

Reinforcement Ranking

Hengshuai Yao and Dale Schuurmans

Department of Computer Science

University of Alberta

Edmonton, AB, Canada T6G2E8

hengshua,dale@cs.ualberta.ca

Abstract

We introduce a new framework for web page ranking—reinforcement ranking—that improves the stability and accuracy of Page Rank while eliminating the need for computing the stationary distribution of random walks. Instead of relying on teleportation to ensure a well defined Markov chain, we develop a reverse-time reinforcement learning framework that determines web page authority based on the solution of a reverse Bellman equation. In particular, for a given reward function and surfing policy we recover a well defined authority score from a reverse-time perspective: looking back from a web page, what is the total incoming discounted reward brought by the surfer from the page’s predecessors? This results in a novel form of reverse-time dynamic-programming/reinforcement-learning problem that achieves several advantages over Page Rank based methods: First, stochasticity, ergodicity, and irreducibility of the underlying Markov chain is no longer required for well-posedness. Second, the method is less sensitive to graph topology and more stable in the presence of dangling pages. Third, not only does the reverse Bellman iteration yield a more efficient power iteration, it allows for faster updating in the presence of graph changes. Finally, our experiments demonstrate improvements in ranking quality.

1 Introduction

Page Rank is a dominant link analysis algorithm for web page ranking [27, 24, 26], which has been applied to a wide range of problems in information retrieval and social network analysis [32, 5, 35, 2]. Under Page Rank, authoritativeness is defined by the stationary distribution of a Markov chain constructed from the web link structure [30, 20, 6, 8, 12, 26, 34]. On each page, a model surfer follows a random link, jumping to the linked page and continuing to follow a random link. Thus, pages are treated as arriving in a Markov chain—the next page visited depends only on the page where the surfer currently visits. The rank of a web page is then defined as

the probability of visiting the page in a long run of this random walk. Unfortunately, this simple protocol does not allow the surfer to proceed from a page that has no outgoing links—such pages are called *dangling* pages. In these cases, the Markov chain derived from the link structure of the Web is not necessarily irreducible or aperiodic, which are required to guarantee the existence of the stationary distribution. To circumvent these problems, a *teleportation* operator is introduced that allows the surfer to escape dangling pages by following artificial links added to the Web graph. Teleportation has been widely adopted by literature, leading to the well accepted stationary distribution formulation of authority ranking, see e.g. [22, 23, 20, 10, 30, 16].

However, if we consider real search behavior, teleportation is obviously artificial. It is unnatural to propagate the score of a page to other unlinked pages, thus teleportation contributes a blind regularization effect rather than any real information. In fact, teleportation contradicts the basic hypothesis of Page Rank: through teleportation, pages that are not linked by a page still receive reinforcement from the page. Teleportation was primarily introduced to guarantee the existence of the stationary distribution. In this paper, we show that teleportation is in fact unnecessary for identifying authoritative pages on the Web. First, contrary to widely accepted belief, teleportation is not required to derive a convergent power iteration for global Page Rank style authority scores. Second, as has been widely adopted in the random surfer interpretation for *Page Rank*, teleportation or even random walk is also unnecessary conceptually. We introduce a new approach to defining web page authority that is based on a novel reinforcement learning model that avoids the use of teleportation while remaining well defined. We prove that the authority function is well posed and satisfies a reverse Bellman equation. We also prove that the induced reverse Bellman iteration, which is more efficient than the Page Rank procedure, is guaranteed to converge for any positive discount factor.

In addition to establishing theoretical soundness, we also show that the reinforcement based authority function is less sensitive to link changes. This allows us to achieve faster updates under graph changes, addressing the Page Rank *updating* problem [4] in an efficient new way. As early as 2000, it was observed that 23% of the web pages changed their index daily [1]. Unfortunately, the Page Rank power iteration does not benefit significantly from initialization with the pre-

Algorithm 1 Standard procedure for computing Page Rank: efficient power iteration method that exploits G 's structure.

Initialize x_0

repeat

$$\begin{aligned} x_{k+1} &= cH^T x_k \\ \omega &= \|x_k\|_1 - \|x_{k+1}\|_1 \\ x_{k+1} &= x_{k+1} + \omega v \end{aligned}$$

until desired accuracy is reached

vious stationary distribution [25]. We prove that our authority function can take better advantage of initialization, and yield faster updates to graph changes. Furthermore, we demonstrate that reinforcement ranking can improve on the authority scores produced by Page Rank in a controlled case study.

2 Page Rank

We first briefly review the formulation of Page Rank [6, 8, 12, 26]. Suppose there are N pages in the Web graph under consideration. Let L denote the adjacency matrix of the graph; i.e., $L(i, j) = 1$ if there is a link from page i to page j , otherwise $L(i, j) = 0$. Let H denote the row normalized matrix of L ; let e be the vector of all ones; and let v denote the *teleportation vector*, which is a normalized probability vector (assume column vectors). Finally, let S be a stochastic matrix such that $S = H + av^T$, where the vector a indicates $a_i = 1$ if page i is dangling and 0 otherwise. Here v is a probability vector that is normally set to either e/N or v . Note that adding av^T to the H matrix artificially ‘‘patches’’ the dangling pages that block the random surfer.

