# Optimality of Fast Matching Algorithms for Random Networks with Applications to Structural Controllability

Mohamad Kazem Shirani Faradonbeh, Ambuj Tewari, and George Michailidis

*Abstract*—Network control refers to a very large and diverse set of problems including controllability of linear time-invariant dynamical systems, where the objective is to select an appropriate input to steer the network to a desired state. There are many notions of controllability, one of them being *structural controllability*, which is intimately connected to finding maximum matchings on the underlying network topology. In this work, we study fast, scalable algorithms for finding maximum matchings for a large class of *random networks*. First, we illustrate that *degree distribution* random networks are realistic models for real networks in terms of structural controllability. Subsequently, we analyze a popular, fast and practical heuristic due to Karp and Sipser as well as a simplification of it. For both heuristics, we establish asymptotic optimality and provide results concerning the asymptotic size of maximum matchings for an extensive class of random networks.

*Index Terms*—Maximum Matching, Karp-Sipser, Structural Controllability, Network Control, Random Networks.

## I. INTRODUCTION

NETWORKS are capable of capturing relationships between a set of entities (vertices) and have found applications in diverse scientific fields including biology, engineering, economics and the social sciences [1], [2], [3]. Network control refers to a very large and diverse set of problems that involve control actions over a network (see for example, [4], [5], [6], [7], [8], [9], [10], [11], [12], [13] and references therein).

A class of control problems involves dynamical systems evolving over time that have inputs and outputs and many results exist for systems that exhibit linear and time-invariant dynamics [11]. One particular notion of control is that of *structural controllability* [1], [14], which was recently explored by Liu et al. [12]. Under this notion, the structural controllability problem reduces to find maximum matchings on appropriate

Mohamad K. S. Faradonbeh: Department of Statistics, 311 West Hall, 1085 South University, Ann Arbor, MI 48109-1107, Phone: 7346478192 (shirany@umich.edu)

Ambuj Tewari: Assistant Professor of Statistics and Electrical Engineering and Computer Science, Department of Statistics, 311 West Hall, 1085 South University, University of Michigan, Ann Arbor, MI 48109-1107, Phone: 7346478192 (tewaria@umich.edu)

George Michailidis: Professor of Statistics and Computer and Information Science and Engineering, Department of Statistics, 102 Griffin-Floyd Hall, University of Florida, Gainesville, FL 32611, Phone: 3523921941 (gmichail@ufl.edu)

matrices as reviewed in Section I-A. The problem of obtaining maximum matchings has been extensively studied in the computer science literature both for deterministic [15] as well as random networks [16]. However, the focus in the literature has been on special classes of *undirected random networks* [17], [18]. For example, using the results of Tao and Vu [19], one can infer that for *dense* Erdos-Renyi random networks, a single controller is sufficient to ensure controllability. Little is known about the performance of matching algorithms in other interesting classes of random networks [20], with an exception of the recent work of Balister and Gerke [21]. We focus on *degree distribution* random network models, and establish results on the minimum number of controllers needed to guarantee structural controllability. We also show that these models are realistic representations for real world applications.

A popular, fast and practical algorithm for matchings on undirected random networks is due to Karp and Sipser [22], which represents the cornerstone of our theoretical investigations and through it we provide generalizations of previous work in the literature to broader classes of undirected random networks. Further, we also extend the results for directed variants of the same classes of random networks.

### A. Structural Controllability and Maximum Matchings

Next, we review some key concepts in structural controllability for linear dynamical systems. Consider a system described by a $n$-dimensional *state* vector $x(t) = (x_1(t), ..., x_n(t))^T \in \mathbb{R}^n$, whose dynamical evolution is described by

$$\frac{dx}{dt} = Ax(t) + Bu(t),$$

where $A \in \mathbb{R}^{n \times n}$ is the system transition matrix, $u(t) = (u_1(t), ..., u_k(t))^T \in \mathbb{R}^k$ captures control actions and $B \in \mathbb{R}^{n \times k}$ is the input matrix. Assuming that the $n$-dimensional system can be represented by vertices on a network $G = (V, E)$ with $V = \{1, 2, ..., n\}$ denoting the set of vertices and $E \subset V \times V$ the set of edges, it can be seen that the non-zero entries in the transition matrix $A$ correspond to the directed edges in $E$. Indeed, for $i, j \in V$ the edge $i \to j$ exists if and only if vertex $i$ influences vertex $j$, i.e. $A_{ji} \neq 0$. Such a system is called *controllable* if for any initial state $x(0) = x_0$ and any desired state $x_d$ for some $T < \infty$ one can find an input matrix $B$ and control vectors $\{u(t)\}_{0 \leq t \leq T}$ so that the system reaches state $x_d$ i.e. $x(T) = x_d$. The minimum $k$ for which system can be controllable is called the minimum number of controllers.

The magnitude of the entries in the transition matrix $A$ captures the interaction strength between the vertices in the network; for example, the traffic on individual communication links in a communications network or the strength of a regulatory interaction in a biological network. The time invariant matrix $B$ indicates which vertices are controlled by an outside controller. Hence, the set of vertices that when applying controllers to them makes the system controllable needs to be identified.

Note that we don't assume any constraint on the number of nonzero elements in the columns of input matrix $B$, i.e. one controller $u_i(t)$ can influence multiple vertices. The case where every controller can be applied to only one vertex in the network (which in turn implies that the goal is to find the minimum number of vertices the controllers can be applied to) is studied by Olshevsky [23]. He shows that finding the exact solution to the problem is NP hard.

The algebraic criterion to check controllability of a time invariant linear dynamical system is Kalman's controllability rank condition, that states that controllability can be achieved, if and only if the matrix $C = [B, AB, ..., A^{n-1}B]$ is full rank; i.e. $\text{rank}(C) = n$. Note that $C \in \mathbb{R}^{n \times nk}$. This algebraic criterion is computationally hard to check, especially for large systems. Further, in many applications, obtaining exact values of $A$ may not be feasible and hence a tractable alternative is needed.

Thus, we say that a time invariant linear dynamical system is *structurally controllable*, if it is possible to select the non-zero values of $A, B$, so that Kalman's rank condition is satisfied [1]. A structurally controllable system is controllable for almost all $A, B$; i.e. the pathological cases for which a structurally controllable network is not controllable has zero Lebesgue measure. The relation between the minimum number of controllers needed to structurally control a network and the size of its maximum matching has been presented in several forms (see [24]). The version used in the current work is the one appearing in Liu et al. [12] "minimum inputs theorem" (stated below). Moreover, similar equivalence between structural controllability and maximum matching when every controller can be applied to only one vertex, is studied by Assadi et al. [25].

Furthermore, Commault and Dion [26] study the problem of using only a single controller applied to as few vertices as possible. More general results regarding the control configuration selection can be found in the work of Pequito et al. [27] which discusses relations to maximum matching problem as well. Other considerations must be taken into account for deciding which approach is most suitable for the practical application under consideration. Finally, we have no assumption regarding self-loops in the network. Cowan et al. [28] study networks, where every vertex influences itself.

Algorithms to find a maximum matching are well studied in the literature and exhibit polynomial time complexity (with respect to the size of the network). A popular one developed by Micali and Vazirani [29] has running time $O\left(|V|^{0.5}|E|\right)$.

Next, for completeness we provide a definition of maximum matching and also state the minimum inputs theorem.

**Definition 1.** *For a directed network $G = (V, E)$, a subset of edges $M$ is a matching, if no two edges in $M$ share a common starting or a common ending vertex. A maximum matching corresponds to a matching of maximum size.*

**Definition 2.** *Given a matching $M$ for directed network $G = (V, E)$, a vertex is matched, if it is an ending vertex of an edge in the matching $M$. Otherwise, it is unmatched.*

**Minimum Inputs Theorem** [12]: *Let $M^*$ be a maximum matching of the network $G = (V, E)$. The minimum number of controllers needed for structural controllability of the network is $\max\{1, n - |M^*|\}$. Moreover, for any matching $M$, the network is structurally controllable using $\max\{1, n - |M|\}$ controllers.*

The upshot of this result is that in order to find the minimum number of required controllers for structural controllability we can equivalently find the size of a maximum matching. Furthermore, one can explicitly find the structure of the input matrix $B$ according to the proof of minimum inputs theorem. In fact, one can see that a matching is formed of a set of directed loops and a set of directed paths. The first vertex of each path is unmatched, while all other vertices are matched. According to Lin's work [1], one controller is used to actuate every unmatched vertex. Amongst matched vertices, some might need to be actuated by a controller, but no new controller is needed, as it suffices to apply any of the previously used controllers to one arbitrary vertex of each loop. So the number of nonzero components in $B \in \mathbb{R}^{n \times k}$, or equivalently the total number of the connections between the inputs and vertices in the network, is at most $n$. More details are provided by Liu et al. (Supplementary Information of [12]).

