Multimedia Tools and Applications
https://doi.org/10.1007/s11042-020-09176-y

Exchange-based diffusion in Hb-Graphs

Highlighting complex relationships in multimedia collections (extended version)

Xavier Ouvrard[1,2] · Jean-Marie Le Goff[1] · Stéphane Marchand-Maillet[2]

Received: 15 March 2019 / Revised: 24 May 2020 / Accepted: 4 June 2020 /
Published online: 18 July 2020

Abstract
Highlighting important information of a network is commonly achieved by using random walks related to diffusion over such structures. Complex networks, where entities can have multiple relationships, call for a modeling based on hypergraphs. But, the limitation of hypergraphs to binary entities in co-occurrences has led us to introduce a new mathematical structure called hyperbaggraphs, that relies on multisets. This is not only a shift in the designation but a real change of mathematical structure, with a new underlying algebra. Diffusion processes commonly start with a stroke at one vertex and diffuse over the network. In the original conference article—(Ouvrard et al. 2018)—that this article extends we have proposed a two-phase step exchange-based diffusion scheme, in the continuum of spectral network analysis approaches, that takes into account the multiplicities of entities. This diffusion scheme allows to highlight information not only at the level of the vertices but also at the regrouping level. In this paper, we present new contributions: the proofs of conservation and convergence of the extracted sequences of the diffusion process, as well as the illustration of the speed of convergence and comparison between classical and modified random walks; the algorithms of the exchange-based diffusion and the modified random walk; the application to two use cases, one based on Arxiv publications and another based on Coco dataset images. All the figures have been revisited in this extended version to take the new developments into account.

Keywords Exchange · Diffusion · Multiset · Hyper-bag-graph · Information retrieval · Ranking

1 Introduction

Multimedia collections, and among them text collections, often require to model complex relationships, that are potentially multi-modal, based either on natural relations or on

✉ Xavier Ouvrard
xavier.ouvrard@cern.ch

[1] CUI, University of Geneva, Battelle (Bat A), Route de Drize, 7, CH-1227, Carouge, Switzerland

[2] CERN, 1 Esplanade des Particules, CH-1211, Geneva, Switzerland

similarity-based groupings. Connections between entities call for a modeling of potentially complex multi-adic relations, that most of the time cannot be reduced to pairwise relationships. Hypergraphs where relations are based on subsets of a given set enhance the modeling of multi-adic relations. A multi-adic relationship can be viewed either as a group or a co-occurrence. Also, hypergraphs can be used to model the complex co-occurrence networks induced by the different facets of an information space—[25].

Nonetheless, in real co-occurrence networks, some elements might repeat themselves more than once, or require an individual weighting at the group level. Sets fail at capturing this information, while multisets, where elements have a multiplicity, naturally handle it. Moving from sets to multisets is not only a change in designation, but an effective mathematical paradigm shift, as the algebra involved behind is not the same. Collections of multisets are then the next step in the modeling. In [26, 27], we have introduced hyper-bag-graphs (hb-graphs for short) as a generalization of hypergraphs to support multisets. A hb-graph is a family of multisets over a same universe, designated as the vertex set. The multisets play the role of hyperedges in hypergraphs and are called hb-edges. Hypergraphs appear as a sub-category of this new mathematical category that hb-graphs constitute.

The step forward consists in highlighting important information conveyed by these complex networks. Traditional approaches in hypergraphs includes random walks—[1, 44]. Particularly, in [1], the authors show that putting hyperedge-based weights on vertices provides a better information retrieval. In these two approaches—[1, 44]—, the focus is mainly put on vertices; but, most of the time, in such modeling, the information carried by the links is semantically significant, as it represents the reference used for building the co-occurrences. Also, having a way to highlight important information from the reference is also interesting. Multimedia databases, such as image databases or document databases are potential applications of such means of highlighting information. For instance, different metadata can be attached to a document such as authors, author keywords, processed keywords, categories, added tags. If the users are able to attach tags to documents, it can be important to weight them individually in the context of each document. The same can apply to an image, with other features that are based on the image analysis. Hb-graphs fit to model such information spaces—[29].

We want to address the following research question: “Can we find a network model and a diffusion process that not only rank vertices but also rank hb-edges in hb-graphs?”. In [24], we have developed an iterative exchange approach in hb-graphs with two-phase steps that allows to extract information not only at the vertex level but also at the hb-edge level. In this article, which is an extended version of [24], we not only present the contributions of [24]—that included the introduction of the exchange-based diffusion process as a means to rank both vertices and hb-edges, and were formalizing the exchanges by using hb-graphs, and presenting a novel visualisation of co-occurrences network—, but also add new contributions.

In [24], we have validated our approach by using lab-generated hb-graphs. We continue here to use this approach, as mimicking real datasets by randomly generated ones is not only a warranty for reproducibility, but also for robustness of the results obtained. We illustrate the extracted information using the exchange process by a hb-graph visualisation that highlights not only vertices but also hb-edges.

We show that the exchange-based diffusion process provides proper coloring of vertices with high connectivity and highlights hb-edges with a normalisation approach—allowing small hb-edges to have a chance to be highlighted. We apply this approach to process the metadata contained in the results retrieved by querying Arxiv through its API in order to visualize the results: we will show how it can be used to allow further query expansion. We give a last use case on Coco dataset images.

In summary, the contributions of this extended version include: the proofs of conservation and convergence of the extracted sequences of the diffusion process, as well as the illustration of the speed of convergence and comparison to classical and modified random walks; the algorithms of the exchange-based diffusion and the modified random walk; the application to two use cases, one based on Arxiv publications and another one based on images of the Coco dataset.

In Section 2, the mathematical background and the related work is given. The construction of the formalisation of the exchange process is presented in Section 3. Results and evaluation are given in Section 4 and future work and conclusion are addressed in Section 5.

2 Mathematical background and related work

For the mathematical background, we give a minimal mathematical formalization. However, the interested reader can refer to [28] which contains all the necessary mathematics and to [26, 27] for a full introduction on hb-graphs.

2.1 Hypergraphs

Hypergraphs have been introduced in [3]; we use nonetheless the definition given in [4], as it relaxes the constraint on the hyperedges to cover the vertex set. A **hypergraph** over a finite set of vertices is defined as a family of subsets of this vertex set. A hypergraph will be said edge-weighted if there exists an application that associates a positive real number to each hyperedge.

Hypergraphs fit to model multi-adicity in structures where the traditional pairwise relationship of graphs is insufficient: they are used in many areas such as social networks in particular in collaboration networks—[22, 23]—, co-author networks—[13] and [37]—, chemical reactions—[38]—, genome—[6]—, VLSI design—[15]—and other applications. Hypergraphs are also used in information retrieval for different purposes such as query formulation in text retrieval [2] and in music recommendation [5]. Several applications of hypergraphs exist based on the diffusion process firstly developed in [44]. In [11], the authors use the diffusion process developed in [44] for 3D-object retrieval and recognition by building multiple hypergraphs of objects based on their 2D-views. In [43], multiple hypergraphs are constructed to characterize the complex relations between landmark images and are gathered into a multi-modal hypergraph that allows the integration of heterogeneous sources to provide content-based visual landmark searches. Hypergraphs are also used in multi-feature indexing to help image retrieval [41]. For each image, a hyperedge gathers the most similar images based on different features. Hyperedges are weighted by average similarity. A spectral clustering algorithm is then applied to divide the dataset into a given number of sub-hypergraphs. A random walk on these sub-hypergraphs retrieves significant images: they are used to build a new inverted index, useful to query images. In [40], a joint-hypergraph learning is achieved for image retrieval, combining efficiently a semantic hypergraph based on image tags with a visual hypergraph based on image features.

Evaluating the importance of vertices in hypergraphs by random walks has been largely studied. In [44], a random walk on a edge-weighted hypergraph is defined by choosing a hyperedge with a probability proportional to its weight and, within that hyperedge, a vertex randomly chosen using a uniform law. This random walk has a stationary state which is shown to correspond to a vector proportional to the vector of vertex degrees in [10]. This process differs from the one we propose: our diffusion process is done in successive steps from a random initial vertex on vertices and hb-edges, taking into account the multiplicities of the vertices inside each hb-edge.

In [1], the authors use a random walk on hypergraphs using weight functions both on hyperedges and vertices. The vertex weights are hyperedge-based: it is achieved using a vector of weights associated to each vertex. The random walk is similar to the one in [44], but additionally takes into account the vertex weight in the probability law for choosing the vertex inside the hyperedge. They show on a publication dataset that this modified random walk gives a ranking of vertices with higher precision than random walks using unweighted vertices. However, this process differs again from our proposal since our process not only enables simultaneous alternative updates of vertices and hb-edges values but also provides hb-edge ranking. We also introduce a new theoretical framework to perform our diffusion process.

Diffusion processes are tightly tied to random walks. In [17], the authors use random walks in hypergraph for image matching. In [19], the authors build higher order random walks in hypergraph and construct a generalised Laplacian attached to the graphs generated from their random walks.

2.2 Multisets

Multisets—also known as bags or msets—have a long use in many domains. But before developing their use in different domains, we firstly give the main definitions on multisets mainly based on [35].

A **multiset** $\mathfrak{A}_m = (A, m)$[1] is a pair composed of a set A of distinct objects—called the **universe** of the multiset—and, of a **multiplicity function** m with a range potentially in the real numbers set. The support $\mathfrak{A}_m^\star$ of the multiset $\mathfrak{A}_m$ corresponds to the elements of the universe that have a non-zero multiplicity. When the range of the multiset is a subset of the non-negative integers, we call it a **natural multiset**. A natural multiset can be viewed as an unordered list of elements with possible repetitions.

The **m-cardinality** of a multiset $\mathfrak{A}_m$, written $\#_m\mathfrak{A}_m$, corresponds to the sum of the multiplicities of the elements of its universe.

Different operations can be defined on multisets of same universe as inclusion, union, intersection and sum. As mentioned in [35], De Morgan's laws on multisets do not hold. Defining complementation and difference requires to fix a limit in m-cardinality to the multisets as given in [12].

Multisets, under the appellation bag, appear in different domains such as text modeling, image description and audio [32]. In text representation, bag of words have been first introduced in [14]: bags are lists of words with repetitions, i.e. multisets of words on a universe. Many applications occur with different approaches. Bags of words have been used for instance in fraud detection [31]. More recently, bags of words have been used successfully for translation by neural nets as a target for the translation as a sentence can be translated in many different ways [20]. In [8], multi-modal bag of words have been used for cross domains sentiment analysis.

Bags of visual words is the transcription to image of textual bags of words; in bags of visual words, a visual vocabulary based on image features is built to allow the description

[1] We systematically use fraktur font for multisets in order to make a clear distinction with sets.

of images as bags of these features. Since their introduction in [36], many applications have been achieved: in visual categorization [7], in image classification and filtering [9], in image annotation [39], in action recognition [30], in land-use scene classification [42], in identifying mild traumatic brain injuries [21] and in word image retrieval [33].

Bags of concepts are an extension of bags of words to successive concepts in a text [16]. A recent extension of these concepts is given in [34] where bag of graphs are introduced to encode in graphs the local structure of a digital object: bags of graphs are declined into bags of singleton graphs and bags of visual graphs. Using the hb-graphs as we propose in this article will allow to extend this approach, by taking advantage of multi-adicity and also of the multiplicity of vertices specific to each hb-edge.

2.3 Hb-graphs

Hb-graphs are introduced in [26]. A **hb-graph** is a family of multisets with the same universe V and with support a subset of V. The msets are called the **hb-edges** and the elements of V the **vertices**. We consider for the remainder of the article a hb-graph $\mathfrak{H} = (V, \mathfrak{E})$, with $V = \left\{v_i : i \in [\![n]\!]\right\}$[2] and $\mathfrak{E} = \left(\mathfrak{e}_j\right)_{j\in[\![p]\!]}$ the family of its hb-edges.

Each hb-edge $\mathfrak{e}_i \in \mathfrak{E}$ has V as universe and a multiplicity function associated to it: $m_{\mathfrak{e}_i} : V \to \mathbb{W}$ where $\mathbb{W} \subset \mathbb{R}^+$. For a general hb-graph, each hb-edge has to be seen as a weighted system of vertices, where the weights of each vertex are hb-edge dependent.

A hb-graph where the multiplicity range of each hb-edge is a subset of the non-negative integer set is called a **natural hb-graph**. A **hypergraph** is a natural hb-graph where the hb-edges have multiplicity one for every vertex of their support.

The **support hypergraph** of a hb-graph $\mathfrak{H} = (V, \mathfrak{E})$ is the hypergraph whose vertices are the ones of the hb-graph and whose hyperedges are the support of the hb-edges in a one-to-one way. We write it $\underline{\mathfrak{H}} = \left(V, \underline{\mathfrak{E}}\right)$, where $\underline{\mathfrak{E}} = (\mathfrak{e}^\star)_{\mathfrak{e}\in\mathfrak{E}}$.

The **m-degree of a vertex** $v_i \in V$ of a hb-graph $\mathfrak{H}$—written $\deg_m(v_i) = d_m(v_i)$—is defined as the sum of the multiplicity of v_i in each hb-edge of the hb-graph.

The matrix $H = \left[m_j(v_i)\right]_{\substack{i\in[\![n]\!]\\ j\in[\![p]\!]}}$ is called the **incident matrix** of the hb-graph $\mathfrak{H}$.

A **weighted hb-graph** $\mathfrak{H}_w = (V, \mathfrak{E}, w_e)$ is a hb-graph $\mathfrak{H} = (V, \mathfrak{E})$ where the hb-edges are weighted by $w_e : \mathfrak{E} \to \mathbb{R}^{+*}$. An unweighted hb-graph is then a weighted hb-graph with $w_e\left(\mathfrak{e}_j\right) = 1$ for all $\mathfrak{e}_j \in \mathfrak{E}$.

A **strict m-path** $u\mathfrak{e}_{j_1}v_{i_1}\ldots\mathfrak{e}_{j_s}v$ in a hb-graph $\mathfrak{H}$ from a vertex u to a vertex v is a vertex / hb-edge alternation, where the intermediate vertices belong to the intersection of the hb-edges immediately surrounding them. In a natural hb-graph, a strict m-path is not unique as many copies of the same vertex can coexist in the intersection. Moreover, in natural hb-graphs, there are two notions of paths: a strict and a large one: some copies of the vertex are possibly not in the intersection of the two surrounding hb-edges and can exist only in one of the two hb-edges.

A strict m-path in a hb-graph corresponds to a unique path in the hb-graph support hypergraph called the **support path**. In this article we abusively call it a path of the hb-graph. The **length of a path** corresponds to the number of hb-edges it is going through.

Representations of hb-graphs can be achieved either by using sub-mset representations or by using edge representations. In the edge representation, an extra-node is added to each hb-edge and the thickness of the link between the extra-node of a hb-edge and the vertices

[2] We note $[\![n]\!] = \{i : 1 \leqslant i \leqslant n \wedge i \in \mathbb{N}\}$.

in the support of the hb-edge is made proportional to their multiplicity in the hb-edge. More details on these representations can be found in [26].

