A CONTROLLED SENSING APPROACH TO GRAPH CLASSIFICATION

Jonathan G. Ligo[†] *George K. Atia*[⋆] *Venugopal V. Veeravalli*[†]

[†] ECE Dept University of Illinois at Urbana-Champaign Urbana, IL 61801
[⋆] EECS Dept University of Central Florida Orlando, FL 32816

ABSTRACT

The problem of classifying graphs with respect to connectivity via partial observations of nodes is posed as a composite hypothesis testing problem with controlled sensing. An observation at a node is a subset of edges incident to the node on the complete graph drawn according to a probability model, which are modeled as conditionally independent given their neighborhoods. Connectivity is measured through average node degree and is classified with respect to a threshold. A simple approximation of the controlled sensing test is derived and simulated on Erdös-Rènyi Model A graphs to characterize error probabilities as a function of expected stopping times. It is shown that the proposed test achieves favorable tradeoffs between the classification error and the number of measurements and further outperforms existing approaches, especially at low target error rates. Furthermore, the proposed test achieves asymptotically optimal error performance, as the error rate goes to zero.

Index Terms— Graph Classification, Controlled Sensing, Complex Networks, Social Networks, Estimation Theory

1 Introduction

Graphs are commonly used to model active or feasible user connections in communication networks, interactions between entities in social networks and propagation paths of diseases between people by representing connections as edges between vertices (also referred to as nodes). Graph connectivity can be measured through a variety of metrics, such as average node degree, clustering coefficients or spectral graph properties that emphasize different notions of connectivity [1].

In many applications, determining if the connectivity of a graph model is high or low gives some insight into the operating point of the system. An example where detecting if a graph is of high connectivity is of interest is the monitoring of a disease spreading. In [2], a disease spreading in a population is modeled as a graph, with connectivity measured through shortest paths between people. In this case, connectivity serves as a measure of how quickly a population can be infected. Classifying this graph to be highly connected can serve as a warning of potential pandemics. On the other hand, in many communication network applications, it is desirable to detect low connectivity. In particular, if a graph that models interference in a wireless network is classified as low connectivity, users can reliably communicate at high rates.

In order to classify graphs into high and low connectivity, it is necessary to have a metric of connectivity as well as a method of observing a graph. Since many graphs arising in practice are large, it is necessary to perform classification by sampling some representative subset of the graph. Sampling is often performed using random walks[3] or randomly sampling nodes in the graph to estimate the graph[1].

In this paper, we pose the problem of classifying a graph by connectivity via sampling nodes as a composite sequential hypothesis test with controlled sensing [4, 5]. In contrast to prior work, mainly from the social network literature [6, 3, 1], which is validated on experimental data sets such as the DBLP authorship graph in [6], the proposed framework allows for classification of graphs with a provably low number of samples when the classification error is desired to be low. While [1, 3] considered a fixed sampling budget procedure, the procedure developed herein sequentially determines the number of samples needed to classify the graph. For a typical sequential hypothesis test, this results in a lower average number of samples used than a fixed sample budget test with the same error probability. The control proposed in this paper finds a favorable trade off between *exploring* the graph and *exploiting* knowledge of the graph in order to improve the quality of graph observations for classification. In contrast to prior work where each edge is assumed to be fully observable [3, 1], we consider more general graph observation models, where both real edges and edges not in the graph ("spurious edges") are probabilistically observed. This handles imperfect observations of graphs and reduces to the fully observable model as a limiting case.

In this paper, we first describe a controlled sensing test to classify graphs by average node degree where edges are probabilistically observable. Then, we compare the controlled sensing test to a random walk based technique, Frontier Sampling (FS) [3], on a graph with probabilistic edge observation. When no spurious edges in the graph are observable, it is shown that the controlled sensing test outperforms FS with respect to error probabilities for a given number of samples in the low and medium edge observation probability regime. The controlled sensing test is also demonstrated on an observation model that allows for spurious edges with respect to different levels of spurious and true edge observation.

