Google, comment ça marche ?
Les mathématiques qui se cachent là-derrière[1]

Françoise Valette-Duchêne, ESNE, CPLN, CIFOM et CPMB

1. Introduction

A la recherche d'applications pouvant illustrer un cours de calcul matriciel destiné à des informaticiens de gestion, je me suis intéressée au fonctionnement du moteur de recherche Google que nous connaissons tous bien.

C'était une excellente idée ! J'ai découvert deux inventeurs géniaux bourrés d'idées pour organiser efficacement, en quelques secondes et pour tous à l'échelle mondiale la recherche d'informations dans les milliards de documents disponibles sur le Web.

Les maths qui se cachent derrière appartiennent à la théorie des graphes, au calcul matriciel avec la recherche de valeurs propres et de vecteurs propres, à la probabilité avec une incursion dans les marches aléatoires, ...

Je donne ici une idée générale simple du fonctionnement en terminant par quelques pistes pour approfondir pour ceux qui resteraient sur leur faim.

2. Principe de Google

La société Google a été fondée en septembre 1998 dans la Silicon Valley en Californie par deux jeunes informaticiens de génie **Larry Page** et **Sergey Brin**. Ils s'étaient rencontrés alors qu'ils étaient étudiants en informatique à l'université de Stanford. Leur travail commun de fin d'étude consistait en la conception d'un moteur de recherche sur le Web, appelé BackRub. Au terme de leurs études, ils se sont associés pour perfectionner BackRub, qui est devenu Google, et en vivre.

Google vise à organiser au mieux la recherche d'informations dans la masse gigantesque de documents disponibles sur le Web pour que dans le monde entier, nous puissions tous accéder aux documents susceptibles de nous être les plus utiles.

Ce moteur de recherche se caractérise par sa puissance, son efficacité, sa rapidité, sa fiabilité, sa simplicité d'utilisation et son objectivité pour obtenir les sites répondant au mieux aux demandes des utilisateurs. De plus, il est gratuit.

Le classement est automatisé. Il n'est pas possible de payer la société pour avancer dans le classement. Google est très attaché à ce principe qui garantit aux internautes un classement objectif et fiable.

La publicité payante existe aussi chez Google mais elle est toujours clairement séparée des résultats de la recherche.

Google offre des résultats en 35 langues.

Environ 20 000 personnes travaillent pour Google qui a réalisé en 2007 un chiffre d'affaires de 16,6 milliards de $ (10,7 milliards d'€) et enregistré un bénéfice de 4,2 milliards de $ (2,7 milliards d'€).

[1] Texte de la conférence présentée lors du Congrès de la SBPMef à Waremme le 27 août 2008. A paraître également dans la revue « Losanges » de la Société Belge des Professeurs de Mathématique d'expression française.

La rapidité des recherches s'explique en partie par l'utilisation de milliers de PC travaillant en réseau.

Avant Google, les moteurs de recherche étaient basés sur la fréquence d'apparitions de mots-clés dans les sites qui étaient considérés comme intéressants lorsque le mot recherché y apparaissait souvent.

La grande innovation des concepteurs de Google a été de regarder le Web comme un immense graphe orienté pondéré dont les noeuds sont les pages et les arêtes sont les renvois à d'autres pages (liens hypertextes), dotés d'un poids plus ou moins grand.

Google en tant que moteur de recherche :

- explore le Web pour localiser les pages en accès public,

- parcourt ces pages à la recherche de mots importants qui serviront à les indexer en vue d'une recherche par mots-clés,

- attribue un score > 0 à chaque page dans la base de données, score d'autant plus élevé que la page est considérée comme importante. L'algorithme de classement des pages par importance est appelé algorithme PageRank. C'est cet algorithme qui fait le succès de Google. L'importance qu'il attribue à une page est d'autant plus grande que beaucoup de pages, elles-mêmes importantes, pointent vers elle avec des arêtes de grand poids.

La recherche de pages Web est basée sur des mots-clés. Et lorsqu'une demande de recherche est adressée à Google, celui-ci ne consulte pas l'intégralité des pages du Web, ce serait beaucoup trop long, d'autant plus qu'il traite chaque jour quelques centaines de millions de demandes ! La durée d'une recherche est habituellement inférieure à une demi-seconde !

Un logiciel analyse pour Google tous les mots de chaque page Web et il ne retient que ceux qu'il considère comme importants. Ces mots importants sont les mots-clés à la base de toute recherche sur Google. Lorsque vous introduisez un mot-clé, par exemple « chat », Google consulte l'index des mots-clés qui permet de dresser très vite la liste des pages associées au mot « chat ».