We give in Fig. 1 an example of such representation of a hb-graph for keywords extracted from sentences in which stop words have been removed. The number of words occurrences differs from one sentence to another: it is given as a multiplicity specific to the corresponding hb-edge that represents the sentence. The universe of the hb-graph is the set of words where the stop words has been removed.

3 Exchange-based diffusion in hb-graphs

We introduce in this section a diffusion process based on the exchange of information between the vertices and the hb-edges. Traditionally, diffusion processes are achieved using an initial stroke on a vertex that propagates over the network structure. Diffusion processes can be approximated using random walks. When the random walk takes place on a network, either graph or hypergraph based, vertices can be ranked by using the number of times they are reached. Teleportation is introduced in these random walks to avoid loops. Several random walks are often necessary in order to average their results.

The idea in the exchange-based diffusion is to propose a mechanism that mimics the behavior of a population where agents—vertices—have equal resources at the beginning and can exchange them only via intermediates—hb-edges—they are belonging to and share the resources according to the multiplicities of these agents.

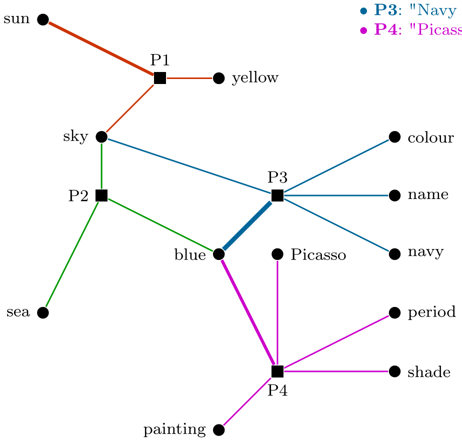
We consider a weighted hb-graph $\mathfrak{H} = (V, \mathfrak{E}, w_e)$ with $|V| = n$ and $|\mathfrak{E}| = p$; we write H its incidence matrix.

At time t, we set a distribution of values over the vertex set:

$$\alpha_t : \begin{cases} V \to [0; 1] \\ v_i \mapsto \alpha_t(v_i) \end{cases} .$$

Four sentences:

- **P1**: "The sun is in the sky and the sun is yellow."
- **P2**: "The sea is blue and the sky is also blue."
- **P3**: "Navy blue and sky blue are blue colour names."
- **P4**: "Picasso had a blue period where his paintings were in blue shade."



	P1	P2	P3	P4
sun	2	0	0	0
sky	1	1	1	0
yellow	1	0	0	0
sea	0	1	0	0
blue	0	1	3	2
colour	0	0	1	0
navy	0	0	1	0
name	0	0	1	0
painting	0	0	0	1
Picasso	0	0	0	1
period	0	0	0	1
shade	0	0	0	1

Fig. 1 An example of hb-graphs: four sentences and their associated bag of words with removed stop words and the incidence matrix of the hb-graph

and a distribution of values over the hb-edge set:

$$\epsilon_t : \begin{cases} \mathfrak{E} \to [0; 1] \\ \mathfrak{e}_j \mapsto \epsilon_t\left(\mathfrak{e}_j\right) \end{cases}.$$

We write $P_{V,t} = (\alpha_t(v_i))_{i \in [\![n]\!]}$ the row state vector of the vertices at time t and $P_{\mathfrak{E},t} = \left(\epsilon_t\left(\mathfrak{e}_j\right)\right)_{j \in [\![p]\!]}$ the row state vector of the hb-edges.

The initialisation is done such that $\sum\limits_{v_i \in V} \alpha_0(v_i) = 1$ and the information value is concentrated uniformly on the vertices at the beginning of the diffusion process and, consequently, each hb-edge has a zero value associated to it. Writing $\alpha_{\text{ref}} = \dfrac{1}{|V|}$, we set for all $v_i \in V$: $\alpha_0(v_i) = \alpha_{\text{ref}}$ and for all $\mathfrak{e}_j \in \mathfrak{E}$, $\epsilon\left(\mathfrak{e}_j\right) = 0$.

We consider an iterative process with two-phase steps. At every time step, the first phase starts at time t and ends at $t + \dfrac{1}{2}$, followed by the second phase between time $t + \dfrac{1}{2}$ and $t + 1$. In Fig. 2, we illustrate this principle using the example in Fig. 1. A more general figure of the principle of this iterative process is given in [24, 28]. The iterative process conserves the overall value held by the vertices and the hb-edges, meaning that we have at any $t \in \left\{\dfrac{1}{2}k : k \in \mathbb{N}\right\}$:

$$\sum_{v_i \in V} \alpha_t(v_i) + \sum_{\mathfrak{e}_j \in \mathfrak{E}} \epsilon_t\left(\mathfrak{e}_j\right) = 1.$$

During the first phase between time t and $t + \dfrac{1}{2}$, each vertex v_i of the hb-graph shares its value $\alpha_t(v_i)$ hold at time t with the hb-edges it is connected to.

In an unweighted hb-graph, the fraction of $\alpha_t(v_i)$ given by v_i of m-degree $d_{v_i} = \deg_m(v_i)$ to each hb-edge is $\dfrac{m_j(v_i)}{\deg_m(v_i)}$, which corresponds to the ratio of the multiplicity of the vertex v_i due to the hb-edge $\mathfrak{e}_j$ over the total m-degree of hb-edges containing v_i in their support.

In a weighted hb-graph, each hb-edge has a weight $w_e\left(\mathfrak{e}_j\right)$. The value $\alpha_t(v_i)$ of the vertex v_i is shared by accounting not only the multiplicity of the vertices in the hb-edge but also the weight $w_e\left(\mathfrak{e}_j\right)$ of the hb-edge $\mathfrak{e}_j$.

The weights of the hb-edges are stored in a column vector:

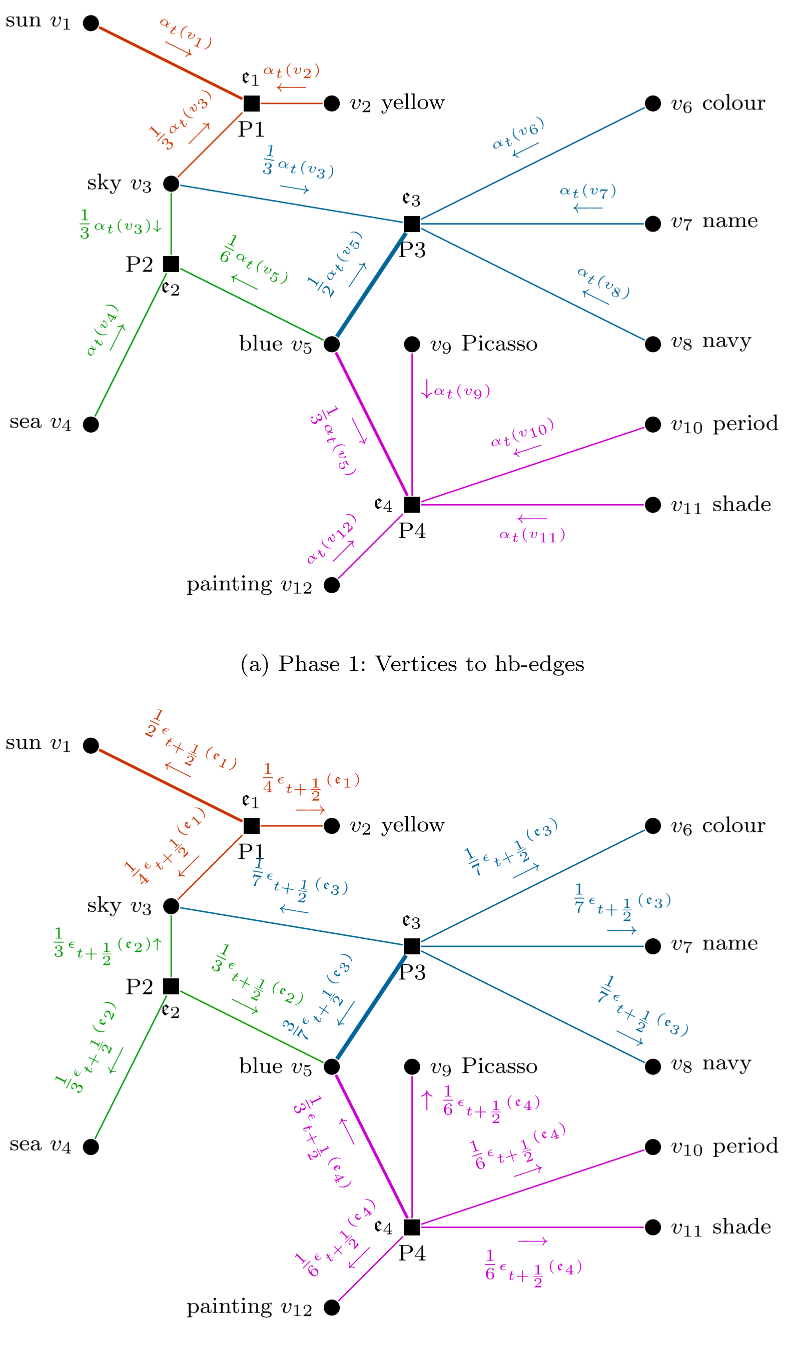
$$w_{\mathfrak{E}} = \left(w_e\left(\mathfrak{e}_j\right)\right)^{\top}_{j \in [\![p]\!]}.$$

We also consider the weight diagonal matrix:

$$W_{\mathfrak{E}} = \operatorname{diag}\left(\left(w_e\left(\mathfrak{e}_j\right)\right)_{j \in [\![p]\!]}\right).$$

We introduce the weighted m-degree matrix:

$$D_{w,V} = \operatorname{diag}\left(\left(d_{w,v_i}\right)_{i \in [\![n]\!]}\right) = \operatorname{diag}\left(H w_{\mathfrak{E}}\right).$$


sun v_1
$\mathfrak{e}_1$
P1
v_2 yellow
sky v_3
P2
$\mathfrak{e}_2$
sea v_4
blue v_5
$\mathfrak{e}_3$
P3
v_6 colour
v_7 name
v_8 navy
v_9 Picasso
v_{10} period
v_{11} shade
$\mathfrak{e}_4$
P4
painting v_{12}
$\alpha_t(v_1)$
$\alpha_t(v_2)$
$\frac{1}{3}\alpha_t(v_3)$
$\frac{1}{3}\alpha_t(v_3)$
$\frac{1}{3}\alpha_t(v_3)$
$\frac{1}{6}\alpha_t(v_5)$
$\frac{1}{2}\alpha_t(v_5)$
$\frac{1}{3}\alpha_t(v_5)$
$\alpha_t(v_4)$
$\alpha_t(v_6)$
$\alpha_t(v_7)$
$\alpha_t(v_8)$
$\alpha_t(v_9)$
$\alpha_t(v_{10})$
$\alpha_t(v_{11})$
$\alpha_t(v_{12})$

(a) Phase 1: Vertices to hb-edges

sun v_1
$\mathfrak{e}_1$
P1
v_2 yellow
sky v_3
P2
$\mathfrak{e}_2$
sea v_4
blue v_5
$\mathfrak{e}_3$
P3
v_6 colour
v_7 name
v_8 navy
v_9 Picasso
v_{10} period
v_{11} shade
$\mathfrak{e}_4$
P4
painting v_{12}
$\frac{1}{2}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_1)$
$\frac{1}{4}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_1)$
$\frac{1}{4}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_1)$
$\frac{1}{3}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_2)$
$\frac{1}{3}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_2)$
$\frac{1}{3}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_2)$
$\frac{1}{7}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_3)$
$\frac{1}{7}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_3)$
$\frac{1}{7}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_3)$
$\frac{1}{7}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_3)$
$\frac{3}{7}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_3)$
$\frac{1}{3}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_4)$
$\frac{1}{6}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_4)$
$\frac{1}{6}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_4)$
$\frac{1}{6}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_4)$
$\frac{1}{6}\epsilon_{t+\frac{1}{2}}(\mathfrak{e}_4)$

(b) Phase 2: Hb-edges to vertices

Fig. 2 Diffusion by exchange: principle of the two phases on the example of Fig. 1: phase 1 occurs from t to $t+\frac{1}{2}$ and phase 2 occurs from $t+\frac{1}{2}$ to $t+1$

where d_{w,v_i} is called the weighted m-degree of the vertex v_i. It is:

$$d_{w,v_i} = \deg_{w,m}(v_i) = \sum_{j \in [\![p]\!]} m_j(v_i)\, w_e\left(\mathfrak{e}_j\right).$$

The contribution of the vertex v_i to the value $\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right)$ attached to the hb-edge $\mathfrak{e}_j$ of weight $w_e\left(\mathfrak{e}_j\right)$ is:

$$\delta\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j \mid v_i\right) = \frac{m_j(v_i)\, w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} \alpha_t(v_i).$$

It corresponds to the ratio of the weighted multiplicity of the vertex v_i in $\mathfrak{e}_j$ over the total weighted m-degree of the hb-edges where v_i is in the support.

We remark that if $v_i \notin \mathfrak{e}_j^\star$:

$$\delta\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j \mid v_i\right) = 0.$$

And the value $\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right)$ is calculated by summing over the vertex set:

$$\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) = \sum_{i \in [\![n]\!]} \delta\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j \mid v_i\right).$$

Hence, we obtain:

$$P_{\mathfrak{E},t+\frac{1}{2}} = P_{V,t} D_{w,V}^{-1} H W_{\mathfrak{E}}. \tag{1}$$

The value given to the hb-edges is subtracted to the value of the corresponding vertex, hence for all $i \in [\![n]\!]$:

$$\alpha_{t+\frac{1}{2}}(v_i) = \alpha_t(v_i) - \sum_{j \in [\![p]\!]} \delta\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j \mid v_i\right).$$

Claim (No information on vertices at $t + \dfrac{1}{2}$) It holds:

$$\forall i \in [\![n]\!] : \alpha_{t+\frac{1}{2}}(v_i) = 0.$$

Proof For all $i \in [\![n]\!]$:

$$\begin{aligned}
\alpha_{t+\frac{1}{2}}(v_i) &= \alpha_t(v_i) - \sum_{j \in [\![p]\!]} \delta\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j \mid v_i\right) \\
&= \alpha_t(v_i) - \sum_{j \in [\![p]\!]} \frac{m_j(v_i)\, w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} \alpha_t(v_i) \\
&= \alpha_t(v_i) - \alpha_t(v_i) \frac{\sum\limits_{j \in [\![p]\!]} m_j(v_i)\, w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} \\
&= 0.
\end{aligned}$$

□

Claim (Conservation of the information of the hb-graph at $t + \dfrac{1}{2}$) It holds:

$$\sum_{v_i \in V} \alpha_{t+\frac{1}{2}}(v_i) + \sum_{\mathfrak{e} \in \mathfrak{E}} \epsilon_{t+\frac{1}{2}}(\mathfrak{e}) = 1.$$

Proof We have:

$$\begin{aligned}\sum_{v_i \in V} \alpha_{t+\frac{1}{2}}(v_i) + \sum_{\mathfrak{e} \in \mathfrak{E}} \epsilon_{t+\frac{1}{2}}(\mathfrak{e}) &= \sum_{\mathfrak{e}_j \in \mathfrak{E}} \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) \\ &= \sum_{\mathfrak{e}_j \in \mathfrak{E}} \sum_{i \in [\![n]\!]} \delta\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j \mid v_i\right) \\ &= \sum_{\mathfrak{e}_j \in \mathfrak{E}} \sum_{i \in [\![n]\!]} \frac{m_j(v_i)\, w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} \alpha_t(v_i) \\ &= \sum_{i \in [\![n]\!]} \alpha_t(v_i) \frac{\sum\limits_{\mathfrak{e}_j \in \mathfrak{E}} m_j(v_i)\, w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} \\ &= \sum_{i \in [\![n]\!]} \alpha_t(v_i) \\ &= 1.\end{aligned}$$

□

During the second phase which starts at time $t+\frac{1}{2}$, the hb-edges share their values across the vertices they hold taking into account the vertex multiplicities within the hb-edge.

