Université d'Aix-Marseille **Analyse numérique, semestre 5**
Licence de maths, 3ème année

Projet PageRank

L'objectif de ce projet est d'étudier sur l'exemple du classement des pages web, la méthode de la puissance. Ce sujet est fortement inspiré d'un sujet d'agreg option B, 2008.

La recherche d'informations pertinentes sur le Web est un des problèmes les plus cruciaux pour l'utilisation de de ce dernier. Des enjeux économiques colossaux sont en jeu, et diverses multinationales se livrent à de grandes manoeuvres. Le leader actuel de ce marché, Google, utilise pour déterminer la pertinence des références fournies, un certain nombre d'algorithmes dont certains sont des secrets industriels jalousement gardés, mais d'autres sont publics. On va s'intéresser ici à l'algorithme PageRank [1], lequel fait intervenir des valeurs propres et vecteurs propres d'une énorme matrice.

Le principe de l'algorithme PageRank

On peut considérer pour simplifier que le Web est une collection de N pages, avec N très grand (de l'ordre de 10^{10} en octobre 2005). La plupart de ces pages incluent des liens hypertextes vers d'autres pages. On dit qu'elles pointent vers ces autres pages. L'idée de base utilisée par les moteurs de recherche pour classer les pages par ordre de pertinence décroissante consiste à considérer que plus une page est la cible de liens venant d'autres pages, c'est-à-dire plus il y a de pages qui pointent vers elle, plus elle a de chances d'être fiable et intéressante pour l'utilisateur final, et réciproquement. Il s'agit donc de quantifier cette idée, c'est-à-dire d'attribuer un score de pertinence à chaque page.

Pour ce faire, on représente le Web comme un graphe orienté : un *graphe* est un ensemble de points, dont certaines paires sont directement reliées par un "lien". Ces liens peuvent être orientés, c'est-à-dire qu'un lien entre deux points u et v relie soit u vers v, soit v vers u : dans ce cas, le graphe est dit orienté. Sinon, les liens sont symétriques, et le graphe est non-orienté. Les points sont généralement appelés sommets, et les liens "arêtes". On peut représenter le graphe par une matrice, qu'on appelle matrice d'adjacence : le coefficient de la i-ème ligne et j-ème colonne est 1 s'il existe une arête allant du sommet i au sommet j, et 0 sinon. Remarquer que si le graphe est non orienté, alors une arête est définie comme liant les sommets i et j sans ordre, et la matrice d'adjacence est symétrique.