The transition probability matrix used by Page Rank is

$$G = cS + (1 - c)ev^T,$$

for a convex combination parameter $c \in (0, 1)$. The matrix G is stochastic, irreducible and aperiodic, and thus its stationary distribution exists and is unique according to classical Markov chain theory. In fact, the Page Rank (denoted by $\bar{\pi}$) is precisely the stationary distribution vector for G , which satisfies $\bar{\pi} = G^T \bar{\pi}$. Page Rank can be interpreted as follows: with probability c the surfer follows a link, otherwise with probability $1 - c$ the surfer teleports to a page according to the distribution v ; the rank of a page is then given by its long run visit frequency. Teleportation is key, since it ensures the chain is irreducible and aperiodic, thus guaranteeing the existence of a stationary distribution for the surfing process.

Unfortunately, the introduction of teleportation causes the matrix G to become completely dense. Power iteration is therefore impractical unless one exploits its special structure in G ; namely that it is a sparse plus two rank one matrices. An efficient procedure for computing Page Rank is given in Algorithm 1 [22, 23]. This algorithm evaluates an equivalent update to $\bar{\pi}_{k+1} = G^T \bar{\pi}_k$, but it avoids using G by implicitly incorporating the scores of the dangling pages and teleportation in computing ω . Note that the issue of accommodating dangling pages in Page Rank has been considered a challenging research issue [13, 6].

3 MDPs and the Value Function

A Markov Decision Process (MDP) is defined by a 5-tuple $(\mathbb{S}, \mathbb{A}, \mathcal{P}^{\mathbb{A}}, \mathcal{R}^{\mathbb{A}}, \gamma)$; where \mathbb{S} denotes a state space; \mathbb{A} is the action space; $\mathcal{P}^{\mathbb{A}}$ is a transition model with $\mathcal{P}^a(s, s')$ being the probability of transitioning to state s' after taking action a at state s ; $\mathcal{R}^{\mathbb{A}}$ is a reward model with $\mathcal{R}^a(s, s')$ being the reward of taking action a in state s and transitioning to state s' ; and $\gamma \in [0, 1)$ is a discount factor [28, 21, 7, 33].

A *policy* π maps a state s and an action a into a probability $\pi(s, a)$ of choosing that action in the state. The *value* of a state s under a policy π is the discounted long-term future rewards received following the policy

$$V^\pi(s) = E_\pi \left\{ \sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, a_{t \geq 0} \sim \pi \right\},$$

where r_t is the reward received by the agent at time t , and E_π is the expectation taken with respect to the distribution of the states under the policy.

The value function satisfies an equality called *Bellman equation*. In particular, for any state $s \in \mathbb{S}$

$$V^\pi(s) = \gamma \sum_{s' \in \mathbb{S}} P^\pi(s, s') V^\pi(s') + \bar{r}^\pi(s), \quad (1)$$

where $P^\pi(s, s')$ is the probability of transitioning from s to s' following the policy, and the $\bar{r}^\pi(s)$ is the expected immediate reward of leaving state s following the policy.

4 The Reinforcement Ranking Framework

We now introduce the *reinforcement ranking* framework, which models search and ranking in terms of an MDP. The framework is composed of the following elements.

The agent and the environment. The agent is a surfer model and the environment is a set of hyperlinked documents on which the surfer explores. That is, we consider the Web to be the environment; the surfer acts by sending requests that are processed by servers on the Web. This is a simple model of everyday surfing that stresses the subjectiveness of the surfer as well as the objective structure of the Web, in contrast to Page Rank which models surfing as a goal-less random walk.

The rewards. According to [33], a reward function ‘‘maps each perceived state (or state-action pair) of the environment to a single number, a reward, indicating the intrinsic desirability of that state’’. Intuitively, a *reward* is a signal that evaluates an action. A surfer can click many hyperlinks on a page. If a clicking leads to a page that satisfies the surfer’s needs, then a large reward is received; otherwise it incurs a small reward. From the perspective of information retrieval, the reward represents information gain from reading a page. The introduction of rewards to search and ranking is important because it highlights the fact that a page has *intrinsic* importance to users. In fact, this is a key difference from what has been pursued in the link analysis literature, which does not normally model pages as having intrinsic values. The reward hypothesis is also important because surfing and search is purposeful in this model—*actions are taken to achieve rewards*. In this paper, we will be considering a special reward function, in which $\mathcal{R}^a(s, s') = r(s')$, where r is a function mapping from the state space to real numbers. This means the reward of transitioning to a state is uniquely determined by the state itself.