Les pages qu'il affiche ne sont pas présentées n'importe comment. Elles sont classées par ordre décroissant d'un score > 0 qu'un logiciel de classement attribue, suivant l'algorithme PageRank, à chaque page répertoriée. Donc, plus le score est élevé, plus la page est considérée comme importante. Ces scores sont recalculés chaque mois pour tenir compte de l'évolution du Web et il faut une semaine de calculs pour attribuer un score à chaque page.

La fonction « J'ai de la chance » de Google présente la première des pages sélectionnées, celle qui a le score le plus élevé.

3. « Mini-Web et Mini-Google »

Pour comprendre comment fonctionne Google, nous allons illustrer les explications sur un « Mini-Web » de 7 pages qui servira de modèle tout au long de l'article (figure 1).

Les nœuds du graphe représentent les 7 pages du Mini-Web et les arcs partant d'un nœud pointent vers les pages qu'il désigne, c'est-à-dire toutes les pages auxquelles la page fait référence. Les auto-références ne sont pas prises en considération.

Une page est notamment considérée comme d'autant plus importante par Google que beaucoup de sites (autres qu'elle-même) la mentionnent, donc que « son fan-club » est gros, ce qui revient à dire qu'il y a beaucoup d'arcs qui aboutissent à la page.

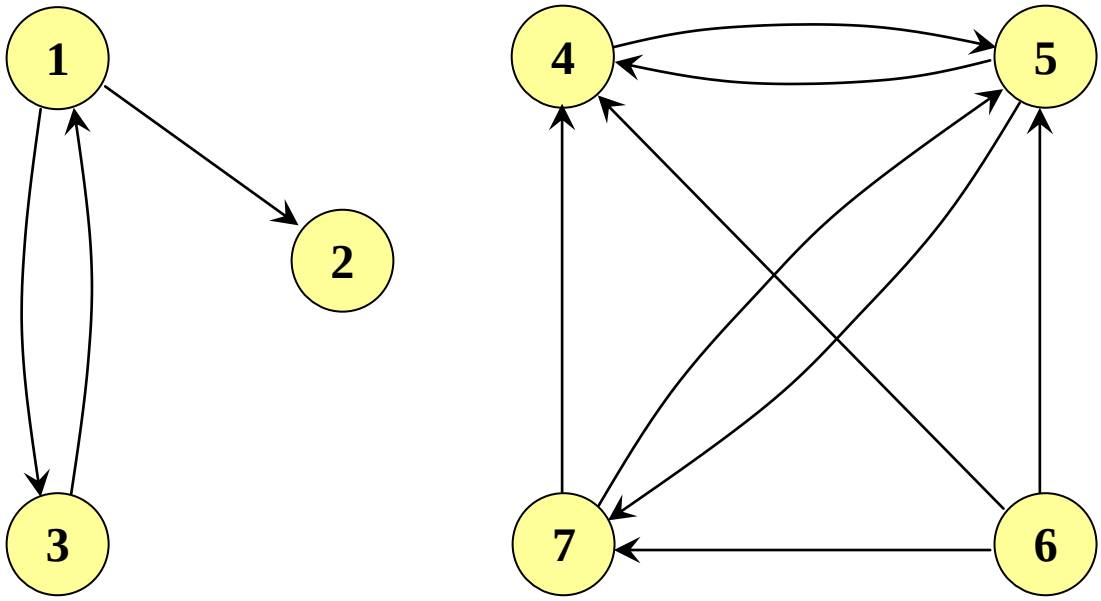

fig 1. Mini-web de 7 pages

Notons *M* la matrice d'adjacence du graphe du Mini-Web.

$$M = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$$

Dans la matrice *M* du Mini-Web :

- une ligne de 0 indique que la page n'en renseigne aucune autre ; c'est une page cul-de-sac (voir la ligne 2 de *M*).

- une colonne de 0 indique qu'aucune page ne pointe vers la page correspondante ; c'est une page qui n'a aucun « supporter » (voir la colonne 6 de *M*).

4. Graphe orienté pondéré de la relation « pointe vers » sur le Web

Les pages du Web renvoient souvent à d'autres pages. Certaines ont beaucoup de supporters, d'autres très peu ou pas du tout.

