PageRank Induced Topology for Real-World Networks

Bruno Gaume
IRIT-UPS
Toulouse F-31062 Cedex 4, France

Fabien Mathieu
France Telecom R & D
38, rue du Général Leclerc 92794 Issy les Moulineaux France

This article presents a stochastic method for studying the structure of large small worlds graphs. The principle of this method is to apply a *PageRank-like* importance algorithm, with a damping factor and an external importance source. By varying the source vector, one obtains a powerful graph visualization tool, which reveals the structural organization of small worlds graphs.

1. Introduction

The discovery that *real-world large networks* from many different domains (sociology, biology, computer science...) share the same characteristics has raised an interest in their studying ([17, 2, 15]).

The associated graphs of such networks are rather sparse (the mean degree stays roughly constant when the number of nodes increases), highly clustered, and there exists short paths that can be found [11, 3]. An hierarchical structure is also revealed by a heavy tail distribution for most parameters[15]. Referring to Milgram's experiment[14], Watts and Strogatz proposed to call highly clustered graphs with low diameter *small worlds* [18].

In Section 2 we describe how, by taking into account distributions of random walks in a graph G as coordinates of its n nodes in $\mathbb{R}^n$, one can fit G with a geometrical structure. This method allows to analyze precisely and efficiently the structure of *small-world-shaped* very large graphs.

Section 3 outlines the limits of this approach, which requires reflexive and symmetric graphs to produce relevant results.

We propose in Section 4 to fit PageRank computation techniques in order to generalize the random walk method described in Section 2 to all graphs without any specific restriction. The adaptability of this technique allows us, for instance, to consider the Web graph, which is neither reflexive nor symmetric.

Lastly, Section 5 demonstrates the main topological characteristics of the geometrical structure obtained. Lower and upper bounds are given for usual cases, along with a complete description of distances for a few canonical cases.

2. From Random walks to Topology

Random walks offer a simple and powerful framework for large graphs structure study. Random walks are very efficient for sparse graphs computation and are thus well suited to real-world large networks. For instance, the PageRank algorithm ([16]) presented in Section 4, computes billions of Web pages importance using the graph structure induced by hyperlinks.

2.1 (Sub)markovian process in a graph

Let $G = (V, E)$ be a graph (directed or not) with n nodes and m edges. In an homogenous markov chain for G, a transition probability is associated to each transition from u to v, with u,v in V. A classical way to represent this chain is a $n \times n$ stochastic matrix A.

The simplest way to take into account the graph structure is to suppose an uniform transition probability distribution for each neighbor (if any) of a given node. Let $\deg(u)$ denotes the degree of u (out-degree if G is a digraph). The matrix $A = (a_{u,v})$ obtained is:

$$A = (a_{u,v})_{u,v \in V}, \text{ with } a_{u,v} = \begin{cases} \dfrac{1}{\deg(u)} \text{ if } u \to v, \\ 0 \text{ else.} \end{cases} \quad (1)$$

Notice that this definition can be extended to weighted graphs by choosing probabilities proportional to weights.

The underlying Markov chain is well defined as long as no node has null degree (otherwise A is substochastic, but not stochastic).

If A is stochastic, for any initial probability distribution P_0 on V and any given integer k, P_0A^k is the result of the random walk of length k starting from P_0 whose transitions are defined by A. More precisely, for any u, v in V, the probability P_k of being in v after a random walk of length k starting from u is equal to $(\delta_u A^k)_v$, where δ_u is the certitude of being in u. Note that P_0A^k has a cash flow interpretation: if P_0 is an initial amount of cash distributed among the nodes of V, and if at each step the cash is redistributed according to A, then P_0A^k is the cash distribution after k steps. This interpretation is useful if A is substochastic, as it is not necessary for the result to be a probability: P_0A^k is just a cash diffusion with (possibly) loss. The flow interpretation is less restrictive than the stochastic interpretation, and many authors have chosen it [4, 1, 5].

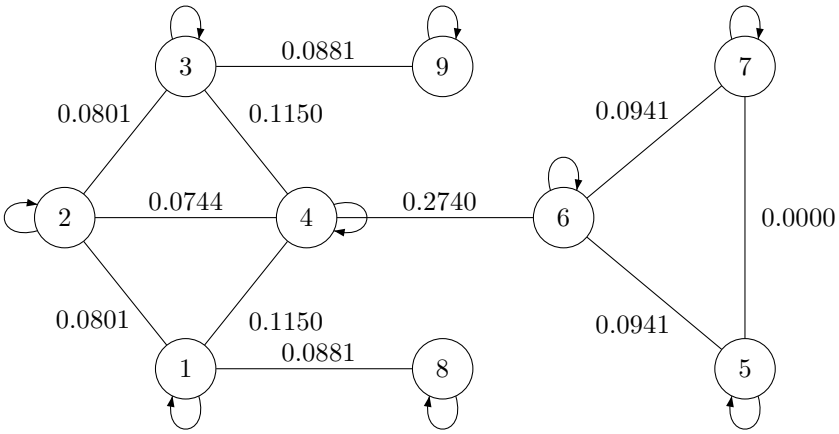

Figure 1. Example of a symmetric reflexive graph. Geometrical lengths of the edges in $\mathbb{R}^9$ using $k = 5$ are indicated.

2.2 Using random walks to extract topology

From now until Section 4, we assume A is stochastic. For any given node u in V, and any positive integer k, $\delta_u A^k$ is then a n-dimensional non-negative vector from the standard $n-1$-simplex Δ^{n-1}. For a given length k, this allows us to assign to each node u some coordinates $C(u,k)$ in Δ^{n-1} defined by:

$$C(u,k) = \delta_u A^k. \tag{2}$$

The idea is that two nodes u and v with similar relations with respect to the rest of the graph, will have similar random walks: they will be close in the n-dimensional representation of G. So using random walks distributions as coordinates should highlight the structural relations in G.

The parameter k may be any integer between 1 and $+\infty$. However, a random walk does not capture the graph structure if k is too small. Conversely, if k is too large, the random walk tends to forget its starting point, so coordinates generally concentrate on one (or a few) unique point. Choosing k with the same order of magnitude as the mean distance in G seems empirically to be a good trade-off [9, 12].

