Improving the Scalability of Search in Networks Through Multiple Random Walks

Mark S. Squillante
Mathematical Sciences Department
IBM Thomas J. Watson Research Center
Yorktown Heights, NY 10598, USA
mss@us.ibm.com

Don Towsley and Sean Barker
Department of Computer Science
University of Massachusetts
Amherst, MA 01003, USA
{towsley, sbarker}@cs.umass.edu

1. INTRODUCTION

Efficient search for and location of content and resources is of fundamental importance in a wide variety of today's networks. These range from peer-to-peer (P2P) networks such as BitTorrent to online social networks such as Facebook, and from ad hoc wireless networks to wireline networks.

Three common search methods employed in such systems include flooding, random walks, and queries to third parties, such as Google. Flooding provides low search latencies, typically on the order of $\log n$, where n denotes the number of nodes in the system, but at the cost of high overhead; see, e.g., [6, 4]. Search based on a single random walk (RW) incurs low overhead, $O(1)$, but can yield a large search latency on the order of $O(n)$ when the content is poorly replicated [4]. Multiple RWs have been shown to reduce latencies (see, e.g., [6]), and adaptive techniques for selecting the number of RWs and the time-to-live (TTL) associated with each RW have been explored through simulation. However, there have been no formal analysis nor theoretical results that address these issues, both within the context of peer-to-peer systems and more generally. Last, search queries to third parties such as Google incur low overhead but at a cost that is very difficult to quantify, namely the loss of privacy.

We explore the use of multiple RWs for content search in a network where a failure occurs with very low probability and can be answered by a query to an external third party that always either has the content or knows where the content is located. We address the following fundamental questions: (*i*) How does the number of RWs affect search time as the system scales in size? (*ii*) What is the communications overhead incurred through the use of multiple RWs? (*iii*) Can the probability of a search failure be made sufficiently small? (*iv*) If a failed query is routed to a third party server, can the load placed on this server be made to scale?

Our goal is to understand how different system and workload parameters affect answers to the above questions. For example, it is necessary to attach a TTL to each RW in order to limit their communications overhead. Moreover, performance is sensitive to the demand pattern for content and to the level of content replication.

The main contributions of our paper are as follows. We extend the model proposed in [4], where users are represented as nodes in a graph and edges correspond to pairs of nodes that know of each other, and use this extended model to characterize the average delay for a successful query as a function of the TTL, the popularity of the content, and the level of replication. We also characterize the query overhead placed on each node, the probability of a failed search and the load placed on an external server due to failed queries. We find that query delays are $O(\log n)$, that the probability of search failure asymptotically tends to zero and that query per node and external server overheads are $O(1)$, provided that the number of RWs and the TTL threshold are set properly when content popularity is balanced by the level of replication. When the level of replication does not match demand, then it is impossible to maintain all metrics of interest unchanged. However, we show that it is possible to maintain low query latency and per node query overhead at the cost of increased external server load.

We formally establish these results by first deriving a set of bounds on the hitting time to a set of nodes via one or more independent RWs on a connected graph. We then exploit these bounds to derive expressions for the asymptotic behavior of the expected delay, search failure probability, server load and peer load as a function of the number of RWs deployed and the per-walk TTL threshold. These theoretical results, which should be of independent interest well beyond the present application, are formally established for the case that users are nodes in a graph with bounded maximum degree and that exhibits expander properties. Simulation experiments are used to investigate various issues of both theoretical and practical interest, validating and quantifying our results both for bounded degree random networks and for networks that exhibit a power law degree distribution.

Although there has been recent theoretical work on multiple RWs that focuses on characterizing either cover times – i.e., time to visit all nodes in the graph (see, e.g., [2]) or hitting times – i.e., time to reach a specific node in the graph (see, e.g., [3]), none of these studies focus on the problem of analyzing the hitting time of multiple RWs to a set of nodes. Our results on hitting times to sets of nodes builds on the work in [3] on hitting times to single nodes and builds on classical ideas from [1] on hitting times of a single RW to sets of nodes. We refer the interested reader to [7] for additional technical details and related work.