Il est assez naturel de mesurer l'importance d'une page en considérant le nombre de pages qui pointent vers elle et leur importance. Mais comment ?

Tout d'abord, aux arcs du graphe du Web, Google attache un poids comme suit : Chaque page dispose d'un poids de 1 à répartir équitablement entre toutes les pages vers lesquelles elle pointe.

La figure 2 présente le graphe pondéré du Mini-Web suivant la règle d'importance de Google. *P* est la matrice des poids associés.

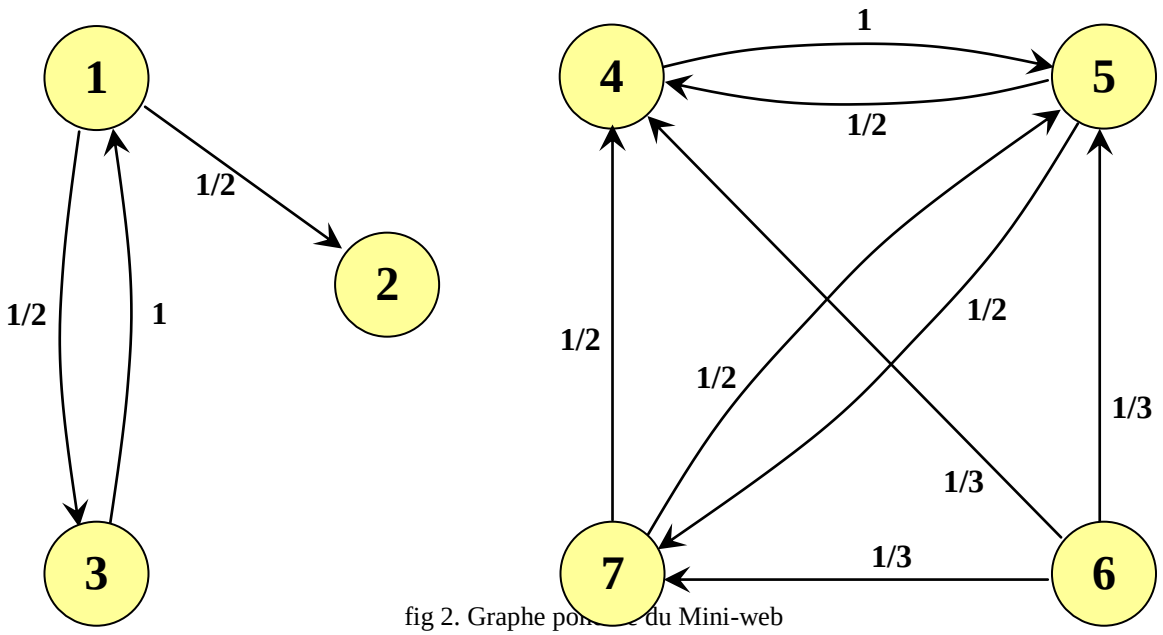

fig 2. Graphe pondéré du Mini-web

$$P = \begin{pmatrix} 0 & 1/2 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1/2 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 1/3 & 1/3 & 0 & 1/3 \\ 0 & 0 & 0 & 1/2 & 1/2 & 0 & 0 \end{pmatrix}$$

La somme des éléments d'une ligne de la matrice *P* des poids vaut 0 ou 1 suivant que la page correspondante pointe vers d'autres ou non.

Donc la somme d'une ligne vaut 1 dès que la page correspondante fait référence à une ou plusieurs autres. Cette page répartit équitablement son poids de 1 entre toutes les pages auxquelles elle fait référence.

Les valeurs des éléments d'une colonne de la matrice *P* des poids dressent la liste des supporters de la page correspondant à la colonne avec les poids ou importances de leurs soutiens (puisqu'un poids est d'autant plus élevé que le « fan » n'a pas cru bon de mentionner beaucoup de pages).

5. Attribution d'un « coefficient d'importance x_i » à chaque page *i* répertoriée sur le Web

Il a paru naturel aux inventeurs de Google de décider que l'importance x_i d'une page *i* est la somme des importances x_j des pages qui pointent vers elle, pondérées par le poids que chaque page *j* accorde à la page *i*.

L'importance x_i d'une page *i* est donc d'autant plus grande que :

- beaucoup de pages pointent vers elle

- elles-mêmes ont une grande importance

- et ces pages lui attribuent un grand poids.

Oui, oui, cette définition semble se mordre la queue : pour calculer une valeur x_i, on utilise toutes les valeurs x_j… Et pourtant, ça marche !