For example, Figure 1 shows a reflexive and symmetric graph with

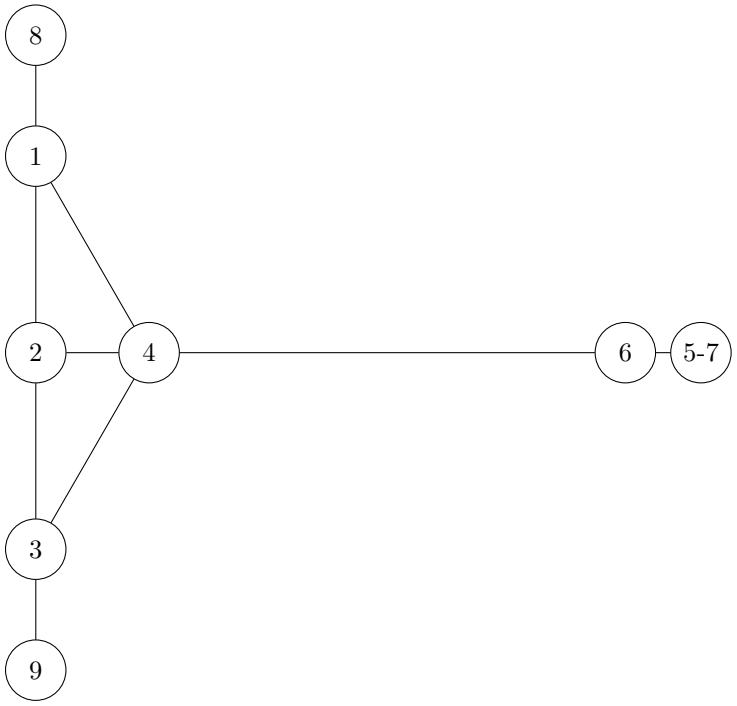

Figure 2. 2-dimensional projection of the graph shown in Figure 1.

9 nodes. Using a random walk of length[1] $k = 5$, distances between adjacent nodes are shown. Nodes 5 and 7 have exactly the same adjacency list: $\{5, 6, 7\}$. This entails their coordinates in $\mathbb{R}^9$ are equal for any random walk of positive length. The edge $(5, 7)$ has therefore a length of zero. Conversely, the edge $(4, 6)$ is the longest, with a length of 0.2740 (for $k = 5$).

For visualization purposes, we choose to use *Principal Component Analysis* (PCA) to produce 2-dimensional projections of our graphs. PCA used on the graph shown in Figure 1 yields Figure 2.

2.3 Example: shortcuts are fast courses by long edges

There are many applications of this geometrical representation (see [9]. Identifying shortcuts through distance refining is one of them. The standard distance between two vertices u and v of a finite graph is the minimum length of the paths connecting them. If no path exists, the distance is infinite. However, in *small worlds* graphs, the standard graph distance is often less interesting: for almost any nodes u and v,

[1] Choosing $k = 5$ is arbitrary in this case, as the graph is too small for estimations of the optimal value of k.

there exists a short path connecting u and v, and it can be difficult to use this distance to distinguish nodes.

Consider a graph G that is plunged in $\mathbb{R}^n$ by the above mentioned method. Any edge e between two nodes r and s has a canonical geometrical weight. This corresponds to the geometrical distance separating r and s in $\mathbb{R}^n$. For example, in Figure 1, using euclidian distance, edge $(2, 4)$ has a weight equal to 0.0744, whereas the weight of edge $(4, 6)$ is equal to 0.2740. Comparing the distances in the weighted and unweighted graphs gives some insight about the graph structure.

Indeed, we note that edges between nodes from different communities are often called short cuts [18, 11, 15]. These edges enable low diameter of small worlds, enhancing the performance of routing algorithms. Note that nodes belonging to different communities are geometrically very distant in $\mathbb{R}^n$ while two nodes belonging to the same community are geometrically close[2]. Hence, we arguably claim that an edge geometrical length is a valuable estimation of its practical importance. In other words, short cuts are long edges. Figure 1 illustrates that the longest edge$((4, 6))$ is obviously the more important, since connectivity between $\{1, 2, 3, 4, 8, 9\}$ and $\{5, 6, 7\}$ depends on it.

2.4 Complexity

We presently focus our interest on u's coordinates using a random walk of length k. We denote $C(u, k) = \delta_u A^k$ the coordinate of u obtained with this method. Recall that the so-called *small worlds* may be very large graphs with billions of nodes. Scalability is therefore an essential design goal. Calculating explicitly A^k seems to be the simplest way; this gives the coordinates of all nodes at the same time. This method, however, is not used in practice, as it does not take advantage of matrix A sparsity. It is indeed much faster to proceed by successive multiplications of a vector by A, each multiplication taking m operations (that is in $O(n)$ for small worlds), so the effective way to get $C(u)$ is to compute recursively the random walk:

- $C(u, 0)$ has the value δ_u for $k = 0$.
- For any positive integer k, we have $C(u, k) = C(u, k - 1)A$

The space and time complexities are reduced compared to A^k computing; calculations are done on n-dimensional vectors plus a single static sparse $n \times n$ matrix, rather than manipulating full $n \times n$ matrices. Multiplying vectors by such matrices is typically $O(n)$ because the number of non-zero entries is proportional to n. Hence, obtaining the

[2]This idea was firstly proposed by [9] for visualization of small worlds and linguistic modeling of lexical wide-area networks and taken up by [12] for classifying the nodes of a small world.

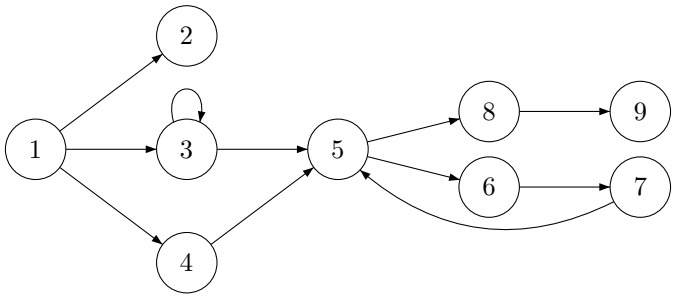

Figure 3. Example of a graph not adapted to the standard random walk approach if unsmoothed.

coordinates of all nodes is $O(kn^2)$ time-consuming[3], whereas classical computations of A^k are typically $O(\log_2(k)n^\alpha)$ time-consuming, with $\alpha > 2$ [8]. As values of k are often small ([9, 12]), computing recursively $C(u, k)$ for all nodes u is strictly equivalent to computing A^k using the recursive method $A^0 = Id$ and $A^k = A^{k-1}A$, but the $C(u, k)$ approach is much more flexible.

