Mapping the forms of meaning in small worlds

Bruno Gaume

Mapping the Forms of Meaning in Small Worlds

Bruno Gaume

IRIT, UPS, 118 route de Narbonne,
31062, Toulouse cedex 4, France
gaume@irit.fr

Abstract. Prox is a stochastic method to map the local and global structures of real-world complex networks which are called Small Worlds. Prox transforms a graph into a Markov chain, the states of which are the nodes of the graph in question. Particles wander from one node to another within the graph by following the graph's edges. It is the dynamics of the particles' trajectories that map the structural properties of the graphs that are studied. Concrete examples are presented in a graph of synonyms to illustrate this approach.

1 Introduction

Recent research in graph theory has brought to light a whole series of properties which most real networks share in common: these characteristics define the Small World (SW) class of graphs. This is the case for protein interaction networks, the graph of the world wide web, the telephone calls graphs, the graphs of co-authors of scientific publications, lexical graphs, etc. These graphs have a very particular topology, in which the relationship between the local structure and the global structure bears no comparison with traditionally-studied random or regular graphs. This accounts for the considerable interest that SWs have generated in different scientific communities. One can hypothesize that these characteristics reflect the properties of the systems which the real networks describe, and that therefore the study of their structures will permit a better understanding of the phenomena from which they come. In the present article we present Prox which is a particularly well-adapted method for drawing the structure of SWs. Part 2 presents a brief summary of the main properties of SW graphs, and provides a rapid overview of the lexical graphs that will be used as concrete examples in this article. In Part 3 we will see how the dynamics of random walks on an SW are substantially constrained and channelled by the topological structure of SWs. In Part 4 we will show that we can map the form of meaning in SW graphs by analysing the dynamic of random walks in SWs. In Part 5 we will be using these dynamics to superpose global information contained in a graph on a topological extraction performed on this graph and projected onto a local map. We discuss the complexity of Prox in Part 6. In part 7 we will show how this method can be applied to creating a model of young children's semantic approximations during the acquisition-phase, with results from a study of the child corpus. Finally in Part 8, we will draw conclusions and consider the perspectives.

2 The properties of real-world complex networks

A presentation of the SWs can be found, for example, in [1]. Real networks are sparse: in a graph with n nodes, the maximum number of possible edges is $O(n^2)$ while the number of edges in real networks is generally $<O(n\log(n))$. In 1998, Watts and Strogatz [2] proposed two indicators to characterise a large sparse graph G: its L and its C, where L = "characteristic path length": the mean of the shortest path between two nodes of G; and C = "clustering coefficient": $C \in [0,1]$, and measures a graph's tendency to possess zones denser in edges. (The more clustered the graph, the more the graph's C approaches 1, whereas in random graphs C is very close to 0). In applying these criteria to different types of graphs, these researchers found that:

– **real networks** have a tendency to have a small L: generally there is at least one short path between any two nodes;
– **real networks** have a tendency to have a large C: this reflects a relative tendency for two neighbours on the same node to be directed inter-connected;
– **random graphs** have a small L: when one constructs a graph randomly with a density of edges comparable to real networks, one obtains graphs with a small L;
– **random graphs** have a small C: they are not formed from aggregates. In a random graph there is no reason why neighbours on a same node are more likely to be connected than any two nodes, hence the weakness of their tendency to form aggregates.

Echoing the "small world phenomenon" [3], Watts and Strogatz proposed calling graphs which have these two characteristics (a small L and a large C) *"small worlds"*, which they found in all the real networks they observed, and which they postulated as universal for real networks. More recent research has shown that most small worlds also have a hierarchical structure. The distribution of the degrees of incidence follows a power law. The probability P(k) that a given node has k neighbours decreases as a power law, $P(k) \approx k^{-\lambda}$, where λ is a constant characteristic of the graph [4], while random graphs conforms to a Poisson Law.

There are several types of lexical graphs, varying according to the semantic relation that defines the graph's edges (the nodes representing the lexical units of a language: from some tens of thousands to hundreds of thousands of elements, depending on the language and the coverage of the corpus used). The two principal types of relations used are:

– **Syntagmatic relationships**, or rather relationships of co-occurrence: one constructs an edge between two words if one finds them close to each other in a large corpus, typically at a maximum distance of two or three words or more, (cf. [5]).

– **Paradigmatic relationships**, particularly of synonymy: using lexical data bases, such as the well-known WordNet [6], one constructs a graph in which two nodes are linked by an edge if the corresponding words have a relationship of synonymy.

All these graphs are clearly of the SW type [7]: they are sparse, show a strong clustering coefficient, a very small characteristic path length, as well as a hierarchical structure (the incidence curve ≈ power law). For example, DicoSyn.Verbe[1] is a symmetric and reflexive graph with roughly 9,000 nodes, 50,000 edges, its L ≈ 4 and its C ≈ 0.3, as is typical of an SW. The curve representing the distribution of the degrees of incidence of its nodes (see Fig. 1) is typical of SW graphs (in log-log, it forms approximately a straight line).

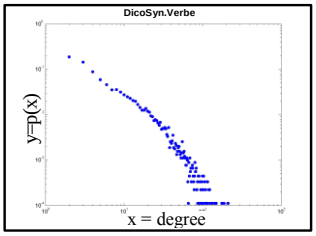

Fig. 1. Log-log curve of the distribution of incidences of the DicoSyn.Verbe nodes

3 Wandering in a small world

Notation:
If U is a vector row with the dimension n, we will note $[U]_i$: the i^{th} value of U;
If M is a matrix mxn then we will note:
$[M]_{i\,k}$: the value located at the intersection of the i^{th} row and the k^{th} column of M;
$[M]_{i\,\bullet}$: the i^{th} vector row of M;
$[M]_{\bullet\,k}$: the k^{th} vector column of M.

