Available online at www.sciencedirect.com
SCIENCE DIRECT®
C. R. Acad. Sci. Paris, Ser. I 340 (2005) 393–397
COMPTES RENDUS MATHEMATIQUE
http://france.elsevier.com/direct/CRASS1/

Numerical Analysis

Extrapolation methods for PageRank computations

Claude Brezinski [a], Michela Redivo-Zaglia [b], Stefano Serra-Capizzano [c]

[a] *Laboratoire Paul Painlevé, UMR CNRS 8524, UFR de mathématiques pures et appliquées, université des sciences et technologies de Lille, 59655 Villeneuve d'Ascq cedex, France*

[b] *Università degli Studi di Padova, Dipartimento di Matematica Pura ed Applicata, Via G.B. Belzoni 7, 35131 Padova, Italy*

[c] *Dipartimento di Fisica e Matematica, Università dell'Insubria – Sede di Como, Via Vallegio 11, 22100 Como, Italy*

Received 17 January 2005; accepted 18 January 2005

Presented by Philippe G. Ciarlet

Abstract

The mathematical problem behind Web search is the computation of the nonnegative left eigenvector of a stochastic matrix P corresponding to the dominant eigenvalue 1. This vector is called the PAGERANK vector. Since the matrix P is ill-conditioned, the computation of PAGERANK is difficult and the matrix P is replaced by $P(c) = cP + (1-c)E$, where E is a rank one matrix and c a parameter. The dominant left eigenvector of $P(c)$ is denoted by PAGERANK(c). This vector can be computed for several values of c and then extrapolated at the point $c = 1$. In this Note, we construct special extrapolation methods for this problem. They are based on the mathematical analysis of the vector PAGERANK(c). ***To cite this article: C. Brezinski et al., C. R. Acad. Sci. Paris, Ser. I 340 (2005).***

Résumé

Méthodes d'extrapolation pour les calculs de PageRank. Le problème mathématique qui est sous-jacent à la recherche sur le Web est le calcul du vecteur propre gauche non négatif d'une matrice stochastique P correspondant à la valeur propre dominante 1. Ce vecteur s'appelle PAGERANK. Puisque la matrice P est mal conditionnée, le calcul de PAGERANK est difficile et la matrice P est remplacée par $P(c) = cP + (1-c)E$, où E est une matrice de rang 1 et c un paramètre. Le vecteur propre gauche dominant de $P(c)$ est dénoté PAGERANK(c). On le calcule pour plusieurs valeurs de c et ensuite on l'extrapole en $c = 1$. Dans cette Note, on construit des méthodes spéciales d'extrapolation pour ce problème. Elles sont basées sur l'analyse mathématique du vecteur PAGERANK(c). ***Pour citer cet article : C. Brezinski et al., C. R. Acad. Sci. Paris, Ser. I 340 (2005).***

E-mail addresses: Claude.Brezinski@univ-lille1.fr (C. Brezinski), Michela.RedivoZaglia@unipd.it (M. Redivo-Zaglia), stefano.serrac@uninsubria.it (S. Serra-Capizzano).

1631-073X/$ – see front matter
doi:10.1016/j.crma.2005.01.015

Version française abrégée

Le problème mathématique qui est sous-jacent à la recherche sur le Web est le calcul du vecteur propre gauche non négatif d'une matrice stochastique P correspondant à la valeur propre dominante 1. Ce vecteur s'appelle PAGERANK. Puisque la matrice P est mal conditionnée, le calcul de PAGERANK est difficile et la matrice P est remplacée par $P(c) = cP + (1-c)E$, où E est une matrice de rang 1 et c un paramètre. Le vecteur propre gauche dominant de $P(c)$ est dénoté PAGERANK(c). Naturellement $P(1) = P$.

Une stratégie consiste à calculer ce vecteur pour plusieurs valeurs de c et ensuite à extrapoler les vecteurs ainsi obtenus en $c = 1$. Naturellement cette stratégie ne pourra donner de bons résultats que si la fonction sur laquelle est basée l'extrapolation est bien choisie, c'est-à-dire si elle est suffisamment proche du comportement exact des vecteurs PAGERANK(c) en fonction de c. Ce comportement à été étudié dans [6] où il est montré que ces vecteurs sont des fractions rationnelles du paramètre c. Dans cet Note, on commence par discuter et approfondir certains aspects de ce résultat. On discute également quelle est la limite quand c tend vers 1 du vecteur PAGERANK(c). Ensuite, à partir de ce résultat, on bâtit deux méthodes d'extrapolation spécialement adaptées.