3. Limitations of the fixed random walk model

Obtaining meaningful results with the random walk method described above requires some graph properties. For example, if there are periodicities interfering with the random walk length, there is a risk to create a phenomenum of resonance which will unnecessary bring closer some nodes and move away others. Another problem arises with the possible existence of nodes with null out-degree (leaves). Since their coordinates according to the random walk is null, they will be merged together. More generally, for a random walk of length k, any node with at least one leaf in its $(k-1)$-neighborhood will undergo a prejudicial probability leak.

For all these reasons, the graphs we want to study is generally trans-

[3]If we just want the coordinates of a subset, the computation time is proportional to the size of the subset.

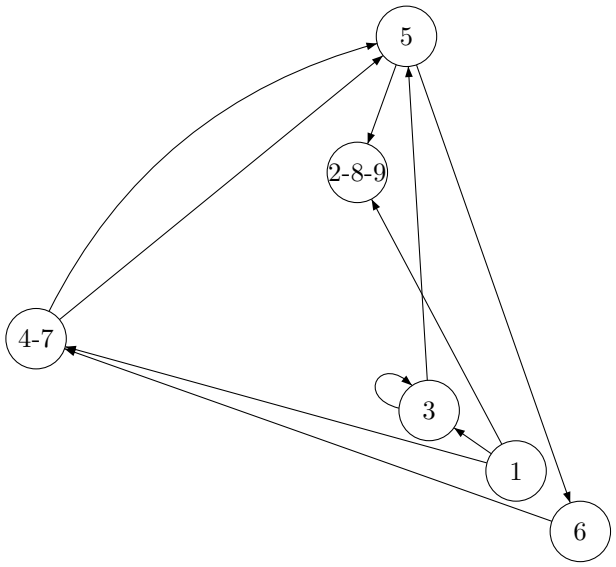

Figure 4. 2-dimensional illustration of the graph from Figure 3 using a random walk over the original graph. k is set to 4.

formed to be reflexive (each node points towards itself) and unoriented. We call *smoothing* this alteration of the graph. To illustrate the benefit of smoothing, let us consider the 9-nodes digraph of Figure 3. This digraph has two leaves (2 and 9), as well as a recurrent 3-periodical strongly connected component formed by nodes 5, 6 and 7. If we use random walk of length $k = 4$[4] on the unsmoothed digraph, the results are awkward: It appears that nodes 2, 8 and 9 are merged (they all have null coordinates); so are nodes 4 and 7, because they point towards the same node, namely 5. It also oddly seems that the nearest node from 6 is 1, whereas 5 and 7 are among the farthest. The 2-dimensional projection of Figure 4 clearly illustrate those flaws.

Most of these problems are solved using the smoothed graph instead of the original digraph, as shown by Figure 5. However, some results remain unsatisfactory. For example, nodes 6 and 7 are merged because they share exactly the same neighborhood (5, 6 and 7). In the same way, even if their neighborhood is different, 3 and 4 produce random walks that are so similar that they cannot be distinguished after using PCA.

It is generally regrettable to lose orientation information contained in digraphs. Directions can be invaluable. For instance, directed graphs are a practical structure for taking account of how Web pages are

[4]Like in Section 2.2, k is arbitrary set (to 4 in this case).

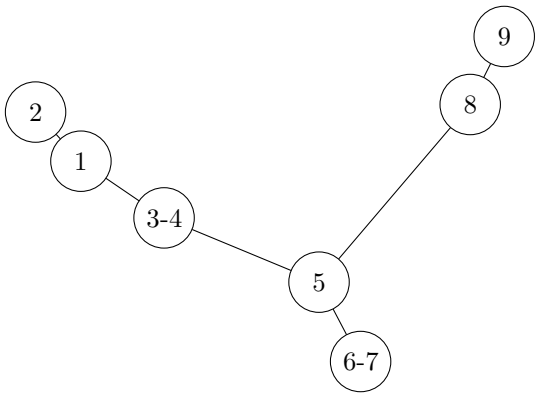

Figure 5. 2-dimensional illustration of the graph from Figure 3 using a random walk over the smoothed graph. k is set to 4.

connected through hyperlinks: nodes represent the pages, and having an edge from page u to page v means u contains a hyperlink that points at v. This means u is aware of v and acknowledges it, but the reverse is not necessary true. This kind of directed acknowledgment is used in most modern ranking algorithms [16]. Making Web graphs symmetrical may be unacceptable. A similar reasoning stands for reflexivity; if some (but not all) nodes point at themselves, we may want not to loose this information.

4. Bringing damping into random walk: the PageRank approach

We propose an alternative method that enables a random walk approach on directed graphs without alteration: the PageRank approach. Basically, PageRank consists in computing the asymptotic presence probability of a random walk (assuming it exists)[5]. Formally, it comes down to seeking solution(s) P of the equation

$$P = PA. \tag{3}$$

With this basic definition, using PageRank to compute coordinates has three major flaws:

[5]For a more complete explanation of PageRank, see [4, 13].

- The random walk is not well defined if there are leaves, and the presence probability may not be unique or even converge. A sufficient condition is that the graph is aperiodic and strongly connected, but most directed graphs are not.
- Convergence of the Markov process is not granted.
- Assuming there is convergence, all nodes within a same recurrent strongly connected component will have identical coordinates and will be undistinguishable.

Many solutions to bypass these issues have been proposed. The Web graphs for which PageRank has been designed are not strongly connected and possess many leaves, or *dangling links* [7, 16]. We will focus on one of the most known variant of PageRank, that suits well our needs and overcomes these drawbacks: PageRank with a damping factor.