The actions and the states. An *action* is the click of a hyperlink on a page. A *state* is a web page. The current state is the current page being visited by the surfer. After a clicking on a hyperlink, the surfer can observe the linked page or a failed connection. For simplicity, we assume that all links are good in this paper. That is, the *next state* is always the page that an action leads to. Therefore, the state space \mathbb{S} is the set of the web pages. The action space on a page s , denoted $\mathbb{A}(s)$, is the set of actions that lead to the linked pages from s . The overall action space is defined by the union of the actions available on each page, i.e., $\mathbb{A} = \cup_{s \in \mathbb{S}} \mathbb{A}(s)$.

The surfing policy and transition model. A surfer policy specifies how hyperlinks are followed at web pages. Based on the above definitions relating web search to an MDP model, we can equate a surfer with a standard MDP policy as specified in Section 3. For web search, we also assume the transitions are *deterministic*; that is, clicking a hyperlink on a particular page always leads to the same successor page, hence $\mathcal{P}^a(s, s') = 1$ for all $a \in \mathbb{A}(s)$. This treatment simplifies the problem without losing generality—it is straightforward to extend our work to the other cases.

4.1 The Authority Function

Given these associations established between web surfing and an MDP, we can now develop a web page authority function in the framework of reinforcement ranking. In particular, we define the authority score of a page to be the rewards accumulated by its predecessors under the surfing policy. That is, for a page $s \in \mathbb{S}$, its authority score under surfing policy π is

$$\begin{aligned} R^\pi(s) &= r(s) + \gamma r^{(1)}(s) + \gamma^2 r^{(2)}(s) + \dots \\ &= \sum_{k=0}^{\infty} \gamma^k r^{(k)}(s), \end{aligned} \quad (2)$$

where γ is the discount factor, $r(s)$ is a reward that is dependent on s , and $r^{(k)}(s)$ is the reward carried from the k -step predecessors of s to s by the policy. Note that in the second equation, $r^{(0)} = r$, and if a page s has no predecessor, $R^\pi(s)$ can be set to $r(s)$ or some other default value. The k -step historical rewards to a state s are defined as follows: $r^{(1)}(s)$ captures the one-step rewards propagated into s

$$r^{(1)}(s) = \sum_{p \in \mathbb{S}} P_{p,s}^\pi r(p);$$

$r^{(2)}(s)$ captures the rewards from the two-step predecessors

$$r^{(2)}(s) = \sum_{p \in \mathbb{S}} P_{p,s}^\pi \sum_{p' \in \mathbb{S}} P_{p',p}^\pi r(p');$$

$r^{(3)}(s)$ captures the 3-step rewards propagated into s

$$r^{(3)}(s) = \sum_{p \in \mathbb{S}} P_{p,s}^\pi \sum_{p' \in \mathbb{S}} P_{p',p}^\pi \sum_{p'' \in \mathbb{S}} P_{p'',p'}^\pi r(p''); \quad \text{etc.}$$

Note that the discount factor γ plays an important role in this model, since it controls the effective horizon over which reward is accumulated. If γ is large, the authority score will consider long chains of predecessors that lead into a page. If γ is small, the authority score will only consider predecessors

that are within a few steps of the page. This gives a new interpretation for the dampening-factor-like in PageRank. Previously it is commonly recognized that the larger the dampening factor in PageRank, the closer the score vector reflects the true link structure of the graph, e.g. see [9, 10, 3, 14, 11]. While the two interpretations do not contradict each other, viewing the dampening/discount factor as a control over the distance of looking back from pages is surely both essential and intuitive.

4.2 The Reverse Bellman Equation

Although the authority score function R^π appears to be similar to a standard value function V^π , they are not isomorphic concepts: the value function (1) is defined in terms of the forward accumulated rewards. The reverse function (2) cannot be reduced to the forward definition (1) because the transition probabilities are not normalized in both directions; they are only normalized in the forward direction. In particular, (1) is an expectation, whereas (2) cannot be an expectation in general. Despite this key technical difference, it is interesting (and ultimately very useful) that the authority function also satisfies a reverse form of Bellman equation.

Theorem 1 (Reverse Bellman Equation) *The authority function R^π satisfies the reverse Bellman equation for all s :*

$$R^\pi(s) = \gamma \sum_{p \in \mathbb{S}} P_{p,s}^\pi R^\pi(p) + r(s). \quad (3)$$

Proof: First observe that the k -step rewards can be expressed in terms of the $(k-1)$ -step rewards; that is

$$r^{(1)}(s) = \sum_{p \in \mathbb{S}} P_{p,s}^\pi r(p), \quad r^{(2)}(s) = \sum_{p \in \mathbb{S}} P_{p,s}^\pi r^{(1)}(p), \quad \text{etc.}$$