As the problem of finding the minimum number of controllers needed for structural controllability of a network reduces to maximum matching type of problems, henceforth we use the (minimum) number of unmatched vertices and the (minimum) number of controllers interchangeably. In this work, we provide results about the size of matchings obtained by different fast algorithms for classes of random networks. In fact, Karp and Sipser [22] proved that, for the classical undirected Erdos-Renyi random network, the $KS$ algorithm is optimal. We generalize their results to a larger class of random networks. The remainder of the paper is organized as follows. In Section II we introduce different classes of random networks subsequently studied in this work. Furthermore, some probabilistic results needed for technical developments are summarized. The main algorithms studied are introduced in Section III and connections to real networks using some numerical examples are provided in Section IV. The key results of the paper are presented in Sections V (analysis and optimality of the algorithms).

**Notations:** For set $S$, let $|S|$ be the number of elements in $S$. $\binom{n}{n_1,...,n_r} = \frac{n!}{n_1!...n_r!}$. For vertices $u, v \in V$ in the network $G = (V, E)$, $\{u, v\}$ ($(u, v)$ or $(v, u)$ in directed or bipartite networks) denotes an edge, so $N = M \cup \{\{u, v\}\}$ means adding the edge $\{u, v\}$ to $M$ gives $N$. $G - \{u, v\}$ means removing vertices $u, v$ (and so all edges connected to them) from network $G$. Further, $\deg_G(v)$ denotes the degree of $v$ in $G$ i.e. the number of edges in $G$ connected to $v$. When there

is no subscript, the network $G$ is identifiable from the context. Finally, for $x \in \mathbb{R}^n$, $n \in \{0, 1, 2, ...\} \cup \{\infty\}$, $\|x\|_1$ is $\ell_1$ norm of $x$: $\|x\|_1 = \sum_{i=1}^{n} |x_i|$.

## II. RANDOM NETWORKS

In this section, we introduce different classes of random network models and then present some general results about convergence and concentration of real-valued functions on networks. In order to have a general framework which includes both directed and undirected networks, we note that every undirected network $G = (V, E)$ can be considered as a directed network in which, for all vertices $i, j$, both edges $i \to j$ and $j \to i$ exist, if and only if the edge $i \leftrightarrow j$ exists in the original undirected network. All statements presented are true for both directed and undirected networks unless explicitly mentioned. For a comprehensive discussion on constructions and properties of (undirected) random networks, see chapter 3 in Durrett [30].

The first model for random networks we consider is the **Erdos-Renyi** ($ER$) model. A directed network $G = (V, E)$, $V = \{1, 2, ..., n\}$ is (drawn from) $ER$ if every edge $i \to j$ for $i, j = 1, 2, ..., n$ is present in the network independently with probability $p^{(n)}$. Analogously, an undirected network $G = (V, E)$, $V = \{1, 2, ..., n\}$ is $ER$ if every edge $i \leftrightarrow j$ for $i, j = 1, 2, ..., n, i \leq j$ is present in the network independently with probability $p^{(n)}$. Henceforth, for $\lambda \in [0, \infty]$, $ER(\lambda)$ is a Erdos-Renyi random network, for which $np^{(n)} \to \lambda$ as $n \to \infty$. In $ER(\lambda)$ random networks, the parameter $\lambda$ corresponds to the average degree.

The next model we consider is the **Uniform Fixed-Size** ($UFS$) model. A directed network $G = (V, E)$, $V = \{1, 2, ..., n\}$ is $UFS$ when the cardinality of the edge set $|E| = k_n$ for some fixed $k_n$, and the $k_n$ directed edges are drawn uniformly among all $n^2$ possible edges. The construction for the undirected network is similar, but the $k_n$ edges are chosen uniformly among all $\frac{n(n+1)}{2}$ possible edges. For $\lambda \in [0, \infty]$, we denote by $UFS(\lambda)$ a random network of the UFS class, for which $\frac{k_n}{n} \to \lambda$ for directed and $\frac{k_n}{n} \to \frac{\lambda}{2}$ for the undirected case, as $n \to \infty$. Once again, the $\lambda$ parameter corresponds to the average degree.

Finally, we introduce the class of **Degree Distribution** ($DD$) random networks. There are a couple of reasons for considering this class. First, it lets us consider networks with degree distributions commonly found in real networks that simpler models such as $ER$ (where the degree distribution is Poisson) cannot model. For example, as shown in Fig. 1, the degree distribution of real world networks can be any arbitrary non-parametric distribution. Studying the degree distribution of more networks leads to a similar conclusion (see the Supplementary Materials).

Second, as empirically shown in Section IV, the number of controllers for structural controllability of a network is, to a large extent, determined by its degree distribution. More details can be found in Section IV as well as the work of Liu et al. [12].

An undirected random network is a member of the $DD$ class, if for a given degree distribution the attachment of edges
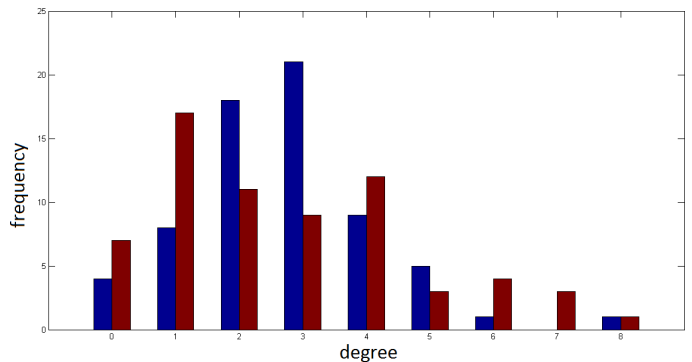


Fig. 1: Histogram of the degree distribution of the social network between prison inmates (blue: input degree, red: output degree). As shown here, it is difficult to model degree distributions of real networks using standard parameteric distributions. Additional information for this network is given in the Supplementary Materials.

is random. Specifically, let $p$ be a probability distribution with support on the set $\{0, 1, 2, ...\}$ of nonnegative integers. We then construct an undirected network $DD(p) = (V, E)$ as follows. Let $V = \{1, 2, ..., n\}$ and for $i \in V$, let vertex $i$ have $D_i$ undirected half-edge(s) (one-half of an edge is connected to vertex $i$). Note that $D_1, ..., D_n$ are the corresponding degrees which are assumed to be independent and identically distributed (iid) with distribution $p$; $\mathbb{P}(D_i = k) = p(k)$. To complete the construction, we then pair all half-edges randomly; i.e. all $\binom{\sum_{i=1}^{n} D_i}{2,...,2}$ possible attachments of half-edges have equal probability.

When the number of half-edges $\sum_{i=1}^{n} D_i$ is an even number, the construction is straightforward and the number of edges will be $\frac{1}{2} \sum_{i=1}^{n} D_i$. When it is an odd number, we pair the half-edges randomly to obtain the network and omit the last single half-edge for which no pairing was established at the end of the construction, so that the number of edges will be $\frac{1}{2}(\sum_{i=1}^{n} D_i - 1)$. Note that the omission or presence of multiple edges will lead to a difference between $D_1, ..., D_n$ and the actual observed degrees $\deg(1), ..., \deg(n)$ once the network construction is completed. However, as Lemma 1 below establishes, the asymptotic empirical degree distribution will be the original degree distribution from which the network was constructed, as long as the expected value of $D_i$, $\mathbb{E}D_i$ is finite.

Viewing an undirected network $DD(p)$ as a directed one, both input and output degrees of vertex $i$ are $\deg(i)$. To construct a directed DD random network, denoted by $DD(p_{in}, p_{out})$, with distinct input ($\deg_{in}$) and output ($\deg_{out}$) degrees, we do the following: once we have iid draws $D_i^{(in)}$ and $D_i^{(out)}$ from the input and output degree probability distributions $p_{in}$ and $p_{out}$ respectively:

$$\mathbb{P}(D_i^{(in)} = k) = p_{in}(k), \ \mathbb{P}(D_i^{(out)} = k) = p_{out}(k),$$

let vertex $i \in V$ have $D_i^{(in)}$ directed half-edges pointing

into vertex $i$ and $D_i^{(out)}$ directed half-edges pointing out from vertex $i$. Next, we pair directed half-edges randomly to have $\min\{\sum_{i=0}^{n} D_i^{(in)}, \sum_{i=0}^{n} D_i^{(out)}\}$ edges and omit the remaining half-edges. The random pairing of half-edges implies that all $\left(\max\left\{\sum_{i=0}^{n} D_i^{(in)}, \sum_{i=0}^{n} D_i^{(out)}\right\}\right)!$ possible pairings of half-edges are equally likely. Note that $D_i^{(in)}, D_i^{(out)}$ do not need to be independent.