Principles of damping factor can be found in [6] and are further explained in [13, 5]. Basically, the standard equation used to describe asymptotic states of a Markov chain is replaced by

$$P = dPA + (1-d)P_0, \tag{4}$$

where d is a positive number less than 1, called *damping factor*, and P_0 a distribution over V. Like P_0A^k, equation (4) has both a stochastic and a flow interpretation (see Section 4.1).

As shown in [13], equation 4 has a unique solution that is a fixed-point of the d-contraction $X \to dXA+(1-d)P_0$. Hence, for $0 < d < 1$, the recursive iteration

$$P_{n+1} = dP_nA + (1-d)P_0 \tag{5}$$

will geometrically (with ratio d) converge towards the solution P for any initial vector P_0. For a given graph G, a unique vector P solution of equation 4 can be associated to any pair constituted by a damping factor d and a distribution P_0. Thus we may refer to P as $P(d, P_0)$.

Note that due to the geometric convergence, there is a need for at most $p = \frac{\ln(\epsilon)}{\ln(d)}$ iterations to compute $P(d, P_0)$ with a precision ϵ using equation 5. As empirically $\epsilon = \frac{1}{n}$ offers a very good precision[13], a reasonable order of magnitude for p is $\frac{\ln(n)}{-\ln(d)}$.

4.1 Interpretations of PageRank with damping

Similarly to P_0A^k in Section 2.1, the solution $P(d, P_0)$ can be interpreted in two ways: as a stochastic distribution or as a cash flow repartition.

If A is stochastic, $P(d, P_0)$ is the stationary probability of a random walk, where a transition defined by A is chosen with a probability d and

a transition according to the distribution P_0 is chosen with probability $(1-d)$. All goes as if the random walk is performed on a weighted complete graph with the same vertices as G and whose edge's weights $w_{u\to v}$ are:

$$w_{u\to v} = \begin{cases} \dfrac{d}{\deg(u)} + (1-d)P_0(v) \text{ if } u \to v \text{ in } G, \\ (1-d)P_0(v) \text{ else.} \end{cases} \quad (6)$$

From a cash flow point of view, $P(d, P_0)$ can be interpreted as follows. Consider a constant external cash source equals to $(1-d)P_0$. If the cash flow is distributed at each step through the graph according to dA (losses greater than $(1-d)$ may occur if A is substochastic), then P is the asymptotic repartition of cash in the vertices.

Both interpretations become more meaningful, if equation (4) is rewritten as an infinite sum:

$$P(d, P_0) = (1-d)P_0 \sum_{i=0}^{\infty} d^i A^i. \quad (7)$$

For each i, $(1-d)P_0 d^i A^i$ represents the result of a random walk of length i starting from P_0 (or a cash distribution with initial values set according to P_0 after i steps) with weight $(1-d)d^i$. As $\sum_{i=0}^{\infty}(1-d)d^i = 1$, $P(d, P_0)$ is the mean over all random walks starting from the distribution P_0 with a geometrical damping of ratio d, since the random walk length increases.

4.2 Using $P(d, \delta_u)$ instead of $\delta_u A^k$

As we have seen, random walks of length i have a weight $(1-d)d^i$ in the computation of $P(d, P_0)$. So the average length of random walks used in equation (7) is $\sum_{i=0}^{\infty}(1-d)d^i i = \frac{d}{1-d}$.

A natural idea is to use equation (4) instead of equation (2) to express u's coordinates, where u is a vertex of G. Instead of using a random walk of fixed length k starting from δ_u, we decide to use $P(\frac{k}{k+1}, \delta_u)$. This is a geometric sum of all cash flows starting from u with a damping set such that the average length is k.

This geometric sum will be referred to as $F(u, k)$:

$$F(u,k) = (1-d)\delta_u \sum_{i=0}^{\infty} d^i A^i, \text{with } d = \frac{k}{k+1}. \quad (8)$$

If leaves make A substochastic, $F(u, k)$ may not be a probability, although it is always a positive non-null vector (because $F(u, k) \geq (1-d)\delta_u$). We choose to make it a probability through normalization for the following reasons: first, it seems more practical to keep all

the nodes on the Δ^{n-1} simplex; second, we have noticed that without normalization, nodes close to leaves tend to be similar and we think this is not desirable.

So from now on, the coordinates $C(u,k)$ used to represent node u will be the normalization of $F(u,k)$:

$$C(u,k) = \frac{F(u,k)}{\|F(u,k)\|_1}. \tag{9}$$

4.3 Advantages of using damping

First, parameter k is not necessarily an integer, it can be any non-negative number. This enables a more sensitive tuning that could not be done before.

More importantly, we insist on the fact that using damping allows all steps of the diffusion to be taken into account, and not only the k^{th} step. When two vertices are compared, if both flows have gone through the same vertices, but not during the same steps, it will still have an impact. That could not be done with a fixed number of steps. Damping allows flows to be compared even in the presence of periodicities or leaves.

As a consequence, equation (9) can be used for *any kind of (di)graph*: there is no need for the graph to be reflexive or symmetrical. Aberrations like the ones observed in Figure 4 are avoided. To our knowledge, there does not exist another way to bypass issues pointed in Section 3, without loosing information (by smoothing). Using damping paves the way for using digraphs like Web graphs. We think it is the essential feature of our approach that makes it worth using it.

In order to realize how powerful the damping approach can be, let's consider again the graph of Figure 3. If we compute the coordinates of the nodes using equation (9) and PCA, we obtain Figure 6. We first notice that we have none of the flaws observed in Figure 4 derived from a fixed-length random walk. More importantly, we have gained information compared to Figure 5 (that used a smoothed version of the original digraph). Neither nodes 3 and 4, nor nodes 6 and 7 are merged: the characteristics of graph G are preserved. Section 5 will prove, based on more general results that give some insight into the disposition of the nodes, that this is not an artifact due to our choice of G.