Therefore, from the definition of R^π in (2), one obtains

$$\begin{aligned} R^\pi(s) &= r(s) + \gamma \left[r^{(1)}(s) + \gamma r^{(2)}(s) + \dots \right] \\ &= r(s) + \gamma \left[\sum_{p \in \mathbb{S}} P_{p,s}^\pi r(p) + \gamma \sum_{p \in \mathbb{S}} P_{p,s}^\pi r^{(1)}(p) + \dots \right] \\ &= r(s) + \gamma \sum_{p \in \mathbb{S}} P_{p,s}^\pi \left[r(p) + \gamma r^{(1)}(p) + \dots \right] \\ &= r(s) + \gamma \sum_{p \in \mathbb{S}} P_{p,s}^\pi R^\pi(p). \quad \square \end{aligned}$$

The standard Bellman equation (1) looks forward from a state to define its value, but equation (3) looks backward from a state to define its authority. Therefore, we call equation (3) the *reverse Bellman equation (RBE)* for short. Similar to PageRank, R^π determines the authority of a page based on its back links. However, R^π is well defined without teleportation. In particular, the surfer model P^π is defined on the link structure only, without any teleportation. Notice that P^π is not necessarily irreducible or aperiodic, in fact it is not even stochastic on rows for dangling pages. Yet, perhaps surprisingly, one is still able to achieve a well defined authority score R^π , which is not possible from the classical Markov chain theory.

Theorem 2 *For $\gamma \in [0, 1)$, any policy π and any bounded reward function r , R^π is finite.*

Algorithm 2 Reverse Bellman iteration for computing R^π : no special treatment is required for dangling pages.

Initialize R_0

repeat

$$R_{k+1} = \gamma(P^\pi)^T R_k + r$$

until desired accuracy is reached

Proof: It can be shown that the spectral radius of γP^π is strictly smaller than that of any well defined policy. Therefore $I - \gamma(P^\pi)^T$ is invertible. Additionally, $(I - \gamma(P^\pi)^T)^{-1} = \sum_{t=0}^{\infty} (\gamma(P^\pi)^T)^t$ by [31, Theorem 1.5]. Therefore, $R^\pi = (I - \gamma(P^\pi)^T)^{-1}r$, hence R^π is finite for any policy and any bounded reward function r . \square

The practical significance of the RBE is that it yields an efficient algorithm for computing R^π , based on a backward version of value iteration as used in dynamic programming and reinforcement learning; see Algorithm 2. To establish the correctness of this algorithm we first need a lemma. Let $\|\cdot\|$ be the L-2 norm, $\|R_1\| = (\sum_{i=1}^N R_1(i)^2)^{1/2}$.

Lemma 1 For any $R \in \mathbb{R}^N$, we have $\|(P^\pi)^T R\| \leq \|R\|$.

Proof:

$$\begin{aligned} \|(P^\pi)^T R\|^2 &= \sum_{i=1}^N \left(\sum_{h=1}^N P_{hi}^\pi R(h) \right)^2 \leq \sum_{i=1}^N \sum_{h=1}^N P_{hi}^\pi R(h)^2 \\ &= \sum_{h=1}^N \sum_{i=1}^N P_{hi}^\pi R(h)^2 = \sum_{h=1}^N R(h)^2 = \|R\|^2. \square \end{aligned}$$

Note here we used the ordinary L-2 norm rather than the weighted L-2 norm, as is common in reinforcement learning.

Theorem 3 For $\gamma \in [0, 1)$ and finite r , Algorithm 2's update has a unique fixed point to which the iteration must converge.

Proof: The proof follows the Banach fixed-point theorem given in [7]. Define $T^\pi : \mathbb{R}^N \rightarrow \mathbb{R}^N$ be a mapping by, $T^\pi(R^\pi) = \gamma(P^\pi)^T R^\pi + r$. T^π is a contraction mapping in the L-2 norm because

$$\|T^\pi(R_1) - T^\pi(R_2)\| = \gamma\|P^\pi(R_1 - R_2)\| \leq \gamma\|R_1 - R_2\|,$$

according to Lemma 1. It follows that the iteration converges to the unique fixed point $R^\pi = T^\pi(R^\pi)$. \square

This approach to computing an authority ranking has several advantages over Page Rank. First, Algorithm 2 does not compute an additional ω factor (which requires $2N$ additional flops per iteration). Second, no special treatment is required for dangling pages, which has generally been considered tricky for Page Rank [13, 6]. Finally, there is a significant improvement in computation cost and sensitivity for Algorithm 2 over Algorithm 1.

5 Sensitivity

To assess the relative sensitivities of Page Rank and reinforcement ranking to changes in the graph topology, we establish a few useful facts. First, an important feature of the reinforcement based authority function is that it *decomposes* over disjoint subgraphs.