Section 2 formally describes the content network model and introduces expander graphs along with some of its relevant properties. We revisit hitting times of RWs on graphs in Section 3 and derive our extensions of these main results to account for multiple RWs. Section 4 presents our main results regarding the scalability of multiple RW search on networks, as well as a brief discussion of simulation results.

MAMA '14 Austin, Texas USA

2. MODEL

2.1 Content network model

We model the network as a connected non-bipartite graph $G = (V, E)$ where $V = \{1, \ldots, n\}$ is the finite set of vertices and $E \subset V \times V$ is the set of undirected edges between vertices. Associated with this system is a set of contents such that zero or more copies of each content is stored in the system; popularity may vary from content to content. We assume search queries are made at each node according to a Poisson process with rate μ. At each Poisson epoch, a query for a given content of interest occurs with probability (w.p.) $p = p(n)$. Search queries at different nodes are assumed to be independent of each other. We further assume that nodes each have a copy of the content of interest w.p. $q(n)$, which may not necessarily equal $p(n)$ *in an asymptotic sense.* Note that copies of the content may move around so long as the fraction of nodes holding a copy is roughly $q(n)$. Without loss of generality, we focus on a single content of interest.

A search query is handled by starting $k = k(n)$ independent RWs that progress through the network until any one of them finds a copy of the content being requested or the number of hops taken by each RW has reached a TTL threshold $T(n)$. In the first case, a copy of the content is returned to the node initiating the RWs. In the second case, the query either fails or is directed to a third party server, which returns the answer. We assume that the RWs execute in lock step and that they all halt as soon as any one of them finds the requested content. The hybrid P2P model of [4] maps into the above model where nodes correspond to peers and queries correspond to the arrival/departure of a peer.

We are interested in three performance metrics for this content network comprised of n nodes: $D(n)$ – the average time required to satisfy a search query; $P_f(n)$ – the probability that a search query fails; and $U(n)$ – the node traffic load measured in search queries handled per unit time. In addition, for the case where a search query failure is routed to a server, we are also interested in $U_s(n)$ – the server traffic load measured in search queries per unit time. Next we provide some background on RWs and expander graphs.

2.2 Random walks and expanders

Let $N(S) = \{i \in V : \exists j \in S \subset V \text{ s.t. } (i,j) \in E\}$ denote the set of neighbors of the nodes in set $S \subset V$. Let d_i denote the degree of vertex i and define $d_{max} := \max_{i \in V} d_i$ to be the maximum degree.

We study a discrete-time (DT) RW on the graph, which is defined to be the DT Markov chain (MC) $\{X(t)\,;\, t = 0, 1, \ldots\}$ with transition probability matrix $\mathbf{P} = [p_{ij}]$ where

$$p_{ij} = \begin{cases} 1/d_i, & (i,j) \in E, \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathbf{P}(t) = [p_{ij}(t)]$ denote the t-th step transition probability matrix associated with the RW, $t = 1, 2, \ldots$, where $p_{ij}(1) = p_{ij}$. Because G is connected and non-bipartite, the finite-state MC is irreducible, aperiodic and reversible. Moreover, the RW admits a unique stationary distribution $\pi = (\pi_1, \ldots, \pi_n)$ with $\pi_i = d_i / \sum_{j \in V} d_j$, $i \in V$. Let $\pi(S)$ denote the stationary probability for the set of states $S \subset V$. Define π_A to be the stationary distribution conditioned to $A \subset V$ via $\pi_A(i) := \pi(i)/\pi(A)$, $i \in A$.

