

Efficiency of random walks for search in different network structures

Gerhard Hasslinger

T-Systems Enterprise Systems
Deutsche-Telekom-Allee 7
D-64295 Darmstadt, Germany

gerhard.hasslinger@telekom.de

Sebastian Kempken

Department of Computer Science
University of Duisburg-Essen
D-47057 Duisburg, Germany

kempken@inf.uni-due.de

ABSTRACT

Dynamic overlay networks on the Internet as well as ad hoc networks often rely on self-organizing distributed communication and construction techniques. When there are no central management facilities and search indices available, flooding is a standard method to collect knowledge about the network structure and the content on the nodes. In recent time, random walks have attracted attention as an alternative search method in P2P networks. The efficiency of the method is usually evaluated by simulation studies.

In this paper we use transient analysis as a simple and scalable approach to examine the properties of random walks. The convergence to steady state and the coverage of the network in the course of the random walk are main characteristics to be analyzed. In our evaluations we consider network examples of different type and size.

Keywords

Random walks, transient analysis, coverage of search

1. INTRODUCTION

Several types of peer-to-peer networks have been build and widely used in the last decade [3][9]. Among them are hybrid systems of different type e.g. with

- centralized servers for search followed by a decentralized data exchange (Napster)
- hierarchical networks usually providing two levels of a few super peers and many normal peers, where super nodes again provide search tables, e.g., eDonkey, eMule, BitTorrent,
- structured homogeneous network approaches like Chord [12], Pastry etc., with index pointers to support search and
- unstructured networks, which do not employ indices to support content search e.g. the Gnutella network or the BubbleStorm approach [13].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SMCTools'07, October 26, 2007, Nantes, France.

Copyright 2007 ICST 978-963-9799-00-4

Randomized techniques appear to be useful in the construction and exploration of self-organizing networks, which often are subject to a high churn of nodes entering and leaving the network. It is discussed in [1][2] how randomness can help to maintain dynamic networks with minimum overhead while preserving sufficient connectivity of the network. Many of the largest known networks build by humans exhibit properties which are generated by random expansion including the Internet, citation indices and networks of social relationships [1].

For extending a network, a random walk can identify points in order to attach new nodes, which are then chosen as independent random samples according to the stationary distribution. Therefore the required length of a random walk until sufficient convergence to steady state has to be evaluated. As a rule of thumb, $L = 3 \ln(|V|)$ is often used in scalefree networks of size $|V|$ [1][2][13].

In P2P networks, which are build as overlays on the Internet, a decomposition into disconnected parts has to be avoided, even if nodes are entering and leaving the network frequently. The BubbleStorm P2P architecture [13] is conceived in order to be highly reliable, such that a simultaneous disappearance of up to 90% of the peers e.g. caused by breakdowns in the underlying transport network leave the rest of the network intact. This includes a replication scheme for data over subsets of nodes in the network ("data bubble"). Therefore each data item can be found on many nodes in a P2P network in order to improve the reliability and the download performance. The most popular data is simultaneously requested from many participants, who would experience a bottleneck if a comparable number of source peers with copies of the data is not available.

The search time for content naturally depends on the number of replicas of data in the network. In the BubbleStorm approach, the size of a "data bubble" is made larger than the square root of the network size and the set of nodes included in a data bubble is almost a random sample among all network nodes. Thus, visiting only a limited portion of the network nodes is sufficient for most searches.

The search by flooding and random walks in P2P networks has been compared and evaluated by several simulation studies [5][7][8][10], some of which point out the random walks are more efficient. The main disadvantage of a single random walk is a long delay while travelling through the network. Flooding spreads search messages in parallel to all nodes in the neighborhood up to some hop distance d . It is often difficult to find an appropriate search radius d in order to reach a predefined number of network

nodes, which contributes to the disadvantage of a large messaging overhead of flooding. Combined methods seem promising to overcome the disadvantages and to extract the best of both approaches [8]. Combined variants include

- several random walks in parallel or branching of a random walk into multiple paths,
- flooding with a predefined budget in order to control the number of visited nodes, which is split up among several paths to different next hops,
- or random walks with an additional flooding step for a small distance d from the last or from several nodes being traversed.

To the authors knowledge, all studies on evaluation of those search methods rely on simulation to estimate the performance of random walks. Instead we utilize transient analysis, yielding the exact probabilistic characteristics of a random walk after each step. The computational effort of this method is linear in the number of network edges and in the number of steps to be traced. Therefore transient analysis is tractable up to the size of popular P2P networks like KaZaa and eDonkey with several millions of peers. This is at least possible for basic random walk analysis as shown in this paper and for several variants, provided that they do not require much memory about previous steps of the walk, which would increase the required state space and complexity of the analysis. Random walks and transient analysis of Markov chains are classical concepts of stochastic modeling with main principles being introduced in books [6][11] and many research papers devoted to special properties, e.g. [14].

This paper is organized in three sections starting with Markov modeling and steady state considerations for random walks, followed by the main part for derivation of the transient analysis and finally a comparison of evaluations of the efficiency of random walks in seven different types of network topologies.

2. MARKOV MODEL AND SOME BASIC PROPERTIES

A network graph $G = (V, E)$ is considered with a set V of nodes and a set E of edges. Usually we assume undirected graphs $(a, b) \in E \Leftrightarrow (b, a) \in E$ or otherwise we clearly indicate cases of directed graphs. We presume that the graph is connected and $d(G) \geq 1$ holds for the degree $d(G)$ of the graph, which is defined as the minimum number of disjoint paths between an arbitrary pair of nodes.