We assume that we have a connected graph, that is reflexive, G = (V,E) with n = |V| nodes and m = |E| edges, and that in this graph a particle can at any moment $t \in \mathbb{N}$ wander randomly on the nodes:
– At time t the particle is on a node $r \in V$;
– When the particle is at time t on a node $r \in V$, it can only reach, at time t+1, a node s randomly and uniformly selected among the neighbours of the node r.

Let Â be the one-step transition matrix from the Markov chain associated to the random walk on the graph. That is to say that, at each stage, the probability of transition from the node $r \in V$ to the node $s \in V$ is equal to $[\hat{A}]_{r\,s} = [A]_{r\,s}/d(r)$, where A is the adja-

[1] *DicoSyn* is a synonym dictionary consisting of seven classic French dictionaries (Bailly, Benac, Du Chazaud, Guizot, Lafaye, Larousse and Robert) from which the synonymic relationships were extracted by ATILF (http://www.atilf.fr/) subsequently homogenized at CRISCO (http://elsap1.unicaen.fr/). *DicoSyn.Verbe* is the graph of the verbs extracted from *DicoSyn*: there is an edge {a,b} if the verbs represented by the nodes a and b are synonyms in *DicoSyn*.

cency matrix of graph G: $A_{r\,s}$=1 if $\{r,s\}\in E$ and 0 otherwise; and d(r) is the degree of node r: $d(r)=\sum_{x\in V}(A_{r\,x})$.

If the initial law of the Markov chain given by the vector row P (i.e., $[P]_r$ is the probability that the particle is on the node r at time t = 0), then $[P\hat{A}^t]_r$ is the probability that the particle is on the node r at time t.

Let $F\subseteq V$, a non-empty set of nodes. We should note that P^F is the vector of n dimensions, such that $[P^F]_r = 1/|F|$, if $r\in F$, and $[P^F]_r = 0$ if $r\notin F$. If the initial law of the Markov chain is given by the vector P^F, this will therefore correspond to a random walk, starting on the nodes of F, each of which is equiprobable. $[(P^F)\hat{A}^t]_s$ is then the probability that the particle will be on the node s at time t when the particle starts its random walk equiprobably on one of the nodes of F at t = 0. One should note that $[(P^{\{r\}})\hat{A}^t]_s = [\hat{A}]_{r\,s}$ which is then the probability that the particle is on the node s at time t when the particle starts its walk on node r at t = 0.

One can demonstrate[2] that if G = (V,E) is a connected and reflextive graph, then:

$$\forall r,s\in V,\ \lim_{t\to\infty}[\hat{A}^t]_{r\,s} = d(s)/\sum_{x\in V}(d(x)). \qquad \textbf{(1)}$$

The probability of being on node s at time t (when t is long enough) no longer depends on the departure node r, but solely on the degree of s and is equal to $d(s)/\sum_{x\in V}(d(x))$.

On the other hand, since L, the characteristic path length, is small in an SW, we know that two nodes are generally linked by at least one relatively short path. However two types of topological configuration can differentiate between two nodes s and u in their relationship from node r.

Configuration 1: the node r can be linked to the node s by many short paths (there is a strong confluence going from r to s);
Configuration 2: the node r can be linked to the node u by only a few short paths (there is only a weak confluence going from r to u).

If formula (1) indicates that when t is long enough, the probability to find itself at time t on the node s does not depend on the departure node, nevertheless the dynamic towards this limit depends strongly on the departure node and the type of confluence that it has towards the node s. For example, when d(s)=d(u), then by formula (1), the sequences $([\hat{A}^t]_{r\,s})_{0\le t}$ and $([\hat{A}^t]_{r\,u})_{0\le t}$ converge towards the same limit $d(s)/\sum_{x\in V}(d(x))$; however these two sequences are not identical. In fact the dynamic of the particle's trajectory on its random walk is completely determined by the graph's topological structure: after t steps, every node s at a distance of t edges or less from the departure node can be reached. When t remains small, the probability of reaching a node at the

[2] See [7], this is a consequence of the Perron-Frobenius theorem [8], since when the graph G=(V,E) is connected and reflexive, $\hat{A}$ the transition matrix of the Markov chain associated with a random walk on graph G is then ergodic.

t^{th} step depends on the number of paths between the departure node and node s, on their length and on the structure of the graph around the intermediary nodes along the way (the more paths there are, the shorter the paths, the weaker the degree of the nodes, the greater the probability of reaching s from the departure node at the t^{th} step – when t remains small). Therefore, if the confluence from node r towards s is stronger than the confluence from node r towards u, then for a random walk of length t that is not too long, there is $[\hat{A}^t]_{r\,s} > [\hat{A}^t]_{r\,u}$. At the beginning of the random walk from a departure node, the particle will more likely pass through the nodes towards which the departure node has strong confluence. For example in DicoSyn.Verbe the nodes "dépiauter" *[to skin, as "to skin a animal"]* and "rêvasser" *["to day-dream"]* have the same number of neighbours (d(dépiauter) = d(rêvasser)), and so, following (1):

$$\lim_{t\to\infty} [\hat{A}^t]_{\text{déshabiller dépiauter}} = \lim_{t\to\infty} [\hat{A}^t]_{\text{déshabiller rêvasser}}.$$

One can however see in Fig. 2 that the two sequences $([\hat{A}^t]_{\text{déshabiller dépiauter}})_{0\leq t}$ and $([\hat{A}^t]_{\text{déshabiller rêvasser}})_{0\leq t}$, are very different for a small t, which shows us that the confluence from "déshabiller" *[to undress]* towards "dépiauter" is stronger than the confluence from "déshabiller" towards "rêvasser".