Ainsi, par exemple, l'importance x_5 attribuée par Google à la page 5 du Mini-Web s'écrit en fonction des importances x_j de toutes les pages :

$$x_5 = \sum_{j=1}^{7} p_{5j} \cdot x_j = x_4 + \frac{1}{3}x_6 + \frac{1}{2}x_7$$

Notons X le vecteur-colonne des x_i.

Les importances x_j des pages sont solutions du système de 7 équations à 7 inconnues :

$$\boldsymbol{{}^tP \cdot X = X.}$$

$${}^tP \cdot X = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/3 & 1/2 \\ 0 & 0 & 0 & 1 & 0 & 1/3 & 1/2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 1/3 & 0 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix}$$ et donc le système devient :

$$\begin{pmatrix} x_3 \\ x_1/2 \\ x_1/2 \\ x_5/2 + x_6/3 + x_7/2 \\ x_4 + x_6/3 + x_7/2 \\ 0 \\ x_5/2 + x_6/3 \end{pmatrix} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \end{pmatrix}$$

Si le système ${}^tP \cdot X = X$ admet une solution X non nulle, elle n'est pas unique puisque tout multiple $k \cdot X$ de X est aussi solution. Il est donc raisonnable d'imposer au vecteur X des importances de chacune des pages répertoriées la condition :

$$\sum_{i=1}^{7} x_i = 1$$

De plus, comme une importance est mesurée par un nombre positif, on impose que tous les x_i sont positifs ou nuls.

Avec ces conditions, la solution du système est (pour économiser un peu de place on donne ce vecteur colonne sous forme de vecteur ligne transposé) : $X = \begin{pmatrix} 0 & 0 & 0 & 3/9 & 4/9 & 0 & 2/9 \end{pmatrix}^T$. C'est le vecteur des importances de chacune des pages répertoriées du Mini-Web.

Le classement des x_i par valeurs décroissantes s'explique, au moins en partie, sur le graphe orienté pondéré, ainsi :

- Les pages 4 et 5 sont celles qui ont le plus de supporters, ce sont aussi celles qui reçoivent les plus grands coefficients d'importance. De plus $x_5 > x_4$, ce qui n'est pas surprenant puisque

le total des poids attribués à la page 5 par les pages qui y font référence (1 + 1/3 + 1/2) est supérieur au total des poids attribués à la page 4 par les pages qui pointent vers elle (1/2 + 1/3 + 1/2).

- x_7 est inférieur à x_4 et à x_5 : Il n'y a que 2 pages qui font référence à la page 7 alors qu'il y a 3 pages qui pointent vers chacune des pages 4 et 5.

- $x_6 = 0$: La page 6 est peu importante car aucune page ne pointe vers elle.

- $x_1 = x_2 = x_3 = 0$: Attention ! Ici, il est tentant de conclure que les pages 1, 2 et 3 sont jugées peu importantes car elles ne sont pointées que par une seule page. Mais ce raisonnement s'avère trop rapide car on peut démontrer que les pages culs-de-sac ainsi que toutes celles d'où part un chemin menant à une page cul-de-sac ont par cette méthode un coefficient d'importance nul. Bien entendu, ceci est gênant car certaines de ces pages pourraient être souvent mentionnées et donc mériter un meilleur score.

6. Adaptation à Google de la recherche de la matrice X des importances de chacune des pages répertoriées

Dans l'exemple du Mini-Web, le système ${}^tP \cdot X = X$ a 7 équations à 7 inconnues. Il est encore facile à résoudre.

Mais pour Google, les choses se corsent puisque le système a environ 10 milliards d'équations et d'inconnues et tel quel, il est pratiquement impossible de le résoudre. Gloups !

Les inventeurs géniaux de Google contournent la difficulté en utilisant le résultat suivant :

Théorème de Perron-Frobenius (1907) : Si T est une matrice carrée dont 1) tous les coefficients t_{ij} sont > 0 et 2) la somme de chaque colonne vaut 1, alors le système $T \cdot X = X$ admet une solution unique à coefficients x_i tous > 0 et dont la somme vaut 1.

Autrement dit, 1 est une valeur propre de la matrice T de vecteur propre X à coefficients x_i tous > 0 et dont la somme vaut 1.

Evidemment, un théorème qui garantit l'existence d'une solution unique, c'est parfait pour un théoricien, encore faut-il l'obtenir cette solution !