Furthermore, in general, the degrees do not need to be iid. In fact, as shown later in the paper, the key asymptotic results we establish are based on the empirical degree distributions which are, by the following lemma, same as the original degree distributions when vertex degrees are iid. However, as long as for all $k = 0, 1, \ldots$, $\lim_{n\to\infty} \frac{|\{i \in V : D_i = k\}|}{n}$ (equivalently $\lim_{n\to\infty} \frac{|\{i \in V : D_i^{(in)} = k\}|}{n}$, $\lim_{n\to\infty} \frac{|\{i \in V : D_i^{(out)} = k\}|}{n}$) are deterministic, our results hold using the resulting asymptotic empirical degree distributions.

**Lemma 1.** *For an undirected network $G = (V, E)$, define the asymptotic empirical degree distribution as*

$$\hat{p}(k) = \lim_{n\to\infty} \frac{|\{i \in V : \deg(i) = k\}|}{n}.$$

*If $G = DD(p)$ is a random network and $\mu = \sum_{k=0}^{\infty} kp(k) < \infty$, then the limit above exists and we have $\hat{p}(k) = p(k)$ for all $k = 1, 2, \ldots$. In general for a network $G = (V, E)$, define the asymptotic empirical input and output degree distributions as*

$$\hat{p}_{in}(k) = \lim_{n\to\infty} \frac{|\{i \in V : \deg_{in}(i) = k\}|}{n}$$

$$\hat{p}_{out}(k) = \lim_{n\to\infty} \frac{|\{i \in V : \deg_{out}(i) = k\}|}{n}.$$

*If $G = DD(p_{in}, p_{out})$ is a random network and $\mu = \sum_{k=0}^{\infty} kp_{in}(k) = \sum_{k=0}^{\infty} kp_{out}(k) < \infty$, then for all $k = 1, 2, \ldots$ the limits above exist and we have $\hat{p}_{in}(k) = p_{in}(k), \hat{p}_{out}(k) = p_{out}(k)$.*

Henceforth, for all networks by $p_{in}, p_{out}$ we mean asymptotic empirical degree distributions $\hat{p}_{in}, \hat{p}_{out}$ respectively.

Next, we define the Lipschitz property for real valued functions defined over networks.

**Definition 3.** *Let $f$ be a real-valued function on the set of directed networks. We say $f$ has the **Lipschitz** property, if $|f(G_1) - f(G_2)| \leq 1$ whenever $G_1 = (V, E_1), G_2 = (V, E_2)$, $E_2 = E_1 \cup \{e\}$; i.e. the value of $f$ will change at most by 1 if one new edge is added to the network.*

**Remark 1.** *Properly defining the norm $\|\cdot\|$ on the networks, one can see the Lipschitz property is the same as the classic notion $|f(G_1) - f(G_2)| \leq \|G_1 - G_2\|$. For additional discussion, see the Supplementary Materials.*

Next, we present convergence and concentration inequalities for functions of random networks which have the Lipschitz property. Specifically, the number of unmatched vertices (or equivalently the number of controllers) obtained by a matching algorithm, has the Lipschitz property as shown in Section V. The consequence of the Lipschitz property for a function defined on a "not too dense" random network, is that it concentrates around its expected value. Further, if the average degree of the network is finite, then the concentration occurs exponentially fast. These results are summarized in the following theorem.

**Theorem 1** (Convergence of real-valued functions for random networks). *Let $G = (V, E)$ belong to the ER, UFS, $DD(p_{in}, p_{out})$ or $DD(p)$ class. For a real-valued function $f$ which has the Lipschitz property, if*

$$\limsup_{n\to\infty} \frac{\mathbb{E}(|E|)}{n^2} = 0,$$

*then $\frac{f(G) - \mathbb{E}(f(G))}{n} \to_P 0$ as $n \to \infty$. If in addition,*

$$\limsup_{n\to\infty} \frac{\mathbb{E}(|E|)\log n}{n^2} = 0,$$

*then $\frac{f(G) - \mathbb{E}(f(G))}{n} \to 0$ a.s. as $n \to \infty$. When $\sup_{n\geq 1} \frac{\mathbb{E}(|E|)}{n} < \infty$ the rate of convergence is exponential; i.e. there is $C > 0$ such that for every $0 < \epsilon < 1$ :*

$$\mathbb{P}\left(\frac{|f(G) - \mathbb{E}(f(G))|}{n} > \epsilon\right) \leq 2\exp(-nC\epsilon^2).$$

## III. MATCHING ALGORITHMS

Before studying the algorithms, we provide a description of networks which will be useful later. As mentioned before, every undirected network can be considered as a directed one. Now to have a better understanding of how the algorithms work we view every directed network as a bipartite network $G = (L, R, E)$ where $L = R = V$, $E \subset L \times R$, $L$ and $R$ are respectively the left and the right side of the bipartite network, and for $l \in L, r \in R$ there is an edge $(l, r) \in E$ if and only if in the original directed network there is an edge from $l$ to $r$: $l \to r$. Henceforth we will only deal with bipartite networks.

Matching algorithms take a network as input and produce a matching as output. Maximum matching algorithms will give a matching of maximum size. Algorithm 1 is the well known Greedy Algorithm that produces a suboptimal matching $M_G$ in general. For an arbitrarily chosen vertex $v$ on the right side, Greedy tries to find an arbitrary vertex $u$ on the left side which is connected to $v$ and has not been used before by any other vertex on the right side. Note that as mentioned above, networks can be assumed to be bipartite. Moreover, clearly the time complexity of Greedy is $O(n)$.

Note that Greedy picks an arbitrary vertex $v \in R$ in every iteration. Because the goal is to find a matching of largest possible size, this strategy for picking a vertex can be improved. First note that for every vertex $v \in R$ of degree one, there is a matching of maximum size in which $v$ is matched. The logic is as follows. Let $u \in L$ be the vertex on the left side connected to $v$, $(u, v) \in E$. There must exist a vertex on the right side, $w \in R$, such that $(u, w) \in E$, such that $w$ is matched to $u$ by a matching $M$ of maximum size; since, if not, adding $(u, w)$ to it leads to a matching of larger size. Now defining a new matching $M'$ which is exactly $M$ with $(u, w)$

---

**Algorithm 1 : Greedy**
Input: $G = (L, R, E)$
Output: matching $M_G(G)$

---

$\quad M_G \leftarrow \emptyset$
$\quad$**while** $E \neq \emptyset$ **do**
$\quad\quad$let $v \in R$
$\quad\quad$**if** $\deg(v) = 0$ **then**
$\quad\quad\quad G \leftarrow G - \{v\}$
$\quad\quad$**else if** for $u \in L$, $(u, v) \in E$ **then**
$\quad\quad\quad G \leftarrow G - \{u, v\}$
$\quad\quad\quad M_G \leftarrow M_G \cup \{(u, v)\}$
$\quad\quad$**end if**
$\quad$**end while**
$\quad$**return** $M_G$

---

removed and $(u, v)$ added, i.e. $M' = M \cup \{(u, v)\} - \{(u, w)\}$, we have $|M| = |M'|$ i.e. $M'$ is a maximum matching as well. Hence as long as we can find a vertex of degree one, we can find a matching of exactly maximum size. In other words: *no mistake occurs as long as a degree one vertex is picked in every iteration* of Greedy (a mistake occurs if in an iteration, the algorithm adds an edge to the matching which is not optimal, i.e. leads to a deviation from maximum matching).

---

**Algorithm 2 : Karp-Sipser**
Input: $G = (L, R, E)$
Output: matching $M_{KS}(G)$

---

$\quad M_{KS} \leftarrow \emptyset$
$\quad$**while** $E \neq \emptyset$ **do**
$\quad\quad$let $v = \arg\min_{w \in L \cup R} \deg(w)$
$\quad\quad$**if** $\deg(v) = 0$ **then**
$\quad\quad\quad G \leftarrow G - \{v\}$
$\quad\quad$**else if** for $u \in L \cup R$, $\{u, v\} \in E$ **then**
$\quad\quad\quad G \leftarrow G - \{u, v\}$
$\quad\quad\quad M_{KS} \leftarrow M_{KS} \cup \{\{u, v\}\}$
$\quad\quad$**end if**
$\quad$**end while**
$\quad$**return** $M_{KS}$

---

This fact is the idea behind Algorithm 2, called the Karp-Sipser Algorithm ($KS$) [22], which produces a matching $M_{KS}$. In every iteration of $KS$, among all vertices a vertex of *minimum* degree is picked.

Regarding time complexity of the $KS$ algorithm, note the following connection with the Greedy one: in every iteration, $KS$ picks a vertex of minimum degree and by using a "Heap" data structure, finding a vertex of minimum degree has complexity $O(1)$ [31]; therefore, the running time of $KS$ is linear. Further, since for real networks, the average degree is finite, for each iteration of the $KS$ algorithm (i.e. excluding up to $o(n)$ iterations) there are $O(n)$ many vertices of degree one, which implies that determining a degree one vertex takes on average in $O(1)$ steps.