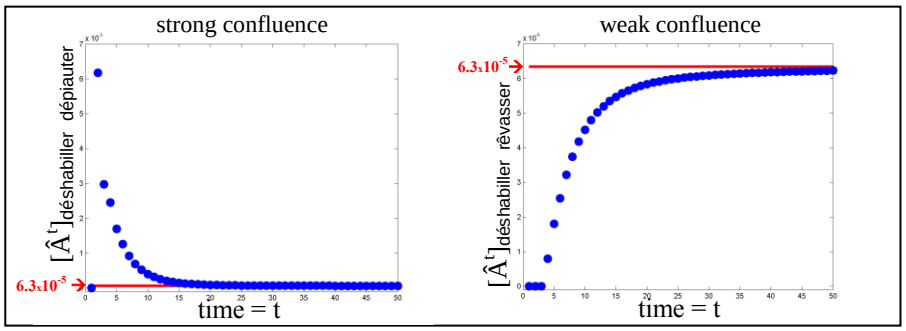

Fig. 2. $([\hat{A}^t]_{\text{déshabiller dépiauter}})_{0\leq t}$ and $([\hat{A}^t]_{\text{déshabiller rêvasser}})_{0\leq t}$ in DicoSyn.Verbe

We can choose therefore t between L and 2L in order to reach nearly all the nodes from whichever happens to be the departure node, without however attaining the limit when the t becomes too large. In the rest of this article, we illustrate this process using DicoSyn.Verbe of the $L \approx 4$, selecting $t = 5$.

4 Forms of meaning

For a given t one can consider $\hat{A}^t$ as a matrix in which $\forall r,s \in V$, $[\hat{A}^t]_{r\,s}$ tells us the confluence level of the node r towards the node s. For example, Fig. 3 is a list in decreasing order of 50 nodes towards which the node "déshabiller" has the strongest confluence (scored at $t = 5$) in the DicoSyn.Verbe graph. In other words, Fig. 3 is the list in decreasing order of the set: $\{x \in V, [\hat{A}^t]_{\text{déshabiller x}} \geq 0.0026\}$ which contains 50 elements.

1→dépouiller *[to skin]*, **2→défaire** *[to open (a packet)]*, **3→démunir** *[to deprive]*, **4→déshabiller** *[to undress]*, **5→découvrir** *[to uncover]*, **6→dénuder** *[to strip]*, **7→montrer** *[to show]*, **8→dévêtir** *[to undress]*, **9→dégarnir** *[to clear (a table)]*, **10 révéler** *[to reveal]*, **11→étaler** *[to spread]*, **12→ôter** *[to take away]*, **13→écorcher** *[to skin (an animal)]*, **14→délacer** *[to unlace]*, **15 dévoiler** *[to unveil]*, **16→démasquer** *[to unmask]*, **17→médire** *[to denigrate]*, **18 dégager** *[to clear up]*, **19→exhiber** *[to put on show]*, **20 afficher** *[to put on display]*, **21 enlever** *[to take away]*, **22 dénouer** *[to untie (a knot)]*, **23 desserrer** *[to loosen]*, **24 désaffubler** *[to take off (strange clothes)]*, **25 arracher** *[to tear off]*, **26 voler** *[to steal]*, **27 dépourvoir** *[to lack]*, **28 développer** *[to develop]*, **29 exposer** *[to expose]*, **30 déchirer** *[to tear up]*, **31 ouvrir** *[to open]*, **32 déchausser** *[to unshoe]*, **33 débarrasser** *[to get rid of]*, **34 déployer** *[to deploy]*, **35 priver** *[to deprive]*, **36 trahir** *[to betray]*, **37 dépiauter** *[to skin (an animal) or to analyse in detail]*, **38 vider** *[to empty]*, **39 frustrer** *[to frustrate]*, **40 démontrer** *[to demonstrate]*, **41 déposséder** *[to dispossess]*, **42 prouver** *[to prove]*, **43 faire voir** *[to make seen]*, **44 tailler** *[to prune]*, **45 peler** *[to peal]*, **46 deviner** *[to guess]*, **47 sevrer** *[to sever]*, **48 dénantir** *[to take away(a possession)]*, **49 tondre** *[to cut (a lawn)]*, **50 couper** *[to cut]*, **…**

Fig. 3. The 50 nodes with the strongest confluence from "déshabiller" in *DicoSyn.Verbe*

In Fig. 3, the words with an arrow → are the 16 neighbours of the verb "déshabiller". One can see that the verb "dépouiller" *[to skin]*, which is a hypernym of "déshabiller", is the verb with which "déshabiller" has the strongest confluence (measured at t = 5). The verb "dépiauter" is 37th, and the verb "défaire" *[to undo or to open (a packet)]* which has a strong incidence (d(défaire) = 81), is a high-level hypernym of "déshabiller", and is 2nd in this table.

Another approach is to consider $\hat{A}^t$ as an $n_x n$ matrix with the coordinates of n vector rows $([\hat{A}^t]_{x\bullet})_{x\in V}$ in $\mathbb{R}^n$. This viewpoint then allows us to plunge the graph G = (V,E) in $\mathbb{R}^n$, where a node $r\in V$ has the vector row $[\hat{A}^t]_{r\bullet}$ for coordinates in $\mathbb{R}^n$. The idea[3] is that two nodes r and s, with the coordinates $[\hat{A}^t]_{r\bullet}$ and $[\hat{A}^t]_{s\bullet}$ in $\mathbb{R}^n$, will be even closer in $\mathbb{R}^n$ when their relationship to the whole graph is similar. For example in Fig. 4, the graph G1 has 9 nodes, and nodes 5 and 7 have exactly the same neighbours: {5, 6, 7}, which entails $\forall t\in\mathbb{N}^*$, $[\hat{A}^t]_{5\bullet} = [\hat{A}^t]_{7\bullet}$, their coordinates in $\mathbb{R}^9$ are equal, and the edge {5,7} has therefore a length of zero, while the edge {4,6} is the longest, with a length of 0.2740 at t = 5.