The contribution to $\alpha_{t+1}(v_i)$ given by a hb-edge $\mathfrak{e}_j$ is proportional to $\epsilon_{t+\frac{1}{2}}$ in a factor corresponding to the ratio of the multiplicity $m_j(v_i)$ of the vertex v_i to the hb-edge m-cardinality:

$$\delta\alpha_{t+1}\left(v_i \mid \mathfrak{e}_j\right) = \frac{m_j(v_i)}{\#_m \mathfrak{e}_j} \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right).$$

The value $\alpha_{t+1}(v_i)$ is then obtained by summing on all values associated to the hb-edges that are incident to v_i :

$$\alpha_{t+1}(v_i) = \sum_{j \in [\![p]\!]} \delta\alpha_{t+1}\left(v_i \mid \mathfrak{e}_j\right).$$

Writing $D_{\mathfrak{E}} = \operatorname{diag}\left(\left(\#_m \mathfrak{e}_j\right)_{j \in [\![p]\!]}\right)$ the diagonal matrix of size $p \times p$, it comes:

$$P_{\mathfrak{E},t+\frac{1}{2}} D_{\mathfrak{E}}^{-1} H^{\top} = P_{V,t+1}. \tag{2}$$

The values given to the vertices are subtracted to the value associated to the corresponding hb-edge. Hence, for all $j \in [\![p]\!]$:

$$\epsilon_{t+1}\left(\mathfrak{e}_j\right) = \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) - \sum_{i \in [\![n]\!]} \delta\alpha_{t+1}\left(v_i \mid \mathfrak{e}_j\right).$$

Claim (The hb-edges have 0 value at $t+1$) It holds:

$$\epsilon_{t+1}\left(\mathfrak{e}_j\right) = 0.$$

Proof For all $i \in \llbracket p \rrbracket$:

$$\begin{aligned}
\epsilon_{t+1}\left(\mathfrak{e}_j\right) &= \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) - \sum_{i \in \llbracket n \rrbracket} \delta\alpha_{t+1}\left(v_i \mid \mathfrak{e}_j\right) \\
&= \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) - \sum_{i \in \llbracket n \rrbracket} \frac{m_j\left(v_i\right)}{\#_m \mathfrak{e}_j} \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) \\
&= \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) \left(1 - \frac{\sum\limits_{i \in \llbracket n \rrbracket} m_j\left(v_i\right)}{\#_m \mathfrak{e}_j}\right) \\
&= 0.
\end{aligned}$$

□

Claim (Conservation of the information of the hb-graph at $t+1$) It holds:

$$\sum_{v_i \in V} \alpha_{t+1}\left(v_i\right) + \sum_{\mathfrak{e}_j \in \mathfrak{E}} \epsilon_{t+1}\left(\mathfrak{e}_j\right) = 1.$$

Proof

$$\begin{aligned}
\sum_{v_i \in V} \alpha_{t+1}\left(v_i\right) + \sum_{\mathfrak{e} \in \mathfrak{E}} \epsilon_{t+1}\left(\mathfrak{e}\right) &= \sum_{v_i \in V} \alpha_{t+1}\left(v_i\right) \\
&= \sum_{v_i \in V} \sum_{j \in \llbracket p \rrbracket} \delta\alpha_{t+1}\left(v_i \mid \mathfrak{e}_j\right) \\
&= \sum_{v_i \in V} \sum_{j \in \llbracket p \rrbracket} \frac{m_j\left(v_i\right)}{\#_m \mathfrak{e}_j} \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) \\
&= \sum_{j \in \llbracket p \rrbracket} \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) \frac{\sum\limits_{v_i \in V} m_j\left(v_i\right)}{\#_m \mathfrak{e}_j} \\
&= \sum_{j \in \llbracket p \rrbracket} \epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right) \\
&= 1.
\end{aligned}$$

□

Regrouping (1) and (2):

$$P_{V,t+1} = P_{V,t} D_{w,V}^{-1} H W_{\mathfrak{E}} D_{\mathfrak{E}}^{-1} H^{\top}. \tag{3}$$

It is valuable to keep a trace of the intermediate state $P_{\mathfrak{E},t+\frac{1}{2}} = P_{V,t} D_{w,V}^{-1} H W_{\mathfrak{E}}$ as it records the importance of the hb-edges.

Writing $T = D_{w,V}^{-1} H W_{\mathfrak{E}} D_{\mathfrak{E}}^{-1} H^{\top}$, it follows from (3):

$$P_{V,t+1} = P_{V,t} T. \tag{4}$$

Claim (Stochastic transition matrix) T is a square row stochastic matrix of dimension n.

Proof Let consider:

$$A = \left(a_{ij}\right)_{i \in \llbracket n \rrbracket j \in \llbracket p \rrbracket} = D_{w,V}^{-1} H W_{\mathfrak{E}} \in M_{n,p}$$

and:

$$B = \left(b_{jk}\right)_{j \in \llbracket p \rrbracket k \in \llbracket n \rrbracket} = D_{\mathfrak{E}}^{-1} H^{\top} \in M_{p,n}.$$

A and B are non-negative rectangular matrices. Moreover:

$$a_{ij} = \frac{m_j\left(v_i\right) w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}}$$

and, it holds:

$$\sum_{j \in \llbracket p \rrbracket} a_{ij} = \frac{\sum\limits_{j \in \llbracket p \rrbracket} m_j\left(v_i\right) w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} = 1.$$

$b_{jk} = \dfrac{m_j\left(v_k\right)}{\#_m\left(\mathfrak{e}_j\right)}$ and it holds:

$$\sum_{k \in \llbracket n \rrbracket} b_{jk} = \frac{\sum\limits_{k \in \llbracket n \rrbracket} m_j\left(v_k\right)}{\#_m \mathfrak{e}_j} = 1.$$

We have: $P_{V,t+1} = P_{V,t} AB$ where:

$$T = AB = \left(\sum_{j \in \llbracket p \rrbracket} a_{ij} b_{jk} \right)_{i \in \llbracket n \rrbracket k \in \llbracket n \rrbracket}.$$

It yields:

$$\begin{aligned} \sum_{k \in \llbracket n \rrbracket} \sum_{j \in \llbracket p \rrbracket} a_{ij} b_{jk} &= \sum_{j \in \llbracket p \rrbracket} a_{ij} \sum_{k \in \llbracket n \rrbracket} b_{jk} \\ &= \sum_{j \in \llbracket p \rrbracket} a_{ij} \\ &= 1. \end{aligned}$$

Hence $T = AB$ is a non-negative square matrix with its row sums all equal to 1: it is a row stochastic matrix. □

Claim (Properties of T) Supposing that the hb-graph is connected, the exchange-based diffusion matrix T is aperiodic and irreducible.

Proof This stochastic matrix is aperiodic, due to the fact that any vertex of the hb-graph retrieves a part of the value it has given to the hb-edge, hence $t_{ii} > 0$ for all $i \in \llbracket n \rrbracket$.

Moreover, as the hb-graph is connected, the matrix is irreducible as all states can be joined from any state. □

Claim The sequence $\left(P_{V,t}\right)_{t\in\mathbb{N}}$, with $P_{V,t} = \left(\alpha_t\left(v_i\right)\right)_{i\in\llbracket n\rrbracket}$, in a connected hb-graph converges to the state vector π_V such that:

$$\pi_V = \left(\frac{d_{w,v_i}}{\sum\limits_{k\in\llbracket n\rrbracket} d_{w,v_k}}\right)_{i\in\llbracket n\rrbracket}.$$

Proof We denote by π an eigenvector of $T = \left(c_{ik}\right)_{i\in\llbracket n\rrbracket k\in\llbracket n\rrbracket}$ associated to the eigenvalue 1. We have $\pi T = \pi$.

Let consider $u = \left(d_{w,v_i}\right)_{i\in\llbracket n\rrbracket}$.

We have:

$$\begin{aligned}
(uT)_k &= \sum_{i\in\llbracket n\rrbracket} d_{w,v_i} c_{ik}\\
&= \sum_{i\in\llbracket n\rrbracket} d_{w,v_i} \sum_{j\in\llbracket p\rrbracket} \frac{m_j\left(v_i\right) w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} \times \frac{m_j\left(v_k\right)}{\#_m\left(\mathfrak{e}_j\right)}\\
&= \sum_{j\in\llbracket p\rrbracket}\sum_{i\in\llbracket n\rrbracket} m_j\left(v_i\right) w_e\left(\mathfrak{e}_j\right) \times \frac{m_j\left(v_k\right)}{\#_m\left(\mathfrak{e}_j\right)}\\
&= \sum_{j\in\llbracket p\rrbracket} w_e\left(\mathfrak{e}_j\right) m_j\left(v_k\right) \frac{\sum\limits_{i\in\llbracket n\rrbracket} m_j\left(v_i\right)}{\#_m\left(\mathfrak{e}_j\right)}\\
&= \sum_{j\in\llbracket p\rrbracket} w_e\left(\mathfrak{e}_j\right) m_j\left(v_k\right)\\
&= d_{w,v_k} = u_k.
\end{aligned}$$

Hence, u is a non-negative eigenvector of T associated to the eigenvalue 1.

For a connected hb-graph, when we iterate over the stochastic matrix T which is aperiodic and irreducible, we are then ensured of convergence to a stationary state: this stationary state is the probability vector associated to the eigenvalue 1. It is unique and is equal to αu such that $\sum\limits_{k\in\llbracket n\rrbracket} \alpha u_k = 1$.

We have $\alpha = \frac{1}{\sum\limits_{k\in\llbracket n\rrbracket} d_{w,v_k}}$ and hence the result. □

Claim The sequence $\left(P_{\mathfrak{E},t+\frac{1}{2}}\right)_{t\in\mathbb{N}}$, with $P_{\mathfrak{E},t+\frac{1}{2}} = \left(\epsilon_{t+\frac{1}{2}}\left(\mathfrak{e}_j\right)\right)_{j\in\llbracket p\rrbracket}$, in a connected hb-graph converges to the state vector $\pi_{\mathfrak{E}}$ such that:

$$\left(\frac{w_e\left(\mathfrak{e}_j\right) \times \#_m\left(\mathfrak{e}_j\right)}{\sum\limits_{k\in\llbracket n\rrbracket} d_{w,v_k}}\right)_{j\in\llbracket p\rrbracket}.$$

Proof As $P_{\mathfrak{E},t+\frac{1}{2}} = P_{V,t} D_{w,V}^{-1} H W_{\mathfrak{E}}$ and that $\lim\limits_{t\to+\infty} P_{V,t} = \pi_V$, the sequence $\left(P_{\mathfrak{E},t+\frac{1}{2}}\right)_{t\in\mathbb{N}}$ converges towards a state vector $\pi_{\mathfrak{E}}$ such that: $\pi_{\mathfrak{E}} = \pi_V D_{w,V}^{-1} H W_{\mathfrak{E}}$.

We have:

$$\begin{aligned}
\pi_{\mathfrak{E}} &= \left(\sum_{i \in \llbracket n \rrbracket} \frac{d_{w,v_i}}{\sum\limits_{k \in \llbracket n \rrbracket} d_{w,v_k}} \times \frac{m_j\left(v_i\right) \times w_e\left(\mathfrak{e}_j\right)}{d_{w,v_i}} \right)_{j \in \llbracket p \rrbracket} \\
&= \left(\sum_{i \in \llbracket n \rrbracket} \frac{m_j\left(v_i\right) \times w_e\left(\mathfrak{e}_j\right)}{\sum\limits_{k \in \llbracket n \rrbracket} d_{w,v_k}} \right)_{j \in \llbracket p \rrbracket} \\
&= \left(\frac{w_e\left(\mathfrak{e}_j\right) \times \sum\limits_{i \in \llbracket n \rrbracket} m_j\left(v_i\right)}{\sum\limits_{k \in \llbracket n \rrbracket} d_{w,v_k}} \right)_{j \in \llbracket p \rrbracket} \\
&= \left(\frac{w_e\left(\mathfrak{e}_j\right) \times \#_m\left(\mathfrak{e}_j\right)}{\sum\limits_{k \in \llbracket n \rrbracket} d_{w,v_k}} \right)_{j \in \llbracket p \rrbracket} .
\end{aligned}$$

All components are non-negative and we check that the components of this vector sum to one:

$$\begin{aligned}
\sum_{j \in \llbracket p \rrbracket} \pi_{\mathfrak{E},j} &= \frac{\sum\limits_{j \in \llbracket p \rrbracket} w_e\left(\mathfrak{e}_j\right) \times \sum\limits_{i \in \llbracket n \rrbracket} m_j\left(v_i\right)}{\sum\limits_{k \in \llbracket n \rrbracket} d_{w,v_k}} \\
&= \frac{\sum\limits_{i \in \llbracket n \rrbracket} \sum\limits_{j \in \llbracket p \rrbracket} w_e\left(\mathfrak{e}_j\right) \times m_j\left(v_i\right)}{\sum\limits_{k \in \llbracket n \rrbracket} d_{w,v_k}} \\
&= \frac{\sum\limits_{i \in \llbracket n \rrbracket} d_{w,v_i}}{\sum\limits_{k \in \llbracket n \rrbracket} d_{w,v_k}} \\
&= 1.
\end{aligned}$$

□

These two claims show that this exchange-based process ranks vertices by their weighted m-degree and hb-edges by their weighted m-cardinality.

We have gathered the two-phase steps of the exchange-based diffusion process in Algorithm 1. The time complexity of this algorithm is $O\left(T\left(\mathrm{d}_{\mathfrak{H}}\mathrm{n} + \mathrm{r}_{\mathfrak{H}}\mathrm{p}\right)\right)$ where $d_{\mathfrak{H}} = \max\limits_{v_i \in V}\left(d_i\right)$ is the maximal degree of vertices in the hb-graph and $r_{\mathfrak{H}} = \max\limits_{\mathfrak{e}_j \in \mathfrak{E}} \left|\mathfrak{e}_j^{\star}\right|$ is the maximal cardinality of the support of a hb-graph. Usually, $d_{\mathfrak{H}}$ and $r_{\mathfrak{H}}$ are small compared to n and p. Algorithm 1 can be refined to determine automatically the number of iterations needed, fixing an accepted error to ensure convergence on the values for the vertices and storing the previous state.