We next define the vertex expansion ratio g_G as $g_G := \min_{S:|S| \leq n/2} |N(S)|/|S|$. Now, consider a sequence of graphs $\{G_n\}$ where G_n is a connected non-bipartite graph with n vertices. We assume that $d_{max}(n) \leq d$ for some constant $d \geq 2$. Let $g(n) = g_{G_n}$ and define $g := \liminf_{n \to \infty} g(n)$. Lastly, we focus on $\{G_n\}$ such that $g > 0$, i.e., the sequence $g(n)$ is bounded away from zero. Such a sequence of graphs is called an *expander family.*

Define the *relaxation time* τ_2 of a RW on graph G to be the inverse of the spectral gap of the RW: $\tau_2 = 1/(1 - \max\{\lambda_2, |\lambda_n|\})$. Here, λ_i denotes the i-th largest eigenvalue of the RW, $i = 1, \ldots, n$, such that $1 = \lambda_1 > \lambda_2 \geq \cdots \geq \lambda_n \geq -1$. Henceforth we assume $\lambda_2 \geq |\lambda_n|$. This can always be guaranteed by allowing the RW to be *lazy*, i.e., requiring a RW at each step to remain at a node w.p. 1/2 [5, Ch 12].

3. HITTING TIME RESULTS

In this section we exploit classical results on reversible MCs to derive bounds on the hitting time to a set $A \subset V$ by $k \geq 1$ independent RWs on the graphs G_n, where we assume $n = |V|$ large and $|A|/|V|$ small. Consistent with our model of Section 2, we focus here on DTMCs noting that analogous hitting-time results for continuous-time (CT) MCs can be similarly derived. All proofs and details are provided in [7].

Consider a subset A of states V with $\pi(A)$ small, in the sense of [1]. Let T_A denote the hitting time to set A; namely, $T_A = \min\{t \geq 0 : X(t) \in A\}$. Let $\mathbb{P}_\beta[\cdot]$ and $\mathbb{E}_\beta[\cdot]$ respectively denote the probabilities and expectations for the MC started at time 0 with probability distribution β; similarly, $\mathbb{P}_u[\cdot]$ and $\mathbb{E}_u[\cdot]$ denote the same operators when the MC starts in state $u \in V$ at time 0.

We first consider the case of $k = 1$ and establish the following key result on the relationship between hitting time and relaxation time. Our derivation relies on classical results for reversible DT and CT MCs [1]. To this end, let $\mathbf{P}^A$ denote the matrix $\mathbf{P}$ restricted to $A^c = V \setminus A$, and hence $\mathbf{P}^A$ is a substochastic matrix; similarly, let $\mathbf{Q}^A$ denote the generator for the corresponding CTMC.

Lemma 3.1. *For a fixed subset $A \subset V$, the hitting time T_A satisfies*

$$\begin{aligned} \mathbb{E}_\pi[T_A] &\leq \tau_2/\pi(A), & (1) \\ \mathbb{P}_\pi[T_A > t] &\leq (1 - \pi(A)/\tau_2)^{t-1}, \qquad t = 1, \ldots. & (2) \end{aligned}$$

Now, we turn to consider the case $k > 1$. Let T_A^k denote the hitting time to set $A \subset V$ by k independent RWs on the graphs G_n. Define $\pi^k := (\pi, \ldots, \pi)$ to be the stationary distribution of k nodes selected independently from the stationary distribution. Lemma 3.1 then allows us to conclude the following result.

Lemma 3.2. *For a fixed subset $A \subset V$, the hitting time T_A^k satisfies*

$$\begin{aligned} \mathbb{P}_{\pi^k}[T_A^k > t] &\leq (1 - \pi(A)/\tau_2)^{k(t-1)}, \qquad t = 1, \ldots, & (3) \\ \mathbb{E}_{\pi^k}[T_A^k] &\leq \frac{1}{1 - (1 - \pi(A)/\tau_2)^k}. & (4) \end{aligned}$$

Our consideration of the subsets $A \subset V$ has been such that $\pi(A)$ is small, or equivalently $\mathbb{E}_\pi[T_A]$ is large, in comparison with the relaxation time τ_2. These conditions will indeed hold for sufficiently large n and sufficiently small $|A|/|V|$ for expander graphs. Formally, our conditions on the subsets $A \subset V$ are such that $\tau_2(n)/\pi(A) = \omega(1)$ and $\tau_2(n)/\mathbb{E}_\pi[T_A] = o(1)$, which can be combined with Lemma 3.2. Focusing on the case where the k RWs start from the same vertex $u \notin A$, our interest is in the behavior of $\mathbb{E}_u[T_A^k]$ as n becomes large, for which we derive the following result.