[3] This idea was firstly proposed by [7] for metrology of small worlds and linguistic modeling of lexical wide-area networks : http://Prox.irit.fr

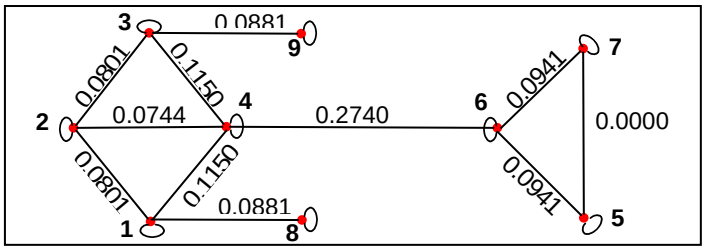

Fig. 4. The geometric length in $\mathbb{R}^9$ of the edges of G1 at t = 5

It is the combination of these two approaches ($\hat{A}^t$ as an nxn matrix of coordinates of n vector rows in $\mathbb{R}^n$, or $\hat{A}^t$ as an nxn confluence matrix between the n nodes of the graph) which will enable us to display a graph globally or locally, while still taking its global structure into account.

The $\hat{A}^t$ matrix, as a matrix of coordinates in $\mathbb{R}^n$, contains information calculated from the whole graph that could be represented in $\mathbb{R}^3$ by means of a Principle Components Analysis (PCA) of $\hat{A}^t$, retaining the first 3 axes. For example, Fig. 5 illustrates the 3D form of the graph G1 of Fig. 4.

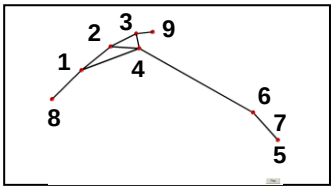

Fig. 5. The map of G1 at t = 5 (on the first 3 axes of the PCA of $\hat{A}^5$)

However the 9000 nodes of our DicoSyn.Verbe graph present an unreadable image when displayed on a screen. We will therefore only show *"one part around"* a set of nodes F. If one wishes to observe the structure of graph G = (V,E) around a non-empty set of nodes F⊆V with a *"map around F"*, then:

(a) One extracts the set $T_{t,F,\alpha} \subseteq V$, where $T_{t,F,\alpha} = \{x \in V, [(P^F)\hat{A}^t]_x \geq \alpha\}$. Let $R_{t,F,\alpha}=|T_{t,F,\alpha}|$. For $\alpha \in [0,1]$, $T_{t,F,\alpha}$ is to some extent a *"topological extraction around F"*, if F = {r} is a singleton, then it is the $R_{t,F,\alpha}$ nodes $x \in V$ of which the confluences towards x are stronger that are selected. For instance, we can see in Fig. 3 the set $T_{5,\{déshabiller\},0.0026}$.

(b) One extracts the matrix $M_{t,F,\alpha}$ from $\hat{A}^t$ matrix, the former being the squared matrix $(R_{t,F,\alpha})x(R_{t,F,\alpha})$, formed by the intersection of the $R_{t,F,\alpha}$ rows $[\hat{A}^t]_{x\bullet}$ with $R_{t,F,\alpha}$ columns $[\hat{A}^t]_{\bullet x}$ such that $x \in T_{t,F,\alpha}$. $M_{t,F,\alpha}$ is a sort of *"topological zoom around F"*.

(c) One then normalizes the $R_{t,F,\alpha}$ rows of $M_{t,F,\alpha}$ (for each $x \in T_{t,F,\alpha}$, replacing the row $[M_{t,F,\alpha}]_{x\bullet}$ with $[M_{t,F,\alpha}]_{x\bullet}/\|[M_{t,F,\alpha}]_{x\bullet}\|$, where $\|.\|$ is the Euclidean norm of $\mathbb{R}^n$). One then performs a PCA on $M_{t,F,\alpha}$, retaining only the first 3 dimensions (which then preserve

most of the information relevant to the reduction to $\mathbb{R}^3$) so as to obtain $D_{t,F,\alpha}$ which is the 3D display of the region[4] in G, maping the *"topologogical confluence region around F"* for a given time t.

Example 1: Fig. 8.a shows us $D_{5,\{jouer\},0.0015}$, the 3D map around the singleton {jouer} *[to play]*, in other words the first 3 coordinates of the PCA of $M_{t,F,\alpha}$ for $t = 5$, $F = \{jouer\}$, $\alpha = 0.0015$, where $R_{t,F,\alpha} = 100$. We can see that the geometric structure of $D_{5,\{jouer\},0.0015}$ (the form that it has now taken) is a good reflection of the polysemic structure of the verb "jouer" with its four principle meanings in French: "tromper" *[to dupe]*, "imiter" *[to imitate]*, "parier" *[to bet]*, "s'amuser" *[to enjoy oneself]*.

Example 2: In asserting $F = \{monter, descendre\}$ *[to go up],[to go down]* and $\alpha = 0.001$, we can display $D_{5,\{monter, descendre\},0.001}$ (Fig. 9.a), where we can see that at the transition-point of the semantic articulation {"descendre", "monter"} we find "sauter" *[to jump]*, although this verb has no direct connection with "monter".

Only the major population centres and the major axes appear on a map of the whole world; if you are looking for more details about a given region, you consult another more detailed regional map. One could say that our local displays (where F is a singleton like {jouer}) plays the role of a regional map: for instance, in the region of "jouer" (Fig. 8.a), it is α that gives us the range of the *"topologogical confluence region around"* "jouer". To obtain a global view of a graph G = (V,E) without at the same time having to show all the nodes, but just the *"capital nodes"* which are at the heart of the major confluences, we just need to assert F = V.

Example 3: Fig. 10 shows $D_{5,V,0.0005}$, the global 3D display around V (all the nodes) for $\alpha = 0.0005$ and $t = 5$, where $R_{t,F,\alpha} = 200$. Using Dicosyn.Verbe, the geometric form obtained in this way has roughly the shape of a tetrahedron, organizing the French verbs in a semantic continuum thereby making four axes apparent (the corners of the tetrahedron): (1) LOCOMOTIVE, (2) POSITIVE, (3) FIXATIVE, (4) NEGATIVE.