We can simplify the KS algorithm and search for a minimum degree vertex among vertices on only one side to derive Algorithm 3 that we call the One-sided Karp-Sipser ($OKS$)

Algorithm and whose output we denote by $M_{OKS}$. As we will see later in the analysis of real networks, the size of the matching given by $OKS$ is usually less than or equal to the size of the matching given by $KS$. The intuition behind this is as follows. It is possible to make a mistake in $OKS$ because of lack of degree one vertices on the right side, but if degree one vertices exist on the left side, $KS$ can still work optimally. Yet, later we will prove (asymptotic) optimality of both algorithms.

---

**Algorithm 3 : One-Sided Karp-Sipser**
Input: $G = (L, R, E)$
Output: matching $M_{OKS}(G)$

---

$\quad M_{OKS} \leftarrow \emptyset$
$\quad$**while** $E \neq \emptyset$ **do**
$\quad\quad$let $v = \underset{w \in R}{\arg\min}\, \deg(w)$
$\quad\quad$**if** $\deg(v) = 0$ **then**
$\quad\quad\quad G \leftarrow G - \{v\}$
$\quad\quad$**else if** for $u \in L$, $(u, v) \in E$ **then**
$\quad\quad\quad G \leftarrow G - \{u, v\}$
$\quad\quad\quad M_{OKS} \leftarrow M_{OKS} \cup \{(u, v)\}$
$\quad\quad$**end if**
$\quad$**end while**
$\quad$**return** $M_{OKS}$

---

## IV. NUMERICAL EXAMPLES

In this section, we present the results of selected numerical analyses on real world networks. We study the number of controllers, or equivalently the number of unmatched vertices for 10 different networks. This provides support to studying $DD$ random networks as a realistic model for control applications. In other words, using Fig. 3, to find the number of controllers needed to structurally control a network, one can assume real networks are in fact $DD$ random networks. The full description of these networks can be found in the Supplementary Materials.

Table I contains the results for 10 networks, including social, internet, web, electronic, neuronal, power grid and transcriptional regulatory networks, enumerated by the first row of the table. The second (third and fourth respectively) row is the number of controllers needed for structural controllability of the corresponding network if $OKS$ ($KS$ and Maximum Matching respectively) algorithm will be used.

The fifth row shows the average number of controllers for a random network generated by degree distribution model for random networks, i.e. input and output degrees of all vertices are the same as the original real-world network but the attachment of half-edges is random. Finally, the last row of the table shows the size of the networks. As seen in the table, $OKS$ and $KS$ perform very close to Maximum Matching for all networks. Moreover, network 9 is the only one for which the performance of $OKS$ is significantly different than $KS$. We suspect that this is because network 9 is the only undirected network.

The numerical results in Table I can be understood better by the following figures. Fig. 2 shows the performance of $OKS$ and $KS$ versus Maximum Matching, i.e. rows 2,3

| | Networks | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Algorithms | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| $|V|-|M_{OKS}|$ | 6 | 51 | 5033 | 703 | 10 | 29 | 59 | 119 | 610 | 565 |
| $|V|-|M_{KS}|$ | 6 | 49 | 5033 | 702 | 9 | 29 | 59 | 119 | 577 | 565 |
| $|V|-|M^*|$ | 6 | 49 | 5033 | 702 | 9 | 29 | 59 | 119 | 575 | 565 |
| average of $|V|-|M_{OKS}|$ for random network | 5.6 | 29.8 | 5006.8 | 616.5 | 10.1 | 24.7 | 48.6 | 100.5 | 454.9 | 557.7 |
| $|V|=n$ | 32 | 297 | 8717 | 1490 | 67 | 122 | 252 | 512 | 4941 | 688 |

TABLE I: The number of controllers given by different algorithms for different networks as well as the average number of controllers for equivalent degree distribution random network and the size of the networks.
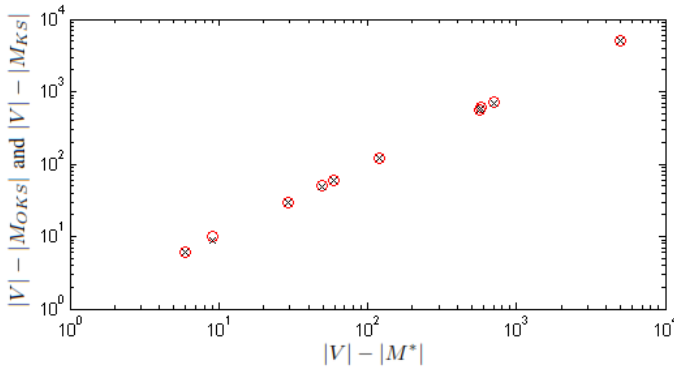


Fig. 2: The number of controllers given by $OKS$ ( $\circ$ ) and $KS$ ( $\times$ ) versus the number of controllers given by Maximum Matching for 10 different real networks.
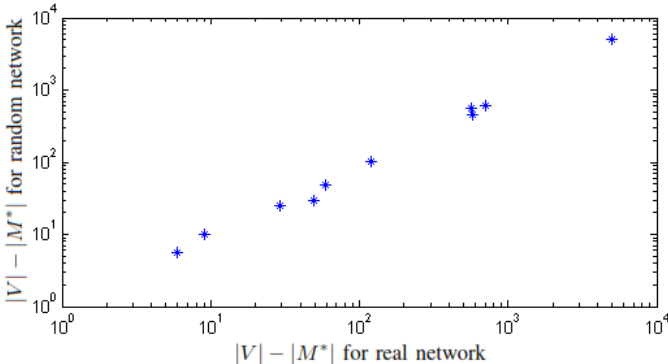


Fig. 3: The number of controllers for randomly rewired networks versus the number of controllers for the original networks.

of the Table I versus row 4. The similarity in performance between $OKS, KS$ and maximum matching algorithms is better depicted in Fig. 2.

The number of controllers for equivalent degree distribution random network versus the original network, i.e. row 5 of the Table I versus row 4 is shown in Fig. 3. According to this plot, degree distribution random networks are sufficiently realistic models for real-world networks in terms of the number of controllers needed for structural controllability.

## V. MATCHING ALGORITHMS IN RANDOM NETWORKS

In this section, we present results about the asymptotic size of matchings produced by the algorithms presented above. We first consider the general case where the asymptotic degree

distribution of the random network is any arbitrary degree distribution with finite mean. Then, more detailed results regarding the special case for a Poisson asymptotic empirical degree distribution will be provided. Even though as seen before, the latter case is not common among real networks, because of classical interests on $ER$ and $UFS$ random networks and the $KS$ algorithm, as well as comprehensiveness, we also study it.

Before proceeding with the analysis of matching algorithms in an extensive class of random networks, we must ensure that the size of the matchings provided by either Greedy, KS, OKS algorithms or any *maximum* matching algorithm has the Lipschitz property in order to have convergence of $\frac{|M_G(G)|}{n}, \frac{|M_{KS}(G)|}{n}, \frac{|M_{OKS}(G)|}{n}$ and $\frac{|M^*(G)|}{n}$ for random network $G$ where $M^*(G)$ is a maximum matching of network $G$. The following lemma establishes the desired Lipschitz property. The size of the matching provided by any of the above algorithms has the Lipschitz property due to the recursive nature of the algorithms. The Lipschitz property for the size of maximum matchings comes from their maximality regardless of the algorithm used to obtain the maximum matching (this follows easily from the definition of maximum matching).

**Lemma 2.** *The real-valued functions $|M_G|, |M_{KS}|, |M_{OKS}|$ and $|M^*|$ which are the size of matchings provided by Greedy, KS, OKS and maximum matching algorithms, respectively, have the Lipschitz property.*

### A. Arbitrary Degree Distribution

We establish the optimality of $OKS$ algorithm which immediately yields optimality of $KS$ as well, for reasons explained before. For this purpose, we follow in the footsteps of Karp and Sipser [22] and embed the dynamics of both input and output degree sequences as the algorithm proceeds in continuous time. This embedding provides differential equations governing the degree sequence vectors. However, in the general degree distribution case, unlike the classic Erdos-Renyi case, the differential equations are defined in arbitrarily high dimensions. So there is little hope of working in fixed small dimension as Karp and Sipser [22] did (their differential equations were 3 dimensional) and new ideas are needed. The key idea in our proof is to use the differential equations to show that the number of iterations when there is no degree one vertex (and so the algorithm can possibly make a mistake) is sublinear (in $n$) which means the fraction of unmatched vertices (or equivalently the relative size of the number of controllers) given by $OKS$ is asymptotically the same as that

of maximum matching. Finally, a set of equations for the relative size of maximum matching according to asymptotic empirical input and output degree distributions is provided.