We follow a random walk through the network as a stepwise process, which proceeds from a node to a neighboring node at the next hop. A random walk R of length L is denoted by the series $R = (r_0, r_1, r_2, \dots, r_L)$ of visited nodes, where an edge $(r_{k-1}, r_k) \in E$ is chosen for the k -th hop ($1 \leq k \leq L$). The degree $d(a)$ of a node $a \in V$ is the number of edges attached to the node

$$d(a) = |\{k \mid k \in V, (a, k) \in E\}| \text{ and } d_{\min} = \min\{d(a) \mid a \in V\}$$

denotes the minimum node degree.

As the basic case, a random walk chooses each edge from node a for the next hop with the same probability

$$\forall a, k; (a, k) \in E: p_{ak} = \text{Prob}(r_{n+1} = k \mid r_n = a) = 1 / d(a). \quad (1)$$

Then $P = (p_{ak})$ denotes the corresponding transition matrix. In this way, the random walk in the network is equivalent to a Markov process [6], where network nodes correspond to the states of the underlying Markov chain and links to transitions between states. In general, arbitrary transition probabilities may be considered,

but for applications in P2P networks the setting (1) seems appropriate.

Since $d(G) \geq 1$, the Markov chain is irreducible. When periodical cases are excluded, where the number of hops on all paths leading from a node in a circle back to the same node are a multiple of some period length $l \geq 2$, then steady state probabilities $p^{(R)}(a)$ exist

$$p^{(R)}(a) = \lim_{n \rightarrow \infty} \text{Prob}(r_n = a) \quad (2)$$

and the long term behaviour becomes independent of the starting point. For bidirectional links and uniform transition probabilities as in equation (1), the steady state probabilities are given by

$$p^{(R)}(a) = \frac{d(a)}{2|E|}. \quad (3)$$

For verification of the steady state solution, equilibrium equations can be set up for the transitions from a node to each next hop [6]:

$$p^{(R)}(a) = \sum_{k: (k,a) \in E} p^{(R)}(k) p_{ka} = \sum_{k: (k,a) \in E} p^{(R)}(k) / d(k)$$

Inserting the solution (3) for $p^{(R)}(k)$ confirms

$$\begin{aligned} p^{(R)}(a) &= \sum_{k: (k,a) \in E} \frac{d(k)}{2|E|} / d(k) \\ &= \sum_{k: (k,a) \in E} \frac{1}{2|E|} = \frac{d(a)}{2|E|} \end{aligned}$$

where in addition the normalization constraint

$$\sum_{a \in V} p^{(R)}(a) = \sum_{a \in V} \frac{d(a)}{2|E|} = 1$$

is met. The result can be interpreted in the way, that the considered random walks utilize all edges in the network with equal probability in the long term. Consequently, the expected frequency of visiting a node becomes proportional to the node degree.

The convergence time of a random walk to steady state strongly depends on the network topology. Therefore we define the probability distribution of the random walk to enter a node at its m -th hop

$$p_m^{(R)}(a) = \text{Prob}(r_m = a)$$

and a measure of the distance from steady state

$$\Delta_m = \sum_{a \in V} |p_m^{(R)}(a) - p^{(R)}(a)|$$

The convergence to steady state can be expressed on behalf of the eigenvalues of the transition matrix P of the random walk. It is discussed e.g. in [5][6][11][14] that an irreducible and aperiodic matrix P of an undirected graph has 1 as the largest eigenvalue and that the second largest eigenvalue λ_2 is real and positive. Then λ_2 is decisive for the convergence to steady state and can be used to obtain the following bound on the convergence speed [7]:

$$\Delta_m \leq 4|E| \frac{\lambda_2^m}{d_{\min}}.$$

A corresponding bound can also be derived for the mean number of nodes that have been visited during a random walk of length L [7].

$$q_m^{(R)}(t) \leq q_{m+1}^{(R)}(t); \quad \lim_{m \rightarrow \infty} q_m^{(R)}(t) = 1.$$

3. TRANSIENT BEHAVIOUR OF RANDOM WALKS

When the exploration of a network by random walks of predefined length is evaluated using simulation [7][8][10][15] the results are subject to confidence levels with long simulation runs being required to achieve tight confidence intervals. In addition, analytically derived bounds for the network coverage and the steady state convergence are valuable to describe the principle behaviour, but do not capture a random walk in detail [4][5][7].

In this paper we follow an alternative classical approach using transient analysis, which stepwise determines the probability distribution of network nodes that a random walk will enter at its m -th hop. When the random walk starts at a specific node s , then we have

$$p_0^{(R)}(s) = 1 \quad \text{and} \quad \forall a \neq s: p_0^{(R)}(a) = 0$$

as the initial distribution. In general, any arbitrary initial distribution can be considered. The transient analysis iteratively computes the distribution of the next hop location. After $p_m^{(R)}(a)$ has been determined, the distribution $p_{m+1}^{(R)}(a)$ for the next step can be computed based on the knowledge of $p_m^{(R)}(a)$:

$$p_{m+1}^{(R)}(a) = \sum_{k: (k, a) \in E} p_m^{(R)}(k) p_{ka} \quad (4)$$

Using equation (4) for the first hop after starting at node s , we obtain

$$p_1^{(R)}(a) = 1/d(s) \quad \text{if } (s, a) \in E \quad \text{and} \quad p_1^{(R)}(a) = 0 \quad \text{if } (s, a) \notin E.$$

If steady state conditions are satisfied, then the iterative computation of $p_1^{(R)}(a)$, $p_2^{(R)}(a)$, ... converges to $p^{(R)}(a)$. The computational complexity for the transient analysis due to equation (4) is proportional to the number $|E|$ of edges in the network for each hop. Therefore the effort to compute $p_m^{(R)}(a)$ is of the order $O(m|E|)$, which makes transient analysis applicable for large scale networks.