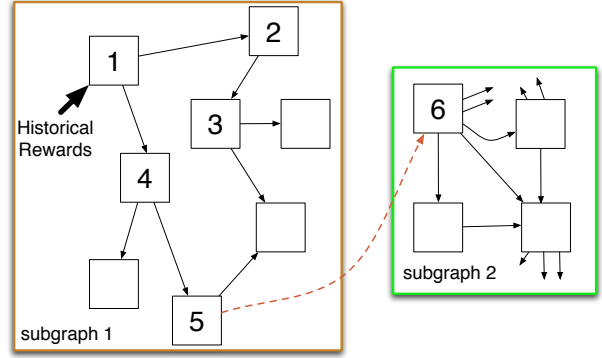


Figure 1: A small graph example.

Proposition 1 (Disjoint Independence) For a graph consisting of separate subgraphs, the R^π vector is given by the union of the local R^π vectors over the disjoint subgraphs. (Straightforward consequence of the definition.)

As the Web grows, subgraphs are often added that have only limited connection to the remainder of the web. In such cases, the R^π score remains largely unchanged, whereas Page Rank is globally affected due to teleportation. In fact, merely increasing graph size affects the Page Rank scores for a fixed subgraph, since the teleportation vector changes.

Another independence property of reinforcement ranking is that the R^π score for *altruistic* subgraphs (subgraphs with only outgoing and no incoming links) is not affected by any external changes to the graph that do not impact altruism.

Proposition 2 (Altruistic Independence) The local R^π vector for an altruistic subgraph cannot be affected by external graph changes, provided no new incoming links to the subgraph are created. (Immediate consequence of the definition.)

Again, Page Rank cannot satisfy altruism independence due to the global effect introduced by teleportation.

Intuitively, separate websites (i.e. separate subgraphs) grow in a nearly independent manner. Reinforcement ranking is more stable with respect to independent subgraph changes, since the stationary distribution of Page Rank must react globally to even local changes. To illustrate the point, consider the example in Figure 1. First, suppose the link from 5 to 6 is not present; in which case the graph consists of two disjoint subgraphs. For reinforcement ranking, any local changes within the subgraphs (including adding new pages) cannot affect the authority scores in the other subgraph, provided no connecting links are introduced between them. However, the stationary distribution for Page Rank must be affected even by disjoint updates. Next, consider the effect of adding a link from 5 to 6, which connects the two subgraphs. In this case, changes to the right subgraph will still not affect the reinforcement scores of the left subgraph if no new links are introduced from the right to the left, whereas Page Rank is affected. Finally, deleting the link from node 1 to node 4 has no influence on node 2 under reinforcement ranking (only the successors of node 4 are influenced), whereas the Page Rank of node 2 will generally change.

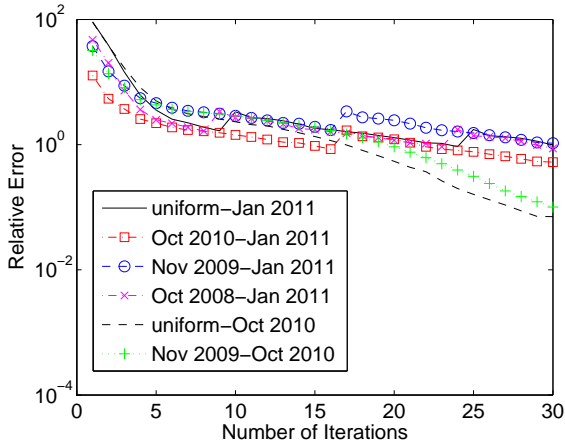


Figure 2: Convergence rate of Page Rank.

The implication is that the reinforcement based authority score is more stable to innocuous changes to the Web graph than Page Rank, which has consequences for both the efficiency of the update algorithms as well as the quality of their respective authority scores, as we now demonstrate.

6 Experimental Results

We conducted experiments on real world graphs (Wikipedia and DBLP) to evaluate two aspects of reinforcement ranking and Page Rank. First, we compared these methods on the *updating problem*: how quickly can the score function be updated given changes to the underlying graph? Second, we investigated the overall quality of the score functions produced. **Sensitivity and the Updating Problem.** Intuitively, the speed with which an iterative method can update its scores for a modified graph is related to the sensitivity of its score function. If the score is not significantly affected by the graph update, then initializing the procedure from the previous scores reduces the number of iterations needed to converge. Conversely, if the new score is significantly different than its predecessor, one expects that many more iterations will be required to converge. Indeed, we find that this is the case: Page Rank demonstrates far more score sensitivity to graph modification, and consequently it is significantly outperformed by reinforcement ranking in the updating problem.