LEMMA 3.3. *For a fixed subset $A \subset V$, an arbitrary vertex $u \notin A$ and large n, the hitting time T_A^k satisfies*

$$\mathbb{E}_u[T_A^k] \le (7/2 \log n + \log k) + \frac{1}{1-(1-\pi(A)/\tau_2)^k}. \qquad (5)$$

Next, if $\{G_n\}$ is an expander family, we then have $\tau_2(n) = O(1)$, due to $\lambda_2 \ge |\lambda_n|$. Combining this asymptotic relationship together with Lemma 3.3 yields

$$\mathbb{E}_u[T_A^k] \;\; \le \;\; O\left(\log n + \log k + \frac{1}{1-(1-\pi(A))^k}\right). \qquad (6)$$

In addition to providing limits on the expected hitting time for k RWs, this bound plays an important role in our choice of the number of RWs. Specifically, we seek large values of $k(n)$ so long as the third term on the right-hand side of (6) grows slower than $O(\log n + \log k)$.

4. PERFORMANCE RESULTS

We now turn our attention to the behavior of $D(n)$, $U(n)$, $P_f(n)$ and $U_s(n)$ in the limit as $n \to \infty$, leveraging the results of the previous sections. Our interest is in the asymptotic behavior of these performance metrics as a function of the number of RWs deployed, $k(n)$, and the per-walk TTL threshold, $T(n)$. To this end, consider a node u that issues a search query w.p. $p(n)$; recall that $p(n)$ may not equal the probability that a node has a copy of the content, $q(n)$. Without loss of generality, our analysis focuses on search queries for a single content of interest. We shall generically use $A \subset V$ to refer to the set of nodes that hold a copy of the requested content. By independence, the stationary probability associated with this set $A \subset V$ is given by $Q(V, A) \; = \; q(n)^{|A|}(1-q(n))^{|V|-|A|}$, $A \subseteq V$. Letting M denote a generic random variable for the number of nodes which hold a copy of the requested content, we then have that M is binomially distributed with population $n-1$ and parameter $q(n)$. Throughout the analysis that follows, use is made of the inequality $\pi(A) \ge |A|/(nd_{max})$, $A \subset V$.

Based on our previous assumptions/results, we consider

$$\begin{aligned} T(n) &= \Theta(\log n \max\{1, \log(nq(n))\}), & (7)\\ k(n) &= \max\Big\{1, \min\Big\{\frac{1}{q(n)\log n}, \frac{n}{\log n}\Big\}\Big\}. & (8)\end{aligned}$$

There are several cases of interest for our analysis of the performance metrics $D(n)$, $U(n)$, $P_f(n)$ and $U_s(n)$. Starting with the case $1/q(n) = O(\log n)$, we have from (8) that $k(n) = O(1)$. The results of [4] then apply, and therefore $D(n) = O(1/p(n))$, $U_s(n) = U(n) = O(1)$ and $P_f(n) \to 0$ as $n \to \infty$. Hence, we now focus on the cases of interest where $k(n)$ grows with n, deriving bounds on the performance metrics $D(n)$, $U(n)$, $P_f(n)$, $U_s(n)$ in terms of $k(n)$, $T(n)$, $p(n)$, $q(n)$, and considering the asymptotic behavior of these bounds under different asymptotic properties for $q(n)$ and $p(n)$. We assume throughout that n is large, $|A|/|V|$ is small, and $\{G_n\}$ is an expander family, hence $\tau_2(n) = O(1)$.