Heureusement, en plus (et surtout), il existe un algorithme pour construire cette solution, algorithme qui est encore applicable en pratique et en un temps raisonnable lorsque la dimension de T vaut quelques milliards !

Cet algorithme relativement standard est le suivant : Si T est une matrice carrée $n \times n$ dont 1) tous les coefficients t_{ij} sont > 0 et 2) la somme de chaque colonne vaut 1, alors pour tout choix d'un vecteur-colonne $Y \neq 0$ de taille n, tel que tous les $y_i \geq 0$ et la somme des y_i vaut 1, $T^k \cdot Y$ converge (exponentiellement vite) vers l'unique solution du système $T \cdot X = X$ à coefficients x_i tous positifs et de somme égale à 1.

L'intérêt est qu'il s'agit d'itérations du produit matriciel par la matrice T, ce qui s'effectue très facilement sur un ordinateur. On arrête l'itération lorsque les écarts absolus entre les coordonnées de $T^k \cdot Y$ et de $T^{k+1} \cdot Y$ deviennent tous inférieurs à une valeur ε préalablement fixée. L'entreprise Google annonce effectuer un nombre d'itérations compris entre 50 et 100.

C'est cette solution X obtenue par l'algorithme PageRank qui associe un score $x_i > 0$ à chaque page i du Web et permet de les classer par scores ou importances croissants.

Il faut environ une semaine pour obtenir le vecteur X des scores des pages du Web et il est recalculé tous les mois pour s'adapter à l'évolution du Web.

Le classement des pages du Web par ordre d'importance établi par Google est donc indépendant des mots-clés que vous introduisez dans votre recherche.

D'une part, Google associe à chaque page répertoriée du Web une liste de mots-clés qui permet de la trouver très vite lorsque le mot est introduit. D'autre part, le score associé à chaque page permet de classer très rapidement les pages sélectionnées par importance décroissante.

C'est formidable, mais

- la matrice ${}^tP$ a des coefficients ≥ 0 et pas nécessairement > 0

- la somme des colonnes de ${}^tP$ vaut 0 ou 1 et pas nécessairement 1.

Le théorème n'est donc pas applicable sauf si on modifie la matrice ${}^tP$ pour qu'elle satisfasse aux deux conditions.

Et ici, les deux pères de Google ont encore des idées géniales !

a) Ils considèrent que le non-choix des pages culs-de-sac équivaut à choisir toutes les pages, (y compris elles-mêmes) avec le même poids et ils corrigent la matrice ${}^tP$ en conséquence en une matrice ${}^tP'$.

Dans notre exemple, cela donne :

$$ {}^tP' = \begin{pmatrix} 0 & 1/7 & 1 & 0 & 0 & 0 & 0 \\ 1/2 & 1/7 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/7 & 0 & 0 & 1/2 & 1/3 & 1/2 \\ 0 & 1/7 & 0 & 1 & 0 & 1/3 & 1/2 \\ 0 & 1/7 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/7 & 0 & 0 & 1/2 & 1/3 & 0 \end{pmatrix} $$

b) Puis, ils introduisent une nouvelle matrice ${}^tP''$ des poids tout à fait égalitaire, c'est-à-dire qui modélise la situation où toutes les pages renvoient à toutes les pages, y compris à elles-mêmes.

Dans l'exemple du Mini-Web, cela donne :

$$ {}^tP'' = \begin{pmatrix} 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \\ 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 & 1/7 \end{pmatrix} $$

c) Enfin, ils calculent la moyenne pondérée T des 2 matrices ${}^tP'$ (situation réelle) et ${}^tP''$ (situation égalitaire) : $T = \lambda\, {}^tP' + (1 - \lambda)\, {}^tP''$

en accordant à ${}^tP'$ un grand poids ($\lambda = 0{,}85$) et à ${}^tP''$ un petit poids ($1 - \lambda = 0{,}15$) :

$$T = 0{,}85\, {}^tP' + 0{,}15\, {}^tP''$$

Mais pourquoi prendre $\lambda = 0{,}85$? Ce choix résulte d'un compromis basé sur des considérations empiriques : une valeur de λ plus petite donne trop de poids à la matrice « égalitaire » tandis qu'une valeur de λ plus proche de 1 augmente le nombre d'itérations nécessaires lorsqu'on applique l'algorithme pour calculer le vecteur X des importances.