4.4 Complexity

Using damping requires extra resources. However, we want to point out that our model's complexity is not excessive compared to fixed-length model's complexity. In both cases, each iteration requires $\bigcirc(m)$. The difference lies in the number of iterations required. For the fixed-length

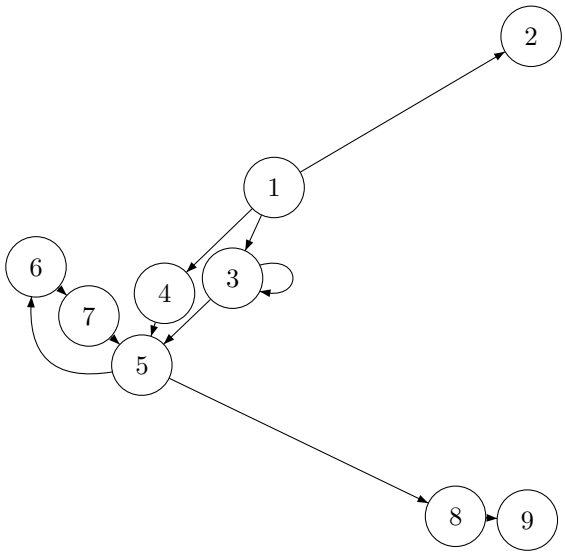

Figure 6. 2-dimensional illustration of the graph from Figure 3 using damping over the original graph. k is set to 4 (thus $d = \frac{k}{k+1} = 0.8$).

model, it is equal to k. For the damping model, we must evaluate the number of iterations needed to have a good approximation of the solution of equation (4). As said before, $\frac{\ln(n)}{-\ln(d)}$ offers a very good precision. Replacing d by $\frac{k}{k+1}$ leads to an estimation of the number of iterations: $k\ln(n)$. The complexity is thus increased by a factor $\ln(n)$. This seems reasonable even for very large graphs. Furthermore, the number of iterations may be reduced at the detriment of precision (the results will not be distorted, though).

5. Characteristics of the PageRank Induced Topology

The topology we have created using equation (9) instead of equation (2) copes with any kind of (di)graph. Moreover, this equation has some interesting properties that can be analyzed, especially if we consider the 1-norm. In the rest of this Section, $C(u)$ denotes the node u coordinates obtained with equation (9) (the parameter k is implicit), and the distance between two nodes u and v will be defined by $\text{dist}(u,v) = \|C(u) - C(v)\|_1$.

5.1 Maximal distance

As all nodes stand on a simplex, it is obvious that the 1-distance between two nodes is at most 2. In fact, as shown by Theorem 1, this

maximal distance 2 characterizes nodes that have nothing in common (nodes whose *spheres of influence* are strictly distinct).

Theorem 1 *Let u and v be two distinct nodes of a graph G. $\mathrm{dist}(u,v)$ is equal to 2 if, and only if, there exists no node w in G such that a path from u to w and a path from v to w exist.*

Proof. Notice that non-null components of $C(u)$ are nodes that can be reached from u. Thus, if there is a node w reachable from both u and v, $C(u)$ and $C(v)$ have at least a common non-null component and $\|C(u) - C(v)\|_1$ is less than 2. Conversely, the non-existence of such w indicates that non-null components of $C(u)$ and $C(v)$ are distinct. It is then straightforward that $\|C(u) - C(v)\|_1$ is equal to 2. ■

5.2 Minimal distance

Unlike the fixed random walk topology seen in Section 2.2, two different nodes would never have the same coordinates if damping is used. We think that it is an important feature to alleviate distinct nodes merging and to keep a minimal distance. Theorem 2 gives this minimal distance:

Theorem 2 *Let u and v be two distinct nodes of a graph G. A lower bound $d_{\min}$ for $\mathrm{dist}(u,v)$ is*

$$d_{\min} = 2\frac{1-d}{1+d}. \tag{10}$$

This lower bound is reached if and only if u's only outgoing link is v and vice versa.

Proof. To prove 2, it is useful to define the notion of *influence*. Influence $d_{u\to v}$ of u on v is defined by the proportion of cash issued from u that can reach v. It can be computed by summing with damping all the paths from u to v that do not pass through v until the end. One way to do this is to consider a diffusion process with damping similar to dA except that cash that reaches v is preserved (without damping) forever. More formally, if we consider the matrix $A_{\not v}$ defined by:

$$(A_{\not v})_{i,j} = \begin{cases} da_{i,j} \text{ if } i \neq v, \\ \delta_v^j \text{ if } i = v, \end{cases} \tag{11}$$

then $d_{u\to v}$ can be defined as the limit of the (u,v) component of powers of $A_{\not v}$:

$$d_{u\to v} = \lim_{l\to\infty} (A_{\not v}^l)_{u,v}. \tag{12}$$

With this definition, we can see that for $u \neq v$, $d_{u\to v}$ is a positive number smaller than d (except for the v^{th} row, $A_{\not v}$ is substochastic with

ratio d. As $u \neq v$, the ratio of cash able to reach v can not be more than d). However, a simpler way to express $d_{u\to v}$ is to consider $F_v(u)$:

$$\begin{aligned} F_v(u) &= \left((1-d)\delta_u \sum_{i=0}^{\infty} d^i A^i \right)_v \\ &= \left((1-d) \sum_{i=0}^{\infty} d^i A^i \right)_{u,v}. \end{aligned} \tag{13}$$

As $F_v(u)$ aggregates with damping all paths from u to v, we can split each of these paths into a path from u to v that does not pass through v and a path from v to v. The sum can, thus, be written as follows:

$$\begin{aligned} F_v(u) &= \left((1-d) d_{u\to v} \delta_v \sum_{i=0}^{\infty} d^i A^i \right)_v \\ &= d_{u\to v} F_v(v). \end{aligned} \tag{14}$$

This leads to this equivalent expression of $d_{u\to v}$:

$$d_{u\to v} = \frac{F_v(u)}{F_v(v)}. \tag{15}$$

Influence of v on u is similarly defined by:

$$d_{v\to u} = \frac{F_u(v)}{F_u(u)}. \tag{16}$$

The next step in our proof is to notice that a lower bound for $F_u(u)$ is the sum of $(1-d)$ and of the cash coming forth and back from v:

$$\begin{aligned} F_u(u) &\geq (1-d) + d_{u\to v} F_u(v) \\ &\geq (1-d) + d_{u\to v} d_{v\to u} F_u(u). \end{aligned} \tag{17}$$

So we can write:

$$F_u(u) \geq \frac{1-d}{1 - d_{u\to v} d_{v\to u}}. \tag{18}$$

This result can be interpreted as follows: due to mutual influences between u and v, an external constant source of cash of value $(1-d)$ received by u is amplified (for u) by a factor of at least $\frac{1}{1-d_{u\to v}d_{v\to u}}$ that corresponds to round-trips between u and v. This amplification lower bound also stands for v.