Besides the complete distribution of the node location after m hops, the probability $q_L^{(R)}(t)$ that a node t has been visited during a random walk of length L is the most important measure for the performance of random network explorations:

$$q_L^{(R)}(t) = \text{Prob}(\exists j, 0 \leq j \leq L: r_j = t).$$

The probabilities $q_L^{(R)}(t)$ indicate how long it will last until the walk has visited half or 90% of the network nodes in order to collect information on the structure or content in the network. The probabilities $q_m^{(R)}(t)$ are still computed in the hop-by-hop approach of equation (4). The only modification required is to introduce an absorbing state at node t , which accumulates the probabilities of visiting that node during the random walk [6]. Therefore the transition equations (1) need to be changed only for the transitions from node t , while all others remain valid:

$$\forall a, k; (a, k) \in E: q_{ak} = 1/d(a) \quad \text{if } a \neq t; \quad (5)$$

$$\forall k; (t, k) \in E: q_{tk} = 0 \quad \text{if } k \neq t; \quad q_{tt} = 1.$$

$q_L^{(R)}(t)$ is determined by applying equation (4) to the modified transitions:

$$q_{m+1}^{(R)}(a) = \sum_{k: (k, a) \in E} q_m^{(R)}(k) q_{ka}. \quad (6)$$

The random walk defined by equations (5) and (6) has the same behaviour at all nodes except for t . Once it has reached t it stays there for ever. Consequently, $q_m^{(R)}(t)$ is monotonously increasing with m . If steady state preconditions are met, node t will be reached sooner or later with probability 1:

This limiting behaviour even holds for irreducible and periodical random walks. On the other hand, $q_m^{(R)}(a)$ with $a \neq t$ can be interpreted as the probability of the random walk to enter node a by the m -th hop without having visited t beforehand. The computational effort to compute $q_m^{(R)}(a)$ for all nodes in the network is again linear in the number of edges and the length of the considered random walk $O(m|E|)$.

If we want to know the length m of a random walk until the probability of having visited node t has reached a predefined level α ($0 \leq \alpha \leq 1$), then the computational effort depends on the network topology and is increasing with the considered probability level α .

Finally, note that $q_m^{(R)}(t)$ can be computed starting from a specific node $s \neq t$, where again

$$q_0^{(R)}(s) = 1 \quad \text{and} \quad \forall a \neq s: q_0^{(R)}(a) = \text{Prob}(r_0 = a) = 0$$

or from an arbitrary initial distribution. In the following evaluation section, we always start with a uniform distribution over the network nodes, such that

$$\forall a: q_0^{(R)}(a) = 1/|V|.$$

4. EVALUATION OF NETWORK TOPOLOGIES

Based on the previously presented transient analysis approach, we have evaluated several types of networks and studied the influence of the network size for each type. In each case, results on the convergence to steady state and on the network coverage are shown in two separate figures. At first we present each network type with its evaluation results and then give short remarks on the comparison of the obtained results.

All the considered cases exhibit regular structures of the topology, whereas most applications of random walks are in unstructured P2P or ad hoc networks. Nevertheless, the clearly structured cases make it easier to comprehend the different behaviour in the convergence to steady state and the suitability of random walks for exploring a network. Preconditions of the analysis of the steady state convergence are discussed in detail in section 4.1 and for the analysis of the network coverage in section 4.2.

Table 1 summarizes main parameters of the network examples.

Table 1: Main parameters of the network examples

| Network type | Number of edges | Min. node degree | Max. node degree | Diameter |
|---|--|------------------|------------------|-------------------------------------|
| Ring | $ V $ | 2 | 2 | $\lfloor V /2 \rfloor$ |
| Chord structure: ring & unidirectional pointers | $ V \log_2 V $ (for $ V = 2^k$) | $\log_2 V $ | $\log_2 V $ | $\log_2 V $ |
| Binary Tree ($ V = 2^k - 1$) | $ V - 1$ | 1 | 3 | $2 \log_2 (V - 1)$ |
| Extension of a binary tree ($ V = 2^k - 1$) | $(\log_2 V - 1) \cdot (V + 1) + 2$ | $\log_2 V - 1$ | $ V - 1$ | 2 |
| 2-dim. grid, wrapped | $2 V $ | $(V = k^2)$ 4 | 4 | $2 \lfloor \sqrt{ V /2} \rfloor$ |
| 3-dim. grid, wrapped | $3 V $ | $(V = k^3)$ 6 | 6 | $2 \lfloor \sqrt[3]{ V /2} \rfloor$ |
| Hyper cube ($ V = 2^k$) | $ V \log_2 V /2$ | $\log_2 V $ | $\log_2 V $ | $\log_2 V $ |
| Full mesh | $ V (V + 1)/2$ | $ V $ | $ V $ | 1 |
| Star | $ V - 1$ | 1 | $ V - 1$ | 2 |

Ring network

At first, a ring network is studied, where each of the nodes $N_1, N_2, \dots, N_{|V|}$ is connected to a neighbor on both sides:

$$(N_j, N_k) \in E \Leftrightarrow (k = (j+1) \bmod |V| \text{ or } k = (j-1) \bmod |V|)$$

All node degrees are equal $d(N_j) = 2$ in this example.