Algorithm 1 Exchange-based diffusion algorithm.

```
Given:
A hb-graph $\mathfrak{H} = (V, \mathfrak{E}, w_e)$ with $|V| = n$ and $|\mathfrak{E}| = p$
Number of iterations: $T$

Initialisation:
For all $v_i \in V : \alpha_i := \frac{1}{n}$
For all $\mathfrak{e}_j \in \mathfrak{E} : \epsilon_j := 0$

DiffuseFromVerticesToHbEdges():
For $j := 1$ to $p$:
    $\epsilon_j := 0$
    For $v_i \in \mathfrak{e}_j^\star$:
        $\epsilon_j := \epsilon_j + \frac{m_j(v_i)\, w_e(\mathfrak{e}_j)}{d_{w,m}(v_i)} \alpha_i$

DiffuseFromHbEdgesToVertices():
For $i := 1$ to $n$:
    $\alpha_i := 0$
    For $\mathfrak{e}_j$ such that $v_i \in \mathfrak{e}_j^\star$:
        $\alpha_i := \alpha_i + \frac{m_j(v_i)}{\#_m \mathfrak{e}_j} \epsilon_j$

Main():
Calculate for all $i : d_{w,m}(v_i)$ and for all $j : \#_m \mathfrak{e}_j$
For $t = 1$ to $T$:
    DiffuseFromVerticesToHbEdges()
    DiffuseFromHbEdgesToVertices()
```

4 Evaluation and use cases

This section firstly addresses the validation of the approach taken on lab-generated hb-graphs. Secondly, this approach is applied to two use cases: one on the processing of the results of Arxiv querying and another one on Coco dataset images.

4.1 Validation on lab-generated hb-graphs

This diffusion by exchange process has been validated on two experiments: the first one generates a random hb-graph to validate the approach and the second compares the results with a classical and a modified random walk on the hb-graph.

Using lab-generated hb-graphs allow to test our diffusion on hb-graphs that have different shapes, and where the connectivity is always guaranteed. The lab-hb-graph generator includes different parameters to ensure both the connectivity, the number of groups—i.e. sub-hb-graphs—and the way of connection of these groups. As it is shown in Fig. 3, we generate $N_{\max}$ vertices. N_0 out of the $N_{\max}$ vertices are regrouped in V_0 and will be used for interconnection between the different groups. The remaining $N_{\max} - N_0$ vertices are then

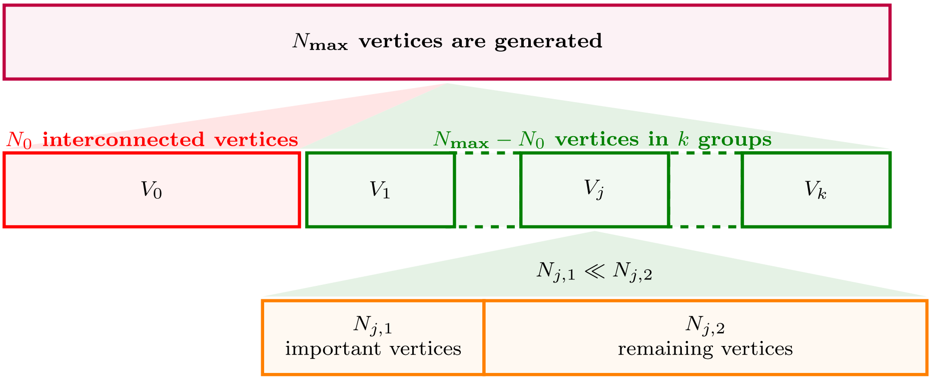

$N_{\max}$ vertices are generated
N_0 interconnected vertices
$N_{\max} - N_0$ vertices in k groups
V_0
V_1
V_j
V_k
$N_{j,1} \ll N_{j,2}$
$N_{j,1}$ important vertices
$N_{j,2}$ remaining vertices

Fig. 3 Random hb-graph generation principle

separated into k subsets $\left(V_j\right)_{j \in \llbracket k \rrbracket}$, corresponding to the vertices of the groups. In each of these k groups V_j, we generate two subsets of vertices: a first set $V_{j,1}$ of $N_{j,1}$ vertices and a second set $V_{j,2}$ of $N_{j,2}$ vertices with $N_{j,1} \ll N_{j,2}$, $j \in \llbracket k \rrbracket$. The number of hb-edges to be built is adjustable and shared between the different groups. The m-cardinality $\#_m (e)$ of a hb-edge is chosen randomly below a maximum tunable threshold. The multiplicity given to a vertex is also a random choice, tunable below a threshold. Vertices in $V_{j,1}$ are the vertices considered as important: their presence is required in a certain number of hb-edges per group; the number of important vertices in a hb-edge is randomly fixed below a maximum number. The completion of each hb-edge is done by choosing vertices randomly in the $V_{j,2}$ set. A vertex can be chosen randomly many times, increasing in this case its multiplicity within the hb-edge using the same random approach. The random choice made into these two groups is tuned to follow a power law distribution. It implies that some vertices occur more often than others. Interconnection between the k components is achieved by choosing vertices in V_0 and inserting them randomly into the hb-edges built.

The exchange-based diffusion is then applied to these generated hb-graphs: we analyze not only the validity of this diffusion process but also propose a visualisation of the results that highlights not only vertices but also hb-edges, both on the hb-graph and on its support hypergraph.

We make the hypothesis that vertices with the highest values of α_T correspond to vertices of the network that are important in the sense of being central for the connectivity. To validate this hypothesis, we are going to define a relative eccentricity of vertices from a subset of the vertex set in the hb-graph.

The eccentricity of a vertex in a graph is the length of a maximal shortest path between this vertex and the other vertices of this graph: extending this definition to hb-graphs is straightforward. If the graph is disconnected then each vertex has infinite eccentricity.

The **relative eccentricity** is then defined as the length of a maximal shortest path starting from a given vertex in a subset S of the vertex set V and ending with any vertices of $V \backslash S$. The relative eccentricity is calculated for every vertex of S provided that it is connected to vertices of $V \backslash S$; otherwise it is set to $-\infty$. The concept of relative eccentricity is illustrated in Fig. 4.

The subset of the vertex set V—written $A_V (s_V)$—is built by using a threshold value s_V : it gathers vertices of V with α_T-value above s_V. Consequently, the subset $B_V (s_V) \overset{\Delta}{=} V \backslash A_V (s_V)$ corresponds to the set of vertices with α_T values below the threshold. The

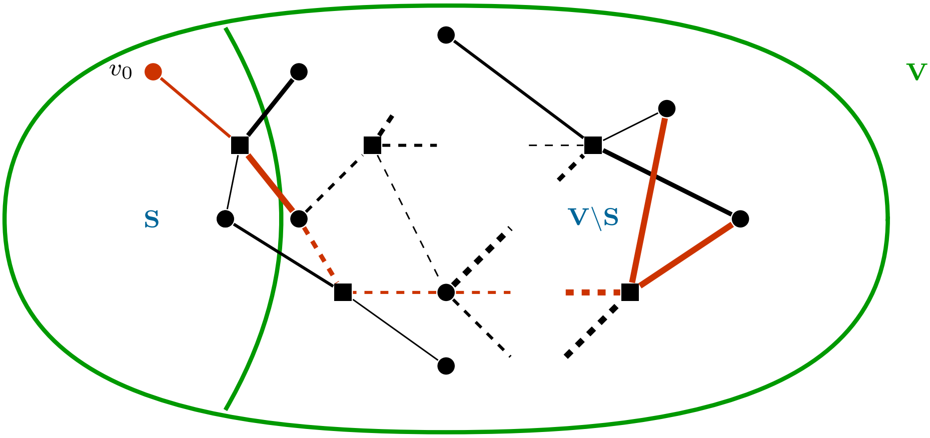


Fig. 4 Relative eccentricity: finding the length of a maximal shortest path in the hb-graph starting from a given vertex v_0 of S and finishing with any vertex in $V\backslash S$

relative eccentricity of each vertex of $A_V(s_V)$ to vertices of $B_V(s_V)$ in the support hypergraph of the corresponding hb-graph is then evaluated.

Assuming that iterations stop at time T, we let s_V vary in incremental steps from 0 to the value $\alpha_{T,\max} \overset{\Delta}{=} \max\limits_{v \in V}(\alpha_T(v))$. In order to have a relative value instead of the raw value s_V, we consider:

$$r_V \overset{\Delta}{=} \frac{s_V}{\alpha_{\text{ref}}}$$

where α_{ref} is the reference normalised value used for the initialisation of the α value of the vertices of the hb-graph $\mathfrak{H}$.

The results obtained by this experiment are shown on the two plots of Fig. 5. The first plot corresponds to the maximal length of the path between vertices of $A_V(s_V)$ and vertices of $B_V(s_V)$ that are connected according to the ratio r_V : this path length corresponds to half of the length of the path observed in the extra-vertex graph representation of the hb-graph support hypergraph as in between two vertices of V there is an extra-vertex that represents the hb-edge (or the support hyperedge). The second curve plots the percentage of vertices of V that are in $A_V(s_V)$ in function of r_V. When r_V increases, the number of elements in $A_V(s_V)$ naturally decreases while they get closer to the elements of $B_V(s_V)$, marking the fact that they are central.

Figures 6 and 7 show that high values of $\alpha_T(v)$ correspond to vertices that are highly connected either by degree or by m-degree.

A similar approach is taken for the hb-edges: assuming that the diffusion process stops at time T, we use the $\epsilon_{T-\frac{1}{2}}$ function to partition the set of hb-edges into two subsets for a given threshold $s_{\mathfrak{E}}$: $A_{\mathfrak{E}}(s_{\mathfrak{E}})$ of the hb-edges that have ϵ values above the threshold and $B_{\mathfrak{E}}(s_{\mathfrak{E}})$ the one gathering the hb-edges that have ϵ values below $s_{\mathfrak{E}}$.

$s_{\mathfrak{E}}$ varies from 0 to $\epsilon_{T-\frac{1}{2},\max} \overset{\Delta}{=} \max\limits_{\mathfrak{e} \in \mathfrak{E}}\left(\epsilon_{T-\frac{1}{2}}(\mathfrak{e})\right)$ by incremental steps while keeping the eccentricity above 0, first of the two conditions achieved. In the hb-graph representation, each hb-edge corresponds to an extra-vertex. Each time, we evaluate the length of the maximal shortest path linking one vertex of $A_{\mathfrak{E}}(s_{\mathfrak{E}})$ to one vertex of $B_{\mathfrak{E}}(s_{\mathfrak{E}})$ for connected vertices in the extra-vertex graph representation of the hb-graph support hypergraph: the

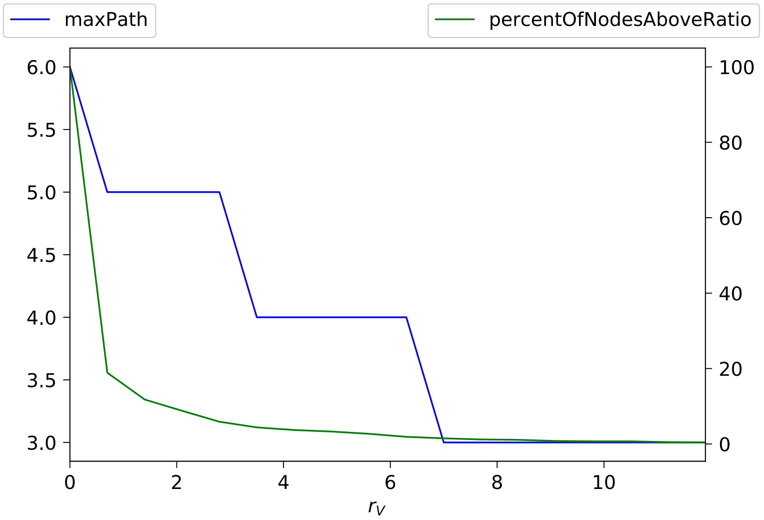


Fig. 5 Maximum path length and percentage of vertices in $A_V(s)$ over vertices in V vs ratio r_V

length of the path corresponds to half of the one obtained from the graph for the same reason as before.

We define the ratio:

$$r_{\mathfrak{E}} \triangleq \frac{s_{\mathfrak{E}}}{\beta_{\text{ref}}}$$

where $\beta_{\text{ref}} \triangleq \frac{1}{|\mathfrak{E}|}$ that corresponds to the normalised value that would be used in the dual hb-graph to initialize the diffusion process. In Fig. 8, we observe for the hb-edges the same trend than the one observed for vertices: the length of the maximal path between two

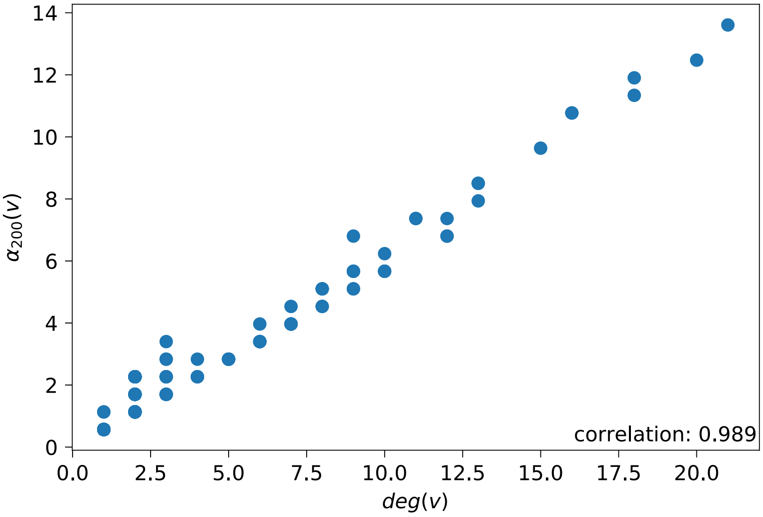


Fig. 6 Alpha value of vertices at step 200 vs degree of vertices

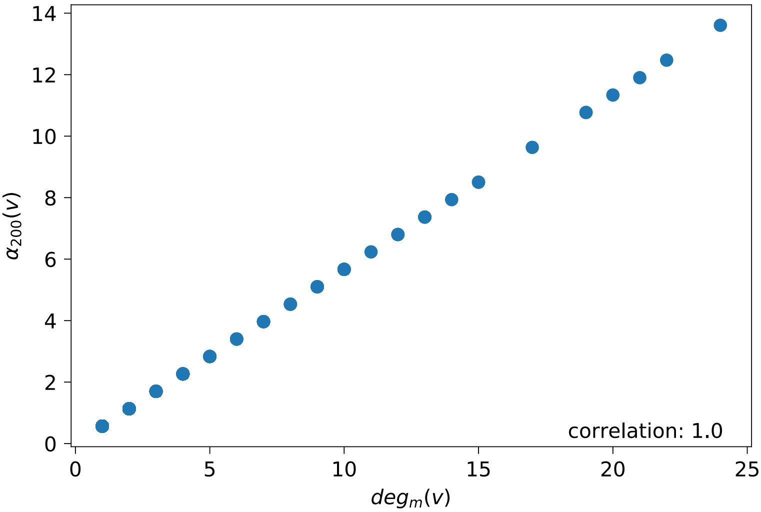


Fig. 7 Alpha value of vertices at step 200 vs m-degree of vertices

hb-edges decreases as the ratio $r_{\mathfrak{E}}$ increases while the percentage of vertices in $A_{\mathfrak{E}}(s_{\mathfrak{E}})$ over V decreases.