We now have all we need to complete our proof:

$$\begin{aligned}
\mathrm{dist}(u,v) &= \left\| \frac{F(u)}{\|F(u)\|_1} - \frac{F(v)}{\|F(v)\|_1} \right\|_1 \\
&\geq \left| \frac{F_u(u)}{\|F(u)\|_1} - \frac{F_u(v)}{\|F(v)\|_1} \right| + \left| \frac{F_v(v)}{\|F(v)\|_1} - \frac{F_v(u)}{\|F(u)\|_1} \right| \\
&\geq \left| \frac{F_u(u)}{\|F(u)\|_1} - \frac{d_{v\to u}F_u(u)}{\|F(v)\|_1} \right| + \left| \frac{F_v(v)}{\|F(v)\|_1} - \frac{d_{u\to v}F_v(v)}{\|F(u)\|_1} \right| \\
&\geq \frac{1-d}{1-d_{u\to v}d_{v\to u}} \left(\left| \frac{1}{\|F(u)\|_1} - \frac{d_{v\to u}}{\|F(v)\|_1} \right| + \left| \frac{1}{\|F(v)\|_1} - \frac{d_{u\to v}}{\|F(u)\|_1} \right| \right) \\
&\geq \frac{1-d}{1-d_{u\to v}d_{v\to u}} \left(\frac{1}{\|F(u)\|_1} - \frac{d_{u\to v}}{\|F(u)\|_1} + \frac{1}{\|F(v)\|_1} - \frac{d_{v\to u}}{\|F(v)\|_1} \right) \\
&\geq \frac{1-d}{1-d_{u\to v}d_{v\to u}} \left(\frac{1-d_{u\to v}}{\|F(u)\|_1} + \frac{1-d_{v\to u}}{\|F(v)\|_1} \right) \\
&\geq (1-d) \left(\frac{2-d_{u\to v}-d_{v\to u}}{1-d_{u\to v}d_{v\to u}} \right). \qquad (19)
\end{aligned}$$

Influences are less than or equal to d, so it appears that $\frac{2-d_{u\to v}-d_{v\to u}}{1-d_{u\to v}d_{v\to u}}$ is minimal for $d_{u\to v} = d_{v\to u} = d$, leading us to:

$$\mathrm{dist}(u,v) \geq (1-d)\left(\frac{2-2d}{1-d^2}\right) = 2\frac{1-d}{1+d}. \qquad (20)$$

We have shown that $2\frac{1-d}{1+d}$ is a lower bound for $\mathrm{dist}(u,v)$. It is easy to verify that this bound is reached if u only links to v and v only links to u. Conversely, this bound can only be reached if both influences are equal to d. This only happens if u only links to v and v only links to u. ■

5.3 Clones

In many real and modelized small-worlds, there exist nodes that have exactly the same outgoing links. Such nodes will be called clones. For instance, clones may occur in Web graphs when pages are the exact copies of others. Distances between clones are rather easy to describe, as shown by Theorem 3:

Theorem 3 *If a node v is a clone of a node u (u and v have exactly the same outgoing links), then* $\mathrm{dist}(u,v) \geq 2(1-d)$. *If no leave is reachable from them, then there is equality.*

Proof. If u and v have the same outgoing links, the expressions of $F(u)$ and $F(v)$ only differ for the first term of the sum (see equation 8), so we have:

$$F(u) - F(v) = (1-d)(\delta_u - \delta_v). \qquad (21)$$

It appears then that $\|F(u)\|_1 = \|F(v)\|_1$, thus we have

$$\text{dist}(u,v) = \|C(u) - C(v)\|_1 = \frac{2(1-d)}{\|F(u)\|_1} \geq 2(1-d). \qquad (22)$$

Equality stands for $\|F(u)\|_1 = 1$ $(= \|F(v)\|_1)$. This is the case if, and only if, no leaf-loss happens during the diffusion, meaning no leaf is accessible from u (or v). ■

Note that the fact u and v are clones does not mean they cannot be differentiated by other nodes. However, there is a case where nodes different from u and v cannot distinguish u from v. Theorem 4 explicits this case.

Theorem 4 *If u and v have the same outgoing links and the same incoming links, then for any $w \notin \{u,v\}$, we have* $\text{dist}(u,w) = \text{dist}(v,w)$.

In other words, even if u and v are different (Theorem 3 says there is a distance of at least $2(1-d)$ between them), the other nodes cannot separate them. We could say that u and v are different, but they are the only ones aware of that fact. This is more subtle compared to the fixed-length random walk approach where u and v just share the same coordinates. Note that just having the same incoming links is insufficient to deduce something. For instance, nodes 2 and 4 of Figure 6 (or nodes 6 and 8) have the same incoming links, yet there is little resemblance between them.

Proof. As u and v are clones, $F(u)$ and $F(v)$ only differ by the u^{th} and the v^{th} components. Having the same incoming links leads to $F_u(u) = F_v(v)$ and $F_v(u) = F_u(v)$, but also to $F_u(w) = F_v(w)$ for any $w \notin \{u,v\}$. This assures that $\text{dist}(u,w) = \|C(u) - C(w)\|_1 = \|C(v) - C(w)\|_1 = \text{dist}(u,w)$. Note that this result does not depend on the choice of the norm (it is just due to a permutation between two components of the coordinates). ■

5.4 Uninfluenced nodes

Another special case is when two distinct nodes u and v do not share a common cycle. That means at least one of those nodes, say v, is not reachable from u. Influence from u to v is thus null. In other words, v is uninfluenced by u. For instance, a node without incoming link receives no influence from the rest of the graph. Distance between such nodes has a lower bound, as shown by Theorem 5.

Theorem 5 *Let u and v be two nodes, such that $d_{u\to v} = 0$. Then* $\text{dist}(u,v)$ *is greater or equal to* $2(1-d)$.

