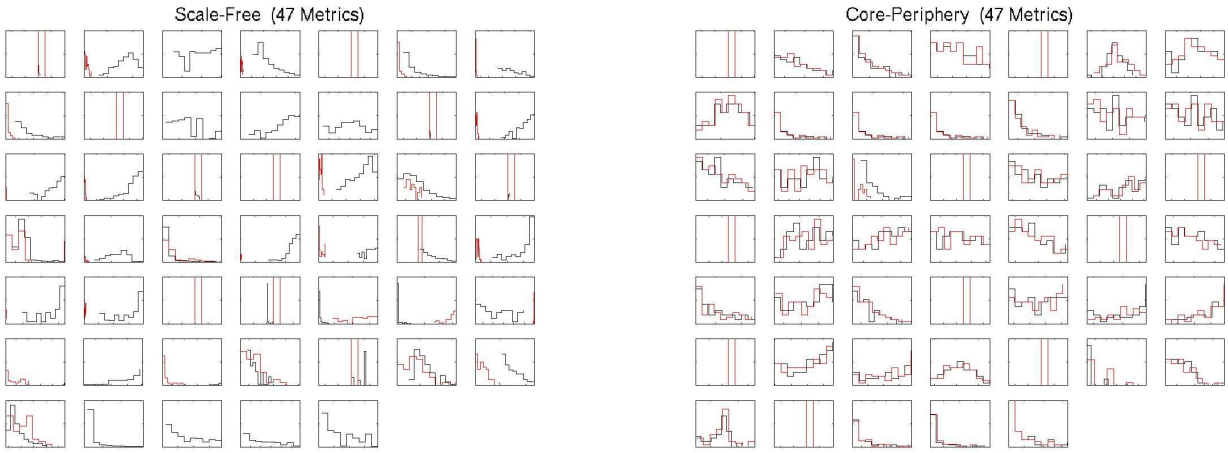

Figure 4: Profiles of 47 metrics of interest, excluding shortest path 48-51, as measured over the sampled networks. The left panel refers to scale free topology, whereas the right panel refers to core-periphery topology. Within each panel, each small plot shows two histograms in different colors, which summarize the metric values of the two different sampling algorithms for each topology type. Metrics are numbered left to right, top to bottom.

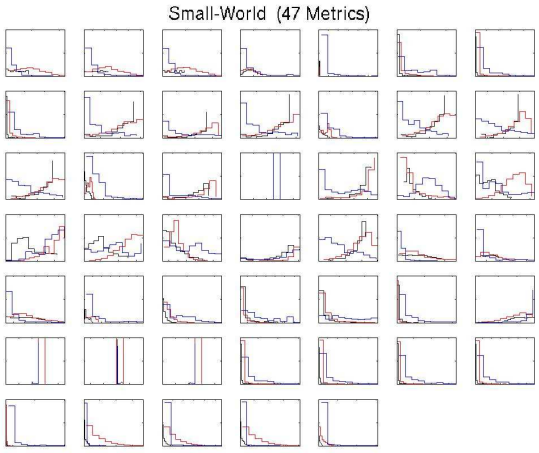

Figure 5: Profiles of 47 metrics of interest for small world topology, excluding shortest path 48-51, as measured over the sampled networks. Each small plot shows three histograms in different colors, which summarize the metric values of the three different sampling algorithms. Metrics are numbered left to right, top to bottom.

types," at the sampling level, as a better starting point for developing models and algorithms for network analysis.

6 Concluding Remarks

We surveyed pure topology types along with existing sampling algorithms that generate them. We introduced novel algorithms that enhance the diversity of samples, and address the case of cellular topologies. We performed statistical studies of the stability and separability of the topological properties of pure types, as captured by a set of network metrics of interest, widely adopted in the social and physical sciences.

We find that the sampling algorithms considered are neither stable to alternative specifications, nor separable in terms of the topological properties they entail. The lack of stability is a cause of concern, and we suggest the practitioners that employ the simple sampling algorithms discussed in this paper to consider more variable schemes, for example, mixtures, in order to obtain more robust p-values and statistics in general. The lack of separability was somewhat anticipated, as real world networks hardly present the variability profile of a single pure type. We conclude by suggesting the assumption of "mixtures of types" as a better starting point for developing models and algorithms for network analysis.

7 Acknowledgements

The author's thinking about issues discussed in this paper has benefited greatly from discussions and collaborations with Dr. Stephen E. Fienberg, Dr. Kathleen M. Carley, Dr. Jon M. Kleinberg, Dr. David M. Blei, Dr. Eric P. Xing, Bradley Malin, and Jeff Reminga.

This work was partially supported by National Institutes of Health (NIH) under Grant 1 R01 AG023141-01, by the Office of Naval Research (ONR) under Dynamic Network Analysis (N00014-02-1-0973), the National Science Foundation (NSF) and the Department of Defense (DOD) under MKIDS (IIS0218466).

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the NIH, the ONR, the NSF, the DOD, or the U.S. government.

References

[1] L. A. Adamic and E. Adar. How to search a social network. *Social Networks*, 27(3):187–203, 2005.

[2] E. M. Airoldi, D. M. Blei, E. P. Xing, and S. E. Fienberg. A latent mixed-membership model for relational data. In *Workshop on Link Discovery: Issues, Approaches and Applications, in conjunction with the 10th International ACM SIGKDD Conference*, 2005.

[3] E. M. Airoldi, W. W. Cohen, and S. E. Fienberg. Bayesian models for frequent terms in text. In *Proceedings of the Classification Society of North America and INFORMS Annual Meetings*, 2005.

[4] R. Albert and A. L. Barabasi. Statistical mechanics of complex networks. *Rebiews of Modern Physics*, 74(47), 2002.

[5] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.