1. Le nom "PageRank" vient du nom de Larry Page, l'un des inventeurs de Google.

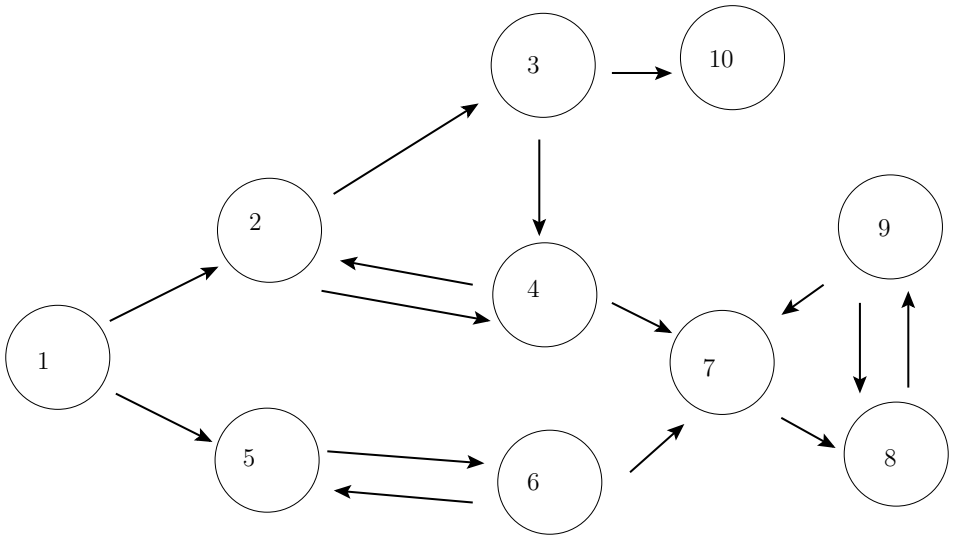

FIGURE 1 – Exemple 1

Pour représenter le Web comme un graphe, on se donne donc un ordre arbitraire sur l'ensemble des pages que l'on numérote ainsi de $i = 1$ à $i = N$: ce sont les sommets du graphe. La structure de connectivité du Web peut alors être représentée par la matrice d'adjacence C de taille $N \times N$ telle que $C_{ij} = 1$ si la page j pointe sur la page i, $C_{ij} = 0$ sinon. Remarquons que c'est un graphe non symétrique. Les liens d'une page sur elle-même ne sont pas significatifs, on pose donc $C_{ii} = 0$. Remarquons que la ligne i de la matrice C contient tous les liens significatifs qui pointent sur la page i, alors que la colonne j contient tous les liens significatifs qui partent de la page j.
Pour illustrer notre étude, on propose les trois exemples de graphe décrits sur les figures 1, 2 et 3. .

1. *Ecrire les matrices C_1, C_2 et C_3 associées respectivement aux exemples 1, 2 et 3.*