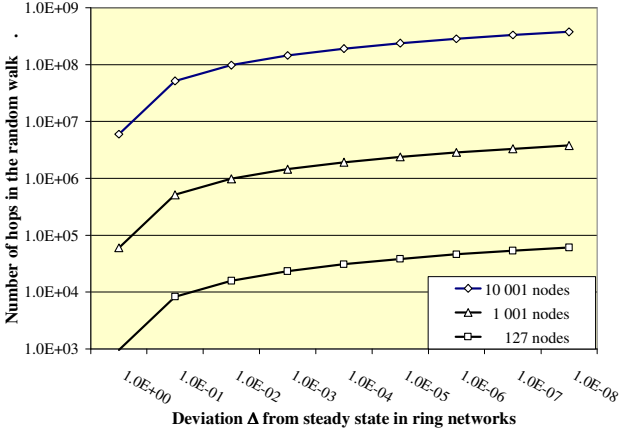


Figure 1: Ring network: Convergence to steady state

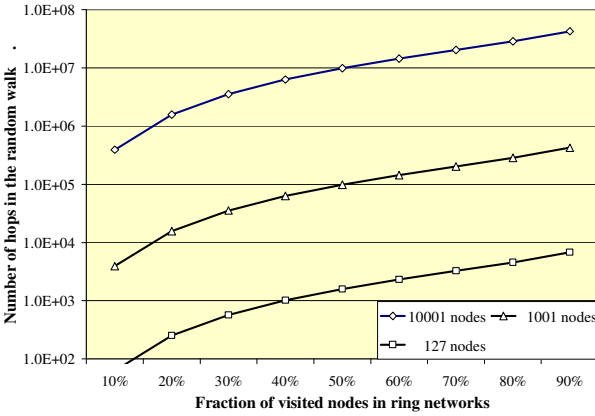


Figure 2: Ring network: Hop count for network coverage

Ring network with pointers

The third example is a directed graph. All nodes have the same degree resembling the structure of the Chord P2P proposal [12] of a ring with fingers pointing to nodes whose distance in the ring equals a power of 2. The node degree is K for a network of size $2^{K-1} < |V| \leq 2^K$. The topology is defined extending the simple ring:

$$\forall j, k; 1 \leq j, k \leq |V| \text{ and } \forall l; 1 \leq l \leq K:$$

$$(N_j, N_k) \in E \Leftrightarrow k = (j + 2^{l-1}) \bmod |V|.$$

The derivation of steady state probabilities of section 2 is valid only in undirected graphs whereas this graph is directed.

However, the network is irreducible, aperiodic and owing to the symmetrical transition matrix, the steady state distribution is uniform

$$p^{(R)}(N_1) = p^{(R)}(N_2) = \dots = p^{(R)}(N_{|V|}) = 1 / |V|.$$

The transient analysis applies to directed as well as undirected graphs.

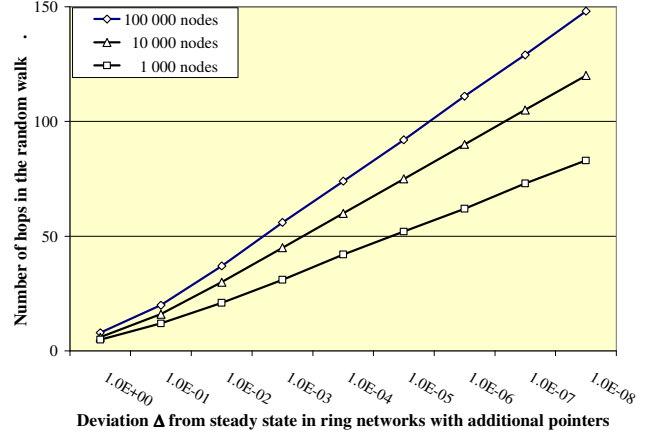


Figure 3: Ring with pointers: Convergence to steady state

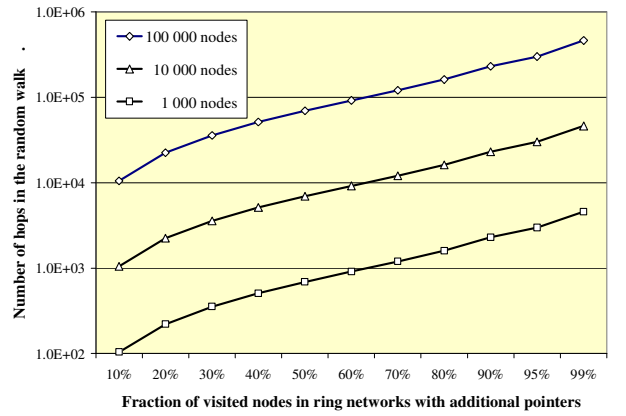


Figure 4: Ring with pointers: Network coverage

Binary tree network

The binary tree structure has $|V| = 2^K - 1$ nodes in total. On a level $k = 1, \dots, K$ there are 2^{k-1} nodes, where the first level corresponds to the root and the last is the leaf level. Then $N_1, \dots, N_{2^{K-1}}$ are the leaves, $N_{2^{K-1}+1}, \dots, N_{2^{K-1}+2^{K-2}}$ are the nodes on the intermediate level $K-1$ up to N_{2^K-1} as the root on the first level. Each node has one edge to a node on the next higher level except for the root and each node has two edges to different nodes on the next lower level except for the leaves. The leaves have node degree 1, all nodes on intermediate levels have degree 3 and the root has degree 2.