To investigate this issue, we ran experiments on a set of real world graphs extracted from Wikipedia dumps taken at different times. In particular, we used graphs extracted from dumps on Oct-2008, Nov-2009, Oct-2010 and Jan-2011. These are large and densely connected graphs; for example, the Jan-2011 graph contains 6,832,616 articles and 144,231,297 links. For both methods, we used a uniform random surfer policy, and a discount/dampening factor of 0.85. For Page Rank, we set the teleportation vector to uniform, and for reinforcement ranking we used uniform rewards. To evaluate a given method’s ability to cope with graph updates, we measured its rate of convergence to the new solution, as well as the relative advantage of initializing from the previous solution versus initializing uniformly. In particular,

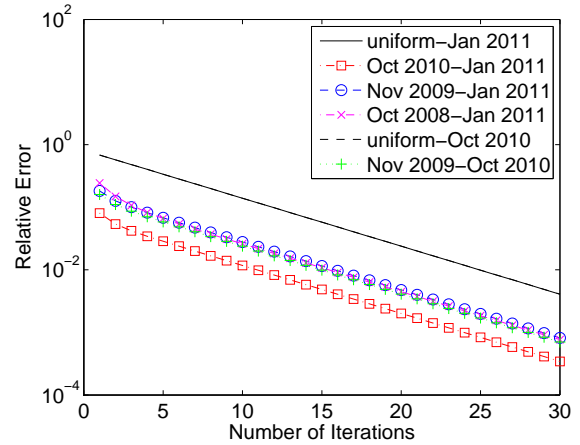


Figure 3: Convergence rate of reinforcement ranking.

the plots show the results for the (initialization, target) pairs:

color	initializer	target	Δ nodes%	Δ links%
red	Oct-2010	Jan-2011	+12, -4	+17, -8
green	Nov-2009	Oct-2010	+18, -5	+39, -20
blue	Nov-2009	Jan-2011	+19, -5	+46, -24
magenta	Oct-2008	Jan-2011	+49, -18	+65, -41

Here, + indicates the percentage of new nodes/links added, and - denotes the percentage nodes/links deleted between the initial and target graphs. Figures 2 and 3 compare the relative rate of convergence of Page Rank versus reinforcement ranking. Note that, given its sensitivity, Page Rank is not able to exploit a previous solution to significantly improve the time taken to converge to a new solution for an updated graph: uniform initialization performs as well. This confirms Google’s report that historical update based power iteration does not improve the accuracy for Page Rank [25]. By contrast, reinforcement ranking exhibits far less sensitivity and therefore demonstrates significantly faster convergence when initialized from a previous graph’s score function. Practically this means that, initialized with a historical update from three months prior, the reinforcement score can be computed about 10 times more accurately than with a uniform initialization.

Ranking Quality. To assess the ranking quality of the two methods we performed an experiment on the DBLP graph [36], which consists of 1,572,278 nodes and 2,083,947 links. We chose this network because citation links are usually reliable, reducing the effects of spam and low quality links. For this experiment, we used the same parameters as before, except that for reinforcement ranking we used a history depth of 3.

To illustrate the ranking quality achieved by Page Rank and reinforcement ranking, we show the highest ranked papers according to each method in Tables 1 and 2 respectively. We used the latest number of citation data retrieved from Google Scholar on March 24, 2013 as the ground truth for paper quality. Note that this oracle considers future citations that are received four years later than the time of the link graph was extracted. In addition, Google Scholar considers much

Table 1: Top papers according to Page Rank.

Rank	Paper Title	#Cites
1	<i>A Unified Approach to Functional Dependencies and Relations</i>	51
2	<i>On the Semantics of the Relational Data Model</i>	167
3	Database Abstractions: Aggregation and Generalization	1518
4	Smalltalk-80: The Language and Its Implementation	5496
5	<i>A Characterization of Ten Hidden-Surface Algorithms</i>	847
6	<i>An algorithm for hidden line elimination</i>	73
7	Introduction to Modern Information Retrieval	9056
8	C4	20913
9	Introduction to Algorithms	30715
10	Compilers: Principles, Techniques, and Tools	11598
11	Congestion avoidance and control	6078
12	A Stochastic Parts Program and Noun Phrase Parser for ...	1314
13	Illumination for Computer Generated Pictures	2504
14	Graph-Based Algorithms for Boolean Function Manipulation	8252
15	Programming semantics for multiprogrammed computations	777
16	Time, Clocks, and the Ordering of Events in a Distributed ...	7720
17	<i>Reentrant Polygon Clipping</i>	373
18	Computational Geometry - An Introduction	8558
19	A Computing Procedure for Quantification Theory	2579
20	A Machine-Oriented Logic Based on the Resolution Principle	4077
21	Beyond the Chalkboard: Computer Support for Collaboration	1079
22	<i>A Stochastic Approach to Parsing</i>	42
23	<i>Report on the algorithmic language ALGOL 60</i>	646

more citation sources than DBLP. Although the results exhibit some noise, it is clear that the Page Rank scores in Table 1 are generally inferior: observe the prevalence of “outlier” papers (*italicized*) that have very few citations. By contrast, the reinforcement based ranking in Table 2 completely avoids papers with low citation counts. Due to the relative purity of the links in this graph, it is reasonable to expect a shallow history depth of 3 should be sufficient to safely identify influential papers in the reinforcement approach. On the other hand, Page Rank which considers long term random walks, appears to be derailed by noise in the graph and produces more erratic results.