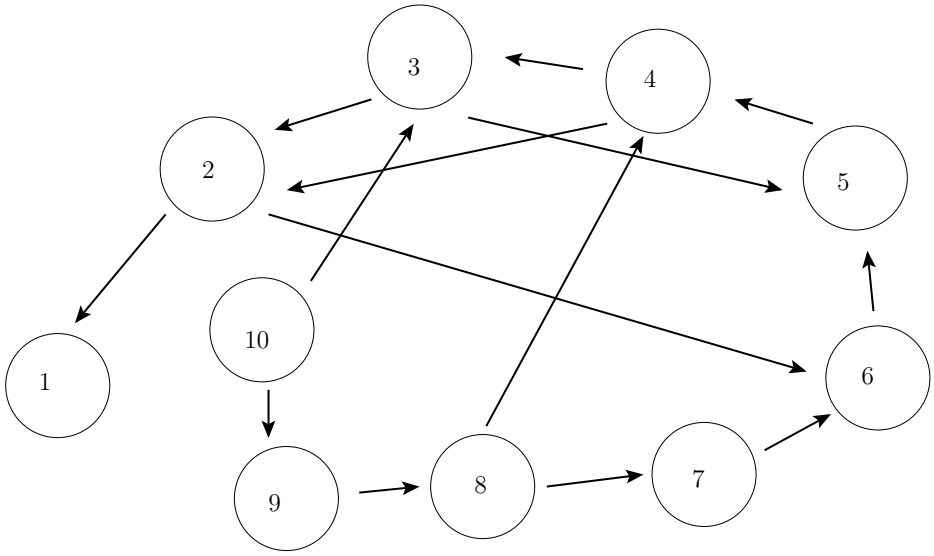

FIGURE 2 – Exemple 2

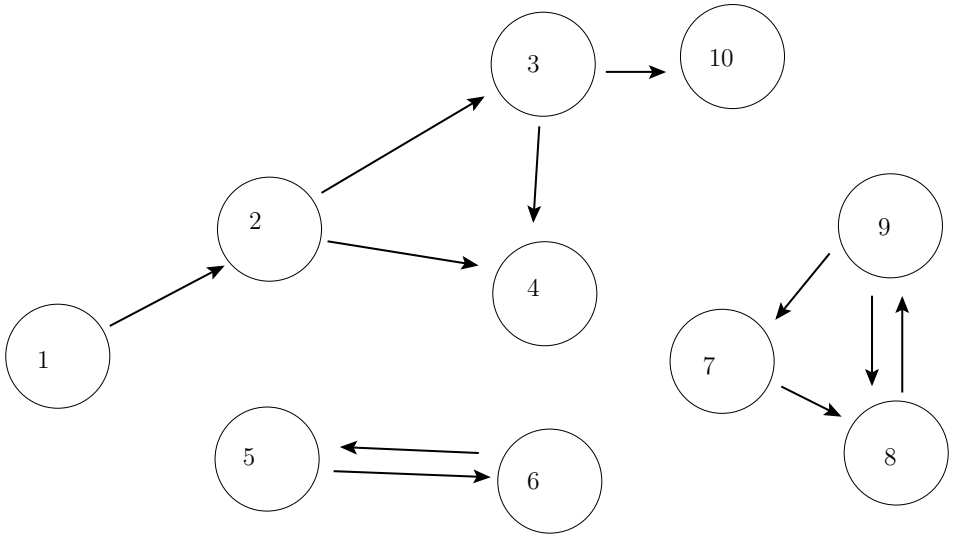

FIGURE 3 – Exemple 3

Corrigé –

$$C_1 = \begin{pmatrix} 0&0&0&0&0&0&0&0&0&0\\ 1&0&0&1&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0&0\\ 0&1&1&0&0&0&0&0&0&0\\ 1&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0\\ 0&0&0&1&0&1&0&0&1&0\\ 0&0&0&0&0&0&1&0&1&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&1&0&0&0&0&0&0&0 \end{pmatrix} \quad C_2 = \begin{pmatrix} 0&1&0&0&0&0&0&0&0&0\\ 0&0&1&1&0&0&0&0&0&0\\ 0&0&0&1&0&0&0&0&0&1\\ 0&0&0&0&1&0&0&1&0&0\\ 0&0&1&0&0&1&0&0&0&0\\ 0&1&0&0&0&0&1&0&0&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0&0&0 \end{pmatrix}$$

$$C_3 = \begin{pmatrix} 0&0&0&0&0&0&0&0&0&0\\ 1&0&0&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0&0&0\\ 0&1&1&0&0&0&0&0&0&0\\ 0&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&1&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&1&0&1&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&1&0&0&0&0&0&0&0 \end{pmatrix}$$

On souhaite attribuer à chaque page i un score $r_i \in \mathbb{R}_+^*$ de façon à pouvoir classer l'ensemble des pages par score décroissant et présenter à l'utilisateur une liste ainsi classée des pages correspondant à sa requête. L'algorithme PageRank part du principe qu'un lien de la page j pointant sur la page i contribue positivement au score de cette dernière, avec une pondération par le score r_j de la page dont est issu le lien (une page ayant un score élevé a ainsi plus de poids qu'une n'ayant qu'un score médiocre) et par le nombre total de liens présents sur ladite page $N_j = \sum_{k=1}^{N} C_{kj}$. On introduit donc la matrice Q définie par $Q_{ij} = \frac{C_{ij}}{N_j}$ si $N_j \neq 0$, $Q_{ij} = 0$ sinon.

2. *Déterminer les matrices Q_1, Q_2 et Q_3 associées aux trois exemples. Vérifier que la somme des coefficients des colonnes non nulles de Q vaut toujours* 1.

3. *Démontrer que toutes les colonnes non nulles de la matrice Q générale sont telles que la somme de leurs coefficients est égale à 1.*

   Corrigé – Soit $j \in \{1, \dots, N\}$. On suppose que la colonne j de Q est non nulle. Ceci mplique que $N_j \neq 0$ et que $Q_{i,j} = C_{i,j}/N_j$ pour tout $i \in \{1, \dots, N\}$. On a donc

$$\sum_{i=1}^{N} Q_{i,j} = \frac{1}{N_j} \sum_{i=1}^{N} C_{i,j} = 1.$$

L'application des principes ci-dessus conduit donc à une équation pour le vecteur $r \in \mathbb{R}^N$ des scores des pages de la forme

$$r_i = \sum_{j=1}^{N} Q_{ij} r_j \text{ c'est à dire } r = Qr, \tag{1}$$

où Q est une matrice dont la somme des coefficients de chaque colonne non nulle est égale à 1. Le problème du classement des pages du Web se trouve ainsi ramené à la recherche d'un vecteur propre d'une énorme matrice, associé à la valeur propre 1. Mais il peut arriver que la matrice Q n'admette pas la valeur propre 1 ce qui invalide quelque peu la philosophie originale de l'algorithme.