2 Hypothesis Testing for Graph Classification

For notation, we use regular font to denote scalars and bold-face to denote vectors. A subscript on a letter which is used for a vector not in bold-face indicates a particular component of the vector.

Interactions between nodes are described through the edges of a fixed underlying graph $G = (V, E)$ with sets $V = V_G$ and $E = E_G$ denoting the vertex (node) and edge sets, respectively. An edge connects two distinct nodes, and there is at most 1 edge between any pair of nodes. The edge with endpoint vertices i and j is denoted e_{ij}, and i and j are said to be adjacent if there is an edge between them. The complement of a graph G,G^C, is the graph (V, E^C) where E^C is the complement of E relative to the set of all possible edges connecting pairs in V. The nodes are labeled $V = \{1, \ldots, N\}$, and $N_G(i)$ denotes the neighborhood of vertex i, which is the set of vertices adjacent to i. The degree of vertex i is $|N_G(i)|$.

This work was supported by the U.S. Defense Threat Reduction Agency through subcontract 147755 at the University of Illinois from prime award HDTRA1-10-1-0086.

We let $\bar{d}_G$ denote the average node degree of graph G. If $\boldsymbol{d}_G$ is the $N \times 1$ vector of node degrees of G, then

$$\bar{d}_G = \frac{1}{N}\boldsymbol{d}_G^\top \cdot \mathbf{1}_N$$

where $\mathbf{1}_N$ is the vector of all ones of length N and $^\top$ denotes transposition. Define the classes of graphs $\mathcal{G}_0$ and $\mathcal{G}_1$ as

$$\begin{aligned}\mathcal{G}_0 &= \{G: \ |V| = N, \bar{d}_G \leq \eta\} \\ \mathcal{G}_1 &= \{G: \ |V| = N, \bar{d}_G > \eta\}\end{aligned} \tag{1}$$

Consider the composite binary hypothesis testing problem,

$$H_0 : G \in \mathcal{G}_0, \ \ H_1 : G \in \mathcal{G}_1$$

where at each time k, a node is selected for observation. Hence, the control U_k at time k is such that $U_k \in \mathcal{U} = V \bigcup \{S\}$, where S denotes a stopping action. If $e_{ij} \in E$, nodes i and j interact with probability p at each time step. Hence, if node i is selected ($U_k = i$), the observation $Y_k \subseteq N_G(i)$ of nodes connected to i is drawn according to the probability mass function (pmf) $P_G^i(y)$, where

$$P_G^i(y) = p^{|y|}(1-p)^{d_i - |y|} \tag{2}$$

This model is referred to as "Observation Model 1" (OM1). A similar model, "Observation Model 2" (OM2), allows for spurious edges to be observed. In this case, the observations are of the form $A \cup B$, where A is a subset of $N_G(i)$ for which each edge is observed with probability p and B is a subset of $N_{G^C}(i)$ where each edge is observed with probability $q < p$. Thus, the observations follow the pmf $P_G^i(y)$, where

$$P_G^i(y) = q^{|y \cap E^C|}(1-q)^{((N-1)-d_i)-|y^C \cap E^C|} p^{|y \cap E|}(1-p)^{d_i - |y^C \cap E|} \tag{3}$$

This reduces to OM1 as $q \searrow 0$ where $0^0 = 1$.

Let u^n and y^n denote the list of controls and observations from time 1 to n respectively. If the test stops at time n, i.e., if $U_n = S$, we make a decision $\delta(y^n, u^n) \in \{0, 1\}$ about the hypothesis. Hence, the sequential test $\gamma = \{\phi_k, \tilde{N}, \delta\}$ consists of a control policy

$$\phi_k : \mathcal{U}^{k-1} \times \mathcal{Y}^{k-1} \to \mathcal{U}, \ k = 0, 1, \ldots \tilde{N} - 1$$

a stopping rule with stopping time $\tilde{N}$, and a decision rule

$$\delta_{\tilde{N}} : \mathcal{U}^{\tilde{N}-1} \times \mathcal{Y}^{\tilde{N}-1} \to \{0, 1\}.$$