Figure 9 shows the high correlation between the value of $\epsilon(\mathfrak{e})$ and the cardinality of $\mathfrak{e}$; Fig. 10 shows that the correlation between value of $\epsilon(\mathfrak{e})$ and the m-cardinality of $\mathfrak{e}$ is even stronger.

The number of iterations needed to have a significant convergence depends on the initial conditions; we tried different initializations, either uniform, or applying some strokes on a different number of nodes. We observed that the more uniform the information on the network is, the less number of iterations for convergence is needed. No matter the configuration, the most important vertices in term of connectivity are always the most

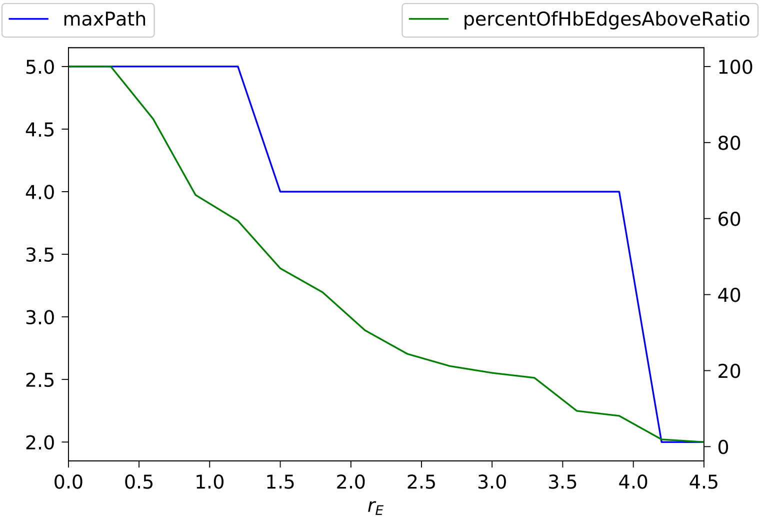


Fig. 8 Maximum path length and percentage of vertices in $A_{\mathfrak{E}}(s)$ vs ratio

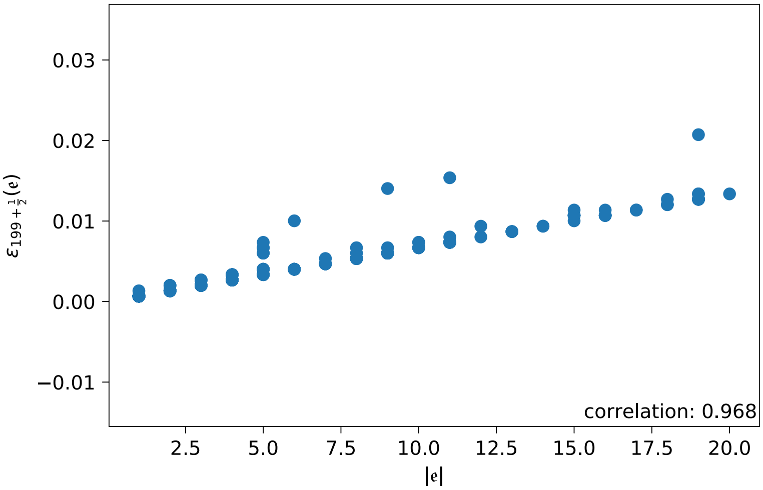


Fig. 9 Epsilon value of hb-edge at stage 199+$\frac{1}{2}$ and cardinality of hb-edge

highlighted ones. Figures 11 and 12 depict the convergence observed on a uniform initial distribution as it is described in the former section. In Fig. 11, we can see how the α-values as observed in Fig. 6 reflect the m-degree of the vertex they are associated with: 200 iterations is more than enough to rank the vertices by m-degree. In Fig. 12, we can observe an analogous phenomena with the ϵ-value associated to hb-edges that reflect the m-cardinality of the hb-edges. Again 200 iterations are sufficient to converge in the studied cases.

The number of iterations needed to converge depends on the structure of the network. In the transitory phase, the vertices need to exchange with the hb-edges; the process requires some iterations before converging and its behavior depends on the node connectivity and the hb-edge composition. It is an open question to investigate on this transitory phase to have more indications on the way the ϵ and the α-values vary.

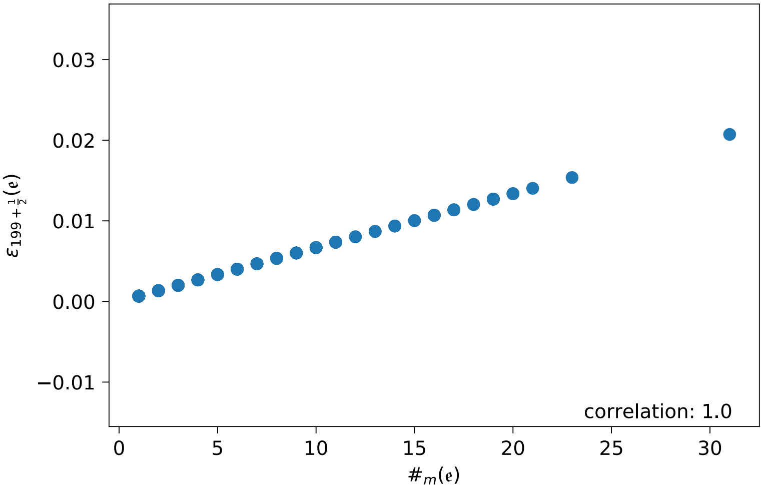


Fig. 10 Epsilon value of hb-edge at stage 199+$\frac{1}{2}$ and (m-)cardinality of hb-edge

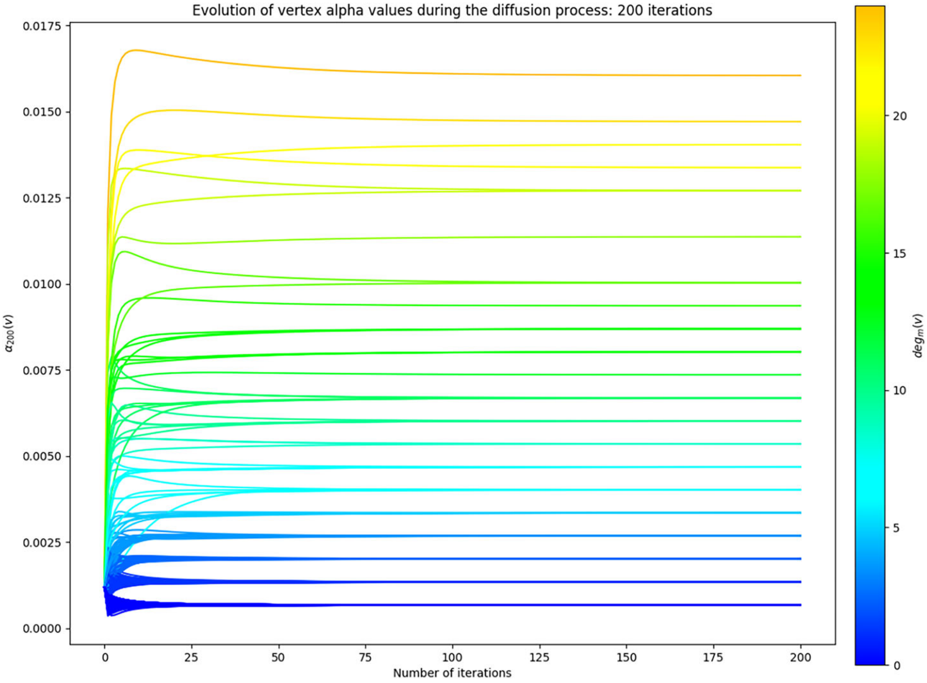


Fig. 11 Alpha value convergence of the vertices vs number of iterations. The plots are colored using a m-degree-based gradient coloring

We show an example of exchange-based diffusion on a lab-generated hb-graph in Fig. 13a and on its support hypergraph in Fig. 13b. The vertices are colored depending on the value of the ratio:

$$c_\alpha(v) \triangleq \frac{\alpha_T(v)}{\alpha_{\text{ref}}}$$

using the scale of colors on the right. Vertices with near zero $c_\alpha(v)$ values—i.e. low $\alpha_T(v)$ values compared to α_{ref}—are dark bluish colored; on the opposite, with high $c_\alpha(v)$ values—i.e. with high $\alpha_T(v)$ values compared to α_{ref}—are yellowish colored; when $c_\alpha(v)$ is close to 1, the vertices are colored in a close turquoise. The hb-edges—i.e. the extra-nodes representing images—are colored with the left gradient color scale according to the value of the ratio:

$$c_\epsilon(\mathfrak{e}) \triangleq \frac{\epsilon_{T-\frac{1}{2}}(\mathfrak{e})}{\epsilon_{\text{norm}}(\mathfrak{e})}$$

where $\epsilon_{\text{norm}}(\mathfrak{e}) \triangleq \sum\limits_{v \in \mathfrak{e}^\star} \frac{m_{\mathfrak{e}}(v)}{\deg_m(v)} \alpha_{\text{ref}}$. $\epsilon_{\text{norm}}(\mathfrak{e})$ corresponds to the value the hb-edge $\mathfrak{e}$ should have in reference to the fraction of α_{ref} given by each vertex and depending on the fraction of its multiplicity versus its m-degree in the hb-edge. Hb-edges are colored using $c_\epsilon(\mathfrak{e})$: when this ratio is close to 0—i.e. when the hb-edges have low $\epsilon_{T-\frac{1}{2}}(\mathfrak{e})$ compared to $\epsilon_{\text{norm}}(\mathfrak{e})$—hb-edges are colored in a blueish hue; when this ratio is high—i.e. when the hb-edges have high $\epsilon_{T-\frac{1}{2}}(\mathfrak{e})$ compared to what was expected with $\epsilon_{\text{norm}}(\mathfrak{e})$—they are colored in a reddish hue. It is worth mentioning that diffusing only on the support hypergraph of a hb-graph highlights only nodes that are highly connected inside a group, the ones being at the intersection

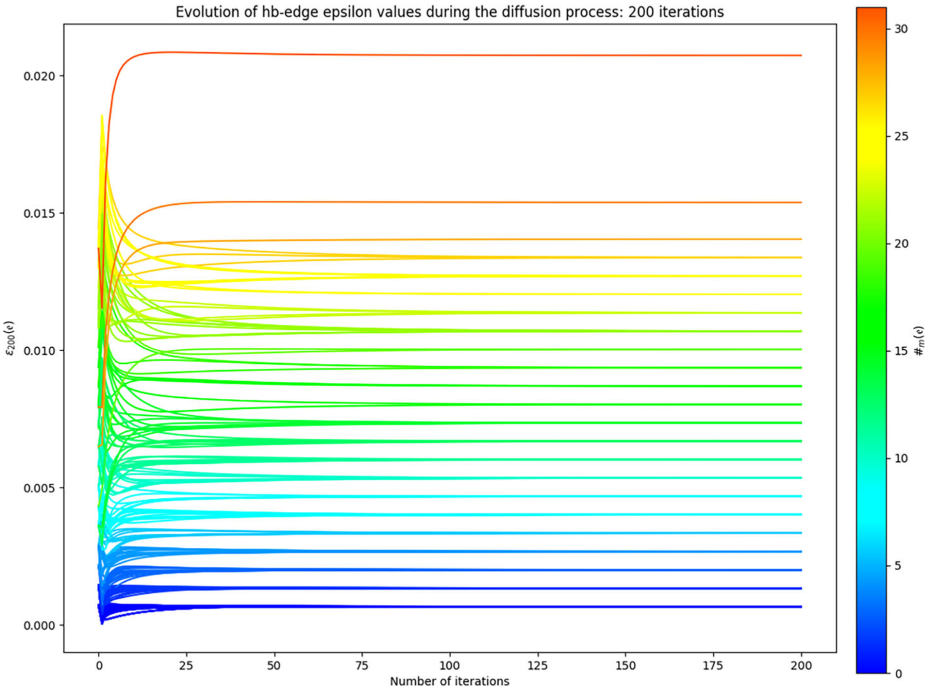
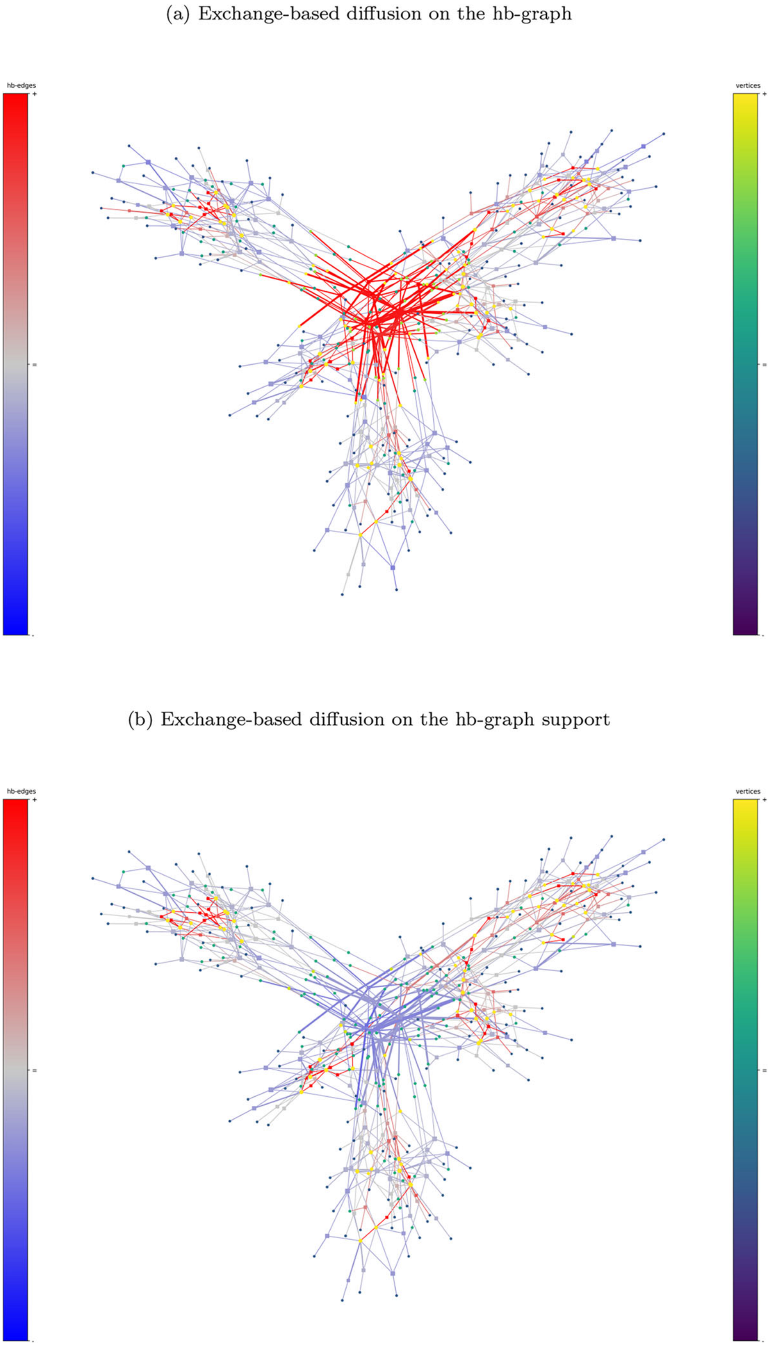


Fig. 12 Epsilon value convergence of hb-edges vs number of iterations. The plots are m-cardinality-based gradient colored

of the different groups have less importance in this case. The diffusion on the hb-graph captures the centrality of these vertices that are peripheral to the groups. Hence, taking the multiplicities into account brings valuable information on the network and on the centrality of some vertices.