4. *Vérifier avec un programme informatique que les matrices Q_1 et Q_3 admettent 1 comme valeur propre mais que ce n'est pas le cas de Q_2.*
5. *Soit Q la transposée d'une matrice stochastique*[2]
   (a) *Montrer que 1 est valeur propre de Q^t et donc de Q.*

   Corrigé – On note $q_{i,j}$ la composante de Q en i-ieme ligne et j-ieme colonne et e le vecteur dont les composantes (notées e_i) sont toutes égales à 1.
   On a $\sum_{i=1}^{N} q_{i,j} = 1$ et donc $\sum_{i=1}^{N} q_{i,j} e_i = e_j$, ce qui montre que le vecteur e est vecteur propre de Q^t pour la valeur propre 1. On en déduit que Q a aussi 1 comme valeur propre (mais e n'est pas, en général, vecteur propre associé à cette valeur propre).

   (b) *Montrer que $\rho(Q^t) = \rho(Q) = 1$.*

   Corrigé – Soit λ une valeur propre de $A = Q^t$, et $x = (x_1, \dots x_N)^t$ un vecteur propre associé. On a alors, en prenant i tel que $|x_i| = \max_j |x_j|$,

$$|\lambda||x_i| \leq \sum_j |a_{ij}||x_j| = \sum_j a_{ij}|x_j| \leq |x_i|.$$

   Donc $|\lambda| \leq 1$. Ceci donne $\rho(Q^t) = \rho(Q) \leq 1$, et donc, par la question précédente, $\rho(Q^t) = \rho(Q) = 1$.

Pour faire en sorte que 1 soit valeur propre, on va donc modifier la matrice Q de manière à ce qu'elle soit la transposée d'une matrice stochastique, et donc en particulier qu'elle n'ait plus de colonne nulle. Pour cela, on note e le vecteur de $\mathbb{R}^N$ dont les composantes (notées e_i) sont toutes égales à 1 et d le vecteur de $\mathbb{R}^N$ de composantes d_j, $j = 1, \dots, N$, avec $d_j = 1$ si $N_j = 0$, $d_j = 0$ sinon. On définit alors la matrice

$$P = Q + \frac{1}{N} e d^t. \tag{2}$$

2. On appelle *matrice stochastique* une matrice dont tous les coefficients appartiennent à $[0, 1]$ et dont la somme des coefficients de chaque ligne vaut 1.

6. *Visualiser sur ordinateur les matrices P_1, P_2 et P_3 associées aux trois exemples et calculer leurs valeurs propres.*

7. *Montrer que le passage de Q à P revient à remplacer les colonnes de zéros de Q par des colonnes dont toutes les composantes sont égales à $\frac{1}{N}$.*

   Corrigé – Soit $j \in \{1, \ldots, N\}$.
   Si la colonne j de Q est une colonne de 0, ceci signifie que $N_j = 0$ et donc que $d_j = 1$. La colonne j de la matrice ed^t est alors une colonne de 1. La colonne j de P est donc une colonne de $1/N$.
   Si la colonne j de Q n'est une colonne de 0, ceci signifie que $N_j \neq 0$ et donc que $d_j = 0$. La colonne j de la matrice ed^t est alors une colonne de 0. La colonne j de P est alors égale à la colonne j de Q.