7 Discussion

A key challenge faced by Page Rank is coping with dangling pages. Although some dangling pages genuinely do not have any outlinks, many are left “dangling” simply because crawls are incomplete. In practice, the number of dangling pages can even dominate the number of non-dangling pages [13]. Page et al. (1998) first removed dangling pages (and the links to them) before computing the Page Rank for the remaining graph, re-introducing dangling pages afterward. Such a process, however, does not compute the Page Rank on the original graph. Moreover, removing dangling pages produces more dangling pages. In general, many approaches have been proposed to solve this problem, but it does not appear to be definitively settled for Page Rank; see, e.g., [13, 6]. This is not a challenge for reinforcement ranking.

Recently, versions of Page Rank have been formulated using linear system theory (e.g., see [15, 6, 25]). However, the justification for these formulations inevitably returns to random walks, teleportation, and the resulting stationary distributions. As we have observed, such foundations tend to lead to globally sensitive ranking methods. Our work explains and justifies a linear system formulation in a different way. We generalize the teleportation vector to rewards that evaluate the intrinsic importance of individual pages. Moreover,

Table 2: Top papers according to R_3 (3-step history).

Rank	Paper Title	#Cites
1	C4	20913
2	Introduction to Algorithms	30715
3	Introduction to Modern Information Retrieval	9056
4	Smalltalk-80: The Language and Its Implementation	5496
5	Compilers: Principles, Techniques, and Tools	11598
6	Graph-Based Algorithms for Boolean Function Manipulation	8252
7	Computational Geometry - An Introduction	8558
8	Congestion avoidance and control	6078
9	Time, Clocks, and Ordering of Events in Distributed Sys...	7720
10	Induction of Decision Trees	11561
11	Mining Association Rules between Sets ...	12342
12	A Performance Comparison of Multi-Hop Wireless ...	4936
13	Fast Algorithms for Mining Association Rules ...	13827
14	Highly Dynamic Destination-Sequenced ... Routing ...	6731
15	A Stochastic Parts Program and Noun Phrase ...	1314
16	Support-Vector Networks	10523
17	A Machine-Oriented Logic Based on Resolution Principle	4077
18	A Theory for Multiresolution Signal Decomposition...	15897
19	An information-maximization approach to blind separation ...	5871
20	The Anatomy of a Large-Scale Hypertextual Web Search ...	10122
21	The Complexity of Theorem-Proving Procedures	4876
22	Combinatorial Optimization: Algorithms and Complexity	7050
23	A Computing Procedure for Quantification Theory	2579

we have related the linear systems formulation to work in dynamic programming and reinforcement learning, via an accumulative rewards-based score function. It has previously been observed that using a c near 1 in this linear formulation still “often” converges, but the reason has not been well understood [15, 16]. However, we have shown that the authority function can be well defined and guaranteed to converge for any discount factor in $(0, 1)$ and any well-defined surfing policy, without using teleportation.

There have been many attempts to formulate teleportation for more sophisticated ranking, such as personalization [27, 20], query-dependent [30, 29], context-sensitive [18, 19], and battling-link-spam ranking [17]. For example, the personalized Page Rank surfer teleports to the bookmarks of a user. However, these practice still rely on the the stationary distribution formulation for convergence. In fact, all these can be even more naturally expressed in a reinforcement ranking framework, and thus convergence guaranteed. For example, the preferences of different users can be modeled by different reward functions over pages, influenced by bookmarks. (Such reward functions can even be learned via inverse reinforcement learning, allowing convenient generalization across a large portion of the graph.) We can also explain why the pages linked by the bookmarked pages also receive a high ranking, a fact first observed by [27]. In particular, the nonzero rewards received by a user on their bookmarked pages are also the historical rewards of the successor pages of the bookmarked pages, hence the successor pages are also rewarded.

8 Conclusion

Formulating and viewing Page Rank as the stationary distribution of random walks has been long recognized and practiced. However, to guarantee the existence, stochasticity, ergodicity, and irreducibility of the underlying Markov chain has to be ensured. This is tricky for the case of Web, where there are many dangling pages, sinks, and pages without any incoming links. These problems are important to the theory and prac-

tice of Page Rank, for which there are many solutions and discussions.

We proposed an authority function based on historical rewards. We used rewards to capture the intrinsic importance of pages, without the need of teleporting and constructing well behaved Markov chains. We related the authority function to the value function in dynamic programming and reinforcement learning, and showed that the authority function satisfies a reverse Bellman equation. Thus, at a high level, our work establishes a theoretical foundation for the recent linear system formulation of Page Rank. We proved that our authority function is well defined for any discount factor in $(0, 1)$ and any surfing policy, by referring not to the stationary distribution theory but to the contraction mapping technique. Given that random walk models, a generalization of Page Rank, have been used in various contexts, we believe our work will contribute to the fields of information retrieval and social networks.