La première de ces méthodes est une méthode scalaire appliquée séparément sur chaque composante des vecteurs PAGERANK(c). On effectue le changement de variable $d = 1/(1-c)$ et l'on utilise le ϱ-algorithme pour effectuer une extrapolation à l'infini composante à composante par des fractions rationnelles dont numérateur et dénominateur sont de degré $k \leqslant n$, où n est la dimension de PAGERANK.

La seconde procédure est vectorielle. On interpole les vecteurs PAGERANK(c) par une fraction rationnelle dont le numérateur est un polyôme de degré k à coefficients vectoriels et le dénominateur un polynôme de degré k à coefficients scalaires. Les coefficients de ces polynômes sont obtenus grâce à la formule d'interpolation de Lagrange et à la résolution d'un système d'équations linéaires de dimension $k+1$.

Ces deux procédures sont complètement justifiées par les résultats théoriques à partir desquelles elles ont été bâties.

1. Introduction

In Web search, in order to determine the importance of each page, one has to compute a left dominant (nonnegative) eigenvector of an $n \times n$ row stochastic matrix P (called the *exact Google matrix*), that is an eigenvector associated to the eigenvalue 1. A vector satisfying these requirements is called PAGERANK. Here when we write that $\mathbf{x}$ is a right eigenvector of a matrix P we mean that $\mathbf{x}$ is nonzero and that $P\mathbf{x} = \lambda\mathbf{x}$ for some scalar λ; when we write that $\mathbf{y}$ is a left eigenvector of a matrix P we mean that $\mathbf{y}$ is nonzero and that $\mathbf{y}^{\mathrm{T}}P = \lambda\mathbf{y}^{\mathrm{T}}$ for some scalar λ (see e.g. [4]).

In fact, for some modelistic and computational reasons explained in [5] (see also [2]), the matrix P is replaced by the *parametric Google matrix*

$$P(c) = cP + (1-c)E, \quad E = \mathbf{e}\mathbf{v}^{\mathrm{T}},$$

where $\mathbf{e} = (1, \ldots, 1)^{\mathrm{T}}$ and $\mathbf{v}$ is a positive vector with $\|\mathbf{v}\|_1 = 1$. The parameter c belongs to $[0, 1)$ and the unique nonnegative left dominant eigenvector with unitary l^1 norm of this matrix is denoted by PAGERANK(c). Procedures for its computation, based on the power method, were given in [5], and interpreted and extended in [2]. However, when c approaches 1, the convergence of these procedures becomes unacceptably slow, and, moreover, the matrix $P(c)$ becomes ill-conditioned (as $(1-c)^{-1}$); as a consequence, the PAGERANK(c) vectors cannot be computed accurately in a reasonable time. We also have to emphasize that for $c = 1$ we lose the uniqueness of the nonnegative left dominant eigenvector with unitary l^1 norm, that is of the vector PAGERANK. So, among these normalized vectors, we decide to choose the vector $\bar{\mathbf{y}}_1$ computed as $\lim_{c\to 1}$PAGERANK(c).

A possible strategy for PAGERANK consists in computing PAGERANK(c) for different values of c, and then extrapolating these vectors at the point $c = 1$. Obviously, the same extrapolation idea could be used if one needs

to compute PAGERANK(c) for a value of $c \neq 1$, but arbitrarily close to 1. This strategy will only be able to deliver good approximations of PAGERANK if the extrapolation function is well chosen, that is if it is as close as possible to the exact behavior of the vectors PAGERANK(c) with respect to c. This behavior was analyzed by Serra-Capizzano in [6], where the following results were proved:

Theorem 1.1. *Let* $\mathbf{e}, \mathbf{x}_2, \ldots, \mathbf{x}_n$ *be the right eigenvectors of the matrix* P*, and* $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$ *its left eigenvectors corresponding to the eigenvalues* $1, \lambda_2, \ldots, \lambda_n$ *with* $1 \geqslant |\lambda_2| \geqslant \cdots \geqslant |\lambda_n|$.

If P *is diagonalizable*

$$\text{PAGERANK}(c) = \mathbf{y}_1 + (1-c)\sum_{i=2}^{n} \frac{\alpha_i}{1-c\lambda_i}\mathbf{y}_i, \tag{1}$$

with $\alpha_i = \mathbf{v}^{\mathrm{T}}\mathbf{x}_i$*, and where* $\mathbf{y}_1$ *is the basic* PAGERANK *vector* (*i.e. when* $c = 1$).