8. *Montrer que P est bien la transposée d'une matrice stochastique, et en déduire que P admet bien la valeur propre 1 et que $\rho(P^t)$ et $\rho(P)$ sont égaux à 1.*

   Corrigé – Soit $j \in \{1, \ldots, N\}$.
   Si la colonne j de Q est une colonne de 0, la colonne j de P est une colonne de $1/N$ et donc la somme des coefficients de cette colonne de P est 1.
   Si la colonne j de Q n'est une colonne de 0, La colonne j de P est alors égale à la colonne j de Q. La question 3 donne alors que la somme des coefficients de cette colonne de P est aussi 1.
   Ceci montre bien que P est la transposée d'une matrice stochastique. La question 5 donne alors que P admet bien 1 comme valeur propre 1 et que $\rho(P^t)$ et $\rho(P)$ sont égaux à 1.

On remarque sur l'exemple 3 que la valeur propre 1 peut être multiple. Or, notre problème consiste à trouver un vecteur propre associé à cette valeur propre et il est alors préférable (en particulier pour l'efficacité des algorithmes de recherche de vecteurs propres) que 1 soit valeur propre simple. On va donc encore modifier la matrice de manière à ce que tous ses coefficients soient strictement positifs. En effet, on a le théorème suivant :

Théorème 1 (Perron-Frobenius, cas des matrices stochastiques) *Soit A une matrice transposée d'une matrice stochastique et qui est de plus strictement positive (c.à.d. dont tous les coefficients sont strictement positifs). Alors $\rho(A) = 1$, et 1 est une valeur propre simple ; de plus, il existe un vecteur propre r strictement positif associé à la valeur propre 1 .*

Ce théorème est important pour nous car il va nous permettre non seulement de construire une matrice avec 1 comme valeur propre simple, mais de plus d'obtenir le vecteur r donnant un classement des pages du web. La question 9 a pour objet de démontrer le théorème de Perron-Frobenius. En fait, sous les hypothèses de ce théorème, on sait déjà

par les questions précédentes que $\rho(A) = 1$ et que 1 est valeur propre. Il reste à montrer que 1 est valeur propre simple et qu'il existe un vecteur propre r strictement positif associé à la valeur propre 1.

9. *Soit $A = B^t$ où B est une matrice stochastique strictement positive.*

   (a) *Montrer que le sous-espace propre associé à la valeur propre 1 de B est $\mathbb{R}e$.*

   Corrigé – On sait déjà, par la démonstration de la question 5a, que $\mathbb{R}e$ est inclus dans le sous-espace propre associé à la valeur propre 1 de B.
   Soit maintenant x un vecteur de $\mathbb{R}^N$ appartenant au sous-espace propre associé à la valeur propre 1 de B. On note $b_{i,j}$ les composantes de B et x_i les composantes de x. On choisit $i \in \{1, \dots, N\}$ tel que $x_i \leq x_j$ pour tout j, de sorte que $(x_j - x_i) \geq 0$ pour tout j. Comme $Bx = x$, on a $\sum_{j=1}^N b_{i,j}(x_j - x_i) = 0$. Or $b_{i,j} > 0$ pour tout j, on a donc nécessairement $x_j = x_i$ pour tout $j \in \{1, \dots, N\}$. Ceci prouve de $x \in \mathbb{R}e$ et donc que le sous-espace propre associé à la valeur propre 1 de B est $\mathbb{R}e$.

   (b) *En déduire que la valeur propre 1 de A est simple.*

   Corrigé – les matrices A et B ont même polynôme caractéristique, elles ont donc les mêmes valeurs propres, comptées avec leur multiplicité (algébrique). Les matrices A et B ont aussi les mêmes valeurs propres, comptées avec leur multiplicité géométrique (c'est, par exemple, une conséquence du théorème du rang et du fait que toute matrice a même rang que sa transposée). La question 9a donne donc que 1 est une valeur propre simple de A en prenant "simple" au sens de la multiplicité géométrique, et c'est bien ce qui est intéressant ici.
   En fait, il est aussi possible de montrer que 1 est une valeur propre simple de A en prenant "simple" au sens de la multiplicité algébrique. Cette démonstration est plus difficile.

