Deterministic Decentralized Search in Random Graphs

Esteban Arcaute[1,⋆], Ning Chen[2,⋆], Ravi Kumar[3], David Liben-Nowell[4,⋆], Mohammad Mahdian[3], Hamid Nazerzadeh[1,⋆], and Ying Xu[1,⋆]

[1] Stanford University. Email: {arcaute,hamidnz,xuying}@stanford.edu
[2] University of Washington. Email: ning@cs.washington.edu
[3] Yahoo! Research. Email: {ravikuma,mahdian}@yahoo-inc.com
[4] Carleton College. Email: dlibenno@carleton.edu

Abstract. We study a general framework for decentralized search in random graphs. Our main focus is on deterministic memoryless search algorithms that use only local information to reach their destination in a bounded number of steps in expectation. This class includes (with small modifications) the search algorithms used in Kleinberg's pioneering work on long-range percolation graphs and hierarchical network models. We give a characterization of searchable graphs in this model, and use this characterization to prove a monotonicity property for searchability.

1 Introduction

Since Milgram's famous "small world" experiment [14], it has generally been understood that social networks have the property that a typical node can reach any other node through a short path (the so-called "six degrees of separation"). An implication of this fact is that social networks have small diameter. Many random graph models have been proposed to explain this phenomenon, often by showing that adding a small number of random edges causes a highly structured graph to have a small diameter (e.g., [3,16]). A stronger implication of Milgram's experiment, as Kleinberg observed [8], is that for most social networks there are decentralized search algorithms that can *find* a short path from a source to a destination without a global knowledge of the graph. As Kleinberg proved, even many of the random graph models with small diameter do not have this property (i.e., any decentralized search algorithm in such graphs can take many steps to reach the destination), while in certain graph models with a delicate balance of parameters, decentralized search is possible. Since Kleinberg's work, there have been many other models that provably exhibit the searchability property [5,7,9,10,12,15]; however, we still lack a good understanding of what contributes to this property in graphs.

In this paper, we look at a general framework for searchability in random graphs. We consider a general random graph model in which the set of edges

⋆ Work performed in part while visiting Yahoo! Research.

leaving a node u is independent of that of any other node $v \neq u$. This framework includes models such as the directed variant of the classical Erdős–Rényi graphs [6], random graphs with a given expected degree sequence (e.g., [4]), long-range percolation graphs [8], hierarchical network models [9], and graphs based on Kronecker products [11,13], but not models such as preferential attachment [2] in which the distribution of edges leaving a node is dependent on the other edges of the graph. It is worth noting that, in a random graph model where edges can have arbitrary dependencies, the search problem includes arbitrarily difficult learning problems as special cases, and therefore one cannot expect to have a complete characterization of searchable graphs in such a model.

Throughout most of this paper, we restrict the class of decentralized search algorithms that we consider to deterministic memoryless algorithms that succeed in finding a path to the destination with probability 1. This is an important class of search algorithms, and includes the decentralized search algorithms used in Kleinberg's work on long-range percolation graphs and hierarchical network models. For this class, we give a simple characterization of graphs that are searchable in terms of a node ordering property. We will use this characterization to show a monotonicity property for searchability: if a graph is searchable in our model, it stays searchable if the probabilities of edges are increased.

The rest of this paper is organized as follows: Section 2 contains the description of the model. Section 3 presents a characterization of searchable random graphs. The monotonicity theorem is presented in Section 4.

2 The Model

Given a positive integer n and an $n \times n$ matrix $\mathbf{P}$ with entries $p_{i,j} \in [0,1]$, we define a directed random graph $G(n,\mathbf{P})$ with the node set $V = \{1,\ldots,n\}$ and with a directed edge connecting node i to node j with probability p_{ij}, independently of all other edges. As we will see later (Remark 1), our results hold for a more general random graph model where the edges originating from a node i can be dependent on each other but are independent of the edges leaving other nodes. However, for the sake of simplicity, we state and prove our results in the $G(n,\mathbf{P})$ model.