References

- [1] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke, and S. Raghavan. Searching the Web. *ACM Transactions on Internet Technology*, 1(1):2–43, 2001.
- [2] L. Backstrom and J. Leskovec. Supervised random walks: predicting and recommending links in social networks. *WSDM*, pages 635–644, 2011.
- [3] R. Baeza-Yates, P. Boldi, and C. Castillo. Generic damping functions for propagating importance in link-based ranking. *Internet Math.*, 3(4):445–478, 2006.
- [4] B. Bahmani, R. Kumar, M. Mahdian, and E. Upfal. PageRank on an evolving graph. *KDD*, pages 24–32, 2012.
- [5] H. Bao and E. Y. Chang. Adheat: an influence-based diffusion model for propagating hints to match ads. *WWW*, pages 71–80, 2010.
- [6] P. Berkhin. A survey on PageRank computing. *Internet Mathematics*, 2(1):73–120, 2005.
- [7] D. P. Bertsekas and J. N. Tsitsiklis. *Neuro-dynamic Programming*. Athena, 1996.
- [8] M. Bianchini, M. Gori, and F. Scarselli. Inside PageRank. *ACM Transactions on Internet Technologies*, 5(1):92–128, 2005.
- [9] P. Boldi, M. Santini, and S. Vigna. PageRank as a function of the damping factor. *WWW*, 2005.
- [10] P. Boldi. Totalrank: ranking without damping. *WWW*, 2005.
- [11] M. Bressan and E. Peserico. Choose the damping, choose the ranking? *J. of Discrete Algorithms*, 8(2):199–213, 2010.
- [12] M. Brinkmeier. PageRank revisited. *ACM Transactions on Internet Technology*, 6(3):282–301, 2006.
- [13] N. Eiron, K. S. McCurley, and J. A. Tomlin. Ranking the web frontier. *WWW*, 2004.
- [14] Hwai-Hui Fu, Dennis K. J. Lin, and Hsien-Tang Tsai. Damping factor in Google page ranking. *Applied Stochastic Models in Business and Industry*, 22:431–444, 2006.
- [15] D. Gleich, L. Zhukov, and P. Berkhin. Fast parallel PageRank: A linear system approach. Technical report, Yahoo! Research Labs Technical Report, YRL-2004-038, 2004.
- [16] D. F. Gleich, A. P. Gray, C. Greif, and T. Lau. An inner-outer iteration for PageRank. *SIAM Journal of Scientific Computing*, 32(1):349–371, 2010.
- [17] Z. Gyöngyi, H. Garcia-Molina, and J. Pedersen. Combating web spam with Trustrank. *VLDB*, 2004.
- [18] T. H. Haveliwala. Topic-sensitive PageRank. *WWW*, 2002.
- [19] T. Haveliwala. *Context-Sensitive Web Search*. PhD thesis, Stanford University, 2005.
- [20] G. Jeh and J. Widom. Scaling personalized web search. *WWW*, 2003.
- [21] L.P. Kaelbling, M.L. Littman, and A.W. Moore. Reinforcement learning: A survey. *JAIR*, 4:237–285, 1996.
- [22] S. Kamvar, T. Haveliwala, and G. Golub. Adaptive methods for the computation of PageRank. Technical report, Stanford University, 2003.
- [23] S. D. Kamvar, T. H. Haveliwala, Christopher D. Manning, and Gene H. Golub. Extrapolation methods for accelerating PageRank computations. *WWW*, 2003.
- [24] J. Kleinberg. Authoritative sources in a hyperlinked environment. *SODA*, 1998.
- [25] A. N. Langville and C. D. Meyer. *Google’s PageRank and Beyond: The Science of Search Engine Rankings*. Princeton University Press, 2006.
- [26] B. Liu. *Web Data Mining: Exploring Hyperlinks, Contents and Usage data*. Springer, 2007.
- [27] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford University, 1998.
- [28] M.L. Puterman. *Markov Decision Processes—Discrete Stochastic Dynamic Programming*. J. Wiley & Sons, Inc., New York, NY, 1994.
- [29] D. Rafiei and A. O. Mendelzon. What is this page known for? Computing web page reputations. *WWW*, 2000.
- [30] M. Richardson and P. Domingos. The intelligent surfer: Probabilistic combination of link and content information in PageRank. *NIPS*, 2002.
- [31] Y. Saad. *Iterative Methods for Sparse Linear Systems*. SIAM, 2003.
- [32] P. Sarkar and A. W. Moore. Fast dynamic reranking in large graphs. *WWW*, 2009.
- [33] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- [34] R. S. Wills and I. C. F. Ipsen. Ordinal ranking for Google’s PageRank. *SIAM J. Matrix Anal. Appl.*, 30(4):1677–1696, 2008.

- [35] E. Yan and Y. Ding. Discovering author impact: A PageRank perspective. *Information Processing & Management*, 47:125–134, 2011.
- [36] R. Yan, J. Tang, X. Liu, D. Shan, and X. Li. Citation count prediction: Learning to estimate future citations for literature. *CIKM*, 2011.