**Theorem 2** (Asymptotic optimality of OKS algorithm). *For network $G = (L, R, E), |R| = |L| = n$ let $|M_{OKS}(G)|$ and $|M^*(G)|$ be the size of matching given by OKS algorithm and the size of maximum matching respectively. Let $G$ be either ER, UFS or DD random network with finite average degree, i.e. $\lim_{n \to \infty} \frac{|E|}{n} = \sum_{i=0}^{\infty} i p_{in}(i) = \sum_{i=0}^{\infty} i p_{out}(i) < \infty$ (where $p_{in}, p_{out}$ are asymptotic empirical degree distributions). Then*

$$\lim_{n \to \infty} \frac{|M_{OKS}(G)|}{n} = \lim_{n \to \infty} \frac{|M^*(G)|}{n}.$$

**Remark 2.** *Under the assumptions of Theorem 2, one can show the asymptotic optimality of KS as well: $\lim_{n \to \infty} \frac{|M_{KS}(G)|}{n} = \lim_{n \to \infty} \frac{|M^*(G)|}{n}$. We omit the proof here as it is very similar to the proof of Theorem 2.*

Note that in Theorem 1 letting $\epsilon = n^{-r}$ for every $r > \frac{1}{2}$ the convergence holds. So the difference between $|M_{OKS}|$ and $|M^*|$ is $O(\sqrt{n})$. Now the following questions arise: $(i)$ what is the size of maximum matching? $(ii)$ how can we compute the answer (asymptotically) without running the algorithm? The following theorem gives the size of maximum matching in terms of input and output degree distributions. For $u \in [0, 1)$ define moment generating functions:

$$\Phi_{in}(u) = \sum_{i=0}^{\infty} p_{in}(i) u^i, \Phi_{out}(u) = \sum_{i=0}^{\infty} p_{out}(i) u^i,$$

$$\phi_{in}(u) = \frac{1}{\mu} \Phi'_{in}(u) = \sum_{i=1}^{\infty} \frac{i p_{in}(i)}{\mu} u^{i-1},$$

$$\phi_{out}(u) = \frac{1}{\mu} \Phi'_{out}(u) = \sum_{i=1}^{\infty} \frac{i p_{out}(i)}{\mu} u^{i-1},$$

where $\mu = \sum_{i=0}^{\infty} i p_{in}(i) = \sum_{i=0}^{\infty} i p_{out}(i)$ is the average degree.

**Theorem 3** (Size of Maximum Matching). *For (either ER, UFS or DD) random network $G = (L, R, E), |R| = |L| = n$ if $\mu < \infty$ let $U^*$ be*

$$
\begin{aligned}
U^* &= \frac{1}{2} \Big[ \Phi_{in}(1 - w_1) + \Phi_{in}(w_2) + \Phi_{out}(1 - w_3) - 2 \\
&+ \Phi_{out}(w_4) + \mu (w_3(1 - w_2) + w_1(1 - w_4)) \Big]
\end{aligned}
$$ (1)

*where $(w_1, w_2, w_3, w_4) \in [0, 1)^4$ is the smallest solution of*

$$\phi_{out}(1 - w_3) = 1 - w_2 \quad, \quad \phi_{in}(w_2) = w_3,$$

$$\phi_{in}(1 - w_1) = 1 - w_4 \quad, \quad \phi_{out}(w_4) = w_1,$$

*then the asymptotic fraction of unmatched vertices is $U^*$:*

$$\lim_{n \to \infty} 1 - \frac{|M^*(G)|}{n} = U^*.$$

Note that the formula (1) first appeared in Liu et al.'s work [12] and was supported by numerical experiments. The result above formally proves its validity.

## B. Poisson Degree Distribution

We first present selected asymptotic and non-asymptotic results about the fraction of matched vertices for the matching given by Greedy. Subsequently we generalize the results provided by Karp and Sipser [22] on the performance of $KS$.

Studying a simple algorithm (Greedy) on a simple random network $G$ (drawn from directed $ER$ where every edge $i \to j$ exists with probability $p$) allows us to obtain explicitly the *non-asymptotic* probability mass function for $|M_G(G)|$. The following theorem also provides the asymptotic behavior of $|M_G(G)|$ for directed $ER$.

**Theorem 4** (Greedy for directed $ER$). *Let the network $G$ be directed $ER$ of size $n$. Then:*

$$\mathbb{P}(|M_G(G)| = n - k) = \frac{\alpha_n(q)^2}{\alpha_k(q)^2 \alpha_{n-k}(q)} q^{k^2},$$

*where $q = 1 - p$ and $\alpha_i(q) = \prod_{j=1}^{i} (1 - q^j)$. For $ER(\lambda)$ (i.e. $np \to \lambda$) if $\lambda = 0$ then $\lim_{n \to \infty} \frac{|M_G(G)|}{n} = 0$. If $\lambda = \infty$ then $\lim_{n \to \infty} \frac{|M_G(G)|}{n} = 1$. For $\lambda \in (0, \infty)$, $|M_G(G)|$ is asymptotically normal:*

$$\mathcal{N} \left( n \frac{\lambda - \log(2 - e^{-\lambda})}{\lambda}, n \frac{1}{4\lambda} \right).$$

Some of the results in Theorem 4 remain valid for a larger class of random networks including directed and undirected networks.

**Theorem 5** (Greedy for asymptotically Poisson degree distributions). *Assume $G$ is one of $ER(\lambda)$, $UFS(\lambda)$, $DD(p)$ and $DD(p, p)$ where probability distribution $p$ is $\text{Poisson}(\lambda)$. If $\lambda = 0$ (respectively $\lambda = \infty$) then $\lim_{n \to \infty} \frac{|M_G(G)|}{n} = 0$ (respectively 1). For $\lambda \in (0, \infty)$, $\lim_{n \to \infty} \frac{|M_G(G)|}{n} = 1 - \frac{\log(2 - e^{-\lambda})}{\lambda}$.*

Similar results hold for $KS$. In fact, Karp and Sipser [22] proved that, for the classical undirected Erdos-Renyi random network (denoted by undirected $ER(\lambda), \lambda \in (0, \infty)$ here), $KS$ is optimal. They split the running of the algorithm into two phases. Phase 1 begins when the algorithms starts and finishes the first time there is no vertex of degree one in the network, when phase 2 starts and proceeds until the algorithm removes all edges from the network. For network $G$ let $U(G), U_1(G)$ and $U_2(G)$ be the number of vertices left unmatched (became of degree zero before being removed from the network) when running maximum matching, phase 1 and phase 2 respectively, $H = (V', E')$ be the remaining network at the beginning of phase 2. Hence $|M^*(G)| = n - U(G), |M_{KS}(G)| = n - U_1(G) - U_2(G), |M_{KS}(H)| = |V'| - U_2(G)$. Since there is no deviation from maximum matching as long as vertices of degree one exists, we have $U_1(G) \leq U(G) \leq U_1(G) + U_2(G)$. Karp and Sipser show $\frac{U_2(G)}{n} \to 0$ as $n \to \infty$ so the algorithm is optimal, i.e. $\lim_{n \to \infty} \frac{|M_{KS}(G)|}{n} = \lim_{n \to \infty} \frac{|M^*(G)|}{n}$. Further, they show $\frac{U_1(G)}{n} \to k(\lambda)$ and $\frac{|V'|}{n} \to h(\lambda)$ and find functions $k, h$ as $k(\lambda) = \frac{\gamma_* + \gamma^* + \gamma_* \gamma^*}{\lambda} - 1, h(\lambda) = \frac{(1 - \gamma_*)(\gamma^* - \gamma_*)}{\lambda}$ where $\gamma_*$ is the smallest root of $\gamma = \lambda \exp(-\lambda e^{-\gamma})$ and $\gamma^* = \lambda e^{-\gamma_*}$. For $\lambda \leq e$ we have $h(\lambda) = 0$ because of $\gamma_* = \gamma^*$. In the following

theorem, we generalize these results to a larger class of random networks.

**Theorem 6** (KS for asymptotically Poisson degree distributions). *Assume G is one of $ER(\lambda)$, $UFS(\lambda)$, $DD(p)$ and $DD(p,p)$ where probability distribution $p$ is Poisson($\lambda$). If $\lambda = 0$ then $\lim_{n \to \infty} \frac{|M_{KS}(G)|}{n} = \lim_{n \to \infty} \frac{|M^*(G)|}{n} = 0$. If $\lambda = \infty$ then $\lim_{n \to \infty} \frac{|M_{KS}(G)|}{n} = \lim_{n \to \infty} \frac{|M^*(G)|}{n} = 1$. For $\lambda \in (0, \infty)$, we have*

$$\lim_{n \to \infty} \frac{|M_{KS}(G)|}{n} = \lim_{n \to \infty} \frac{|M^*(G)|}{n} = 1 - k(\lambda).$$

*Furthermore, $\frac{U_2(G)}{n} \to 0$ , $\frac{U_1(G)}{n} \to k(\lambda)$ and $\frac{|V'|}{n} \to h(\lambda)$ as $n \to \infty$.*

**Remark 3.** *Note that the results in Theorem 6 are consistent with Theorem 3 as follows. Letting $p_{in}, p_{out}$ be Poisson($\lambda$), calculations show $U^* = k(\lambda)$ because of $\mu = \lambda$, $\Phi_{in}(u) = \Phi_{out}(u) = \phi_{in}(u) = \phi_{out}(u) = \exp(\lambda(u-1))$.*

Results presented in Theorems 5 and 6 are based on the fact that in all mentioned random networks, the asymptotic empirical degree distribution is *Poisson*.