Proof. The v^{th} component of $C(v)$ verify $C_v(v) \geq F_v(v) \geq (1-d)$, thus we also have $\sum_{w \neq v} C_w(v) \leq d$. For u, we have $C_v(u) = 0$ and $\sum_{w \neq v} C_w(u) = 1$. This leads to:

$$\begin{aligned} \mathrm{dist}(u,v) &= \|C(u) - C(v)\|_1 \\ &\geq C_v(v) - C_v(u) + \sum_{w \neq v} C_w(u) - \sum_{w \neq v} C_w(v) \\ &\geq 2(1-d). \end{aligned} \tag{23}$$

This lower bound is reached, for instance, if v has no incoming link and a single outgoing link to u and no leaf is reachable from u. ■

We note the lower bound for uninfluenced nodes is the same as for clones, that is $2(1-d)$. An interpretation of this *coincidence* is that $2(1-d)$ is a critical value. It may be a good idea to consider nodes that are distant by less than $2(1-d)$ as close; being closer than $2(1-d)$ means there is a non-trivial structural proximity with strong reciprocal influence. For example, it is impossible to go below $2(1-d)$ by duplicating the outgoing links.

5.5 Cycles

We have seen in Section 5.2 that the cycle of length 2 breaks the $2(1-d)$ barrier. We can wonder what happens for a cycle of arbitrary length. Fortunately, a cycle is a simple structure and distances between its nodes are explicitly given by Theorem 6.

Theorem 6 *Let G be a cycle of length $n \geq 2$ and l a positive integer less than n. If u and v are nodes of G such that v is the l^{th} successor of u, then* $\mathrm{dist}(u,v)$ *is* $2\frac{(1-d^l)(1-d^{n-l})}{1-d^n}$.

Proof. If nodes of G are relabeled from 0 to $n-1$ starting from u, then u is labeled 0 and v is labeled l. In order to prove that $\mathrm{dist}(u,v) = 2\frac{(1-d^l)(1-d^{n-l})}{1-d^n}$, we have to express coordinates of u and v:

$$\begin{cases} C(u)=C(0)=\frac{1-d}{1-d^n}(\ \ 1\ \ ,\ldots,d^{l-1},d^l,\ldots,\ d^{n-1}), \\ C(v)=C(l)=\frac{1-d}{1-d^n}(d^{n-l},\ldots,d^{n-1},1,\ldots,d^{n-1-l}). \end{cases} \tag{24}$$

Calculation of $\mathrm{dist}(u,v)$ is then straightforward:

$$\begin{aligned}\mathrm{dist}(u,v) &= \|C(u)-C(v)\|_1 \\ &= \frac{1-d}{1-d^n}\left((1-d^{n-l})\sum_{i=0}^{l-1}d^i + (1-d^l)\sum_{i=0}^{n-1-l}d^i\right) \\ &= \frac{1-d}{1-d^n}\left((1-d^{n-l})\frac{1-d^l}{1-d} + (1-d^l)\frac{1-d^{n-l}}{1-d}\right) \\ &= 2\frac{(1-d^l)(1-d^{n-l})}{1-d^n}. \end{aligned} \tag{25}$$

■

Theorem 6 shows that the minimal distance in cycles is obtained for consecutive nodes and has value

$$\mathrm{dist}(0,1) = 2(1-d)\frac{1-d^{n-1}}{1-d^n}. \tag{26}$$

As $\frac{1-d^{n-1}}{1-d^n}$ is strictly less than 1, consecutive nodes are below the critical value $2(1-d)$. As a special case, $n=2$ leads to the already known minimal distance $d_{\min} = 2\frac{1-d}{1+d}$.

On the other hand, maximal distance is reached for opposite nodes and is equal to:

$$\mathrm{dist}(0,\lceil\frac{n}{2}\rceil) = 2\frac{(1-d^{\lceil\frac{n}{2}\rceil})(1-d^{n-\lceil\frac{n}{2}\rceil})}{1-d^n}. \tag{27}$$

The maximal distance is less than $2(1-d)$ only for $n=2$ and $n=3$ (where it is the same as the minimal distance). For greater values, we can consider that extrema of a cycle are not intimately connected.

Lastly, we notice that distances increase when the cycle length tends towards infinity. Minimal distance asymptotically tends towards $2(1-d)$, and maximal distance tends towards 2.

5.6 Cliques

According to Theorem 3, in a complete graph (with loops), all nodes are $2(1-d)$ distant (they have exactly the same outgoing links). One can ask if a smaller universal distance can be achieve between a set of nodes. The answer is yes, as shown by Theorem 7.

Theorem 7 *Let G be a complete symmetric digraph without loop of size n. The distance between any two distinct nodes u and v of G is*

$$\mathrm{dist}(u,v) = 2(1-d)\frac{n-1}{n-1+d}. \tag{28}$$

Proof. Let u be a node of G. The u^{th} component of $C(u)$ will be noted x, while the value of other components (that have all the same value for symmetry reasons) will be noted y. The relations between x and y are:

$$x + (n-1)y = 1 \text{ } (C(u) \text{ is a probability}), \tag{29}$$

$$x = (1-d) + dy \text{ (flow passing through } u). \tag{30}$$

The unique solution of this system is $x = \frac{(n-1)(1-d)+d}{n-1+d}$ and $y = \frac{d}{n-1+d}$. Considering that for $v \neq u$, $C(v)$ is just an inversion of the u^{th} and the v^{th} component of $C(u)$, we have

$$\begin{aligned} \text{dist}(u,v) = 2\,|x-y| &= 2\frac{(n-1)(1-d)+d}{n-1+d} - \frac{d}{n-1+d} \\ &= 2(1-d)\frac{n-1}{n-1+d}. \end{aligned} \tag{31}$$

■

For $n = 2$, we find $d_{\min}$ (a 2-clique without loop is a 2-cycle). The interpretation that a complete graph without loop brings the nodes closer than the same complete graph with loops (with ratio $\frac{n-1}{n-1+d}$) is that a loop on a node u tends to move away u from the nodes reachable from u by reenforcing the u^{th} component of u. This explains for instance why in Figure 6, node 3 is farther from 5 that node 4.