Example 4: Fig. 11.a shows us $D_{5,V,0}$, applied to G2, (the two-dimensional 10x10 grid) and Fig. 11.b shows us $D_{5,V,0}$ applied to a G3 graph with eight nodes made up of four nodes forming a clique and 4 hanging nodes.

5 From global confluences to local forms through colours

In Fig. 8.b (the global map $D_{5,V,0.0005}$) the darker a node x, the larger is $[\hat{A}^t]_{jouer\ x}$ (the stronger is the confluence of the node "jouer" towards the node x). We can readily see that the node "jouer" is a highly polysemic verb, and it covers many meanings distributed across the whole semantic space. To construct $D_{5,\{jouer\},0.0015}$ (Fig 8.a), one starts

[4] For each non-empty set $F \subseteq V$, if $\alpha = 0$, then $T_{t,F,\alpha} = V$ and $R_{t,F,0} = n$; it will then be the n nodes of the whole graph which will be displayed in $D_{t,F,0}$.

by extracting $R_{5,\{jouer\},0.0015} = 100$ nodes in $T_{t,\{jouer\},0.0015}$, which are then displayed in $D_{5,\{jouer\},0.0015}$ (the local display around "jouer"), but then we can no longer know from which semantic region of $D_{5,V,0.0005}$ a displayed node has been extracted. In order to make up for this lack of available information, we will see how to display the relationship between the local form with the whole graph relative to a set of given nodes. We can choose, for example the four nodes in the corners of the tetrahedron {fuir, exciter, fixer, briser} *[to flee, to excite, to fix, to break]* in *DicoSyn.Verbe* and select 4 colour vectors[5] $\{B_{fuir}, B_{exciter}, B_{fixer}, B_{briser}\}$. For each node s in our graph we can assign a colour C_s in the following way:

$$C_s = \frac{([\hat{A}^t]_{s\ fuir})B_{fuir} + ([\hat{A}^t]_{s\ exciter})B_{exciter} + ([\hat{A}^t]_{s\ fixer})B_{fixer} + ([\hat{A}^t]_{s\ briser})B_{briser})}{[\hat{A}^t]_{s\ fuir} + [\hat{A}^t]_{s\ exciter} + [\hat{A}^t]_{s\ fixer} + [\hat{A}^t]_{s\ briser}}$$

C_s is the barycentre of the four colour vectors, weighted by the confluences of s towards each of the 4 respective nodes. The colour C_s of a node s therefore reflects the strength of the confluences that s has with the 4 nodes: fuir, exciter, fixer, briser (if we were to choose other nodes, we would observe other confluences). One can see in Fig. 8.a that it is the nodes {voler, coulisser, marcher, passer, couler} *[to fly, to slide, to walk, to pass, to flow]* – all verbs involving movement) that are closer to colour B_{fuir} (blue), which has been associated with "fuir" located at the LOCOMOTIVE angle of the tetrahedron. This shows us then that these 5 nodes have their strongest confluence with the nodes in this semantic region, the LOCOMOTIVE verbs.

6 Complexity

Fig. 6 provides a summary of the procedures for calculating the map $D_{t,F,\alpha}$ in which the $R_{t,F,\alpha}$ nodes are displayed.

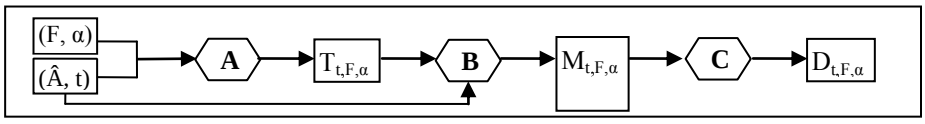

Fig. 6. Flow-chart: $(\hat{A}, t, F, \alpha) \rightarrow D_{t,F,\alpha}$

– Procedure A: One first calculates the vector $U \leftarrow \mathtt{Row}(\hat{A},t,P^F)$ (see Fig. 7), then one extracts the set $T_{t,F,\alpha} \subseteq V$, such that $T_{t,F,\alpha} = \{x \in V, U_x \geq \alpha\}$.
– Procedure B: For each of the $R_{t,F,\alpha}$ nodes $x \in T_{t,F,\alpha}$, one calculates $H_{x\bullet} \leftarrow \mathtt{Row}(\hat{A},t,P^{\{x\}})$ (see Fig. 7), then one constructs $M_{t,F,\alpha}$, the squared matrix $(R_{t,F,\alpha})x(R_{t,F,\alpha})$ made up of the $R_{t,F,\alpha}$ columns $H_{\bullet x}$, such that $x \in T_{t,F,\alpha}$.
– Procedure C: One normalizes each of the $R_{t,F,\alpha}$ rows of the matrix $M_{t,F,\alpha}$, then one conducts a PCA from which one retains the first 3 axes in $D_{t,F,\alpha}$.

[5] The colours coded in RGB are treated as vectors of dimension 3.

By exploiting the sparse structure of Â, the $n_x n$ matrix with m non null values, the computing time and the memory work-space required to calculate $U_{out} \leftarrow \mathtt{Row}(\hat{A},t,U_{in})$ are respectively O(n+tm) and O(2n+m). The cumulative computing time of the two procedures A and B is then $O((R_{t,F,\alpha})(n+tm))$, and memory work-space is O(2n+m).