The test is designed to minimize the expected stopping time under constraints on the error probabilities, i.e.,

$$\min_{\gamma} \mathbb{E}_i[\tilde{N}], \ \ i \in \{0, 1\} \tag{4}$$

3 Sequential Test

The problem is related to the controlled sensing framework developed in [4, 7] and Chernoff's procedure in [8] to sequentially test between two composite hypotheses. Specifically, the problem is readily posed as a controlled hypothesis testing problem where the control can shape the quality of the observation at each time step. Let $\mathcal{G}_i$ be the hypothesis which contains $\widehat{G}$, the estimate of the graph G, and $\mathcal{G}_j$ the alternative hypothesis which does not contain $\widehat{G}$. We propose the following sequential test:

1. At each time k, find the Maximum-Likelihood Estimate (MLE) of G, $\widehat{G} = \widehat{G}(y^k, u^k)$.
2. Find $\widehat{i}(k)$, the estimate of the hypothesis at time k, which is 1 if $\bar{d}_{\widehat{G}} > \eta$ and 0 otherwise.
3. The controller stops at time k and declares $\widehat{i}(k)$ if

$$\min_{\widetilde{G}: \widetilde{G} \in \mathcal{G}_j} \log \frac{P_{\widehat{G}}(y^k, u^k)}{P_{\widetilde{G}}(y^k, u^k)} > \log \beta, \tag{5}$$

where $P_G(y^k, u^k)$ is the joint distribution of the observations and the controls induced by the observation model $P_G^u(y)$ and the causal control distributions $q(u_k | u^{k-1}, y^{k-1})$ specified by the control policy. The graph $\widetilde{G}$ is the nearest graph under the alternative hypothesis. Thus, the left hand side (LHS) of (5) is simply the likelihood ratio of the joint distributions given the current graph estimate and the nearest graph in the alternative hypothesis. If OM1 is used, it is sufficient to stop if $\widehat{i}(k) = 1$. The parameter β is a design threshold.

If the decision is to continue sampling, the controller chooses a control action U_{k+1} drawn from the distribution

$$q_{k+1}^*(u) \ \triangleq \ \mathbb{P}\{U_{k+1} = u | \widehat{I}_k = 0\}$$

where the probability vector $\boldsymbol{q}_{k+1}^*$ is obtained as a solution to the following maxmin optimization problem

$$\max_{q(u), u \in V} \ \min_{\widetilde{G}: \widetilde{G} \in \mathcal{G}_j} \sum_{u=1}^{N} c(u, u^k, \widehat{G}) q(u) D(P_{\widehat{G}}^u, P_{\widetilde{G}}^u) \tag{6}$$

where $D(P_1, P_2)$ denotes the Kullback-Leibler (KL) distance between the distributions P_1 and P_2 [9] and $c(u, u^k, \widehat{G})$ is a user-designed positive weighting function whose purpose will be discussed in section 4.

If the KL distance between distributions is zero under at least one control, we modify the test by using a uniform control at times $\lceil a^l \rceil$ for $l \in \mathbb{N}$ and $a > 1$ fixed. By [5], this test has asymptotically optimal error decay with the sample size under some technical conditions omitted due to space constraints.

A simple modification to this procedure is to truncate the test to a fixed number of samples, akin to the truncated sequential probability ratio test (SPRT) to sequentially decide between two hypotheses with independent and identically distributed observations. In the case of the truncated SPRT, the test retains good stopping times and error performance (for suitably large number of samples) while avoiding pitfalls such as sample paths where a large number of samples are needed to make a decision [10].

It is straightforward to extend the test to the case where the controls are subsets of V to sample multiple nodes at each time. The graph estimate and stopping rule are identical, while the control policy has the same structure. Details are omitted due to space constraints.