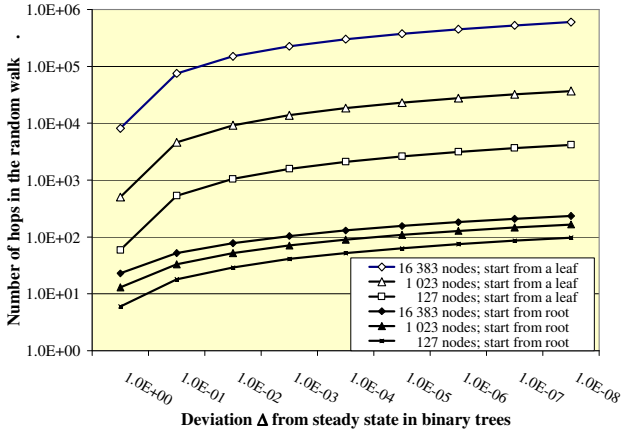


Figure 5: Binary tree: Convergence to steady state

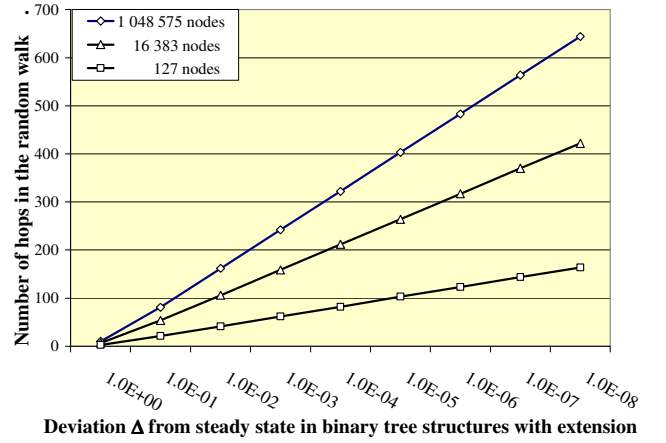


Figure 7: Extended binary tree: Steady state convergence

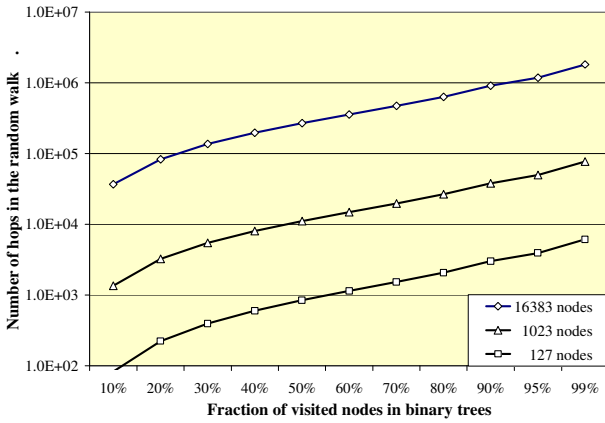


Figure 6: Binary tree: Hop count for network coverage

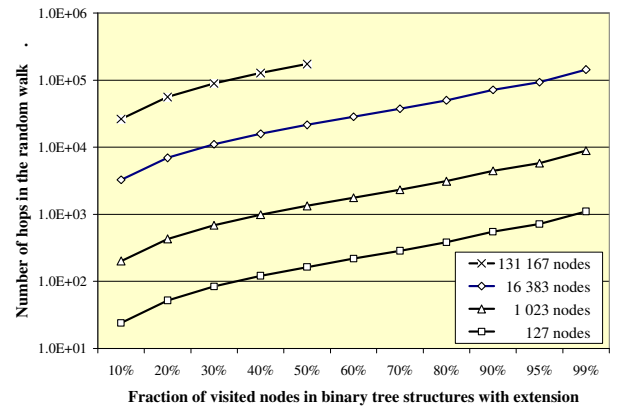


Figure 8: Extended binary tree: Hop count for coverage

Binary tree structure with additional links

In addition, we consider the case of a network with the same level structure as for the binary tree network and more links. In particular, each non-leaf node has links to all nodes in the subtree corresponding to this node. Thus the root is connected to all other nodes and has degree $2^K - 2$ and a node on a level $k = 1, \dots, K-1$ has links to 2^{j-k} distinct nodes on each lower level j .

Both second level nodes are connected to $2^{K-1} - 2$ nodes on lower levels in their sub-trees and to the root, which establishes degree $2^{K-1} - 1$, etc. Since all links are bidirectional, a leaf is connected to one node on each of the higher levels and has degree $K-1$. The degrees of nodes in the network are spread over almost the entire possible range.

2-dimensional grid

In the 2-dimensional grid each node has a degree of 4, assuming that edges are wrapping around the ends of the grid in both directions. In this way, the network structure becomes homogeneous. The network size is assumed to be a square number. In order to determine the edges of the network, let's denote a node by its coordinates in the grid:

$$\forall n_1, n_2; 1 \leq n_1, n_2 \leq \sqrt{|V|} : (n_1, n_2) \in V$$

Then $((n_1, n_2), (m_1, m_2)) \in E \Leftrightarrow$

$$(n_1 = m_1 \text{ and } |n_2 - m_2| \bmod \sqrt{|V|} = \pm 1) \text{ or}$$

$$(n_2 = m_2 \text{ and } |n_1 - m_1| \bmod \sqrt{|V|} = \pm 1).$$

If the network size $|V|$ is an even number, then the random walk is periodical and alternates between nodes in two subsets of size $|V|/2$. Then a modified convergence criterion may be applied according to equation (7-8) considering the arithmetic mean over two hops of the random walk. In the examples, the network size is odd and then a stationary distribution exists.