To compare our exchange-based diffusion process to a baseline we consider a classical random walk. In this classical random walk, the walker who is on a vertex v chooses randomly a hb-edge that is incident with a uniform probability law and when the walker is on a hb-edge $\mathfrak{e}$, he chooses a vertex inside the hb-edge randomly with a uniform probability law. We let the possibility of teleportation to an other vertex from a vertex with a tunable value γ: $1-\gamma$ represents the probability to be teleported. We choose the classical value $\gamma = 0.85$. We count the number of passages of the walker through each vertex and each hb-edge. We stop the random walk when the hb-graph is fully explored. We iterate N times the random walk, N varying.

To improve the results of the classical random walk we propose a modified random walk—described in Algorithm 2—on the hb-graphs with random choice of hb-edges when the walker is on a vertex v with a distribution of probability $\left(\frac{w_e(\mathfrak{e}_i)\, m_i(v)}{\deg_{w,m}(v)}\right)_{i \in [\![p]\!]}$ and a random choice of the vertex when the walker is on a hb-edge $\mathfrak{e}$ with a distribution of probability $\left(\frac{m_{\mathfrak{e}}(v_i)}{\#_m(\mathfrak{e})}\right)_{i \in [\![n]\!]}$. We let the possibility of teleportation as it is done in the classical random walk. Similarly to the classical random walk, we count the number of passages of the walker through each vertex and each hb-edge. We also stop the random walk when the



(a) Exchange-based diffusion on the hb-graph

(b) Exchange-based diffusion on the hb-graph support

Fig. 13 Exchange-based diffusion in a hb-graph (**a**) and its support hypergraph (**b**) after 200 iterations of Algorithm 1: highlighting important hb-edges. Simulation with 848 vertices (chosen randomly out of 10 000) gathered in 5 groups of vertices (with 5, 9, 14, 16 and 9 important vertices and 2 important vertices per hb-edge), 310 hb-edges (with cardinality of support less or equal to 20), 10 vertices in between the 5 groups. Extra-vertices have square shape and are colored with the hb-edge color scale

hb-graph is fully explored. We iterate N times the random walk with various values of N. Assigning a multiplicity of 1 to every vertex and a weight of 1 for every hb-edge—with the vertex degree and the hb-edge cardinality instead of the multiplicity—retrieves the classical random walk from the modified random walk.

Algorithm 2 Modified random walk in hb-graphs

```
Given:
A hb-graph $\mathfrak{H} = (V, \mathfrak{E}, w_e)$ with $|V| = n$ and $|\mathfrak{E}| = p$
Number of Random walks: $T_{\mathrm{RW}}$
A teleportation threshold: $\gamma_{\mathrm{th}}$

Initialisation:
$\forall v \in V : n_V(v) := 0$
$\forall \mathfrak{e} \in \mathfrak{E} : n_{\mathfrak{E}}(\mathfrak{e}) := 0$
$Q :=$ deep copy $(V)$
$v_0 :=$ random $(v \in Q)$
$n_V(v_0) := 1$
$Q := Q \backslash \{v_0\}$

OneRW():
While $Q \neq \emptyset$:
    $\gamma_{\mathrm{rand}} :=$ random $([0; 1],$ weight $=$ uniform$)$
    if $\gamma_{\mathrm{rand}} < \gamma_{\mathrm{th}}$:
        # Visit of incident edges
        $\mathfrak{e}_{\mathrm{c}} :=$ random $\left(\mathfrak{e} \in \mathfrak{E} : v_{\mathrm{c}} \in \mathfrak{e}^{\star},\ \text{weight} = \left(\frac{w_e(\mathfrak{e}_j)\, m_{\mathfrak{e}_j}(v_0)}{\deg_{w_e,m}(v_0)}\right)_{\mathfrak{e}_j \in \mathfrak{E}}\right)$
        $n_V(\mathfrak{e}_{\mathrm{c}}) := n_V(\mathfrak{e}_{\mathrm{c}}) + 1$
        # Choice of the next vertex
        $v_0 :=$ random $\left(v \in V : v \in \mathfrak{e}_{\mathrm{c}}^{\star},\ \text{weight} = \left(\frac{m_{\mathfrak{e}_{\mathrm{c}}}(v)}{\#_m(\mathfrak{e}_{\mathrm{c}})}\right)_{v \in V}\right)$
        If $v_0 \in Q$:
            $Q := Q \backslash \{v_0\}$
            $n_V(v_0) := n_V(v_0) + 1$
    else:
        # Case of teleportation
        $v_0 :=$ random $\left(v \in V : v \in \mathfrak{e}_{\mathrm{c}}^{\star},\ \text{weight} = \left(\frac{m_{\mathfrak{e}_{\mathrm{c}}}(v)}{\#_m(\mathfrak{e}_{\mathrm{c}})}\right)_{v \in V}\right)$
        $Q := Q \backslash \{v_0\}$
        $n_V(v_0) := n_V(v_0) + 1$

Main():
For $i := 0$ to $T_{\mathrm{RW}}$ :
    OneRW()
$\forall v \in V : \overline{n_V}(v) = \frac{n_V(v)}{T_{\mathrm{RW}}}$
$\forall \mathfrak{e} \in \mathfrak{E} : \overline{n_{\mathfrak{E}}}(\mathfrak{e}) = \frac{n_{\mathfrak{E}}(\mathfrak{e})}{T_{\mathrm{RW}}}$
```

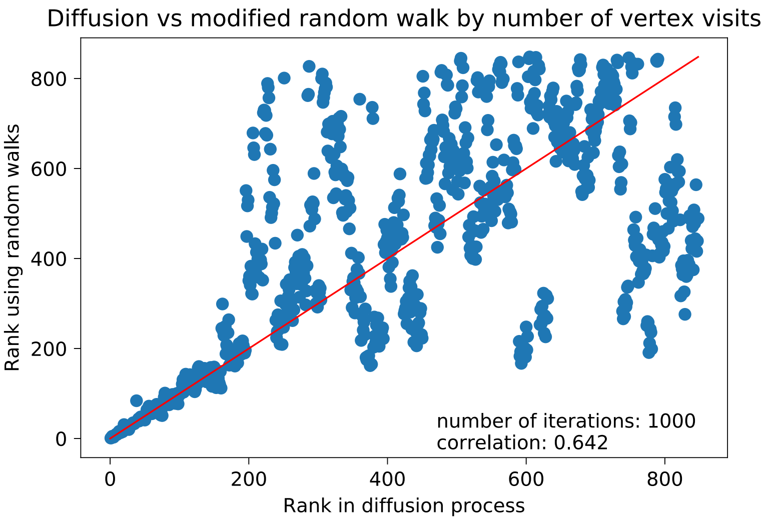


Fig. 14 Comparison of the rank obtained by a thousand modified random walks after total discovery of the vertices in the hb-graph and rank obtained with 200 iterations of the exchange-based diffusion process

Figure 14 shows that there is a good correlation between the rank obtained by a thousand modified random walks and after two hundreds iterations of our diffusion process, especially for the first two hundred vertices of the network, which is generally the ones targeted. The lack of anti-correlation between the rank obtained by the random walk with the degree of the vertices and the m-degree of vertices as shown respectively in Figs. 15 and 16

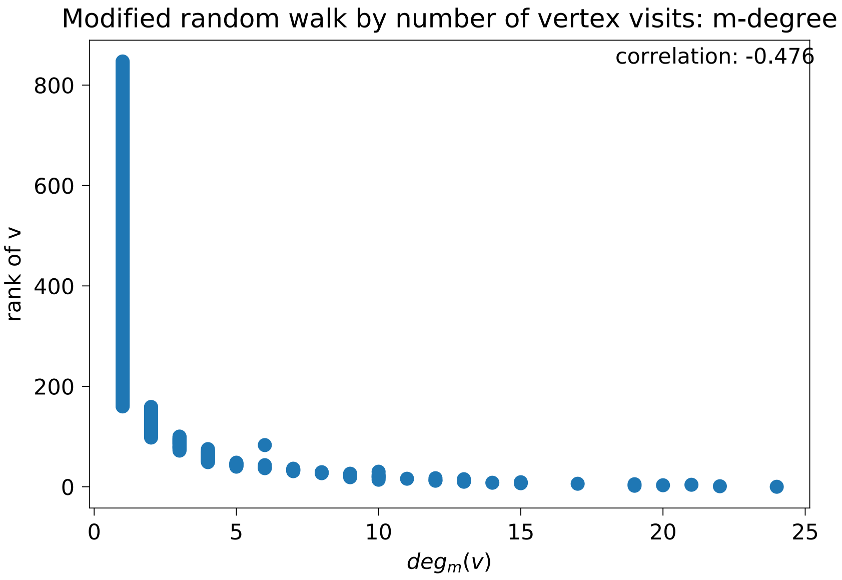


Fig. 15 Comparison of the rank obtained by a thousand modified random walks after total discovery of the vertices in the hb-graph and m-degree of vertices

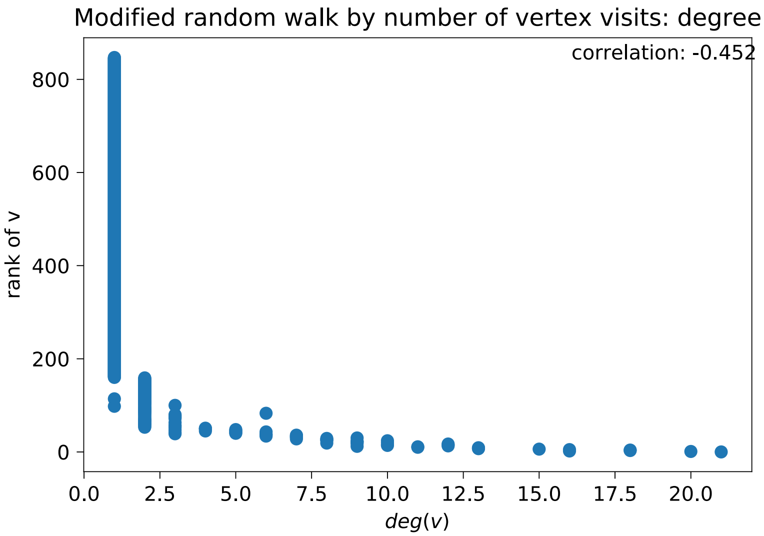


Fig. 16 Comparison of the rank obtained by a thousand modified random walks after total discovery of the vertices in the hb-graph and degree of vertices

is mainly due to the vertices with low m-degrees / degrees, but this lack decreases with the modified random walk.

We can remark in Fig. 17 that the correlation is a bit lower with a thousand classical random walks due to the fact that there are more vertices that are seen as differently ranked in between the two approaches. In Fig. 18, we can see that the ranks in the classical random walk relies more on the degree than on the m-degree as shown in Fig. 19, especially for vertices with small (m-)degrees; but there is still a misclassification for lower (m-)degree vertices.

We have compared the three methods from a computational time perspective; the results are shown in Table 1. The diffusion process is clearly faster; the modified random walk, essentially related to the overhead due to the large number of divisions, requires longer than the classical random walk. A lot of optimization can be foreseen to make this modified random walk run faster. The random walks can be easily parallelized; it is also the case for the diffusion process. The number of iterations in the diffusion process can also be optimized. These issues will be addressed in future work.

4.2 Two use cases

4.2.1 Application to Arxiv querying

We use the standard Arxiv API[3] to perform searches on Arxiv database. When performing a search, the query is transformed into a vector of words which is the basis for the retrieval of documents. The most relevant documents are retrieved based on a similarity measure between the query vector and the word vectors associated to individual documents. Arxiv relies on Lucene's built-in Vector Space Model of information retrieval and the Boolean

[3] https://arxiv.org/help/api/index

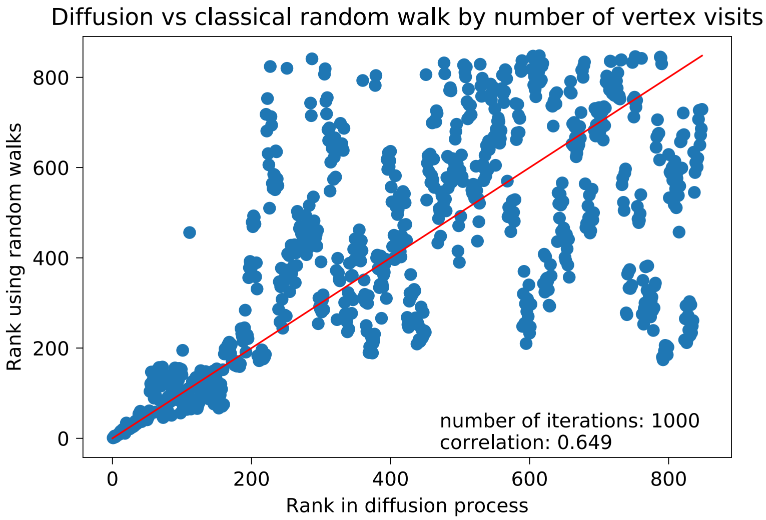


Fig. 17 Comparison of the rank obtained by a thousand classical random walks after total discovery of the vertices in the hb-graph and rank obtained with 200 iterations of the exchange-based diffusion process

model.[4] The Arxiv API returns the metadata associated to documents with highest scores for the query performed.

This metadata, filled by the authors during their submission of a preprint, contains different information such as authors, Arxiv categories and abstract.

We process these abstracts using TextBlob, a natural language processing Python library[5] and extract the nouns using the tagged text.

Nouns in the abstract of each document are scored with TF-IDF, the Term Frequency - Invert Document Frequency. Even if it is a classical measure, we just remind here its definition:

$$\text{TF-IDF}\,(x,d) = \text{TF}(x,d) \times \text{IDF}\,(x,d)$$

with $\text{TF}(x,d)$ the relative frequency of x in d and $\text{IDF}\,(x,d)$ the invert document frequency.