3.1 Control Policy

The optimization problem in (6) can be viewed as a zero sum game. Let $\widehat{G} \in \mathcal{G}_i$. Then, player 1, the maximizer, tries to choose a distribution $\boldsymbol{q}$ over the vertex set V, while player 2 chooses the nearest graph in $\mathcal{G}_j$ for $j \neq i$. We will show that this can be done by inserting (resp. removing) edges to (resp. from) $\widehat{G}$ if $\widehat{G} \in \mathcal{G}_0$ (resp. $\mathcal{G}_1$). Since a graph can have $O(N^2)$ edges on N vertices, the number of edges which can be inserted or removed from $\widehat{G}$ can be significantly larger than N. In particular, when η is chosen sufficiently high and

the underlying graph is sufficiently sparse, there exist enough edges in G^C (G) such that the average node degree can be increased (decreased) so that player 1 is forced to adopt a uniform control. By good design of $c(u, u^k, \widehat{G})$, this can be alleviated by trading exploration with exploitation. Let $\widehat{\boldsymbol{d}}$ and $\widetilde{\boldsymbol{d}}$ be the degree vectors of the graphs $\widehat{G}$ and $\widetilde{G}$ respectively. In the case of model 1,

$$D(P^u_{\widehat{G}}, P^u_{\widetilde{G}}) = \sum_{i=0}^{\widehat{d}_u} \binom{\widehat{d}_u}{i} p^i (1-p)^{\widehat{d}_u - i} \log \left(\frac{p^i (1-p)^{\widehat{d}_u - i}}{p^i (1-p)^{\widetilde{d}_u - i}} \right)$$
$$= \alpha'_u (\widetilde{d}_u - \widehat{d}_u) \log \frac{1}{1-p}$$

where α'_u is 1 if $\widehat{d}_u > 0$ and is 0 otherwise. Note that this distance is infinite if $N_u(\widehat{G}) \not\subseteq N_u(\widetilde{G})$, since every edge observed is known to be in the underlying graph. Similarly, under OM2, $D(P^u_{\widehat{G}}, P^u_{\widetilde{G}}) = \alpha'_u r(\widetilde{d}_u - \widehat{d}_u)$ for some constant r (regardless of the relation of N_u in both graphs). Thus, in both cases, (6) can be written as:

$$\max_{q(u), u \in V} \min_{\widetilde{G}: \widetilde{G} \in \mathcal{G}_j} \sum_{u=1}^{N} \alpha_u q(u) \widetilde{d}_u$$

where $\alpha_u = \alpha'_u c(u, u^k, \widehat{G})$.

In practice, it can be useful to replace α'_u with

$$\alpha'_u = \begin{cases} 1 & \text{if } \widehat{d}_u > 0 \\ f(u, u^k, \widehat{G}) & \text{o.w.} \end{cases}$$

where f is an experimenter-designed non-negative weighting function chosen to penalize unexplored nodes and encourage exploiting explored nodes. While the choice of $f(u, u^k, \widehat{G})$ and $c(u, u^k, \widehat{G})$ do not change the asymptotic error performance, a good choice will improve non-asymptotic error performance.

We begin with the case where $\widehat{G} \in \mathcal{G}_0$. The optimization problem is easily posed in terms of the incidence matrix of $\widehat{G}^C$, $M_{\widehat{G}^C}$ [11]. The incidence matrix of G is a matrix with $|E_G|$ columns, where the column corresponding to edge $e_{ij} \in E_G$ has 1's in indices i and j and zeros elsewhere. Note that the sum of the i-th row of an incidence matrix is the degree of the i-th vertex. Let $M'_{\widehat{G}^C}$ denote the matrix obtained from $M_{\widehat{G}^C}$ by multiplying the i-th row of $M_{\widehat{G}^C}$ by $\alpha_i, i = 1, \ldots, N$. P_N is the N-dimensional probability simplex and $IS = \{\boldsymbol{x} \in \{0,1\}^{|E_{\widehat{G}^C}|} : \boldsymbol{x} \text{ has } \lceil |\eta - \bar{d}_{\widehat{G}}| \frac{N}{2} \rceil \text{ non-zero entries}\}$ (where IS stands for "insertion set" since we are inserting edges into $\widehat{G}^C$). Hence, the problem is

$$\max_{\boldsymbol{q} \in P_N} \min_{x \in IS} \boldsymbol{q} M'_{\widehat{G}^C} \boldsymbol{x}. \tag{7}$$