In the general case

$$\text{PAGERANK}(c) = \mathbf{y}_1 + \sum_{i=2}^{n} w_i(c)\mathbf{y}_i \tag{2}$$

with

$$w_2(c) = (1-c)\alpha_2/(1-c\lambda_2),$$
$$w_i(c) = \big[(1-c)\alpha_i + c\beta_i w_{i-1}(c)\big]/(1-c\lambda_i), \quad i = 3, \ldots, n,$$

and β_i *equal to* 0 *or* 1.

So, the coefficients $w_i(c)$ are rational expressions in c which tend to 0 when c tends to 1.

Let us now discuss in more details the nonuniqueness of the nonnegative left dominant eigenvector with unitary l^1 norm PAGERANK(1). The exact Google matrix P is reducible. Thus its left nonnegative dominating eigenvector with unitary l^1 norm is not unique. On the contrary, the left nonnegative dominating eigenvector PAGERANK(c) with unitary l^1 norm of $P(c)$ with $c \in [0, 1)$ is not only unique, but it is strictly positive.

Moreover, since PAGERANK(c) is a rational expression without poles at $c = 1$ (see Theorem 1.1), there exists a limit of PAGERANK(c) as c tends to 1. Clearly this vector is unique and, consequently, we are solving a special PAGERANK problem. So, we have to characterize the limit vector we are looking for, and to know if this special PAGERANK vector (defined in the convex set of all the possible PAGERANK vectors) has a specific meaning.

A careful study of the set of all normalized PAGERANK(1) vectors was carried out in [6], and, by using Corollary 2.4 of [6], it was proved that formula (2) (corresponding to formula (2.10) in [6]) can be written as

$$\text{PAGERANK}(c) = \mathbf{y}_1 + \alpha_2\mathbf{y}_2 + \cdots + \alpha_t\mathbf{y}_t + \sum_{i=t+1}^{n} w_i(c)\mathbf{y}_i,$$

with $\lim_{c\to 1} w_i(c) = 0$ for $i = t+1, \ldots, n$, and where t denotes the algebraic and geometric multiplicity of the eigenvalue $\lambda = 1$ for the matrix P.

Therefore the unique vector $\bar{\mathbf{y}}_1 = \text{PAGERANK}(1) = \mathbf{y}_1 + (\mathbf{v}^{\mathrm{T}}\mathbf{x}_2)\mathbf{y}_2 + \cdots + (\mathbf{v}^{\mathrm{T}}\mathbf{x}_t)\mathbf{y}_t$ that we will compute as $\lim_{c\to 1}$PAGERANK(c) is only one of the nonnegative normalized dominating eigenvectors of the exact Google matrix. Moreover, this vector depends on $\mathbf{v}$, a configuration which is correct since the personalization vector $\mathbf{v}$ decides which PAGERANK vector is chosen.

Now we are convinced that our formulae concern a vector which is worth computing and therefore the next step is to build extrapolation methods based on the preceding results. The idea of the extrapolation procedure is to start from values of PAGERANK(c) for different values of c (possibly far away from 1), then to compute the unknowns appearing in (1) or (2), and finally to compute $\bar{\mathbf{y}}_1$. In practice, since the dimension n is huge, the expressions in

Theorem 1.1 are simplified by replacing n in the summations by a much smaller value $k+1$. Hence, the preceding strategy will produce approximations of $\bar{\mathbf{y}}_1$ depending on k. This procedure is based on an idea quite similar to the approach followed in [3] for solving ill-conditioned linear systems.

2. Extrapolation methods

Let us first treat the diagonalizable case. Replacing n by $k+1$ in (1) and reducing to the same denominator, we see that we obtain a rational function with a denominator of degree k in c, and a numerator of degree k in c, with vector coefficients. An approximation of $\bar{\mathbf{y}}_1$ will be obtained by interpolating values of PAGERANK(c) by such a rational function and then computing its value at the point $c=1$. A similar result holds by making the change of variable $d=1/(1-c)$ and extrapolating at infinity. An algorithm which performs such an extrapolation is the ϱ-algorithm (see [1] for more details and a FORTRAN subroutine).