Both the computing time and the memory work-space required for procedure C only depend on $R_{t,F,\alpha}$, (the dimension of the squared matrix $M_{t,F,\alpha}$), which never needs to be large for a relevant mapping either for the global or the local displays in the SWs, thanks to their hierarchical structures and strong C. By increasing $R_{t,F,\alpha}$ by K, the maximum number of nodes that can be displayed on a map (K is O(100) and depends on the maximum cognitive load that is accepted), one can consider that the computing time and the memory work-space for procedure C are constants. Then the time and space needed for calculating $D_{t,F,\alpha}$ are $O((R_{t,F,\alpha})(n+tm)) \leq O(K(n+tm))$ and O(2n+m) respectively. But generally in an SW, m<nlog(n) and one therefore arrives at computing time in O(Ktnlog(n)) and memory work-space in O(nlog(n)).

$U_{out} \leftarrow$ **function** Row(M,t,U_{in})
  $U_{out} \leftarrow U_{in}$;
  for i **from** 1 **to** t
  $U_{out} \leftarrow (U_{out})M$;
  end
end
	input:
	  M: sparse matrix$_{n \times n}$ of real
	  t: integer
	  U_{in}: full vector$_n$ of real
	output:
	  U_{out}: full vector$_n$ of real

Fig. 7. Algorithm: $U_{out} \leftarrow$ **function** Row(M, t, U_{in})

7 Small worlds and small words

Let G=(V,E), a lexical network containing n words. For each vertice $r \in V$, one can rank all the n vertex of V in decreasing order resulting $[\hat{A}^t]_{r\bullet}$ on V : $\forall r,x \in V$, $1 \leq rank_r(x) \leq n$ and $([\hat{A}^t]_{rx} < [\hat{A}^t]_{ry}) \Rightarrow (rank_r(y) < rank_r(x))$.

The ranking of words calculated by Prox, based on different lexical networks, are consistent with "semantic approximation by analogy" produced by young children. For example, the spontaneous utterance "I undress a tree" *[I peal the bark off a tree]*, produced by a 2½ year-old child, demonstrates a partial matching between these two verbs that is consistent with the low rank of "déshabiller" *"undress"* relative to "écorcer" *"peel"*: $rank_{écorcer}(\mathtt{déshabiller}) << n$.

Here are some examples of "semantic approximation by analogy" (taken from the corpus [2]).

*« je **déshabille** l'orange »* 36 mois (l'enfant épluche une orange)
[child **Déshabiller**/Adult **Eplucher**] (145^e)
"I undress an orange" age : 36 months (the child peels an orange)
[child **Undress**/adult **Peel**] (145^{th})

*« le livre est **cassé** »* 26 mois (le livre est déchiré) [child **Casser**/Adult **Déchirer**] (6^{e})
"The book is broken" age : 26 month (the book is torn) [child **Break**/adult **Tear**] (6^{th})

*« il faut la **soigner** la voiture »* 38 mois (il faut réparer la voiture)
[child **Soigner**/Adult **Réparer**] (332^{e})
"The car has to be cared for" age : 38 months (the car has to be repaired)
[child **Care for**/adult **Repair**] (332^{nd})

The ranking shown in the examples above indicated the rank of the verb uttered by the child relative to the verb uttered by the adult ($rank_{adult\_world}$(child_world)) to describe the same scene, calculated by Prox using DicoRob.verbe, the verb graph extracted from DicoRob[6] :

The child first learns words that correspond to "capitals cities", and uses these describe a large area: the child attempting to communicate event A [for example: tearing up a book] for which he does not dispose of the constituted verbal category (1) would make an analogy with a past event B [breaking a glass] already stored in memory with a lexical entry *"break"* and (2) using this analogy, says *"the book is broken"* to commicate event A. Then the child progressively acquire the words corresponding to "cities" less important than "the capital city", thereby refining the precision of his designation.

This analysis, based on a corpus of 230 "semantic approximations by analogy" produced by young children (aged 1.8 to 4.2 years) shows that the mean rank of the verb uttered by the child (such as *"break"*) relative to the "correct" word (like *"tear"*) is 239 (mean{$rank_{adult\_world}$(child_world)}=239), which is relatively low given the 10,860 verbs in the graph extracted from the "Grand Robert" [10].

8 Conclusion and perspectives

There are many applications[7] of this geometrical representation presented in Part 4. Identifying shortcuts through distance refining is one of them. The standard distance between two vertices r and s 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 often loses its interest: for almost any nodes r and s, there exists a short path connecting r and s, and it can be difficult to use this distance to distinguish nodes.

Consider a graph with n nodes G=(V,E) is plunged in $\mathbb{R}^n$ by the method mentioned above in Part 4. Any edge e between two nodes r and s has a canonical geometrical

[6] DicoRob is a graph constructed from the "Grand Robert 1994": Vertex are the entries in the Robert and there is a edge $r \leftrightarrow s$ if and only if r=s or if r is contained in the definition of s, or s is contained in the definition of r.

[7] See [7] for applications in linguistics, psycholinguistics and data processing.

weight. This correspond to the geometrical distance separating r and s in $\mathbb{R}^n$. For example, in Figure 4 , using euclidian distance, edge (2,4) has a weight equals to 0.0744, whereas the weight of edge (4,6) is equal to 0.2740. Comparing the distances in the graph gives some insight about the graph structure. Indeed we note that edges between nodes from different communities are often called short cuts ([1], [2], [11]). These edges enables low diameter of small worlds that enables better routing algorithm performances. Notice that nodes belonging to different communities are geometrically very distant in $\mathbb{R}^n$ while two nodes belonging in the same community are geometrically close. It is thus reasonable to claim that an edge geometrical length is a valuable estimation of its practical importance. In other words, *short cuts are long edges*. Figure 4, illustrate that the longest edge (4,6), is obviously the more important, since connectivity between {1,2,3,4,8,9} and {5,6,7} relies on it.