We fix two nodes $s,t \in V$ of $G(n,\mathbf{P})$ as the *source* and the *destination*. For $v \in V$, let $\Gamma(v)$ denote the set of out-neighbors of u in G. We investigate the existence of a decentralized search algorithm that finds a path from s to t of at most a given length d in expectation.[1] We restrict our attention to *deterministic memoryless algorithms*. A deterministic memoryless algorithm can be defined as a partial function $A : V \times 2^V \to V$. Such an algorithm A defines a path $v_0, v_1, v_2, \ldots$ on a given graph G as follows: $v_0 = s$, and for every $i \geq 0$, $v_{i+1} =$

[1] Alternatively, we could ask for which graphs a decentralized search algorithm can find a path between *every* pair of nodes s and t, or between a *random* pair of nodes s and t. Our techniques apply to these alternative formulations of the problem as well. The only point that requires some care is that the orderings in the characterization theorem can depend on s and t.

$A(v_i, \Gamma(v_i))$. The length of this path is defined as the smallest integer i such that $v_i = t$. If no such i exists, we define the length of the path as infinity.

We are now ready to define the notion of searchability. For a given matrix $\mathbf{P}$, source and destination nodes s and t, and a number d, we say that $G(n, \mathbf{P})$ is *d-searchable* using a deterministic memoryless algorithm A if the expected length of the path defined by A on $G(n, \mathbf{P})$ is at most d. Note that this definition requires that the algorithm find a path from s to t with probability 1.

3 A Characterization of Searchable Random Graphs

In this section, we provide a complete characterization of searchable random graphs. We begin by defining a class of deterministic memoryless search algorithms parameterized by two orderings of V, and then prove that if a graph is d-searchable, it is also d-searchable using an algorithm from this narrow class.

Definition 1. *Let σ, π be two orderings (i.e., permutations) of the node set V. We define a deterministic memoryless algorithm $A_{\sigma,\pi}$ corresponding to these orderings as follows: for every $u \in V$, $A_{\sigma,\pi}(u, \Gamma(u))$ is defined as the maximum element according to π of the set $\{v \in \Gamma(u) : \sigma(v) > \sigma(u)\} \cup \{u\}$.*

In other words, algorithm $A_{\sigma,\pi}$ never goes backwards according to the ordering σ, and, subject to this restriction, makes the maximum possible progress according to π.

We are now ready to state our main theorem.

Theorem 1 *For a given probability matrix $\mathbf{P}$, source and destination nodes s and t, and number d, if $G(n, \mathbf{P})$ is d-searchable using a deterministic memoryless algorithm A, then there exist two orderings σ and π of V such that $G(n, \mathbf{P})$ is d-searchable by using $A_{\sigma,\pi}$.*

To prove this theorem, we will first construct the ordering σ using the structure of the search algorithm A. Next, we define an ordering π using σ. Finally, we use induction with respect to the ordering σ to show that the expected length of the path defined by $A_{\sigma,\pi}$ on $G(n, \mathbf{P})$ is not more than the one defined by A.

We assume, without loss of generality, that for every set $S \subseteq V$, $A(t, S) = t$. In other words, we assume that A never leaves t once it reaches this node.

Define a graph H with the node set V as follows: for every pair $u, v \in V$, the edge (u, v) is in H if and only if this edge is on the path from s to t defined by A on some realization of $G(n, \mathbf{P})$ (i.e., on some graph that has a non-zero probability in the distribution $G(n, \mathbf{P})$). We have the following important lemma.

Lemma 1. *The graph H is acyclic.*

Proof. Assume, for contradiction, that H contains a simple cycle C. Note that by the definition of H, if an edge (u, v) is in H, then u must be reachable from s in H. Therefore, every node of C must be reachable from s in H. Let v^* be a node in C that has the shortest distance from s in H, and $s = v_0, v_1, \ldots, v_\ell = v^*$

be a shortest path from s to v^* in H. Also, let $v^* = v_\ell, v_{\ell+1}, \ldots, v_k, v_{k+1} = v^*$ denote the cycle C. Therefore, $v_0, v_1, \ldots, v_k$ are all distinct nodes, and for every $i \in \{0, \ldots, k\}$, there is an edge from v_i to v_{i+1} in H.