Writing n_d the total number of terms in document d and n_x the number of occurrences of x :

$$\text{TF}(x,d) = \frac{n_x}{n_d}$$

and writing N the total number of documents and $n_{x\in d}$ the number of documents having an occurrence of x, we have:

$$\text{IDF}\,(x,d) = \log_{10}\left(\frac{N}{n_{x\in d}}\right).$$

Scoring each noun in each abstract of the retrieved documents generates a hb-graph $\mathfrak{H}_Q$ of universe the nouns contained in the abstracts. Each hb-edge contains a multiset of nouns extracted from a given abstract with a multiplicity function that represents the TF-IDF score of each noun.

The exchange-based diffusion process is then applied to the hb-graph $\mathfrak{H}_Q$. We show two typical examples for the same query: the first 50 results in Fig. 20 and the first 100 results

[4] https://lucene.apache.org/core/2_9_4/scoring.html

[5] https://textblob.readthedocs.io/en/dev/

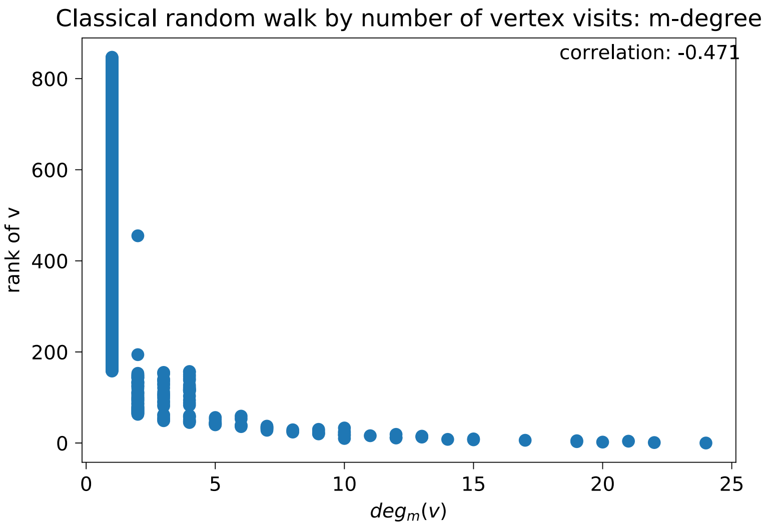


Fig. 18 Comparison of the rank obtained by a thousand classical random walks after total discovery of the vertices in the hb-graph and m-degree of vertices

in Fig. 21. The number of iterations needed to reach convergence is less than 10 in these two cases; with 500 results, around 10 iterations are needed for all hb-edges but one where 30 iterations are needed.

As the hb-edges correspond to documents in Arxiv database, we compare the central documents obtained in the results of the queries: we observe that the ranking obtained based on the $\epsilon_{49+\frac{1}{2}}$ differs significantly from the ranking by pertinence given by Arxiv API. In the

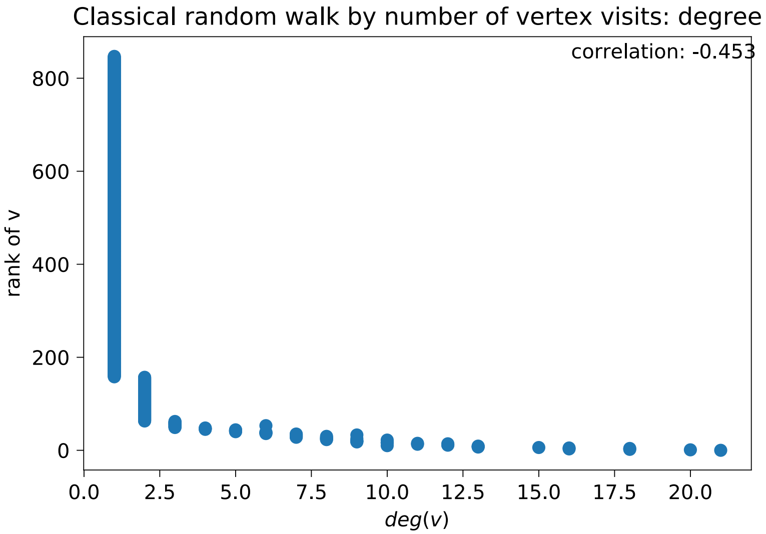


Fig. 19 Comparison of the rank obtained by a thousand classical random walks after total discovery of the vertices in the hb-graph and degree of vertices

Table 1 Time taken for doing 100, 200, 500 and 1000 iterations of the diffusion algorithm and 100, 200, 500 and 1000 classical and modified random walks on different hb-graphs

$\lvert\mathfrak{E}\rvert$	$\lvert V\rvert$	k	N_1	N_0	Type of algorithm	100	200	500	1000
55	106	1	5	5	classical random walk	0.40 ± 0.05	0.78 ± 0.07	1.92 ± 0.10	3.82 ± 0.14
55	106	1	5	5	diffusion	0.05 ± 0.02	0.08 ± 0.02	0.20 ± 0.04	0.39 ± 0.06
55	106	1	5	5	modified random walk	0.71 ± 0.06	1.43 ± 0.09	3.56 ± 0.17	7.12 ± 0.23
55	132	3	5	5	classical random walk	0.49 ± 0.05	0.96 ± 0.06	2.36 ± 0.08	4.71 ± 0.12
55	132	3	5	5	diffusion	0.05 ± 0.02	0.09 ± 0.02	0.21 ± 0.04	0.42 ± 0.05
55	132	3	5	5	modified random walk	0.89 ± 0.06	1.77 ± 0.09	4.43 ± 0.13	8.85 ± 0.19
55	91	5	5	5	classical random walk	0.30 ± 0.04	0.59 ± 0.05	1.44 ± 0.06	2.85 ± 0.07
55	91	5	5	5	diffusion	0.04 ± 0.02	0.07 ± 0.02	0.16 ± 0.03	0.31 ± 0.04
55	91	5	5	5	modified random walk	0.55 ± 0.05	1.09 ± 0.06	2.71 ± 0.09	5.42 ± 0.14
305	534	1	5	5	classical random walk	4.05 ± 0.16	8.07 ± 0.26	20.10 ± 0.45	40.17 ± 0.85
305	534	1	5	5	diffusion	0.29 ± 0.06	0.57 ± 0.08	1.35 ± 0.09	2.64 ± 0.10
305	534	1	5	5	modified random walk	6.86 ± 0.28	13.71 ± 0.41	34.16 ± 0.75	68.28 ± 1.21
305	491	3	5	5	classical random walk	3.51 ± 0.13	6.98 ± 0.21	17.39 ± 0.38	34.77 ± 0.70
305	491	3	5	5	diffusion	0.27 ± 0.05	0.53 ± 0.09	1.25 ± 0.11	2.43 ± 0.11
305	491	3	5	5	modified random walk	6.02 ± 0.22	12.03 ± 0.41	30.10 ± 0.73	60.23 ± 1.34
305	499	5	5	5	classical random walk	3.31 ± 0.15	6.58 ± 0.20	16.38 ± 0.34	32.72 ± 0.51
305	499	5	5	5	diffusion	0.24 ± 0.04	0.47 ± 0.06	1.12 ± 0.06	2.18 ± 0.08
305	499	5	5	5	modified random walk	5.86 ± 0.26	11.70 ± 0.37	29.26 ± 0.58	58.51 ± 0.89

exchange-based diffusion, the ranking sorts documents depending on their respective word weights and their centrality as we have seen in the experimental part on random hb-graphs.

Moreover, we have observed that when the number of results retrieved increases the top 5, top 10 documents sometimes change drastically depending on the retrieval of new documents that are more central with respect to the words they contain. If the gap seems small with a few documents retrieved, it increases as the number of documents increases. Increasing the number of results reveals the full theoretical hb-graph obtained from the whole dataset of the query performed, and hence, reveals the subjects central to this dataset.

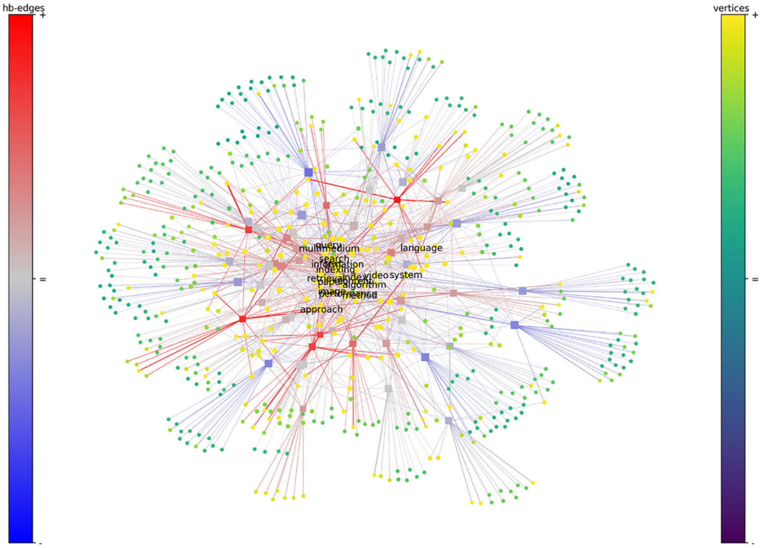


Fig. 20 Querying Arxiv. The search performed is “content-based multimedia indexing” for which 50 most relevant articles have been retrieved with 100 iterations. Top 10: 1: multimedium; 2: video; 3: search; 4: retrieval; 5: image; 6: indexing; 7: paper; 8: index; 9: method; 10: system

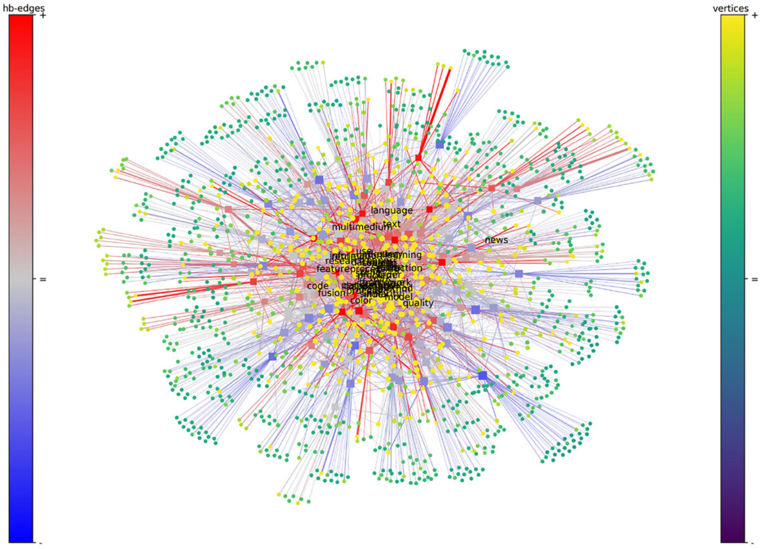


Fig. 21 Querying Arxiv. The search performed is "content-based multimedia indexing" for which 100 most relevant articles have been retrieved with 100 iterations on the exchange-based diffusion. Top 10: 1: paper; 2: index; 3: multimedium; 4: image; 5: method; 6: video; 7: retrieval; 8: performance; 9: indexing; 10: system

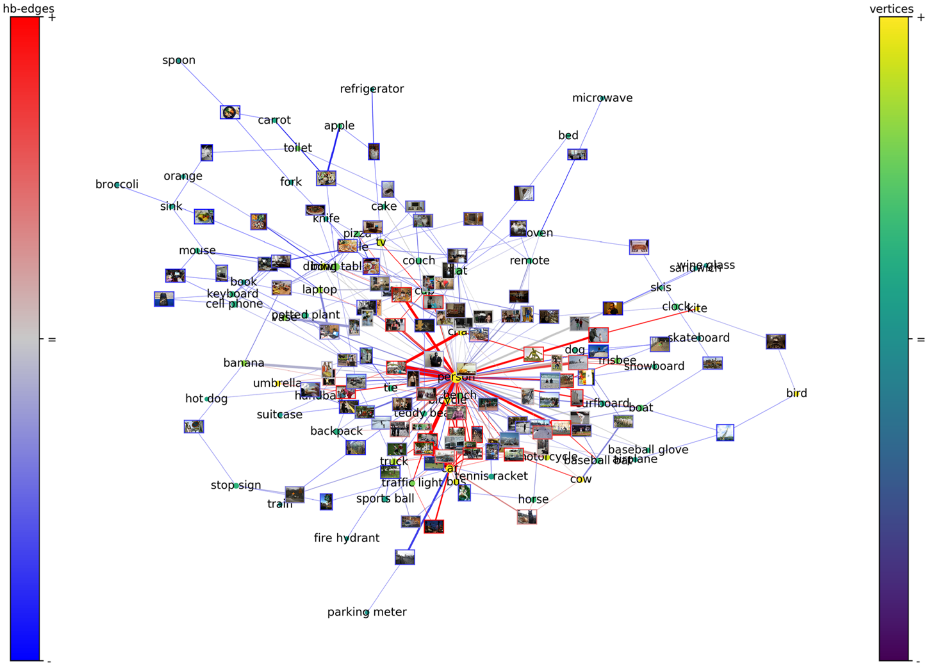
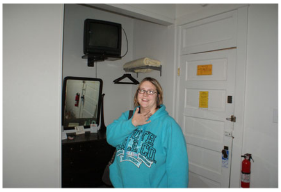
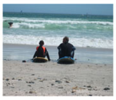


Fig. 22 Exchange-based diffusion on the sub-hb-graph first component with 175 images of a hb-graph of 199 images of the COCO 2014 training dataset

		
Reference	237245	282045
Objects detected	Person: 1; TV: 1.	Persons: 3; Surfboard:2.
Image rank	1 out of 175	72 out of 175

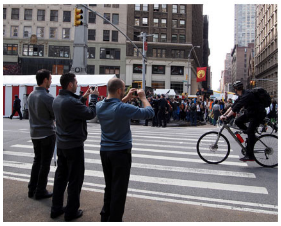
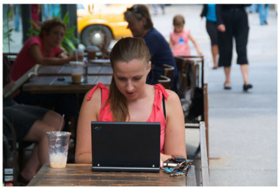
		
Reference	348954	347167
Objects detected	Persons: 7; Bicycle: 1; Traffic light: 1; Backpack: 1.	Persons: 8; Cups:2; Laptop:1
Image rank	27 out of 175	17 out of 175

Fig. 23 Image examples: references are the COCO training dataset 2014 image references

Hence the diffusion process can highlight importance of documents with respect to central subjects when processing the results of the query.

4.2.2 Application to an image database

We have applied the exchange-based diffusion to a database of images. We have used a hb-graph modeling of the objects detected on individual images to build a network of co-occurrences. Each image has been processed using a Retina neural network to label the objects it contains, and each object is then counted in its own category. The database used is the 2014 training set of the COCO dataset[6] [18]. The use of a pre-trained Retina net[7] allows to give bounding boxes corresponding to concepts, with a probability associated to it. We then choose a threshold below which we reject the bounding box: it has been fixed at 0.5, as it is proposed by the library developer. Hence, we can associate to each image its concepts and their multiplicity.