5.7 Leaves

If u is a leaf, then $C(u) = \delta_u$: leaves are placed at the ends of the simplex. As shown by Theorem 8, there is a minimal distance between a leaf and other nodes of the graph that is only reached by predecessors of u.

Theorem 8 *Let u be a leaf of G. For any node v distinct from u,* $\text{dist}(u,v)$ *is greater or equal to $\frac{2}{1+d}$, with equality if, and only if, v's only outgoing link is u.*

Proof. As $C(u) = \delta_u$, we have

$$\begin{aligned} \text{dist}(u,v) &= \|C(u) - C(v)\|_1 \\ &= |C_u(u) - C_u(v)| + \sum_{w \neq u} |C_w(u) - C_w(v)| \\ &= 1 - C_u(v) + \sum_{w \neq u} C_w(v) = 2(1 - C_u(v)). \end{aligned} \tag{32}$$

Consequently, finding a lower bound for $\text{dist}(u,v)$ comes down to finding an upper bound for $C_u(v)$. If we consider that $F_u(v) = d_{v \to u} F_u(u) =$

$d_{v\to u}(1-d)$, we can majorize $C_u(v)$:

$$\begin{aligned} C_u(v) = \frac{d_{v\to u}(1-d)}{\|F(u)\|_1} &\leq \frac{d_{v\to u}(1-d)}{F_u(u)+F_u(v)} \\ &\leq \frac{d_{v\to u}(1-d)}{(1-d)+(1-d)d_{v\to u}} \leq \frac{d_{v\to u}(1-d)}{(1-d)(1+d_{v\to u})} \\ &\leq 1-\frac{1}{(1+d_{v\to u})}. \end{aligned} \tag{33}$$

We know that an upper bound for $d_{v\to u}$ is d, so $C_u(v)$ is less than $1-\frac{1}{1+d}$ and $\mathrm{dist}(u,v)$ (that equals $2(1-C_u(v))$) is greater than $\frac{2}{1+d}$.

We have shown that $\frac{2}{1+d}$ is a lower bound for $\mathrm{dist}(u,v)$. It is easy to verify that this bound is reached if v only links to u. Conversely, equality implies that $d_{v\to u}$ equals d, and that can only happen if v only links to u. ■

6. Conclusion

The main technical differences between the fixed-length model and the damping model are summarized in Table 1.

The principal advantages of the method described above are:

- We apply a PageRank-like importance algorithm, with damping factor and importance. It enables working on weighted (di)graph, without having to modify the graph by making it symmetrical and reflexive, which is necessary if one uses a simple random walk.
- Since the damping factor is a real number, it offers a continuous framework for establishing the average length of random walks, whereas, if one uses a simple random walk with a fixed length, the framework is discrete and, therefore, less flexible.
- As we saw in Section 4.4, this approach has a relatively limited complexity since for a graph G with n nodes the complexity is only increased by a factor of $\ln(n)$ relative to the classic fixed-length random walks, without either a damping factor or importance source.
- When we consider the distributions of the random walks in a graph G with n nodes, we can then fit the graph into $\mathbb{R}^n$. This allows us to use the whole panoply of geometrical tools to analyze the graph structure.

This last point allows us, for example, to develop tools for visualizing and navigating in *small-world shaped* real-world graphs such as the Web with its associated relevant metrology, and also to develop new models of small worlds generation[10].

[6] Complexity to compute coordinates of a single node.

	Fixed model	Damping model
parameter	$k = \frac{d}{1-d}$	$d = \frac{k}{k+1}$
to be used with	smoothed graphs	any kind of graphs
iterative equation	$P_{n+1} = P_n A$	$P_{n+1} = dP_n A + (1-d)P_0$
# of iterations	k	$\bigcirc(k \ln(n))$
average length	k	$\frac{d}{1-d}$
Complexity[6]	km	$\bigcirc(km \ln(n))$

Table 1. Fixed Model vs Fading Model: recapitulation.

References

[1] S. Abiteboul, M. Preda, and G. Cobena. Adaptive on-line page importance computation. In *Proc. 12th International World Wide Web Conference*, pages 280–290, 2003.

[2] R. Albert and A.-L. Barabasi. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47, 2002.

[3] L. Barrière, P. Fraigniaud, E. Kranakis, and D. Krizanc. Efficient routing in networks with long range contacts. *Lecture Notes in Computer Science*, 2180:270+, 2001.

[4] M. Bianchini, M. Gori, and F. Scarselli. Inside pagerank. In *ACM Transactions on Internet Technology*, 2003.

[5] P. Boldi, M. Santini, and S. Vigna. Pagerank as a function of the damping factor. In *Proc. of the Fourteenth International World Wide Web Conference, Chiba, Japan, 2005. ACM Press.*, 2005.

[6] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine. *Computer Networks and ISDN Systems*, 30(1–7):107–117, 1998.

[7] A. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web. *Comput. Networks*, 33(1-6):309–320, 2000.

[8] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. In *STOC '87: Proceedings of the nineteenth annual ACM conference on Theory of computing*, pages 1–6, New York, NY, USA, 1987. ACM Press.

[9] B. Gaume. Balades aléatoire dans les petits mondes lexicaux. *Information Engineering Sciences*, 4(2), 2004.

[10] B. Gaume, C. Barré, and F. Mathieu. From random graphs to small worlds by random walks. *To be published*, 2006.

[11] J. Kleinberg. The Small-World Phenomenon: An Algorithmic Perspective. In *Proceedings of the 32nd ACM Symposium on Theory of Computing*, 2000.

[12] M. Latapy and P. Pons. Computing communities in large networks using random walks. Technical report, Submitted preprint, arxiv.org/cond-mat/0412368.

[13] F. Mathieu. *Web Graphs and PageRank-like Measurements*. PhD thesis, Université Montpellier 2 - LIRMM, December 2004.

[14] S. Milgram. The small world problem. *Psychology Today*, 61:60 – 67, May 1967.

[15] M. Newman. The structure and function of complex networks, 2003.

[16] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. Technical report, Computer Science Department, Stanford University, 1998.

[17] S. H. Strogatz. Exploring complex networks. *Nature*, 410:268–276, March 8 2001.

[18] D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.