Fast ranking algorithm for very large data

Dohy Hong
Alcatel-Lucent Bell Labs
Route de Villejust
91620 Nozay, France
{dohy.hong}@alcatel-lucent.com

Abstract—In this paper, we propose a new ranking method inspired from previous results on the diffusion approach to solve linear equation. We describe new mathematical equations corresponding to this method and show through experimental results the potential computational gain. This ranking method is also compared to the well known PageRank model.

Keywords—Large sparse matrix, Iteration, Fixed point, PageRank, Ranking.

I. INTRODUCTION

Inspired from the previous research results on the diffusion approach [2], [4] to solve fixed point problem in linear algebra, we propose here a new data ranking definition and algorithm. This result can be seen as a mix of PageRank solution [5], diffusion approach [2] and path diversity idea [3].

In Section II, we define the notations and the theoretical framework. Section III show the first experimental results, including the comparison to PageRank ranking.

II. ALGORITHM DESCRIPTION

A. Notations

We will use the following notations:
- \( P \in \mathbb{R}^{N \times N} \) a real matrix;
- \( I \in \mathbb{R}^{N \times N} \) the identity matrix;
- \( J \), the matrix with all entries equal to zero except for the \( i \)-th diagonal term: \((J)_ii = 1\);
- \( \Omega = \{1, \ldots, N\} \);
- \( L = \{i_1, i_2, \ldots, i_n, \ldots\} \) the sequence of nodes for the update (diffusion): \( i_k \in \Omega \);
- \( E = (1, \ldots, 1)^T \in \mathbb{R}^N \); 
- \( L_1 \)-norm: if \( X \in \mathbb{R}^N \), \(|X| = \sum_{i=1}^{N} |x_i|\);
- We assume that \( P \) is the matrix associated to the directed graph on \( \Omega \), for instance, the PageRank matrix (i.e. the transition matrix multiplied by the damping factor [5]).

B. Fast ranking algorithm

The proposed ranking algorithm is based on the iteration of the double equations on history \( H_n \) and fluid \( F_n \) vectors:

\[
H_0 = 0 \\
H_n = H_{n-1} + (\text{int})F_n \quad (1)
\]

and

\[
F_0 = \alpha(1, \ldots, 1)^T \\
F_n = F_{n-1} + (P - I)J_{i_n}((\text{int})F_{n-1}) \quad (2)
\]

The above equations can be easily interpreted as: we apply exactly the algorithm of D-iteration [2], but we only diffuse the integer part of fluids. The Jacobi iterations of the above equations are defined by:

\[
H_n = H_{n-1} + ((\text{int})F_n - F_n) \\
F_n = F_{n-1} + (P - I)((\text{int})F_{n-1}).
\]

If \( P \) is a non negative matrix with spectral radius less than unity, it is obvious to see that \( H_n \) and \( H_n \) converge (\( H_n \) is non-decreasing bounded by PageRank vector) in a finite number of steps (if not, \( H_n \) would be unbounded) and the proposed ranking (FR) method is based on \( H_\infty + F_\infty. F_\infty \) can be seen as a tie-breaker, but one may also use \( H_\infty \). One may also consider a personalized PageRank flavour extension replacing \( F_0 \) by any other initial vector \( V \). One of this approach’s advantage is to be not very dependent on the choice of the damping factor, both for the ranking and the computation speed (cf. Table III).

To solve the above equations, we will apply the diffusion approach [2]. This means in particular that this computation method will be naturally suited for the asynchronous parallel computation, as it was for D-iteration.

Note that if one would diffuse all fluid retained in \( F_\infty \), \( H_\infty \) would be exactly the PageRank vector. The motivations of using \((\text{int})F_{n-1}\) instead of \( F_{n-1} \) are:

- ranking quality improvement: indeed, as it has been shown in [3], we think that the original PageRank vector may be too much influenced by what we could call self-estimation. In presence of loops (self-loop or loops of longer length), a part of scores that are inherited will be returned to the sender, which is not necessarily the desired property;
- computation/convergence acceleration cf. Figure I;
- computation/convergence acceleration for the ranking updates when the graph (or matrix \( P \)) evolves in time.

If \( \alpha \) goes to infinity, the proposed ranking vector converges to PageRank vector. Therefore, PageRank can be seen as a particular case \( \alpha \rightarrow \infty \) of our model where \( \alpha \) tunes the desired influence of loops on the ranking score. Finally, we have also the following interesting bound on the error:

**Theorem 1**: \( (1 - \frac{d}{2})H_\infty \) is an approximation of the PageRank vector with \( L_1 \)-norm error bounded by \( 1/(\alpha - 1) \).

**Proof**: The proof is based on the monotone property of the diffusion. If we denote by \( H(\alpha E, \beta) \) the limit of FI with
initial condition $\alpha E$ and diffusion of the $\beta$-integer part of $F_n$: $(\text{int})(F_n/\beta) \times \beta$, then we have $H(\alpha E, 1) = \alpha H(E, 1/\alpha) = \alpha^2 H(E/\alpha, 1/\alpha^2)$ etc. Let’s call $X$ the PageRank vector. Then, $X$ can be obtained from the diffusion of $F(\alpha E, 1)$ plus $H(\alpha E, 1)$. Note that the diffusion of $F(\alpha E, 1)$ can be denoted by $H(F(\alpha E, 1), 0)$ (implying $X = H((1-d)/NE, 0) = (1-d)/NH(E, 0)$). Now using $H(F(\alpha E, 1), 0) \leq H(E, 0)$, we have:

$$H(\alpha E, 1) \leq \frac{\alpha N}{(1-d)} X = H(\alpha E, 1) + H(F(\alpha E, 1), 0)$$

and

$$H(\alpha E, 1) \leq \frac{\alpha N}{(1-d)} X \leq H(\alpha E, 1) + H(E, \alpha^{-1}) + H(\alpha^{-1}E, \alpha^{-2}) + ...$$