That is, when player 1 picks a distribution, player 2's policy is to insert edges in $\widehat{G}$ until the new graph is in $\mathcal{G}_1$. Player 2 does so by adding edges which do not exist in $\widehat{G}$ whose endpoints are of lowest sum weight, akin to the optimization for the stopping rule shown later. Since each edge inserted increases the average node degree by $\frac{2}{N}$, $\lceil |\eta - \bar{d}_{\widehat{G}}| \frac{N}{2} \rceil$ edges must be inserted into $\widehat{G}$ to get a graph in $\mathcal{G}_1$.

A satisfactory relaxation of (7) for computational purposes is

$$\max_{\boldsymbol{q} \in P_N} \min_{\{x \in \mathbb{R}^{|E_{\widehat{G}^C}|} : x \geq 0, \mathbf{1}^\top x = \lceil |\eta - \bar{d}_{\widehat{G}}| N/2 \rceil\}} \boldsymbol{q} M'_{\widehat{G}^C} \boldsymbol{x}.$$

This can be rewritten to the equivalent linear program (LP):

$$\max_{\boldsymbol{q}, v} v$$

$$\text{subject to: } \begin{cases} \sum_u q_u [M'_{\widehat{G}^C}]_{uj} \geq v & j = 1, \ldots, |E_{\widehat{G}^C}| \\ \sum_u q_u = 1, \quad 0 \leq q_u \leq 1 \end{cases}$$

which can be solved using standard LP techniques. For OM1, this completely specifies the control policy as $\widehat{G} \notin \mathcal{G}_1$.

The case for $\widehat{G} \in \mathcal{G}_1$ in OM2 is similar. We see that $D(P^u_{\widehat{G}}, P^u_{\widetilde{G}}) = \alpha |r| (\widehat{d}_u - \widetilde{d}_u)$. Proceeding in the derivation as before, $M_{\widehat{G}^C}$ is replaced with $-M_{\widehat{G}}$ (to correspond to edge removals) to form the optimization problem. The resultant optimization problem is the same, except with $M'_{\widehat{G}^C}$ replaced with $M'_{\widehat{G}}$ and IS is defined identically, except as a subset of $\mathbb{R}^{|E_{\widehat{G}}|}$.

3.2 Maximum-Likelihood Graph Estimation

In this section, we propose a simple MLE of G - more advanced estimators can improve detection performance at higher computational cost with more accurate graph representation, particularly in the case of more sophisticated models. A graph G is given by its adjacency matrix A_G, where $[A_G]_{ij} = 1$ if $e_{ij} \in E_G$ and is 0 otherwise.

Under OM1, it is clear that the MLE of the graph, $\widehat{G}$, is simply the graph consisting of all edges observed up to the current time (since no edge observed is spurious). Thus, we consider OM2.

At time k and for all $i \in V$, define $\mathcal{T}_i(k) = \{j \in \{1, \ldots, k\} : U_j = i\}$ as the set of all times up to k when node i is selected. We assume that the observations of the various nodes are independent conditioned on their respective neighborhoods

$$P(y^k | G) = \prod_{i=1}^{N} P(y_{\mathcal{T}_i(k)} | N_i(G)) \tag{8}$$

where $y_{\mathcal{T}_i(k)} = \{y_j : j \in \mathcal{T}_i(k)\}$.

Define $\mathcal{T}_{ij}(k) = \mathcal{T}_i(k) \cup \mathcal{T}_j(k)$. This is the number of times edge e_{ij} can be *potentially observed* up to time k. Denote the number of times edge e_{ij} is actually observed up to time k by $l_{ij}(k)$. If $e_{ij} \in E$ (resp. $e_{ij} \notin E$), the probability of the observation sequence is $p^{l_{ij}(k)}(1-p)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)}$ (resp. $q^{l_{ij}(k)}(1-q)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)}$).