## VI. CONCLUSION

We proved that the $OKS$ algorithm is (asymptotically) optimal for determining a set of vertices where controllers should be applied to. Indeed, first benefiting from the connection between structural controllability of networks and maximum matching problems (minimum inputs theorem) we introduced simple fast matching algorithms $OKS$ and $KS$. Further, using topologies extracted from real networks, we empirically showed that the minimum number of controllers for structural controllability heavily depends on the degree distribution of the network, which in turn implies that the assumption of a random network with specified degree distribution is reasonable for many real world networks. Finally, new proof techniques introduced in this study enable the rigorous analysis of the the performance of a class of fast matching algorithms for random networks.

Ruths and Ruths [32] showed that existing random network models, while capturing the key features for predicting the minimum number of controllers, are not predicting more detailed control profiles of real networks. This calls for the development of new random network models that match control profiles, and associated fast control algorithms.

## VII. ACKNOWLEDGEMENTS

## APPENDIX
### Proof of Theorem 2

The complete appendix including proofs of all established results can be found in the Supplementary Materials.

We prove the theorem in two steps. In Step 1, first we embed the dynamics of input and output degree sequences during the algorithm in a continuous time Markov process

for a random network with bounded degrees. Then, we find explicit expressions for differential equations governing the dynamics of degree sequences for infinitely large networks. Then, we show that for a finite random network, the dynamics of degree sequences can be approximated by the solution of the presented initial value problem. Finally, we show that this solution spends zero time in the cases where there is no vertex of degree one so the number of iterations of the algorithm there is no vertex of degree one on the right side of the random network is sublinear w.r.t. the size of the network. In Step 2, we generalize the proof to the unbounded, but finite mean, degree sequences.

**Step 1: Embedding in a continuous time Markov process.** Assume in addition that the asymptotic empirical degree distributions are bounded: $p_{in}(i) = p_{out}(i) = 0$ for $i > N$ i.e. for every vertex $v \in L \cup R$ we have $\deg(v) \leq N$. First, we embed the dynamics of the algorithm in a continuous time Markov process. To go to continuous time, define $G^n(t)$ as the $n$-vertex network at time $t \in \mathbb{R}$ where state changes $G^n = G^n - \{u,v\}$ occur at $\text{Exp}(n)$ interarrival times. More precisely, let $\tau_1, \tau_2, \ldots$ be i.i.d $\text{Exp}(n)$ random variables, i.e. the probability density function is $ne^{-nt}$ for $t \geq 0$ so $\mathbb{E}(\tau_i) = \frac{1}{n}$. The first state change $G^n = G^n - \{u,v\}$ occurs at time $t = \tau_1$, the second one occurs at $t = \tau_1 + \tau_2$ and so forth. Now we construct a Markov process on $\mathbb{R}^{2N}$ which describes the performance of the algorithm. The transition kernel of the Markov process will be described later. Define $X^{(n)}(t), Y^{(n)}(t) \in \mathbb{R}^N$ as:

$$X_k^{(n)}(t) = \frac{1}{n}\big|\{v \in R : \deg(v) = k \text{ in } G^n(t)\}\big|,$$

$$Y_k^{(n)}(t) = \frac{1}{n}\big|\{v \in L : \deg(v) = k \text{ in } G^n(t)\}\big|,$$

for $k = 1, 2, \ldots, N$. In addition let $m = m(X^{(n)}(t))$ be the minimum degree of vertices in $R(t)$: $m(X^{(n)}(t)) = \min\{k : X_k^{(n)}(t) \neq 0\}$ so letting $v_1 = v, u_1 = u$ whenever a state change occurs we have $\deg(v_1) = m$. Let $(u_i, v) \in E$ for $i = 1, \ldots, m$, $K = \deg(u_1)$ and $(u, v_j) \in E$ for $j = 1, \ldots, K$. So for a network of size $n$ we have the following conditional degree distributions for vertices $u_1, \ldots, u_m, v_2, \ldots, v_K$:

$$\mathbb{P}_n\left(\deg(u_i) = k | \mathcal{A}_{i-1}(t)\right) = \frac{nkY_k^{(n)}(t) - k\sum_{j=1}^{i-1} 1_{\deg(u_j) = k}}{n\sum_{k=1}^{N} kY_k^{(n)}(t) - \sum_{j=1}^{i-1}\deg(u_j)},$$

$$\mathbb{P}_n\left(\deg(v_i) = k | \mathcal{B}_{i-1}(t)\right) = \frac{nkX_k^{(n)}(t) - k\sum_{j=1}^{i-1} 1_{\deg(v_j) = k}}{n\sum_{k=1}^{N} kX_k^{(n)}(t) - \sum_{j=1}^{i-1}\deg(v_j)},$$

where $\mathcal{A}_i(t) = (\deg(u_1), \ldots, \deg(u_i), Y^{(n)}(t))$, $\mathcal{B}_i(t) = (\deg(v_1), \ldots, \deg(v_i), X^{(n)}(t))$. Note that since interarrival times are i.i.d exponential random variables, $X^{(n)}(t), Y^{(n)}(t)$ are continuous time Markov processes.

Letting $\tilde{X}^{(n)}, \tilde{Y}^{(n)} \in \mathbb{R}^N$ be the corresponding vectors after one state change for $x, y \in \mathbb{R}^N$ define functions $\mathbb{F}^n, \mathbb{G}^n : \mathbb{R}^{2N} \to \mathbb{R}^N$ as:

$$\mathbb{F}^n(x, y) = n\mathbb{E}_n(\tilde{X}^{(n)} - X^{(n)} | X^{(n)} = x, Y^{(n)} = y),$$

$$\mathbb{G}^n(x, y) = n\mathbb{E}_n(\tilde{Y}^{(n)} - Y^{(n)} | X^{(n)} = x, Y^{(n)} = y),$$

where $\mathbb{E}_n$ is expected value w.r.t $\mathbb{P}_n$. Since the process is Markov, probability distribution of $\tilde{X}^{(n)}, \tilde{Y}^{(n)}$ depends only on $X^{(n)}, Y^{(n)}$.

**Asymptotic initial value problem:** Define $\mathbb{P}(\deg(u_i) = k) = \frac{kY_k^{(n)}(t)}{\sum_{k=1}^N kY_k^{(n)}(t)}, \mathbb{P}(\deg(v_i) = k) = \frac{kX_k^{(n)}(t)}{\sum_{k=1}^N kX_k^{(n)}(t)}$. Note that $\mathbb{P}_n, \mathbb{P}$ can be defined for every $x, y \in \mathbb{R}^N$ with non-negative components. Now, for arbitrary $x, y$, some algebra gives:

$$|\mathbb{P}_n(\deg(u_i) = k | \mathcal{A}_{i-1}(t)) - \mathbb{P}(\deg(u_i) = k)| \leq \frac{C_1}{n}, \quad (2)$$

$$|\mathbb{P}_n(\deg(v_i) = k | \mathcal{B}_{i-1}(t)) - \mathbb{P}(\deg(v_i) = k)| \leq \frac{C_2}{n}. \quad (3)$$

For $C_1 = \frac{2N^2}{\sum_{k=1}^N ky_k}, C_2 = \frac{2N^2}{\sum_{k=1}^N kx_k}$. Define:

$$\mathbb{F}(x, y) = n\mathbb{E}(\tilde{X}^{(n)} - X^{(n)} | X^{(n)} = x, Y^{(n)} = y),$$

$$\mathbb{G}(x, y) = n\mathbb{E}(\tilde{Y}^{(n)} - Y^{(n)} | X^{(n)} = x, Y^{(n)} = y),$$

where $\mathbb{E}$ is expected value w.r.t $\mathbb{P}$. Since

$$n\tilde{X}_k^{(n)} = nX_k^{(n)} + \sum_{j=2}^K [1_{\deg(v_j)=k+1} - 1_{\deg(v_j)=k}] - 1_{k=m}, \quad (4)$$

$$n\tilde{Y}_k^{(n)} = nY_k^{(n)} + \sum_{j=2}^m [1_{\deg(u_j)=k+1} - 1_{\deg(u_j)=k}] - 1_{k=K}, \quad (5)$$

inequalities (2), (3) yield

$$\|\mathbb{F}^n(x, y) - \mathbb{F}(x, y)\|_1 \leq \frac{4N^4}{\sum_{k=1}^N kx_k} \frac{1}{n}, \quad (6)$$

$$\|\mathbb{G}^n(x, y) - \mathbb{G}(x, y)\|_1 \leq \frac{4N^4}{\sum_{k=1}^N ky_k} \frac{1}{n}, \quad (7)$$

and,

$$\mathbb{F}(x, y) = (\frac{\|A^2 y\|_1}{\|Ay\|_1} - 1)\frac{1}{\|Ax\|_1}(SAx - Ax) - 1_{m(x)},$$

$$\mathbb{G}(x, y) = -\frac{Ay}{\|Ay\|_1} + \frac{m(x) - 1}{\|Ay\|_1}(SAy - Ay),$$

where $A$ and $S$ are moment matrix and shift matrix respectively, i.e., $A, S \in \mathbb{R}^{N \times N}$, $A_{ij}$ is $i$ for $i = j$ and $0$ otherwise, and $S_{ij}$ is $1$ for $i = j - 1$ and $0$ otherwise, $m(x) = \min\{k : x_k \neq 0\}$ and $1_m \in \mathbb{R}^N$ is the vector in which $m$-th component is $1$ and all others are $0$. Note that $\|AX^{(n)}(t)\|_1 = \|AY^{(n)}(t)\|_1$ because for finite $n$ always

$|E(t)| = n\sum_{k=1}^N kX_k^{(n)}(t) = n\sum_{k=1}^N kY_k^{(n)}(t)$. Besides, transition kernel of the Markov process can be formulated by $\mathbb{P}_n$ according to (4), (5).