Dans le cas du Mini-Web : $T = 0{,}85\, {}^tP' + 0{,}15\, {}^tP''$

$$T = \begin{pmatrix} 3/140 & 1/7 & 61/70 & 3/140 & 3/140 & 3/140 & 3/140 \\ 25/56 & 1/7 & 3/140 & 3/140 & 3/140 & 3/140 & 3/140 \\ 25/56 & 1/7 & 3/140 & 3/140 & 3/140 & 3/140 & 3/140 \\ 3/140 & 1/7 & 3/140 & 3/140 & 25/56 & 32/105 & 25/56 \\ 3/140 & 1/7 & 3/140 & 61/70 & 3/140 & 32/105 & 25/56 \\ 3/140 & 1/7 & 3/140 & 3/140 & 3/140 & 3/140 & 3/140 \\ 3/140 & 1/7 & 3/140 & 3/140 & 25/56 & 32/105 & 3/140 \end{pmatrix}$$

Il est évident que cette matrice *T* remplit les conditions du théorème (*T* est une matrice carrée dont tous les coefficients t_{ij} sont > 0 et la somme des éléments de chaque colonne vaut 1). Ouf ! Sauvés !

7. Algorithme de résolution

Appliquons l'algorithme de résolution au Mini-Web.

Et puisqu'il faut choisir un vecteur *Y* non nul, autant le choisir très simple et pourquoi pas égalitaire ? Prenons donc : $Y = (1/7 \quad 1/7 \quad 1/7 \quad 1/7 \quad 1/7 \quad 1/7 \quad 1/7)^T$. Après 30 itérations, la valeur obtenue est : $X = (0.0851 \quad 0.0655 \quad 0.0655 \quad 0.2514 \quad 0.3264 \quad 0.0293 \quad 0.1764)^T$. Il est possible de comparer ce résultat à celui obtenu par un calcul exact (tableau 1).

Page	*Nombre de supporters*	*Total des poids*	*Calcul exact de X*	*Calcul de X par l'algorithme*	*Constatations pour les 2 solutions*
1	1	1	0	0,0851	< 0,1 et 4ème nombre par valeurs décroissantes
2	1	0,5	0	0,0655	< 0,1
3	1	0,5	0	0,0655	< 0,1
4	3	1,33	0,3333…	0,2514	2ème nombre par valeurs décroissantes
5	3	1,83	0,4444…	0,3264	nombre le plus grand
6	0	0	0	0,0293	nombre le plus petit
7	2	0,83	0,2222…	0,1764	3ème nombre par valeurs décroissantes
		Total	1	0,9996	

Tableau 1 : Comparaison des vecteurs d'importance *X* obtenus par résolution d'un système d'équations et par application de l'algorithme.

8. Interprétation probabiliste du PageRank

Imaginons un processus aléatoire : un utilisateur du Web dispose d'une pièce de monnaie pipée, qui donne « pile » avec une probabilité 0,85, et « face » avec une probabilité 0,15. Il décide de surfer au hasard à partir de la page de la SBPMef, de la manière suivante : il lance la pièce

- Si elle tombe sur « pile », il choisit au hasard (c-à-d : de façon équiprobable) un des liens de la page de la SBPMef, et clique dessus, ce qui l'amène sur une nouvelle page ;

- si elle tombe sur « face », l'utilisateur se rend au hasard sur une page quelconque du Web.

Notre surfer répète alors indéfiniment l'expérience aléatoire à partir de la dernière page où il est arrivé. Attention, s'il est arrivé sur une page cul-de-sac, on convient qu'il repart au hasard vers une page quelconque du Web. L'importance d'une page, mesurée par l'algorithme PageRank, est la probabilité que ce processus aléatoire arrive sur la page en question, au départ de la page de la SBPMef.

Bibliographie[1]

Brin, S. & Page, L. (1998). The anatomy of a large-scale hypertextual Web search engine. *Proceedings of the seventh international conference on World Wide Web* 7, 107-117. http://www-db.stanford.edu/~backrub/google.htm

Bryan, K. & Leise, T. (2006). The $ 25.000.000.000 Eigenvector. The Linear Algebra behind Google. *SIAM Reviews*, 48 (3), 569-581.

http://www.rose–hulman.edu/~bryan/googleFinalVersionFixed.pdf

[1] Le lecteur intéressé à la saga humaine qui a accomagné celle du pageRank pourra aussi consulter : Vise, D.A. (2005). *The Google story*. Oxford: Pan Books. Un historique est également disponible à l'adresse http://fr.wikipedia.org/wiki/PageRank (ndlr).