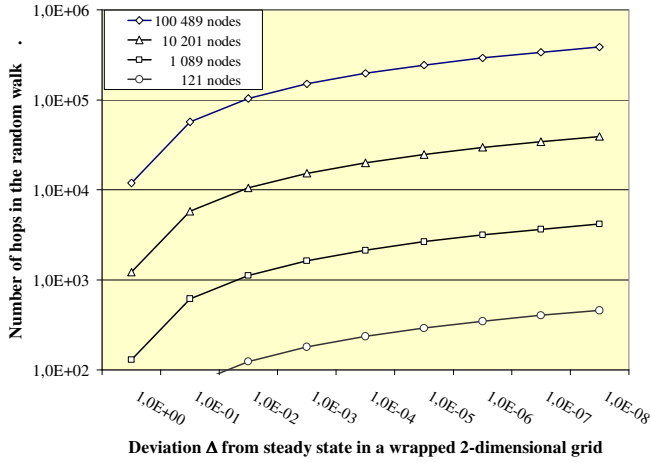


Figure 9: 2-dimensional grid: Steady state convergence

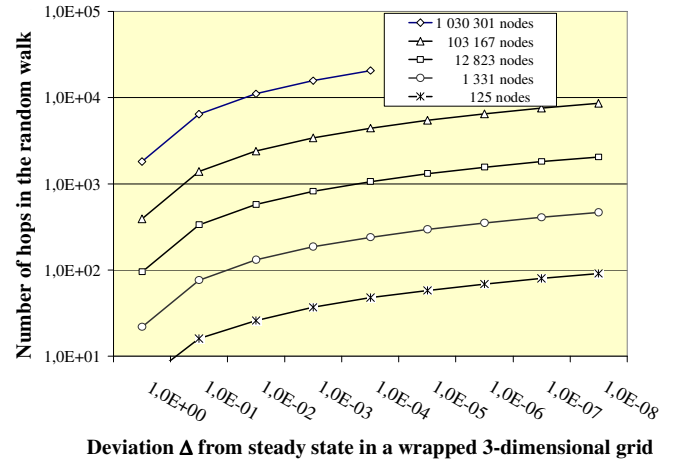


Figure 11: 3-dimensional grid: Steady state convergence

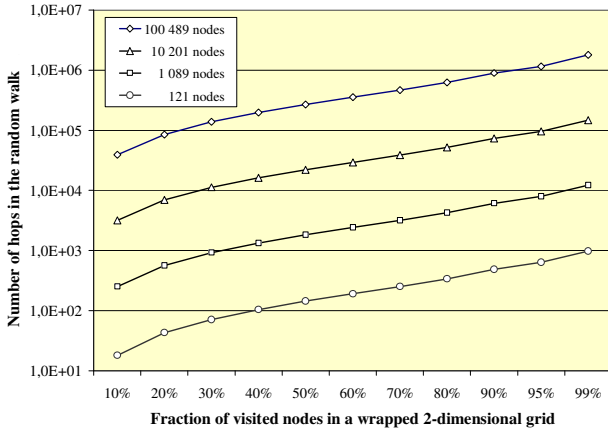


Figure 10: 2-dimensional grid: Hop count for network coverage

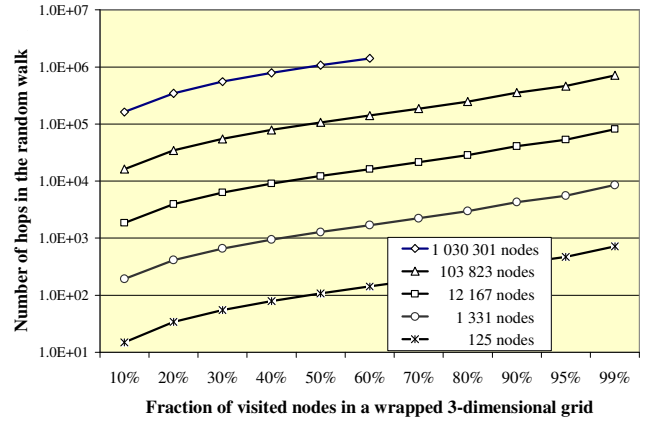


Figure 12: 3-dimensional grid: Hop count for network coverage

3-dimensional grid

The 3-dimensional grid has similar properties as the previous 2-dimensional case. With edges being wrapped around, the network structure is homogeneous and each node has a degree of 6. The network size is assumed to be a cubic number. In order to determine the edges of the network, let's denote a node by its coordinates in the grid:

$$\forall n_1, n_2, n_3; 1 \leq n_1, n_2, n_3 \leq \sqrt[3]{|V|}: (n_1, n_2, n_3) \in V$$

Then $((n_1, n_2, n_3), (m_1, m_2, m_3)) \in E \Leftrightarrow$

$$(n_1 = m_1, n_2 = m_2 \text{ and } |n_3 - m_3| \bmod \sqrt[3]{|V|} = \pm 1) \text{ or}$$

$$(n_1 = m_1, n_3 = m_3 \text{ and } |n_2 - m_2| \bmod \sqrt[3]{|V|} = \pm 1) \text{ or}$$

$$(n_2 = m_2, n_3 = m_3 \text{ and } |n_1 - m_1| \bmod \sqrt[3]{|V|} = \pm 1).$$

If the network size $|V|$ is an even number, then the random walk is periodical and alternates between nodes in two subsets of size $|V|/2$. Then a modified convergence criterion may be applied according to equation (7-8) considering the arithmetic mean over two hops of the random walk. In the examples, the network size is odd and then a stationary distribution exists.

Hyper-Cube

As a last example, we consider a hypercube whose network size is a power of 2: $|V| = 2^k$. This is again a homogeneous network type with degree k for all nodes. The hypercube network structure can be defined recursively. Let G and H be two hypercubes of size $|V|/2 = 2^{k-1}$ with nodes $g_1, g_2, \dots, g_{|V|/2}$ and $h_1, h_2, \dots, h_{|V|/2}$, respectively. $E_{k-1}^{(G)}$ and $E_{k-1}^{(H)}$ are corresponding sets of edges. Then a hypercube of size $|V| = 2^k$ can be build by all the nodes $g_1, g_2, \dots, g_{|V|/2}$ and $h_1, h_2, \dots, h_{|V|/2}$ and edges $E_{k-1}^{(G)}$ and $E_{k-1}^{(H)}$ in G and H and 2^{k-1} additional edges $(g_1, h_1), (g_2, h_2), \dots, (g_{|V|/2}, h_{|V|/2})$.

Random walk in a hypercube are again periodical, alternating between nodes in two subsets of size $|V|/2$. The modified convergence criterion of equation (7-8) is used based on the arithmetic mean of two subsequent hops of the random walk.