Let (c_n) be a sequence of values of c. We set $d_n=1/(1-c_n)$ and $\boldsymbol{\varrho}_0^{(n)}=$ PAGERANK(c_n). We also set $\boldsymbol{\varrho}_{-1}^{(n)}=0$ for all n. The vector ϱ-algorithm consists in computing the vectors $\boldsymbol{\varrho}_k^{(n)}$, component by component, by the relation

$$(\boldsymbol{\varrho}_{k+1}^{(n)})_i=(\boldsymbol{\varrho}_{k-1}^{(n+1)})_i+\frac{d_{n+k+1}-d_n}{(\boldsymbol{\varrho}_k^{(n+1)})_i-(\boldsymbol{\varrho}_k^{(n)})_i},\quad k=0,1,\ldots,\ \text{and } n=0,1,\ldots,$$

where $(\boldsymbol{\varrho}_k^{(n)})_i$ is the ith component of the vector $\boldsymbol{\varrho}_k^{(n)}$. The vectors with an odd lower index are intermediate computations without an interesting meaning, while the vector $\boldsymbol{\varrho}_{2k}^{(n)}$ is the value at infinity of the vector rational function with a numerator and a denominator of degree k which interpolates PAGERANK$(c_n),\ldots,$ PAGERANK(c_{n+2k}). Thus the vectors $\boldsymbol{\varrho}_{2k}^{(n)}$ are approximations of $\bar{\mathbf{y}}_1$.

Let us now describe another algorithm for vector rational extrapolation. We will interpolate PAGERANK(c) by the vector rational function $p(c)=\mathbf{P}_k(c)/Q_k(c)$, where $\mathbf{P}_k$ and Q_k are polynomials of degree k. The coefficients of $\mathbf{P}_k$ are vectors, while those of Q_k are scalars.

Thus, we have to solve the interpolation problem

$$Q_k(c_i)\mathbf{p}_i=\mathbf{P}_k(c_i),\qquad i=0,\ldots,k,$$

with $\mathbf{p}_i=$ PAGERANK(c_i). The polynomials $\mathbf{P}_k$ and Q_k can be computed by the Lagrange's formula

$$\mathbf{P}_k(c)=\sum_{i=0}^{k}L_i(c)\mathbf{P}_k(c_i)$$

$$Q_k(c)=\sum_{i=0}^{k}L_i(c)Q_k(c_i)$$

with

$$L_i(c)=\prod_{\substack{j=0\\ j\neq i}}^{k}\frac{c-c_j}{c_i-c_j},\quad i=0,\ldots,k.$$

We have

$$\mathbf{P}_k(c)=\sum_{i=0}^{k}L_i(c)Q_k(c_i)\mathbf{p}_i.$$

Let $c \neq c_i$ for $i = 0, \ldots, k$. It holds

$$\mathbf{p}(c) = \text{PAGERANK}(c) = \sum_{i=0}^{k} L_i(c) a_i \mathbf{p}_i,$$

with $a_i = Q_k(c_i)/Q_k(c)$.

Let now $\mathbf{s}_0, \ldots, \mathbf{s}_k$ be linearly independent vectors. The coefficients a_i can be computed by solving the system $k+1$ of linear equations in $k+1$ unknowns

$$\sum_{i=0}^{k} L_i(c)(\mathbf{p}_i, \mathbf{s}_j) a_i = \big(\text{PAGERANK}(c), \mathbf{s}_j\big), \quad j = 0, \ldots, k.$$

Then an approximation of $\bar{\mathbf{y}}_1$ is obtained by computing the value at the point $c = 1$ of our interpolating vector rational function, that is

$$\bar{\mathbf{y}}_1 \simeq \sum_{i=0}^{k} L_i(1) a_i \mathbf{p}_i.$$

In the general case, formula (2) also justifies these two procedures with $k = 1$.

Acknowledgements

The work of the second and third author was partially supported by MIUR, grant number 2004015437.

References

[1] C. Brezinski, M. Redivo Zaglia, Extrapolation Methods. Theory and Practice, North-Holland, Amsterdam, 1991.
[2] C. Brezinski, M. Redivo Zaglia, On the acceleration of PageRank computations, submitted for publication.
[3] C. Brezinski, M. Redivo Zaglia, G. Rodriguez, S. Seatzu, Extrapolation techniques for ill-conditioned linear systems, Numer. Math. 81 (1998) 1–29.
[4] G.H. Golub, C.F. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, 1983.
[5] S.D. Kamvar, T.H. Haveliwala, C.D. Manning, G.H. Golub, Extrapolations methods for accelerating PageRank computations, WWW2003, May 20–24, 2003, Budapest, Hungary.
[6] S. Serra-Capizzano, Jordan canonical form of the Google matrix: a potential contribution to the PageRank computation, SIAM J. Matrix Anal., in press. See a preliminary version as Technical Report SCCM-4-3, Stanford University, Stanford, 2004.