**Approximating the dynamics of the degree sequences by the solution of asymptotic initial value problem:** Now we can use *Kurtz's Theorem* [33]. Given functions $\mathbb{F}, \mathbb{G} : \mathbb{R}^{2N} \to \mathbb{R}^N$ and positive constant $T$, define $x(t), y(t) : [0, T] \to \mathbb{R}^N$ as the solution of the initial value problems

$$\dot{x} = \mathbb{F}(x, y), x_k(0) = p_{in}(k), k = 1, \ldots, N, \quad (8)$$

$$\dot{y} = \mathbb{G}(x, y), y_k(0) = p_{out}(k), k = 1, \ldots, N, \quad (9)$$

and let $\mathcal{E} = \{z \in \mathbb{R}^N \text{ such that: } \epsilon < \sum_{k=1}^N kz_k \leq N\}$. Suppose the following statements hold:

1) $\lim_{n \to \infty} \sup_{z_1, z_2 \in \mathcal{E}} \|\mathbb{F}^n(z_1, z_2) - \mathbb{F}(z_1, z_2)\|_1 = 0$.
2) $\lim_{n \to \infty} \sup_{z_1, z_2 \in \mathcal{E}} \|\mathbb{G}^n(z_1, z_2) - \mathbb{G}(z_1, z_2)\|_1 = 0$.
3) for all $k = 1, \ldots, N$, $\lim_{n \to \infty} X_k^{(n)}(0) = p_{in}(k)$.
4) for all $k = 1, \ldots, N$, $\lim_{n \to \infty} Y_k^{(n)}(0) = p_{out}(k)$.
5) functions $\mathbb{F}, \mathbb{G}$ are Lipschitz (in the classic sense).

Then

$$\lim_{n \to \infty} \mathbb{P}_n \left( \exists t \in [0, T] : m(X^{(n)}(t)) \neq m(x(t)) \right) = 0. \quad (10)$$

Letting $T = T(\epsilon) = \sup\{t : \sum_{k=1}^N kx_k(t) > \epsilon, \sum_{k=1}^N ky_k(t) > \epsilon\}$ for some arbitrary $\epsilon > 0$, the first two conditions are satisfied by (6), (7). By the definition of asymptotic empirical degree distributions $\lim_{n \to \infty} X_k^{(n)}(0) = p_{in}(k)$ and $\lim_{n \to \infty} Y_k^{(n)}(0) = p_{out}(k)$. On the other hand, the initial value problems (8), (9) have unique solutions since defining metric $d$ on $\mathbb{R}^N$ as $d(x, y) = \|x - y\|_1 + 1_{m(x) \neq m(y)}$, $\mathbb{F}, \mathbb{G}$ are Lipschitz with respect to this metric i.e. there is a $B < \infty$ such that for all $x, x', y, y' \in \mathbb{R}^N$

$$d(\mathbb{F}(x, y), \mathbb{F}(x', y')) < B(d(x, x') + d(y, y')),$$

$$d(\mathbb{G}(x, y), \mathbb{G}(x', y')) < B(d(x, x') + d(y, y')).$$

Note that stopping time at $T(\epsilon)$, i.e. when $\epsilon n$ edges are remaining in the network to be removed by the algorithm, will not cause any problem since continuing the algorithm from that point on cannot add more than $\epsilon$ edges to the matching on a scale relative to the number $n$ of vertices.

**Properties of the asymptotic initial value problems:** The useful fact about the solutions of (8), (9) is that Lebesgue measure of the set $\{0 < t < T : m(x(t)) > 1\}$ is zero. Suppose it is not. So there are $0 < t_1 < t_2$ such that $x_1(t) = 0$ for all $t \in [t_1, t_2]$ so $\frac{dx_1(t)}{dt} = 0$ for all $t \in (t_1, t_2)$. But

$$\frac{dx_1(t)}{dt} = \mathbb{F}_1(x, y) = (\frac{\|A^2 y\|_1}{\|Ay\|_1} - 1)\frac{1}{\|Ax\|_1}(2x_2(t) - x_1(t))$$

implies $x_2(t) = 0$ for all $t \in (t_1, t_2)$ which means $m(x(t)) > 2$ for all $t \in (t_1, t_2)$. Repeating this argument for $x_2$ now we will get $m(x(t)) > 3$ and so on. Therefore, $m(x(t)) > N$ which is impossible. Thus, the set $\{0 < t < T : m(x(t)) > 1\}$ has zero Lebesgue measure.

**Sublinearity of the number of iterations of the algorithm with no degree one vertex:** Let $J^{(n)} \subset \{1, 2, \ldots, n\}$ be the set of indices $i$ of iterations of $OKS$ such that after the $i$-th iteration the minimum degree on the right side of the network is larger than one. Since the set $\{0 < t < T : m(x(t)) > 1\}$ has zero Lebesgue measure by (10) Lebesgue measure of the set $\{0 < t < T : m(X^{(n)}(t)) > 1\}$ goes to 0 as $n$ grows. Because the Lebesgue measure of the set $\{0 < t < T : m(X^{(n)}(t)) > 1\}$ is $\sum_{i \in J^{(n)}} \tau_{i+1}$ we have $\lim_{n \to \infty} \sum_{i \in J^{(n)}} \tau_{i+1} = 0$ but by the Law of Large Numbers $\lim_{n \to \infty} \frac{1}{|J^{(n)}|} \sum_{i \in J^{(n)}} n\tau_{i+1} = \mathbb{E}(n\tau_1) = 1$. Therefore $\lim_{n \to \infty} \frac{|J^{(n)}|}{n} = \lim_{n \to \infty} \frac{|J^{(n)}|}{n} \frac{1}{|J^{(n)}|} \sum_{i \in J^{(n)}} n\tau_{i+1} = \lim_{n \to \infty} \sum_{i \in J^{(n)}} \tau_{i+1} = 0$ i.e. the number of iterations of $OKS$ algorithm for which there is no vertex of degree one on the right side of the network is sublinear w.r.t. the size of the network.

**Sublinearity of difference between the output of $OKS$ and maximum matching:** Using $\lim_{n \to \infty} \frac{|J^{(n)}|}{n} = 0$ we prove that the size of the matching provided by the $OKS$ algorithm is away from maximum matching size by a sublinear factor. Starting the algorithm, as long as the minimum degree on the right side of the network is one, $OKS$ makes no mistake, i.e. the size of the matching by $OKS$ is the same as the size of maximum matching. When the minimum degree is $m = m(X^{(n)}(t)) > 1$ it is possible that $OKS$ picks a vertex on the left side which is not the optimal choice. We make it optimal by manipulating the network: if $v \in R, u \in L, \deg(v) = m$ are the chosen vertices in the iteration of the algorithm to be removed from the network, $M_{OKS} = M_{OKS} \cup \{(u,v)\}$, manipulate the network by removing all other $m - 1$ edges connected to $v$. Since $|M^*|$ has the Lipschitz property, removing these $m - 1$ edges will change the size of the maximum matching by at most $m - 1$. Since $m$ is the minimum degree and the average degree is bounded, $m - 1$ is bounded as well. On the other hand, the number of iterations that $OKS$ will face such cases is sublinear w.r.t. the size of the network, so the whole number of possible errors, or in other words, the whole deviation from maximum matching made by $OKS$ is sublinear, i.e. $\lim_{n \to \infty} \frac{|M_{OKS}(G)|}{n} = \lim_{n \to \infty} \frac{|M^*(G)|}{n}$.