Starting with the metric $D(n)$, we establish [7]:

$$D(n) = O(\log n) + T(n)(1-q(n))^{n-1}.$$

In the case that $1/q(n) = o(n)$, the second term goes to zero as $n \to \infty$ and we have $D(n) = O(\log n)$. When $1/q(n) = \Omega(n)$, then $T(n) = O(\log n)$ from (7) and $D(n) = O(\log n)$. Turning to the performance metric $U(n)$, we show [7]:

$$U(n) \le \mu p(n) k(n) O(\log n).$$

In the case that $1/q(n) = o(n)$, then from (8) $k(n)$ is of the order $1/(q(n)\log n)$, which yields $U(n) = O(p(n)/q(n))$; note that the communication overhead does not scale if $p(n)/q(n) = \omega(1)$. When $1/q(n) = \Omega(n)$, there is a probability bounded away from zero that the search will fail, and therefore $U(n) = O(np(n))$ in this case.

Considering the metrics $P_f(n)$ and $U_s(n)$, the case $1/q(n) = O(\log n)$ implies $k(n) = 1$ and the arguments in [4] are easily extended to establish $P_f(n) \to 0$ as $n \to \infty$ and $U_s(n) = O(p(n)/q(n))$. When $1/q(n) = \Omega(n)$, we show [7] that $P_f(n)$ is bounded away from zero as $n \to \infty$ and that $U_s(n) = O(np(n))$. Lastly, for $1/q(n) = o(n)$ and $1/q(n) = \omega(\log n)$, we establish [7] that, as $n \to \infty$,

$$P_f(n) \to 0 \quad \text{and} \quad U_s(n) \le O(p(n)/q(n)).$$

In the case $p(n) = \Theta(q(n))$, we obtain that $D(n) = O(\log n)$, $U(n) = U_s(n) = O(1)$, and $P_f(n) \to 0$ as $n \to \infty$.

To empirically validate and quantify the properties of multiple RW search in large-scale content networks, we also study the behavior of such systems through a simulation of a P2P network that employs many properties of these content networks along the lines of [4], considering both expander graphs and power law graphs generated by preferential attachment. Our simulation results show that search query latency, average network load and average server load across various scenarios are respectively $O(\log n)$, $O(1)$ and $O(1)$ for *both* expander graph and power-law graph overlay networks, which consistently match our theoretical results. Even when $p \ne q$, especially $p > q$, the query latency and network load continue to match our theoretical results. These results further demonstrate that multiple RWs effectively control query latency and bound network/server overheads, thus ensuring the system remains scalable to very large network sizes without overwhelming individual users, even when sparsely-available items are requested by many users. Additional details and results are provided in [7].

5. REFERENCES

[1] D. Aldous, J. Fill. Chapter 3: Reversible Markov chains. `http://www.stat.berkeley.edu/~aldous/RWG/Chap3.pdf`, September 2002.

[2] N. Alon, C. Avin, M. Koucký, G. Kozma, Z. Lotker, M. Tuttle. Many random walks are faster than one. *Comb., Prob. and Comp.*, 20:481–502, 2011.

[3] K. Efremenko, O. Reingold. How well do random walks parallelize? Preprint, October 2010.

[4] S. Ioannidis, P. Marbach. On the design of hybrid peer-to-peer systems. In *Proc. ACM SIGMETRICS*, pp. 157–168, 2008.

[5] D. Levin, Y. Peres, E. Wilmer. *Markov Chains and Mixing Times.* American Mathematical Society, 2009.

[6] Q. Lv, P. Cao, E. Cohen, K. Li, S. Shenker. Search and replication in unstructured peer-to-peer networks. In *Proc. Supercomputing*, pp. 84–95, 2002.

[7] M. Squillante, D. Towsley, S. Barker. Improving the scalability of search in networks through multiple random walks. arXiv preprint (2014). Submitted.