Thus, $\widehat{G}$ at time k is $[A_{\widehat{G}}]_{ij} = 1$ if $p^{l_{ij}(k)}(1-p)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)} > q^{l_{ij}(k)}(1-q)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)}$ and $[A_{\widehat{G}}]_{ij} = 0$ otherwise or if $i = j$. The MLE can be calculated in $O(N)$ time.

3.3 Stopping Rule

The form of (5) under OM1 is simple:

$$\log \frac{P_{\widehat{G}}(y^k, u^k)}{P_{\widetilde{G}}(y^k, u^k)} = \sum_{j=1}^{k} \log \frac{P_{\widehat{G}}(y_j, u_j)}{P_{\widetilde{G}}(y_j, u_j)}$$
$$= -\log(1-p) \sum_{j=1}^{k} (\widetilde{d}_{u_j} - \widehat{d}_{u_j})$$
$$= -\log(1-p) \sum_{i=1}^{N} (\widetilde{d}_i - \widehat{d}_j) |\mathcal{T}_i(k)|$$

Collecting constants gives the stopping rule

$$\min_{\widetilde{G} \in \mathcal{G}_1, E_{\widetilde{G}} \supset E_{\widehat{G}}} \sum_{i=1}^{N} (\widetilde{d}_i - \widehat{d}_j) |\mathcal{T}_i(k)| > \log \beta \tag{9}$$

This can be solved in quadratic (in N) time by noting that we can find the minimizer by inserting edges into $\widehat{G}$, and adding edge e_{ij} increases the sum by $|\mathcal{T}_i(k)| + |\mathcal{T}_j(k)| = |\mathcal{T}_{ij}(k)|$ independent of the other edges (so insertion order does not matter). $E_{\widehat{G}^C}$ and $|\mathcal{T}_{ij}(k)|$ can be calculated by finding the non-diagonal zeros of $A_{\widehat{G}}$

and updating $\mathcal{T}_{ij}(k)$ after every sample. To find $\widetilde{G}$, sort the edges by $|\mathcal{T}_{ij}(k)|$ in ascending order and insert the first $\lceil |\eta - \bar{d}_{\widehat{G}}|N/2 \rceil$ edges into $\widehat{G}$. Thus, the LHS of the stopping rule is the sum of the $\lceil (\eta - \bar{d}_{\widehat{G}})\frac{N}{2} \rceil$ smallest values of $|\mathcal{T}_{ij}(k)|$.

For OM2, we start with the case where $\widehat{G} \in \mathcal{G}_0$. Then,

$$\log \frac{P_{\widehat{G}}(y^k, u^k)}{P_{\widetilde{G}}(y^k, u^k)} = \log \left(\frac{\prod_{e_{ij} \in E_{\widehat{G}}} p^{l_{ij}(k)} (1-p)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)}}{\prod_{e_{ij} \in E_{\widetilde{G}}} p^{l_{ij}(k)} (1-p)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)}} \times \frac{\prod_{e_{ij} \in E_{\widehat{G}^C}} q^{l_{ij}(k)} (1-q)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)}}{\prod_{e_{ij} \in E_{\widetilde{G}^C}} q^{l_{ij}(k)} (1-q)^{|\mathcal{T}_{ij}(k)| - l_{ij}(k)}} \right) \quad (10)$$

The numerator is solely a function of $\widehat{G}$ and can be precomputed. Inserting or removing an edge from a graph corresponds to removing or adding the edge to the complement graph. We can split the moving of edges from $\widetilde{G}^C$ to $\widetilde{G}$ into two cases for the minimization. Let $\delta_{ij}(k)$ denote the change in (10) when edge e_{ij} is moved from $\widehat{G}^C$ to $\widehat{G}$. If $e_{ij} \in E_{\widehat{G}^C}$ and $|\mathcal{T}_{ij}(k)| = 0$, $\delta_{ij}(k) = 0$. If $e_{ij} \in E_{\widehat{G}^C}$ and $|\mathcal{T}_{ij}(k)| \neq 0$, $\delta_{ij} = l_{ij}(k) \log \frac{q}{p} + (|\mathcal{T}_{ij}(k)| - l_{ij}(k)) \log \frac{1-q}{1-p}$. Note that δ_{ij} is independent of edges other than e_{ij}.