[6] B. Bollobás. *Random Graphs*. Academic Press, London, 1985.

[7] B. Bollobás. *Random Graphs*. Academic Press, New York, 2nd edition, 2001.

[8] S. P. Borgatti, K. M. Carley, and D. Krackhardt. Robustness of centrality measures under conditions of imperfect data. *Social Networks*, 2005. Forthcoming.

[9] S. P. Borgatti and M. G. Everett. Models of core / periphery structures. *Social Networks*, 21:375–395, 1999.

[10] R. L. Breiger, S. A. Boorman, and P. Arabie. An algorithm for clustering relational data with applications to social network analysis and comparison to multidimensional scaling. *Journal of Mathematical Psychology*, 12:328–383, 1975.

[11] K. M. Carley. Smart agents and organizations of the future. In L. Lievrouw and S. Livingstone, editors, *The Handbook of New Media*, pages 206–220, 2002.

[12] K. M. Carley and A. Newell. The nature of the social agent. *Journal of Mathematical Sociology*, 19(4):221–262, 1994.

[13] K. M. Carley and J. Reminga. ORA: Organizational Risk Analyzer, 2004. Available for download at http://www.casos.cs.cmu.edu/projects/ora/.

[14] W. W. Cohen. Minorthird: methods for identifying names and ontological relations in text using heuristics for inducing regularities from data, 2004. http://minorthird.sourceforge.net.

[15] D. J. Cook and L. B. Holder. Graph-based data mining. *IEEE Intelligent Systems*, 15(2):32–41, 2000.

[16] P. S. Dodds, D. J. Watts, and C. F. Sabel. Information exchange and the robustness of organizational networks. *Proceedings of the National Academy of Sciences*, 100(21):12516–12521, 2003.

[17] M. Dombroski, P. Fishbeck, and K. Carley. Estimating the shape of covert networks. In *Proceedings of the 8th International Command and Control Research and Technology Symposium*, 2003.

[18] P. Domingos. Prospects and challenges for multirelational data mining. *SIGKDD Explorations*, 5(1):80–83, 2003.

[19] P. Erdös and A. Rényi. On the evolution of random graphs. *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, 5:17–61, 1960.

[20] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *SIGCOMM*, pages 251–262, 1999.

[21] S. E. Fienberg, M. M. Meyer, and S. Wasserman. Statistical analysis of multiple sociometric relations. *Journal of the American Statistical Association*, 80:51–67, 1985.

[22] T. Frantz and K. M. Carley. A formal characterization of cellular networks. Technical Report CMU-ISRI-05-109, School of Computer Science, Canregie Mellon University, 2005.

[23] T. Frantz and K. M. Carley. Relating network topology to the robustness of centrality measures. Technical Report CMU-ISRI-05-117, School of Computer Science, Canregie Mellon University, 2005.

[24] L. Getoor. Link mining: A new data mining challenge. *SIGKDD Explorations*, 5(1):84–89, 2003.

[25] L. Getoor, N. Friedman, D. Koller, and A. Pfeffer. Learning probabilistic relational models. In S. Dzeroski and N. Lavrac, editors, *Relational Data Mining*, pages 307–335, 2001.

[26] L. Getoor, N. Friedman, D. Koller, and B. Taskar. Learning probabilistic models with link uncertainty. *Journal of Machine Learning Research*, 2002.

[27] T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer-Verlag, 2001.

[28] L. B. Holder and D. J. Cook. Graph-based relational learning: Current and future directions. *SIGKDD Explorations*, 5(1):90–93, 2003.

[29] D. Jensen. Statistical challenges to inductive inference in linked data. In *Proceedings of the 17th International Workshop on Artificial Intelligence and Statistics*, 1999.

[30] D. Jensen, J. Neville, and B. Gallagher. Why collective inference improves relational classification. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2004.

[31] C. Kemp, T. L. Griffiths, and J. B. Tenenbaum. Discovering latent classes in relational data. Technical Report AI Memo 2004-019, MIT, 2004.

[32] D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.

[33] J. Kleinberg. Navigation in a small world. *Nature*, 845, 2000.

[34] J. Kleinberg. Small-world phenomena and the dynamics of information. In *Advances in Neural Information Processing Systems 14*, 2001.

[35] J. Kleinberg, S. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. The web as a graph: Measurements, models and methods. In *Proceedings of the International Conference on Combinatorics and Computing*, pages 1–17, 1999.

[36] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2005.

[37] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *Journal of Mathematical Sociology*, 1:49–80, 1971.

[38] S. Milgram. The small world phenomenon. *Psychology Today*, 1(61), 1967.

[39] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[40] A. Ng, A. Zheng, and M. I. Jordan. Link analysis, eigenvectors, and stability. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2001.

[41] C. Papadimitriou. Computational aspects of organization theory. In *Lecture Notes in Computer Science*. Springer-Verlag, 1997.

[42] C. Papadimitriu and E. Servan-Schreiber. The origins of the deadline: Optimizing communication in organizations. In *Complexity in Economics*, 1999.

[43] M. P. H. Stumpf, C. Wiuf, and R. M. May. Subnets of scale-free networks are not scale-free: Sampling properties of networks. *Proceedings of the National Academy of Sciences*, 102(12):4221–4224, 2005.

[44] B. Taskar, E. Segal, and D. Koller. Probabilistic classification and clustering in relational data. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2001.

[45] X. Wang, N. Mohanty, and A. McCallum. Group and topic discovery from relations and text. In *Workshop on Link Discovery: Issues, Approaches and Applications, in conjunction with the 10th International ACM SIGKDD Conference*, 2005.

[46] S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.

[47] D. J. Watts and S. H. Strogatz. Collective dynamics of "small-world" networks. *Nature*, 393:440–442, 1998.