Therefore

$$H(\alpha E, 1) \leq \frac{\alpha N}{(1-d)} X \leq H(\alpha E, 1) (1 + \alpha^{-1} + \alpha^{-2} + ...) = \frac{\alpha}{\alpha - 1} H(\alpha E, 1)$$

And

$$0 \leq \frac{\alpha N}{(1-d)} X - H(\alpha E, 1) \leq \frac{1}{\alpha - 1} H(\alpha E, 1)$$

which can be rewritten as:

$$0 \leq X - \left(\frac{1-d}{\alpha N}\right) H(\alpha E, 1) \leq \frac{1}{\alpha - 1} \left(\frac{1-d}{\alpha N}\right) H(\alpha E, 1).$$

Since $|H(\alpha E, 1)| \leq \alpha N/(1-d)$, we have:

$$|X - \left(\frac{1-d}{\alpha N}\right) H(\alpha E, 1)| \leq \frac{1}{\alpha - 1}.$$ 

This means that choosing $\alpha = 1000$ would give an error very close to 0.1% for norm $L_1$ but also for each coordinate (the exact bound is $1/999 = 0.001001001...$).

Note that we also have:

$$|X - \left(\frac{1-d}{\alpha N}\right)(H(\alpha E, 1) + F(\alpha E, 1))| \leq \frac{1}{\alpha - 1} - \frac{1-d}{N} |F(E, \alpha^{-1})|.$$ 

III. EXPERIMENTAL EVALUATION

For the experimental evaluation purpose, we took the web graph imported from the dataset uk-2007-05 @1000000 (available on ¶) which has 41,247,159 links on $10^6$ nodes. Below we vary $N$ from $10^3$ to $10^6$ extracting from the dataset the information on the first $N$ nodes. Few graph properties are summarized in Table I:

| L/N | D/N | E/N | O/N | N | max in | max out |
|-----|-----|-----|-----|---|-------|--------|
| $10^4$ | 12.9 | 0.041 | 0.032 | 0.236 | 716 | 130 |
| $10^5$ | 12.5 | 0.008 | 0.145 | 0.114 | 7982 | 751 |
| $10^6$ | 31.4 | 0.027 | 0.016 | 0.175 | 34764 | 3782 |

Table II shows the comparative evaluation of the computation cost in number of iterations (one iteration is here defined as a use of $L$ coordinates of $P$ in the computation) with a target precision of $1/N$ (for $L_1$ norm). We compared the Jacobi iteration, D-iteration (DI, cf. [2]) and the fast ranking algorithm (FI) we propose in this paper. The convergence becomes very slow when the damping factor $d$ is being close to 1 to compute the PageRank vector, whereas our ranking vector can be obtained very efficiently whatever the choice of $d$.

Figure ¶ shows the convergence speed (in number of iterations) of Jacobi, D-iteration (DI) and the proposed (FI) methods: our approach reaches the limit in all cases in less than 2.2 iterations. The convergence efficiency is simply not comparable.

Figures 2 and 3 compare the ranking results obtained with FR (using $H + F$), LOC (local computation: rank equal to the number of incoming links) to PageRank vector on the top $x\%$: on y-axis, it counts the number of common nodes in the top $x\%$ between two ranking methods, then it is divided by the number of compared elements (nodes). The fifth curve shows the common elements proportion we observe between two PageRank vectors using a damping factor of 0.9 and 0.8. Our FR ranking vector can be seen as an approximation of PageRank vector, since it converges to PageRank vector for...
large $\alpha$: however, by its definition, it tends to eliminate self-ranking aspects due to the presence of loops (a part of scores that I give to children nodes is coming back to me). Therefore, the parameter $\alpha$ is meant to tune the influence of loops in the ranking score and PageRank can be seen as a particular case $\alpha \to \infty$. Globally, we see that our ranking vector preserves very well the top ranked web sites (for $N = 10^6$, we see that $\alpha = 2$ is close enough already to PageRank vector, always above 92%), because they are likely to be pointed by many different and relevant other web sites: FR ranking vector includes by its definition features and ideas of the path diversity mechanism proposed in [3], when $\alpha$ is closer to 1, but with a computation cost that is greatly reduced (whereas the ideas in [3] requires more computation cost than PageRank vector computation).

Even though it is hard to justify theoretically, the author believe that a choice of $\alpha$ between 1 and 2 are the most appropriate in terms of the optimal compromise between computation cost and ranking relevancy.

Applying the diffusion method on this new ranking vector, we showed that a very efficient computation can be obtained while targeting a relevant ranking score as PageRank.


**REFERENCES**

[1] http://law.dsi.unimi.it/datasets.php.
[2] D. Hong. D-iteration method or how to improve gauss-seidel method. arXiv, http://arxiv.org/abs/1202.1163, February 2012.
[3] D. Hong. Statistical reliability and path diversity based pagerank algorithm improvements. arXiv, http://arxiv.org/abs/1202.2393, Feb 2012.
[4] D. Hong, F. Mathieu, and G. Burnside. Convergence of the d-iteration algorithm: convergence rate and asynchronous distributed scheme. arXiv, http://arxiv.org/abs/1301.3007, January 2013.
[5] L. Page, S. Brin, R. Motwani, and T. Winograd. The pagerank citation ranking: Bringing order to the web. Technical Report Stanford University, 1998.

**IV. CONCLUSION**

In this paper, we proposed a new data ranking method and compared its efficiency to the computation of PageRank vector.