By the definition of H, for every $i \in \{0, \ldots, k\}$, there is a realization of $G(n, \mathbf{P})$ in which A traverses the edge (v_i, v_{i+1}). This means that there is a realization of $G(n, \mathbf{P})$ in which the set $\Gamma(v_i)$ of out-neighbors of v_i is S_i^*, for some set S_i^* such that $A(v_i, S_i^*) = v_{i+1}$. Recall that in $G(n, \mathbf{P})$, all edges are present independently at random, and thus the random variables $\Gamma(u)$ are independent. Hence, since v_i's are all distinct and for each i, there is a realization satisfying $\Gamma(v_i) = S_i^*$, there must be a realization in which $\Gamma(v_i) = S_i^*$ for *all* i. In this realization, the algorithm A falls in the cycle C, and therefore will never reach t. Thus the path found by A in this realization is infinitely long, and therefore the expected length of the path found by A is infinite. This is a contradiction. □

By Lemma 1, we can find a topological ordering of the graph H. Furthermore, since by assumption t has no outgoing edge in H, we can find a topological ordering that places t last. Let σ be such an ordering, i.e., σ is an ordering of V such that (i) t is the maximum element of V under σ; (ii) for every edge (u, v) in H, we have $\sigma(v) > \sigma(u)$; and (iii) all nodes not in H precede s and are ordered arbitrarily, i.e., $\sigma(s) > \sigma(v)$ for any such node v. By the definition of H, these conditions mean that the algorithm A (starting from the node s) never traverses an edge (u, v) with $\sigma(u) > \sigma(v)$.

Given the ordering σ, we define numbers r_u for every $u \in V$ recursively as follows: $r_t = 0$, and for every $u \neq t$,

$$r_u = \begin{cases} 1 + \displaystyle\sum_{S \subseteq T_u, S \neq \emptyset} q_{u,S} \cdot \min_{v \in S}\{r_v\} & \text{if } q_{u,\emptyset} = 0 \\ \infty & \text{if } q_{u,\emptyset} > 0, \end{cases} \tag{1}$$

where $T_u := \{v : \sigma(v) > \sigma(u)\}$ and, for a set $S \subseteq T_u$, we write

$$q_{u,S} := \left(\prod_{v \in S} p_{uv}\right)\left(\prod_{v \in T_u \setminus S} (1 - p_{uv})\right)$$

to denote the probability that the subset of nodes of T_u that are out-neighbors of u is precisely S. Note that the above formula defines r_u in terms of r_v for $\sigma(v) > \sigma(u)$, and therefore the definition is well founded.

We can now define the ordering π as follows: let $\pi(u) > \pi(v)$ if $r_u < r_v$. Pairs u, v with $r_u = r_v$ are ordered arbitrarily by π.

The final step of the proof is the following lemma, which we will prove by induction using the ordering σ. To state the lemma, we need a few pieces of notation. For a search algorithm B, let $d(B, u)$ denote the expected length of the path that the algorithm B, started at node u, finds to t. Also, let V_0 denote the set of non-isolated nodes of H—i.e., V_0 is the set of nodes that the algorithm A (started from s) has a non-zero chance of reaching.

Lemma 2. *Let σ and π be the orderings defined as above. Then for every node $u \in V_0$, we have that $d(A, u) \geq d(A_{\sigma,\pi}, u) = r_u$.*

Proof (sketch). We prove this statement by induction on u, according to the ordering σ. The statement is trivial for $u = t$. We now show that for $u \in V_0 \setminus \{t\}$ if the statement holds for every node $v \in V_0$ with $\sigma(v) > \sigma(u)$, then it also holds for u. Observe that for any deterministic memoryless algorithm B,

$$d(B, u) = 1 + \sum_{S \subset V, S \neq \emptyset} q'_{u,S} \cdot d(B, B(u, S)), \tag{2}$$

where $q'_{u,S} := (\prod_{v \in S} p_{uv})(\prod_{v \in V \setminus S}(1 - p_{uv}))$ is the probability that the set of out-neighbors of u in $G(n, \mathbf{P})$ is precisely S. This statement follows from the fact that the algorithm B is memoryless, and the fact that $q'_{u,\emptyset} = 0$ since $u \in V_0$. Applying Equation (2) to $A_{\sigma,\pi}$ and using the fact that, by definition, $A_{\sigma,\pi}(u, S)$ only depends on u and $S \cap T_u$, we obtain

$$d(A_{\sigma,\pi}, u) = 1 + \sum_{S \subseteq T_u, S \neq \emptyset} q_{u,S} \cdot d(A_{\sigma,\pi}, A_{\sigma,\pi}(u, S)). \tag{3}$$

We have that $d(A_{\sigma,\pi}, A_{\sigma,\pi}(u, S)) = r_{A_{\sigma,\pi}(u,S)}$ by the induction hypothesis. Also, by the definition of $A_{\sigma,\pi}$ and π, we have that $r_{A_{\sigma,\pi}(u,S)} = \min_{v \in S}\{r_v\}$. Combined with Equation (3) and the definition of r_u, this shows $d(A_{\sigma,\pi}, u) = r_u$, as desired.