   (c) *Soit f un vecteur propre de A pour la valeur propre 1 .*

      i. *Montrer que $|f_i| < \sum_j a_{ij}|f_j|$ sauf si les f_j sont tous de même signe.*

      Corrigé –
      Soit $i \in \{1, \dots, N\}$. On suppose $f_i \geq 0$. On a alors, comme $a_{i,j} > 0$,

$$f_i = \sum_{j=1}^N a_{i,j} f_j = \sum_{j=1}^N a_{i,j}(f_j^+ - f_j^-) \leq \sum_{j=1}^N a_{i,j}(f_j^+ + f_j^-) = \sum_{j=1}^N a_{i,j}|f_j|.$$

      Si il existe j tel que $f_j < 0$, on a $a_{i,j}(f_j^+ - f_j^-) < a_{i,j}(f_j^+ + f_j^-) = a_{i,j}|f_j|$ et donc

$$|f_i| = f_i < \sum_{j=1}^N a_{i,j}|f_j|.$$

      On a donc bien $|f_i| < \sum_{j=1}^N a_{i,j}|f_j|$ sauf si $f_j \geq 0$ pour tout j.
      Le cas $f_i < 0$ se ramène au cas précédent en raisonnant sur $-f$.

ii. *En raisonnant sur $\sum_i |f_i|$, en déduire que les f_j sont tous de même signe.*

Corrigé – Si les f_j n'ont pas tous le même signe, on a, pour tout i, $|f_i| < \sum_{j=1}^N a_{i,j}|f_j|$. En sommant sur i et en utilisant le fait que $\sum_{i=1}^N a_{i,j} = 1$, on obtient alors $\sum_{i=1}^N |f_i| < \sum_{j=1}^N |f_j|$. Ce qui est impossible car $f \neq 0$.

Dans le but d'obtenir une matrice strictement positive, on effectue alors une dernière modification sur la matrice en choisissant un nombre $0 < \alpha < 1$ et en posant

$$A_\alpha = \alpha P + (1-\alpha)\frac{1}{N} ee^t. \tag{3}$$

10. *Ecrire un programme calculant les matrices $A_{\alpha,1}$, $A_{\alpha,2}$ et $A_{\alpha,3}$ associées aux trois exemples pour $\alpha = 0.1$ et $\alpha = 0.5$. Calculer les modules des valeurs propres de ces matrices et classez les par ordre décroissant).*

11. *Montrer que A_α est toujours la transposée d'une matrice stochastique, et qu'elle est de plus strictement positive. En déduire que $\rho(A_\alpha) = 1$, que 1 est une valeur propre simple de A_α et qu' il existe un vecteur propre r_α strictement positif associé à la valeur propre 1.*

Corrigé – La matrice A_α est toujours la transposee d'une matrice stochastique (car P et $(1/N)ee^t$ le sont) et strictement positive (car $ee^t > 0$) Le fait que $\rho(A_\alpha) = 1$ est donné par la question 8, le fait que 1 est valeur propre simple est donné par la question 9b. Soit r un vecteur propre (non nul) associé à la valeur propre 1. La question 9c donne que les composantes de r ont toutes le même signe. on peut donc supposer que les composantes de r sont toutes positives. il reste à montrer qu'elles sont sont toutes strictement positives. Soit $i \in \{1, \ldots, N\}$. On a $r_i = \sum_{j=1}^N a_{i,j} r_j$. Si $r_i = 0$, on a alors $\sum_{j=1}^N a_{i,j} r_j = 0$. Comme $a_{i,j} > 0$ et $r_j \geq 0$ pour tout j ceci donne que $r_j = 0$ pour tout $j \in \{1, \ldots, N\}$. Ce qui est impossible car $r \neq 0$.

Finalement, PageRank calcule un tel vecteur propre $r_\alpha \in \mathbb{R}^N$, normalisé d'une façon ou d'une autre, qui est tel que

$$r_\alpha = A_\alpha r_\alpha, \tag{4}$$

dont les N composantes fournissent le classement recherché des pages du Web. On remarquera que pour N grand, la matrice A_α n'est qu'une petite perturbation de la matrice Q. On sait combien cette stratégie s'est révélée efficace, puisque Google a totalement laminé les moteurs de recherche de première génération, comme Altavista, lesquels ont essentiellement disparu du paysage.