This type of approach, by exploiting the Small World structure of lexical networks, provides a new conceptual framework for establishing tools of lexical metrology. For example, traditionally, verb categorization has been built on the basis of syntactic structures and restrictive selection (the semantic properties of the verb arguments); this does not take account of the analogical relations between verbs. Thus "soigner" *"to care for"* and "ravaler" *"to clean the outside of"* are not, in this approach, grouped together under the same category because of the differences in their restrictive selection (complement / animate vs. inanimate/. On the contrary, our work makes it possible to group these two verbs together in the same "aggregate", labeled REMETTRE-EN-ÉTAT *TO-PUT-BACK-IN-ORDER*, these compounds being themselves structures by domain:

REMETTRE-EN-ETAT/CORPS → soigner, …
REMETTRE-EN-ETAT/BÂTIMENT → ravaler, …
REMETTRE-EN-ETAT/VÊTEMENT → rapiécer, …
TO-PUT-BACK-IN-ORDER /BODY → to provide care to …
TO-PUT-BACK-IN-ORDER /BUILDING → to clean the outside of …
TO-PUT-BACK-IN-ORDER /CLOTHING → to sew up …

We are currently developing a « proxemic » electronic dictionary from TLF-i[8] with his associated metrology. With this dictionary it will be possible to find a verb such as "peel" without knowing the word one is looking for by using a known analogous verb such as *"undress"* and a word designating the domain as *"tree"*. In fact, if one looks at the definition of "écorcer" *"peel"* there appear words like « écorce », "arbre", "grain", "fruit" *"bark", "tree", "kernel", "fruit"* which can be seen as close when Prox is queried for substantives. Thus, among the verbs close to "déshabiller" *"undress"* which are themselves close to "arbre" *"tree"* one finds:

DÉSHABILLER/ARBRE → tailler, décortiquer, démascler, entailler, écorcer, effeuiller, émonder, inciser
UNDRESS/TREE → *prune, derind, strip, tap, debark, unleaf, cut off the deadwood, incise, clip*

[8] http://atilf.atilf.fr/tlf.htm

8 References

1. Newman M.E.J.: The structure and fonction of complex networks, (2003), http://www.santafe.edu/~mark/recentpubs.htmls
2. Watts D.J., Strogatz S.H.: Collective dynamics of 'small-world' networks. In Nature 393: 440-442, (1998), http://tam.cornell.edu/SS_nature_smallworld.pdf
3. Milgram S.: The small world problem. Psychol. Today, 2:60-67, (1967)
4. Barabási A.-L., Albert R., Jeong H., and Bianconi G.: Power-Law Distribution of the World Wide Web. In Science, 287:2115a, (2000), http://www.nd.edu/~networks/comments.pdf
5. Manning C.D., Schütze H.: Foundations of Statistical Natural Language Processing MIT Press, (2002)
6. Fellbaum C. (ed.): WordNet, an Electronic Lexical Database, MIT Press, (1998)
7. Gaume B., Balades Aléatoires dans les Petits Mondes Lexicaux, In *I3 Information Interaction Intelligence* vol.4 - n°2 - 2004, CEPADUES édition http://www.revue-i3.org/volume04/numero02/index.htm
8. Bermann A., Plemons R.J.: Nonnegative Matrices in the Mathematical Sciences. Classics in applied Mathematics, (1994)
9. Duvignau K.: La métaphore berceau et enfant de la langue. Phd These, Toulouse II, (2002)
10. Duvignau K., Gaume B.: Linguistic, Psycholinguistic and Computational Approaches to the Lexicon. Cognitive Systems, March, 6-2 (3): 255-269, (2004), www.esscs.org
11. Kleinberg : The Small-World Phenomenon: An Algorithmic Perspective. In *Proceeding of 32nd ACM Symposium on Theory of Computing*, 2000

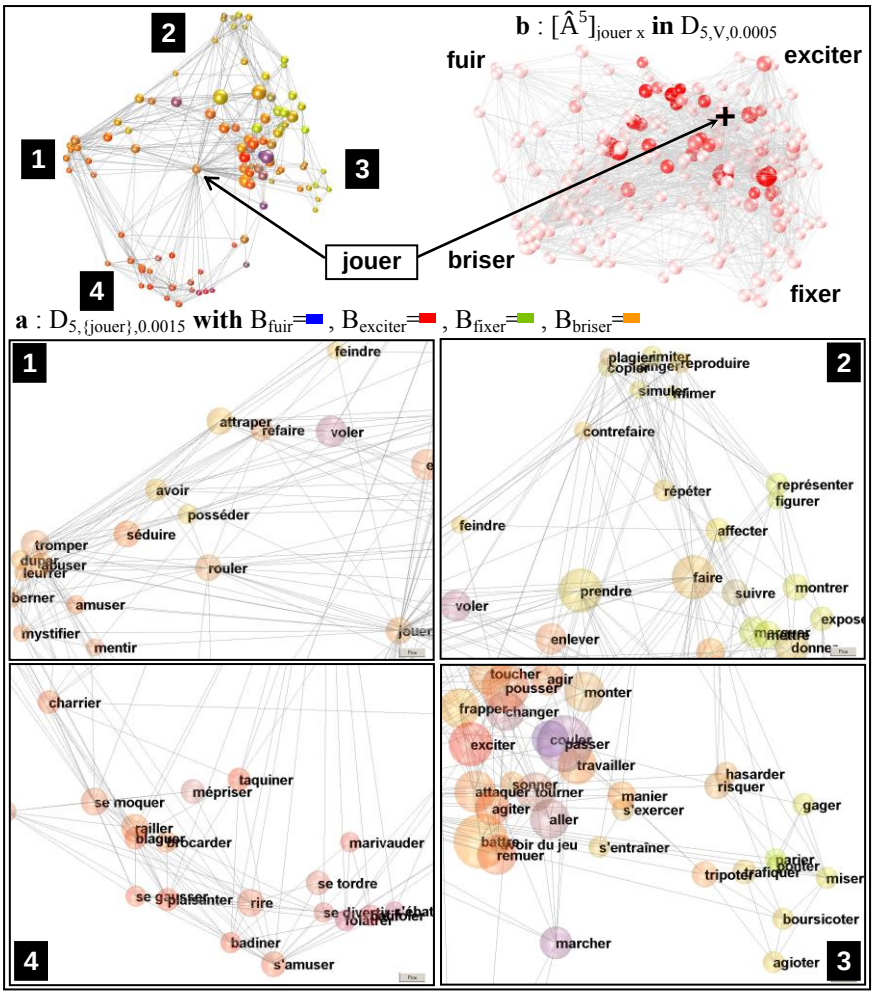

Fig. 8. $D_{5,\{jouer\},0.0015}$ in DicoSyn.Verbe, the conceptual form of "jouer", $R_{5,\{jouer\},0.0015}$=100