To prove $d(A, u) \geq r_u$, note that since $A(u, S) \in S \cap T_u \cap V_0$, we have

$$d(A, A(u, S)) \geq \min_{v \in S \cap T_u \cap V_0}\{d(A, v)\}.$$

By the induction hypothesis, we have that $d(A, v) \geq r_v$ for every $v \in T_v \cap V_0$. Therefore, we have that $d(A, A(u, S)) \geq \min_{v \in S \cap T_u \cap V_0}\{r_v\}$. Substituting this in Equation (2), we obtain

$$\begin{aligned} d(A, u) &\geq 1 + \sum_{S \subset V, S \neq \emptyset} q'_{u,S} \cdot \min_{v \in S \cap T_u \cap V_0}\{r_v\} \\ &= 1 + \sum_{S \subseteq T_u, S \neq \emptyset} q_{u,S} \cdot \min_{v \in S \cap T_u \cap V_0}\{r_v\} \\ &\geq r_u. \end{aligned}$$

This completes the proof of the induction step. □

Proof (of Theorem 1). Define the graph H, the ordering σ, the values r_u, and the ordering π as above. By Lemma 2, we have that $d(A_{\sigma,\pi}, s) \leq d(A, s)$. Since $G(n, \mathbf{P})$ is d-searchable using A by assumption, we have that $d(A, s) \leq d$. Hence we have $d(A_{\sigma,\pi}, s) \leq d$, as desired. □

Remark 1. It is not hard to see that the only property of $G(n, \mathbf{P})$ that was used in the above proof was the fact that the random variables $\Gamma(u)$ (the set of out-neighbors of u) are independent. Therefore, the above proof (with minor modifications in the definitions of $q_{u,S}$ and $q'_{u,S}$) also works for a more general model of random graphs. This includes the directed ACL graphs [1] and the long-range percolation graphs.

Note that in the above proof, the second ordering π was defined in terms of the first ordering σ and $\mathbf{P}$. Therefore, the condition for the searchability of $G(n, \mathbf{P})$ can be stated in terms of only one ordering σ as follows:

Corollary 2 *$G(n, \mathbf{P})$ is d-searchable if and only if there is an ordering σ on the nodes for which $r_s \leq d$, where r is defined as in (1).*

It is not hard to see that even though the expression on the right-hand side of (1) has exponentially many terms, given σ, the value of r_u can be computed in polynomial time for every u. Therefore, the above corollary reduces the problem of d-searchability of $G(n, \mathbf{P})$ to a node-ordering property with a tractable objective function.

4 The Monotonicity Property

Armed with the characterization theorem of the previous section, we can now prove the following natural monotonicity property for searchability.

Theorem 3 *Let $\mathbf{P}$, $\mathbf{P}'$ be two $n \times n$ probability matrices such that for every i and j, we have $p_{ij} \leq p'_{ij}$. Fix the source and destination nodes s and t. Then, if $G(n, \mathbf{P})$ is d-searchable for some d, so is $G(n, \mathbf{P}')$.*

Proof (sketch). By Corollary 2, since $G(n, \mathbf{P})$ is d-searchable, there is an ordering σ such that the value r_s defined using Equation (1) is at most d. To show d-searchability of $G(n, \mathbf{P}')$, we apply the same ordering σ. Let $\{r'_u\}$ denote the values computed using Equation (1), but with $\mathbf{P}$ replaced by $\mathbf{P}'$. All we need to do is to show that $r'_s \leq d$ and then use Corollary 2. To do this, we prove by induction that for every $u \in V_0$, we have $r'_u \leq r_u$. This statement is trivial for $u = t$. We assume it is proved for every $v \in V_0$ with $\sigma(v) > \sigma(u)$, and prove it for u.