Calcul effectif du score des pages web

On décrit maintenant des méthodes pour approcher ce vecteur r.

12. *Programmer la méthode de la puissance : pour* $r_0 \neq 0$

$$q_k = A_\alpha r_{k-1},\ r_k = \frac{q_k}{\|q_k\|_1}. \tag{5}$$

On normalisera les vecteurs en utilisant la norme 1.
Approcher le vecteur r *solution de* $r = A_\alpha r$ *pour nos trois exemples pour* $\alpha = 0.1$.
On pourra discuter le choix du test d'arrêt utilisé pour stopper l'algorithme.
Vérifier numériquement que $\|r_k - r\|_2 \leq Cte|\alpha\mu_2|^k$ *où* μ_2 *est la seconde plus grande valeur propre (en module) de* P.

On remarque que les matrices A_α sont des matrices pleines alors que la matrice initiale Q était creuse. Il est en pratique hors de question d'assembler cette matrice A_α.

13. *Montrer que si* $z \in \mathbb{R}^N$, $z \geq 0$ *avec* $\|z\|_1 = 1$, *alors* $y = A_\alpha z = \alpha Q z + \frac{1-\|\alpha Q z\|_1}{N} e$ *et* $y \geq 0$.

Corrigé – Soit $z \in \mathbb{R}^N$. *On suppose que* $z \geq 0$ *avec* $\|z\|_1 = 1$. *Comme* $P = Q + (1/N)ed^t$, *on a* $A_\alpha = \alpha P + (1-\alpha)(1/N)ee^t = \alpha Q + (\alpha/N)ed^t + ((1-\alpha)/N)ee^t$. *On a donc*

$$A_a z = \alpha Q z + ae \text{ avec } a = \frac{\alpha}{N} d^t z + \frac{1-\alpha}{N} e^t z.$$

Comme $Q \geq 0$, $a \geq 0$ *et* $\alpha \in [0,1]$, *on en déduit que* $\alpha Q z \geq 0$ *et* $a \geq 0$. *On a donc* $A_\alpha z \geq 0$ *et*

$$\|A_\alpha z\|_1 = \alpha \|Q z\|_1 + a\|e\|_1 = \alpha \|Q z\|_1 + aN. \tag{6}$$

Si S *est la transposée d'une matrice stochastique, on a (en notant* $s_{i,j}$ *les composantes de* S *et* z_i *celles de* z*),*

$$\|Sz\|_1 = \sum_i \Big(\sum_j s_{i,j} z_j\Big) = \sum_j \Big(\sum_i s_{i,j}\Big) z_j = \sum_j z_j = \|z\|_1 = 1.$$

Avec $S = A_\alpha$, *on en déduit que* $\|A_\alpha z\|_1 = 1$ *et* (6) *donne alors* $a = \frac{1-\alpha\|Qz\|_1}{N}$ *et donc* $A_\alpha z = \alpha Q z + \frac{1-\|\alpha Q z\|_1}{N} e$.

On a ainsi ramené le calcul du produit matrice pleine-vecteur original à un produit matrice creuse-vecteur et à une évaluation de norme, effectivement calculables à l'échelle du Web (à condition de disposer de ressources informatiques conséquentes, quand même).

14. *Programmer alors l'algorithme*

```
Choisir r0
Tant que s >tol, faire
  r1=alpha Q r0
  beta=1-norm(r1,1)
  r2=r1+beta/N*e
  s=norm(r2-r0,1)
  r0=r2
 retourner r0
```

Construire un exemple de réseau de pages webs de "grande taille" $N = 1000$ *(la taille réelle du WWW est de plusieurs milliards !. . .) et comparer la vitesse de cet algorithme avec l'algorithme de la puissance.*