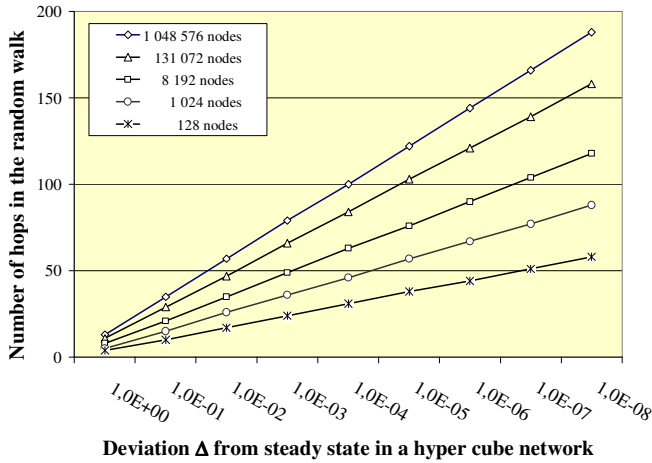


Figure 13: Hyper-Cube: Steady state convergence

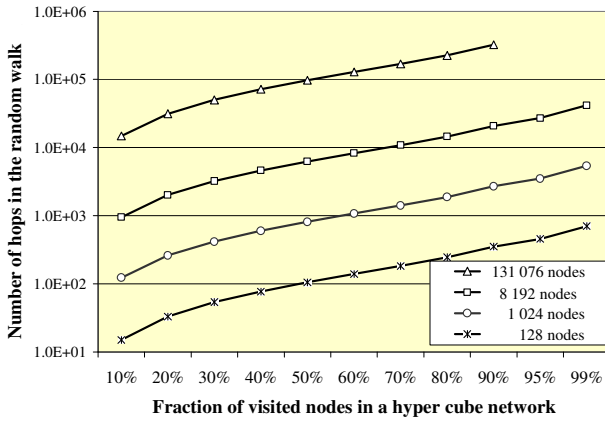


Figure 14: Hyper-Cube: Hop count for network coverage

Note, that in some cases the network structure can be exploited to obtain explicit analytical solutions. For example, the random walk in the ring network leads to a binomial distribution of the location of a random walk after m hops being wrapped around the ring, which comes close to a corresponding Gaussian distribution for a large number of hops. In the binary tree network, the distribution of the current level also follows a binomial structure with reflective bounds on both sides. Starting from the root, all nodes in the tree or subtree are visited with the same probability. We also mention the case of a fully meshed network including a circle from each node back to itself. Again, this is not a usual structure of a P2P network, but it can be considered as being most favourable for random walks, since the uniform steady state distribution of the current location is already reached after one hop. When a random walk has visited n nodes, then a new node is entered in the next step with probability $(|V| - n) / |V|$. Therefore the mean number $E(h_k)$ of hops until the random walk has visited k distinct nodes is given by

$$E(h_k) = 1 + \frac{|V|}{|V|-1} + \frac{|V|}{|V|-2} + \dots + \frac{|V|}{|V|-k+1}. \quad (7)$$

The topology of a star with a central node also leads to a simple analysis. Starting from the center, a random walk next visits any

other node with probability $1/(|V| - 1)$ and then returns to the center. This process is periodical, where the walk stays at the center after an even number of steps and at any other node with equal probabilities after an odd number of steps. The number of hops to visit $k+1$ distinct nodes in the star network is almost twice the expression (7) being observed in the full mesh.

4.1 Steady state analysis

In each network example, we let the random walk start at a specific node and observe the time of convergence to steady state. For homogeneous nodes as in the ring, grid and hypercube networks, convergence to steady state develops equivalently from any arbitrary starting point. In the binary tree network, the behaviour is different for a start from the root as compared to a start from a leaf. Iterative application of equation (4) yields the distribution $p_m^{(R)}(a)$ of the location after m hops. The steady state solution $p^{(R)}(a)$ is known from equation (3) and we use

$$\Delta_m = \sum_{a \in V} |p_m^{(R)}(a) - p^{(R)}(a)|$$

as a measure of the distance from the steady state.

There are several cases, where the random walk is periodical with period length 2. This behaviour is observed for the binary tree, alternating between nodes on an even level and nodes on an odd level in subsequent hops. Other examples of a random walk to alternate between different subsets of network nodes are

- the simple ring with an even number of nodes
- both grid networks with an even number of nodes
- and the hypercube.

Then a steady state distribution according to equation (2) does not exist. Instead, we consider the mean of both distributions in two subsequent hops

$$\tilde{p}_m^{(R)}(a) = \frac{p_m^{(R)}(a) + p_{m+1}^{(R)}(a)}{2}, \quad (7)$$

which converges to $\tilde{p}^{(R)}(a) = \lim_{m \rightarrow \infty} \tilde{p}_m^{(R)}(a)$ for alternating random walks with corresponding distance measure

$$\Delta_m = \sum_{a \in V} |\tilde{p}_m^{(R)}(a) - \tilde{p}^{(R)}(a)|. \quad (8)$$

Steady state analysis results are shown for each network type and for different network sizes in the figures. In all cases, we observe a linear convergence, where the distances Δ_m reduce by about some constant factor with increasing m . The convergence speed differs by orders of magnitude, where the ring with pointers and the hypercube get below a distance measure of 10^{-5} within less than 120 hops even for a network size up to $|V| = 1\,000\,000$.

On the other hand, the ring network is still above 0.05 after 10 000 hops already for $|V| = 127$ nodes. Moreover, the hop count until convergence even increases quadratically with the length of the ring. Rings and linear chains have the largest network diameter, defined as the longest of the shortest paths between arbitrary pairs of network nodes. Small diameters and higher node degrees are favourable for the convergence of random walks as well as their ability to cover a large portion of the network nodes in short time.