We have

$$\begin{aligned} r'_u &= 1 + \sum_{S \subseteq T_u, S \neq \emptyset} \prod_{v \in S} p'_{uv} \prod_{v \in T_u \setminus S} (1 - p'_{uv}) \cdot \min_{v \in S} \{r'_v\} \\ &\leq 1 + \sum_{S \subseteq T_u, S \neq \emptyset} \prod_{v \in S} p'_{uv} \prod_{v \in T_u \setminus S} (1 - p'_{uv}) \cdot \min_{v \in S} \{r_v\} \end{aligned}$$

Let $1, 2, \ldots, k$ be the nodes of T_u, ordered in such a way that $r_1 \geq r_2 \geq \cdots \geq r_k$. It is not hard to see that the above expression can be written as follows.

$$r'_u \leq 1 + r_1 - \sum_{i=1}^{k-1} \mathbf{Pr}_{G(n, \mathbf{P}')}[\Gamma(u) \cap \{i+1, \ldots, k\} \neq \emptyset] \cdot (r_i - r_{i+1})$$

The coefficient of $(r_i - r_{i+1})$ in the above expression is the probability of the event that the set of nodes that have an edge from u in $G(n, \mathbf{P}')$ contains at least one of the nodes $i+1, \ldots, k$. This event is monotone; therefore the probability of

this event under $G(n, \mathbf{P})$ is less than or equal to the probability under $G(n, \mathbf{P}')$. Therefore,

$$r'_u \leq 1 + r_1 - \sum_{i=1}^{k-1} \mathbf{Pr}_{G(n,\mathbf{P})}[\Gamma(u) \cap \{i+1, \ldots, k\} \neq \emptyset] \cdot (r_i - r_{i+1}).$$

This completes the proof of the induction step, since the right-hand side of the above inequality is precisely r_u. □

Note that, simple as the statement of Theorem 3 sounds, we do not know whether a similar statement holds for randomized memoryless algorithms. On the other hand, we proved the monotonicity property for randomized algorithms with memory; the proof can be found in [13].

Acknowledgments

We thank Amin Saberi for many useful discussions.

References

1. W. Aiello, F. Chung, and L. Lu. A random graph model for power law graphs. *Experimental Mathematics*, 10(1):53–66, 2001.
2. A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(15):509–512, 1999.
3. B. Bollobás and F. R. K. Chung. The diameter of a cycle plus a random matching. *SIAM Journal on Discrete Mathematics*, 1(3):328–333, 1988.
4. F. Chung and L. Lu. The average distance in a random graph with given expected degrees. *Internet Mathematics*, 1(1):91–114, 2003.
5. P. Duchon, N. Hanusse, E. Lebhar, and N. Schabanel. Could any graph be turned into a small world? *Theoretical Computer Science*, 355(1):96–103, 2006.
6. P. Erdős and A. Rényi. On random graphs I. *Publications Mathematics Debrecen*, 6:290–297, 1959.
7. P. Fraigniaud. Greedy routing in tree-decomposed graphs. In *Proceedings of the 13th Annual European Symposium on Algorithms*, pages 791–802, 2005.
8. J. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, pages 163–170, 2000.
9. J. Kleinberg. Small-world phenomena and the dynamics of information. *Advances in Neural Information Processing Systems*, 14:431–438, 2001.
10. R. Kumar, D. Liben-Nowell, and A. Tomkins. Navigating low-dimensional and hierarchical population networks. In *Proceedings of the 14th Annual European Symposium on Algorithms*, pages 480–491, 2006.
11. J. Leskovec, D. Chakrabarti, J. Kleinberg, and C. Faloutsos. Realistic, mathematically tractable graph generation and evolution, using Kronecker multiplication. *Proceedings of the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 133–145, 2005.

12. D. Liben-Nowell, J. Novak, R. Kumar, P. Raghavan, and A. Tomkins. Geographic routing in social networks. *Proceedings of the National Academy of Sciences*, 102(33):11623–11628, 2005.
13. M. Mahdian and Y. Xu. Stochastic Kronecker graphs. *Proceedings of the 5th Workshop on Algorithms and Models for the Web-Graph*, 2007.
14. S. Milgram. The small world problem. *Psychology Today*, 1:61–67, 1967.
15. A. Slivkins. Distance estimation and object location via rings of neighbors. In *Proceedings of the 16th ACM Symposium on Principles of Distributed Computing*, pages 41–50, 2005.
16. D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.