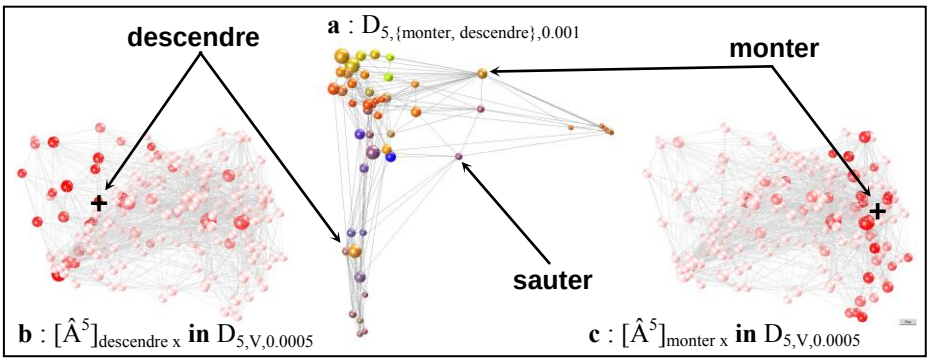

Fig. 9. $D_{5,\{monter,descendre\},0.001}$: Topological zoom on {monter,descendre}, $R_{5,\{monter,descendre\},0.001}$=50

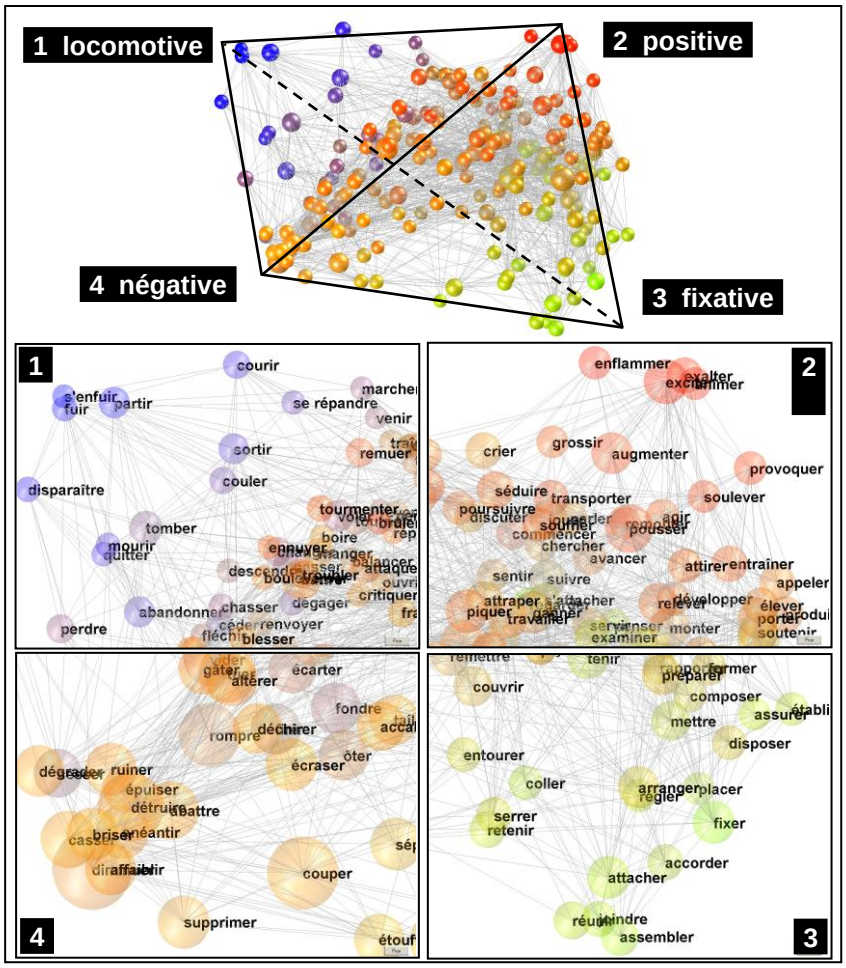

Fig. 10. The tretrahedron of DicoSyn.Verbe (9043 French verbs): $D_{5,V,0.0005}$, $R_{5,V,0.0005}$=200

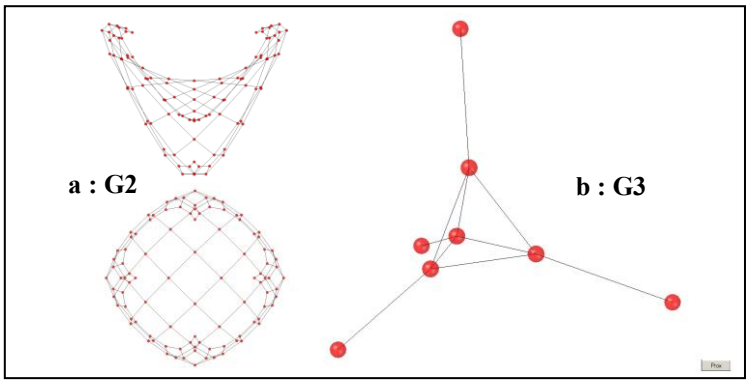

Fig. 11. Global display of two laboratory graphs (at t=5)