Thus, the LHS of (5) under OM2 can be calculated by noting that $E_{\widehat{G}} \subset E_{\widetilde{G}}$ and starting with $E_{\widehat{G}} = E_{\widetilde{G}}$. Then, the left hand side is the sum of the $\lceil (\eta - \bar{d}_{\widehat{G}})\frac{N}{2} \rceil$ smallest δ_{ij} corresponding to $e_{ij} \in E_{\widehat{G}^C}$. This can be solved in $O(N^2)$ time.

In the case where $\widehat{G} \in \mathcal{G}_1$, replace all references to $\widetilde{G}^C$ with $\widetilde{G}$, and note that $E_{\widetilde{G}} \subset E_{\widehat{G}}$. Since we move edges from $\widetilde{G}$ to $\widetilde{G}^C$, replace δ_{ij} with $-\delta_{ij}$, and the rest of the algorithm is identical.

In practice, it is useful to start the algorithm with some initial observations of each node (or a subset of nodes) in order to reduce the stopping time as in remark 7.1 of [8].

4 Discussion and Results

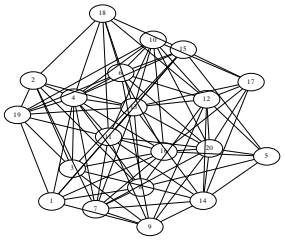

Fig. 1: Graph G with 20 nodes with average node degree 8.9

For concreteness and conciseness, we present classification performance on an Erdös-Rènyi (ER) generated graph with uniform edge probability $\frac{1}{2}$ on 20 nodes with average node degree 8.9 shown in Fig.1. ER graphs are of interest from a performance analysis perspective as proving properties of an appropriate family of ER graphs shows that the property holds for almost all graphs [11]. As per remark 7.1 in [8], the procedure presented is of interest when η is close to $\bar{d}_G$. Thus, we show results for $\eta = 8.8$ and $\eta = 9.0$ with the tests truncated to 1000 samples. $c(u, u^k, \widehat{G})$ is the number of times a node has been sampled up to time k, or 1 if it has not been sampled, i.e.,

$$c(u, u^k, \widehat{G}) = \begin{cases} |\mathcal{T}_u(k)| & \text{if } |\mathcal{T}_u(k)| > 0 \\ 1 & \text{o.w.} \end{cases}$$

The standard deviation of all probabilities presented down to 10^{-3} is at least an order of magnitude below the probabilities. The results

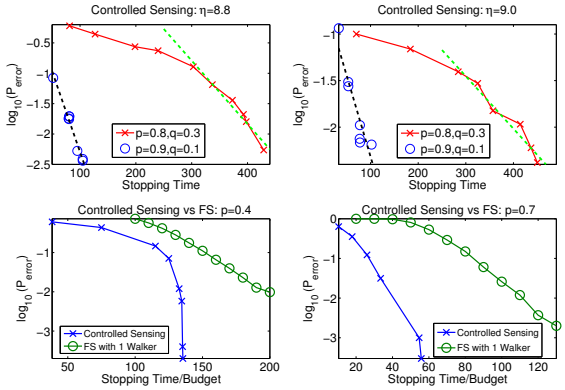

Fig. 2: First Row: Controlled sensing with spurious observations. Second Row: Controlled sensing versus FS without spurious observations. Note that connected lines are drawn only for readability.