A remarkably large difference is apparent in figure 5 for the random walk convergence in a binary tree from different starting points at the root or at a leaf. Starting from the root, all nodes at the same level are entered afterwards with equal probability in each step, which is according to the stationary distribution. Then it is sufficient to consider convergence between the complete levels, which is fast since there are less than 20 levels up to a network

size of more than a million nodes. Starting from a leaf, the mean time to enter the root is much longer, since the probability to choose an edge to the next higher level is only 1/3. Then half of the network cannot be reached without traversing the root. Therefore the distribution of node locations remains unbalanced over a long time, adhering to the subtrees where the starting point belongs to.

4.2 Efficiency of random walks for network exploration

In addition to the convergence to steady state, transient analysis with an absorbing node is used to see how long a random walk must be followed until it has visited a predefined portion of the network nodes.

Therefore we assume a uniform starting distribution in each considered case, i.e. $\forall a \in V: p_0^{(R)}(a) = 1 / |V|$.

One figure of each example shows the number of hops required until a specific network node has been explored by the random walk with 10%, 20%, ..., 90%, 95% and 99% probability.

For the binary tree and its extension, those probabilities differ for nodes on each level. We show the results for an arbitrary leaf. The other networks are homogeneous, such that the exploration process is equivalent for all network nodes and the probability levels are equal to the mean portion of the network that has been explored for the obtained hop count.

Again, the random walk is most efficient for the ring with pointers, followed by the hypercube, the extended binary tree evaluated for leaf nodes, the 3-dimensional and the 2-dimensional grid. In case of the binary tree, exploration times by a random walk are about ten times higher in the mean. Finally, it is not unexpected that the simple ring has the worst performance.

The dependency of the hop count on the network size is roughly linear in most cases, except for a quadratic increase for the ring, as has also been observed for convergence in the ring to steady state. Since content or other information in P2P networks is usually replicated on a number of nodes, it is often sufficient to cover only a small part of the network. Especially the most popular data, to which most of the searches refer to, is distributed over many peers. Therefore results for 10% or less coverage are relevant, which are much more promising than the expected time until an almost complete coverage.

Numerical stability of the analysis method is supposed to be favourable, since no subtractions are involved. We only checked the sum of the probabilities of the result, which showed deviations of less than 10^{-10} in all examples. A verified computation using interval arithmetic [14] is considered as an extension of the analysis.

CONCLUSIONS

Transient analysis is investigated as a standard method to evaluate the efficiency of random walks for network exploration, providing an exact probabilistic characterization of the hop-by-hop progress. The favourable computational complexity makes the method

scalable for large size network. Therefore the dependency of the performance on the network size is evaluated from a hundred to 100 000 or partly up to a million nodes. Improved random walks including some memory e.g. in order to avoid stepping back to the previous node or combined with flooding, still can be treated by transient analysis as a potential subject for further study.

REFERENCES

- [1] R. Albert and A.-L. Barabasi, Statistical Mechanics of Complex Networks, *Rev. Mod. Phys.* 74, 47 (2002) 1-54
- [2] A.-L. Barabási and R. Albert, Emergence of scaling in random networks, *Science* 286 (1999) 509-512
- [3] K. Cho, K. Fukuda, H. Esaki and A. Kato, The impact and implications of the growth in residential user-to-user traffic, *Proc. ACM SIGCOMM Conf., Pisa, Italy* (2006)
- [4] D. Coppersmith, U. Feige and J. Shearer, Random Walks on Regular and Irregular Graphs, *SIAM Journal on Discrete Mathematics* 9/2 (1996) 301-308
- [5] U. Feige and Y. Rabinovich, Deterministic approximation of the cover time, *Random Structures and Algorithms*, 23/1 (2003) 1-22
- [6] W. Feller, An introduction to probability theory and its application, Vol. 2, J. Wiley (1971)
- [7] C. Gkantsidis, M. Mihail and A. Saberi, Random walks in peer-to-peer networks, *Proc. IEEE Infocom* (2004)
- [8] C. Gkantsidis, M. Mihail and A. Saberi, Hybrid search schemes for unstructured peer-to-peer networks, *Proc. IEEE Infocom* (2005)
- [9] G. Hasslinger, ISP Platforms under a heavy peer-to-peer workload, *Proc. Peer-to-Peer Systems and Applications*, Eds.: R. Steinmetz and K. Wehrle Springer LNCS 3485 (2005) 369-382
- [10] Q. Lv, P. Cao, E. Cohen, K. Li and S. Shenker: Search and replication in unstructured peer-to-peer networks, *Proc. Internet. Conf. on Supercomputing*, ACM (2002)
- [11] M.F. Neuts, *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*, Dover Pub. Inc. (1995)
- [12] I. Stoica, R. Morris, D. Liben-Nowell, D. Karger, M. Kaashoek, F. Dabek and H. Balakrishnan, Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications, *IEEE/ACM Transactions on Networking* Vol. 11/1, (2003) 17-32
- [13] W. Terpstra, J. Kangasharju, C. Leng and A. Buchmann, BubbleStorm: Resilient, probabilistic and exhaustive P2P search, *ACM SIGCOMM*, Kyoto, Japan (2007)
- [14] D. Traczinski, W. Luther and G. Hasslinger, Polynomial factorization of servers with semi-Markovian workload: Performance and numerical aspects of a verified solution, *Stochastic Models*, Vol. 21/3 (2005) 643-668
- [15] M. Zhong and K. Shen, Popularity-biased random walks for peer-to-peer search under the square root principle, *Proc. 5th Internat. Workshop on P2P systems, IPTPS* (2006)