**Step 2: Generalization to unbounded degree.** Now to generalize the proof to cases where the asymptotic empirical degree distributions are not bounded, we use the classical technique of truncation. For arbitrary $\epsilon > 0$, let $N$ be large enough such that $\sum_{k=N+1}^{\infty} k p_{in}(k) < \frac{\epsilon}{2}$, $\sum_{i=N+1}^{\infty} k p_{out}(k) < \frac{\epsilon}{2}$. In random network $G$ remove some edges in order to have no vertex of degree larger than $N$ to get random network $H$ which has bounded asymptotic empirical degree distributions. By Step 1,

$$\lim_{n \to \infty} \frac{|M_{OKS}(H)|}{n} = \lim_{n \to \infty} \frac{|M^*(H)|}{n}. \tag{11}$$

Because by Lemma 2 both functions $|M^*|, |M_{OKS}|$ have the Lipschitz property and asymptotically the number of edges removed from $G$ to get $H$ is less than $n\epsilon$:

$$\lim_{n \to \infty} \frac{\left||M^*(H)| - |M^*(G)|\right|}{n} < \epsilon, \tag{12}$$

$$\lim_{n \to \infty} \frac{\left||M_{OKS}(H)| - |M_{OKS}(G)|\right|}{n} < \epsilon. \tag{13}$$

Now (11), (12), (13) imply the desired result.

## References

[1] C.-T. Lin, "Structural controllability," *Automatic Control, IEEE Transactions on*, vol. 19, no. 3, pp. 201–208, 1974.

[2] I. Rajapakse, M. Groudine, and M. Mesbahi, "What can systems theory of networks offer to biology?" *PLoS computational biology*, vol. 8, no. 6, p. e1002543, 2012.

[3] R. Srikant, *The mathematics of Internet congestion control*. Springer Science & Business Media, 2004.

[4] E. Dinits, "Algorithm of solution to problem of maximum flow in network with power estimates," *Doklady Akademii Nauk SSSR*, vol. 194, no. 4, p. 754, 1970.

[5] L. R. Ford and D. R. Fulkerson, "Maximal flow through a network," *Canadian journal of Mathematics*, vol. 8, no. 3, pp. 399–404, 1956.

[6] A. V. Goldberg and R. E. Tarjan, "A new approach to the maximum-flow problem," *Journal of the ACM (JACM)*, vol. 35, no. 4, pp. 921–940, 1988.

[7] T. B. Crabill, D. Gross, and M. J. Magazine, "A classified bibliography of research on optimal design and control of queues," *Operations Research*, vol. 25, no. 2, pp. 219–232, 1977.

[8] L. Tadj and G. Choudhury, "Optimal design and control of queues," *Top*, vol. 13, no. 2, pp. 359–412, 2005.

[9] C. H. Papadimitriou and J. N. Tsitsiklis, "The complexity of optimal queuing network control," *Mathematics of Operations Research*, vol. 24, no. 2, pp. 293–305, 1999.

[10] J. Nino-Mora, "Dynamic allocation indices for restless projects and queueing admission control: a polyhedral approach," *Mathematical programming*, vol. 93, no. 3, pp. 361–413, 2002.

[11] D. G. Luenberger, *Introduction to dynamic systems*. John Wiley & Sons New York, 1979.

[12] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.

[13] I. Rajapakse, D. Scalzo, and M. Groudine, "Losing control: Cancer's catastrophic transition," *Nucleus*, vol. 2, no. 4, pp. 249–252, 2011.

[14] J.-M. Dion, C. Commault, and J. Van Der Woude, "Generic properties and control of linear structured systems: a survey," *Automatica*, vol. 39, no. 7, pp. 1125–1144, 2003.

[15] A. Gibbons, *Algorithmic graph theory*. Cambridge University Press, 1985.

[16] B. Bollobás, "Random graphs, volume 73 of cambridge studies in advanced mathematics," 2001.

[17] A. Frieze and P. Melsted, "Maximum matchings in random bipartite graphs and the space utilization of cuckoo hash tables," *Random Structures & Algorithms*, vol. 41, no. 3, pp. 334–364, 2012.

[18] J. Aronson, A. Frieze, and B. G. Pittel, "Maximum matchings in sparse random graphs: Karp-Sipser revisited," *Random Structures and Algorithms*, vol. 12, no. 2, pp. 111–177, 1998.

[19] T. Tao and V. Vu, "Random matrices have simple spectrum," *arXiv preprint arXiv:1412.1438*, 2014.

[20] T. Bohman and A. Frieze, "Karp-Sipser on random graphs with a fixed degree sequence," *Combinatorics, Probability and Computing*, vol. 20, no. 05, pp. 721–741, 2011.

[21] P. Balister and S. Gerke, "Controllability and matchings in random bipartite graphs," *Surveys in Combinatorics 2015*, vol. 424, p. 119, 2015.

[22] R. M. Karp and M. Sipser, "Maximum matching in sparse random graphs," in *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. IEEE, 1981, pp. 364–375.

[23] A. Olshevsky, "Minimal controllability problems," *Control of Network Systems, IEEE Transactions on*, vol. 1, no. 3, pp. 249–258, 2014.

[24] C. Commault, J.-M. Dion, and J. W. van der Woude, "Characterization of generic properties of linear structured systems for efficient computations," *Kybernetika*, vol. 38, no. 5, pp. 503–520, 2002.

[25] S. Assadi, S. Khanna, Y. Li, and V. M. Preciado, "Complexity of the minimum input selection problem for structural controllability," *IFAC-PapersOnLine*, vol. 48, no. 22, pp. 70–75, 2015.

[26] C. Commault and J.-M. Dion, "The single-input minimal controllability problem for structured systems," *Systems & Control Letters*, vol. 80, pp. 50–55, 2015.

[27] S. Pequito, S. Kar *et al.*, "A framework for structural input/output and control configuration selection in large-scale systems," 2013.

[28] N. J. Cowan, E. J. Chastain, D. A. Vilhena, J. S. Freudenberg, and C. T. Bergstrom, "Nodal dynamics, not degree distributions, determine the structural controllability of complex networks," *PloS one*, vol. 7, no. 6, p. e38398, 2012.

[29] S. Micali and V. V. Vazirani, "An $O(|V|^{0.5}|E|)$ algorithm for finding maximum matching in general graphs," in *Foundations of Computer Science, 1980., 21st Annual Symposium on*. IEEE, 1980, pp. 17–27.

[30] R. Durrett, *Random graph dynamics*. Cambridge university press Cambridge, 2007, vol. 200, no. 7.

[31] T. H. Cormen and C. Charles, "C. e, leiserson, and rl rivest, introduction to algorithms," 1990.

[32] J. Ruths and D. Ruths, "Control profiles of complex networks," *Science*, vol. 343, no. 6177, pp. 1373–1376, 2014.

[33] T. G. Kurtz, "Solutions of ordinary differential equations as limits of pure jump Markov processes," *Journal of applied Probability*, vol. 7, no. 1, pp. 49–58, 1970.

**George Michailidis** (M05) received the Ph.D. degree in mathematics from the University of California, Los Angeles, CA, USA, in 1996. He was a Post-Doctoral Fellow with the Department of Operations Research, Stanford University, Stanford, CA, from 1996 to 1998. He joined the University of Michigan, Ann Arbor, MI, USA, in 1998, where he spent 17 years a faculty members in Statistics, Electrical Engineering, and Computer Science. He is currently the Director of the Informatics Institute at the University of Florida and a faculty in the Departments of Statistics and Computer and Information Science and Engineering. He is a Fellow with the Institute of Mathematical Statistics, American Statistical Association, and the International Statistical Institute. His current research interests include stochastic network modeling and performance evaluation, queuing analysis and congestion control, statistical modeling and analysis of internet traffic, network tomography, and analysis of high dimensional data with network structure. Prof. Michailidis served as the Editor-in-Chief of the Electronic Journal of Statistics from 2013-2015 and serves on a number of editorial boards. He served as Symposium Chair for the Support for Storage, Renewable Sources, and Microgrids Symposium for SmartGridComm 2012 and is the Secretary of the IEEE Subcommittee on SmartGrid Communications.

**Mohamad Kazem Shirani Faradonbeh** received the B.S. degree in electrical engineering from Sharif University of Technology, Tehran, Tehran, Iran, in 2012. He is currently a Ph.D. student in the Department of Statistics, University of Michigan, Ann Arbor, MI, USA. His research interests include statistical analysis of random networks, adaptive network control, and optimal control of high dimensional networked structures.

**Ambuj Tewari** is an assistant professor in the Department of Statistics and the Department of Electrical Engineering and Computer Science (by courtesy) at the University of Michigan, Ann Arbor, MI, USA. He is also affiliated with the Michigan Institute for Data Science (MIDAS). He obtained his Ph.D. degree under the supervision of Peter Bartlett at the University of California at Berkeley. His research interests lie in machine learning including statistical learning theory, online learning, reinforcement learning and control theory, network analysis, and optimization for machine learning. He collaborates with scientists to seek novel applications of machine learning in mobile health, learning analytics, and computational chemistry. His research has been recognized with paper awards at COLT 2005, COLT 2011, and AISTATS 2015. He was the recipient of an NSF CAREER award in 2015.