for OM1 with $p = 0.4$ and $p = 0.7$ are given in the bottom row of Fig.2.The controlled sensing test with expected stopping time $E[N]$ performs strictly better than the FS with budget $E[N]$ with $\eta = 8.8$ (since false alarms are not possible under this model, this is the hardest value of η to classify) in the sense of lower error probabilities. This is in part due to the stopping rule, which accounts for p while FS explicitly assumes $p = 1$. The control is also tailored to capture the structure of G which controls the average node degree rather than the general structure of G as in the case of FS. There is also a threshold phenomena in detection, where the probability of error falls off at a very high rate when controlled sensing has (on average) observed enough edges to conclude the graph has average node degree greater than η. It was also found that FS offered little improvement under this graph model until the number of walkers was on the order of N since it is unlikely then for a walker to get trapped in a small neighborhood in the graph.

Under OM2, FS is not directly applicable due to the spurious edges allowing a walker to transition between non-neighbors in G. Thus, we compare two controlled sensing tests to study the relative performance difference between tests with different observation probabilities for both true and spurious edges ($p = 0.8, q = 0.3$ and $p = 0.9, q = 0.1$) in the bottom row of 2. Lowering q and increasing p significantly reduces the number of samples needed to achieve a given error probability. The dashed least-squares fit lines shown for the tails of the data indicate that in these regimes the error probability decays approximately exponentially. This is consistent with the asymptotic exponential decay of the error probability with the stopping time in Chernoff's procedure and controlled sensing [8, 5].

5 Conclusions and Future Work

In this paper, we proposed a controlled sensing based test for classifying a graph based on connectivity using probabilistic observations of its nodes. This test was shown to outperform classic random walk based approaches at low target error rates. The asymptotic optimality of the proposed test follows from the optimality of the modified Chernoff test[5]. Future work includes developing suboptimal distributed controlled sensing tests that admit simpler computations and that can be easily parallelized. Another key direction for future work is to exploit the sparsity present in many networks such as the DBLP authorship data set and to develop approximate algorithms for other connectivity measures.

6 References

[1] J. Leskovec and C. Faloutsos, "Sampling from large graphs," in *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, New York, NY, 2006, pp. 631 – 636, ACM.

[2] J-P Onnela and N. A. Christakis, "Spreading paths in partially observed social networks," *Phys. Rev. E*, vol. 85, no. 3, pp. 036106, Mar 2012.

[3] B. Ribeiro and D. Towsley, "Estimating and sampling graphs with multidimensional random walks," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, New York, NY, 2010, IMC '10, pp. 390–403, ACM.

[4] G. K. Atia and V. V. Veeravalli, "Controlled sensing for sequential multihypothesis testing," in *Information Theory Proceedings (ISIT), 2012 IEEE International Symposium on*, Cambridge, MA, July 2012, pp. 2196 – 2200.

[5] S. Nitinawarat, G. K. Atia, and V. V. Veeravalli, "Controlled sensing for multihypothesis testing (submitted)," *IEEE Trans. Autom. Contr.*, May 2012.

[6] K. Avrachenkov, N. Litvak, M. Sokol, and D. Towsley, "Quick detection of nodes with large degrees," Research Report 7881, INRIA, Sophia Antipolis, France, Feb 2012.

[7] S. Nitinawarat, G. K. Atia, and V. V. Veeravalli, "Controlled sensing for hypothesis testing," in *Proc. of the 37-th Int. conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, Kyoto, Japan, Mar 2012, IEEE.

[8] H. Chernoff, "Sequential design of experiments," *Ann. Math. Statist.*, vol. 30, pp. 755–770, 1959.

[9] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, NY: John Wiley and Sons, Inc., 2006.

[10] B.C. Levy, *Principles of Signal Detection and Parameter Estimation*, Springer, New York, NY, 2008.

[11] D. B. West, *Introduction to Graph Theory*, Prentice Hall, 2 edition, 2001.

[12] Stephen Boyd and Lieven Vandenberghe, *Convex Optimization*, Cambridge Univ. Press, Cambridge, U.K., Mar. 2004.