Two hb-graphs can be build. First, a hb-graph of images $\mathfrak{H}_{\text{Im}}$, where the vertex set is constituted of the different concepts—objects—that the image holds and where a hb-edge is related to an image, regrouping the different concepts with their respective multiplicity. The second hb-graph is the hb-graph of concepts $\mathfrak{H}_{\text{Co}}$: the vertex set corresponds to the image set and a hb-edge regroups the images holding the concept with a multiplicity that

[6] http://cocodataset.org/#home.

[7] https://github.com/fizyr/keras-retinanet

corresponds to the number of times the corresponding concept occurs in the image. These two hb-graphs are dual one of the other. We now focus on the hb-graph of images.

198 images of the COCO 2014 training dataset have been randomly selected, building the original image hb-graph. To ensure connectivity, only the first main component of the original image hb-graph is kept: it is constituted of 175 images. This component is designated as the hb-graph in the remainder. We then enhance the diffusion on this connected hb-graph. A typical result is presented in Fig. 22: the concepts are the vertices, the images represent the extra-vertices corresponding to the hb-edges. The coloration of vertices—i.e. the nodes of the concepts— and of hb-edges—i.e. the extra-nodes representing images—is the same than the one used in Fig. 13. Images containing persons are more reddish than images without persons, as the concept of person is central to the first component. But a lot of the images highlighted in red with persons contain other concepts, that are seen as important. It is the case for the image Reference 237245 in Fig. 23 which shows one person with one TV, two concepts that are central. Nonetheless, if the second concept is less important

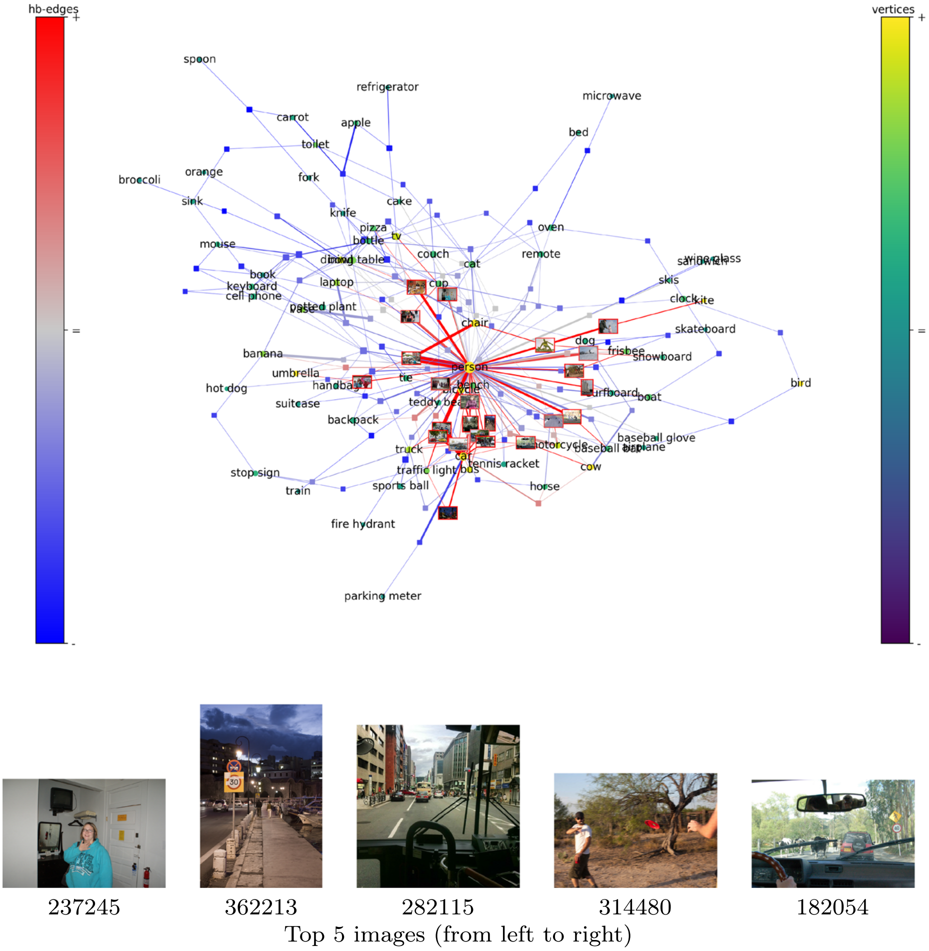


Fig. 24 Most 20% important images as detected by the exchange-based diffusion on the sub-hb-graph first component with 175 images of a hb-graph of 199 images of the COCO 2014 training dataset

the importance of the image decreases, as shown for instance in the image Ref. 282045 in Fig. 23 which contains 2 persons and 2 surfboards—the image is seen as less important as the concept of surfboard is less important than the one of TV. It is worth mentioning that images containing a lot of persons are not systematically highlighted in red—for instance, image Ref. 348954 in Fig. 23 with 7 persons, 1 bicycle, 1 traffic light and 1 backpack is seen as less important than image Ref 347167 in Fig. 23 with 8 persons, 2 cups and 1 laptop. The closer to red the images are, the more central to the sample drawn they are; hence, these images can potentially be used to make a summary of this sample, by selecting for instance the top 20% images based on their importance in the exchange-based diffusion, based on the c_{ϵ} ($\mathfrak{e}$)-value calculated based on the diffusion process, as it is shown in Fig. 24. This strategy for summarizing can be refined with more complex strategies in order to fully covered the dataset concepts: it is kept as future work.

5 Future work and conclusion

Through this study, hb-graphs by enabling multiplicities of elements that are hb-edge based have proven to be efficient in retrieving the important part of a co-occurrence network. The two-phase step diffusion proposed enhances the possibility of retrieving information not only for vertices but also for hb-edges. The two use-cases show the potential of such approaches.

Different applications can be thought in particular in the search of tagged multimedia documents for refining the results and scoring of documents retrieved. Using tagged documents ranking by this means could help in creating visualisation summary. Our approach is seen as a strong basis to refine the approach of [41]; it can also be viewed as a mean to make query expansion and disambiguation by using additional highly scored words in the network and as a way of making some recommendation based on the scoring of a document based on its main words.

Acknowledgments This work is part of the PhD of Xavier OUVRARD, done at UniGe, supervised by Stéphane MARCHAND-MAILLET and founded by a doctoral position at CERN, in the Collaboration Spotting team, supervised by Jean-Marie LE GOFF. The authors are thankful to Roderic BOUESSE (University of Geneva) to have selected the tool needed for the image use case.

References

1. Bellaachia A, Al-Dhelaan M (2013) Random walks in hypergraph. In: Proceedings of the 2013 international conference on applied mathematics and computational methods, Venice, Italy, pp 187–194
2. Bendersky M, Croft WB (2012) Modeling higher-order term dependencies in information retrieval using query hypergraphs, vol 941, ACM Press
3. Berge C (1973) Graphs and hypergraphs, vol 7. North Holland, Amsterdam

4. Bretto A, Theory Hypergraph (2013) Mathematical engineering. Springer International Publishing, Heidelberg
5. Bu J, Tan S, Chen C, Wang C, Wu H, Zhang L, He X (2010) Music recommendation by unified hypergraph: combining social media information and music content. In: Proceedings of the international conference on Multimedia - MM '10, (Firenze, Italy), vol 391. ACM Press
6. Chauve C, Patterson M, Rajaraman A (2013) Hypergraph covering problems motivated by genome assembly questions. In: Combinatorial algorithms, vol 8288. Springer, Berlin, pp 428–432
7. Csurka G, Dance C, Fan L, Willamowski J, Bray C (2004) Visual categorization with bags of keypoints. In: Workshop on statistical learning in computer vision, ECCV, vol 1, pp 1–2. Prague
8. Cummins N, Amiriparian S, Ottl S, Gerczuk M, Schmitt M, Schuller B (2018) Multimodal bag-of-Words for Cross Domains Sentiment Analysis. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), (Calgary, AB), p 2018. IEEE
9. Deselaers T, Pimenidis L, Ney H (2008) Bag-of-visual-words models for adult image classification and filtering. In: 2008 19th international conference on pattern recognition, (Tampa, FL, USA), pp 1–4. IEEE
10. Ducournau A, Bretto A (2014) Random walks in directed hypergraphs and application to semi-supervised image segmentation. Comput Vis Image Underst 120:91–102
11. Gao Y, Wang M, Tao D, Ji R, Dai Q (2012) 3-D Object retrieval and recognition with hypergraph analysis. IEEE Trans Image Process 21:4290–4303
12. Girish K, Jacob JS (2012) On multiset topologies. Theory Appl Math Comput Sci 2(1):37–52
13. Grossman JW, Ion PD (1995) On a portion of the well-known collaboration graph. In: Congressus Numerantium, pp 129–132
14. Harris ZS (1954) Distributional structure. WORD 10:146–162
15. Karypis G, Aggarwal R, Kumar V, Shekhar S (1999) Multilevel hypergraph partitioning: applications in VLSI domain. IEEE Trans Very Large Scale Integr VLSI Syst 7:69–79
16. Kim H, Kim H, Cho S (2017) Bag-of-concepts: Comprehending document representation through clustering words in distributed representation. Neurocomputing 266:336–352
17. Lee J, Cho M, Lee KM (2011) Hyper-graph matching via reweighted random walks. In: Computer Vision and Pattern Recognition (CVPR) IEEE Conference on, p 2011
18. Lin T-Y, Maire M, Belongie S, Hays J, Perona P, Ramanan D, Dollár P, Zitnick CL (2014) Microsoft coco: Common objects in context. In: European conference on computer vision, pp 740–755. tex.organization: Springer
19. Lu L, Peng X (2011) High-ordered random walks and generalized laplacians on hypergraphs, Springer
20. Ma S, Sun X, Wang Y, Lin J (2018) Bag-of-words as target for neural machine translation, arXiv:1805.04871
21. Minaee S, Wang S, Wang Y, Chung S, Wang X, Fieremans E, Flanagan S, Rath J, Lui YW (2017) 2017 Identifying mild traumatic brain injury patients from MR images using bag of visual words. In: IEEE Signal Processing in Medicine and Biology Symposium (SPMB), (Philadelphia, PA), pp 1–5. IEEE
22. Newman MEJ (2001) Scientific collaboration networks. I. Network construction and fundamental results. Phys Rev E 64:016131
23. Newman MEJ (2001) Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality. Phys Rev E 64:016132
24. Ouvrard X, Le Goff J-M, Marchand-Maillet S (2018) Diffusion by exchanges in hb-graphs : highlighting complex relationships. In: CBMI proceedings
25. Ouvrard X, Le Goff J-M, Marchand-Maillet S (2018) Hypergraph modeling and visualisation of complex co-occurence networks. Electron Notes Discrete Math 70:65–70
26. Ouvrard X, Le Goff J-M, Marchand-Maillet S (2018) Adjacency and tensor representation in general hypergraphs. Part 2: multisets. Hb-graphs and Related e-adjacency Tensors. arXiv:1805.11952
27. Ouvrard X, Le Goff J-M, Marchand-Maillet S (2019) On Hb-graphs and their Application to General Hypergraph e-adjacency Tensor. In: MCCCC 32 special volume of the journal of combinatorial mathematics and combinatorial computing to be published
28. Ouvrard X, Le Goff J-M, Marchand-Maillet S (2019) Diffusion by exchanges in hb-graphs: highlighting complex relationships extended version.arXiv:1809.00190v2
29. Ouvrard X, Le Goff J-M, Marchand-Maillet S (2020) The hyperbaggraph dataedron: an enriched browsing experience of datasets. In: LNCS, 46th international conference on current trends in theory and practice of computer science (SOFSEM 2020)
30. Peng X, Wang L, Wang X, Qiao Y (2016) Bag of visual words and fusion methods for action recognition:, Comprehensive study and good practice. Comput Vis Image Und 150:109–125
31. Purda L, Skillicorn D (2015) Accounting variables, deception, and a bag of words: assessing the tools of fraud detection. Contemp Account Res 32:1193–1223

32. Schmitt M, Janott C, Pandit V, Qian K, Heiser C, Hemmert W, Schuller B (2016) A bag-of-audio-words approach for snore sounds' excitation localisation. VDE
33. Shekhar R, Jawahar C (2012) Word image retrieval using bag of visual words. In: 2012 10th IAPR international workshop on document analysis systems, (Gold Coast, Queenslands, TBD, Australia), pp 297–301. IEEE
34. Silva FB, Werneck RDO, Goldenstein S, Tabbone S, Torres RDS (2018) Graph-based bag-of-words for classification. Pattern Recogn 74:266–285
35. Singh D, Ibrahim A, Yohanna T, Singh J (2007) An overview of the applications of multisets. Novi Sad J Math 37(3):73–92
36. Sivic J, Zisserman A (2003) Video google: a text retrieval approach to object matching in videos. In: Proceedings 9th IEEE international conference on computer vision, (Nice, France), vol 2, pp 1470–1477. IEEE
37. Taramasco C, Cointet J.-P., Roth C (2010) Academic team formation as evolving hypergraphs. Scientometrics 85:721–740
38. Temkin ON, Zeigarnik AV, Bonchev D (1996) Chemical reaction networks: a graph-theoretical approach. CRC Press, Boca Raton
39. Tsai C-F (2012) Bag-of-words representation in image annotation: a review. ISRN Artif Intell 2012:1–19
40. Wang Y, Zhu L, Qian X, Han J (2018) Joint hypergraph learning for tag-Based image retrieval. IEEE Trans Image Process 27:4437–4451
41. Xu Z, Du J, Ye L, Fan D (2016) Multi-feature indexing for image retrieval based on hypergraph. In: 2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS), (Beijing, China). IEEE, pp 494–500
42. Zhao J, Collins C, Chevalier F, Balakrishnan R (2013) Interactive exploration of implicit and explicit relations in faceted datasets. IEEE Trans Vis Comput Graph 19:2080–2089
43. Zhu L, Shen J, Jin H, Zheng R, Xie L (2015) Content-based visual landmark search via multimodal hypergraph learning. IEEE Trans Cybern 45:2756–2769
44. Zhou D, Huang J, Schölkopf B. (2007) Learning with hypergraphs: Clustering, classification, and embedding. In: Advances in neural information processing systems, pp 1601–1608

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Xavier Ouvrard graduated engineer INPG - EFPIG - Paper making section -, PhD in Computer Science and MSc in Process Engineering has been an Associate Professor in Mathematics in an International School and IT network administrator. Returning to computer science studies, he has just hold his PhD defense at the University of Geneva (Switzerland) in March 2020. He is now Senior Fellow at CERN in the Radio-protection group of the HSE department. His research during his PhD was focused on the modeling, the analyse and visualisation of complex co-occurence networks to allow knowledge discovery inside textual datasets. He has recently introduced an e-adjacency tensor for general hypergraphs and hyper-bag-graphs, that are families of multisets.

Jean-Marie Le Goff is a senior applied physicist and computer scientist that focuses on applying advanced IT techniques and concepts to Particle Physics. He first studied how to move objects over the internet using CORBA to service the control system middleware of the Large Hadron Collider (LHC) experiments at CERN. He then worked on extending the concepts of Unified Modeling Language (UML) layers with a description-driven layer for classes and objects which led to the development of a software (C.R.I.S.T.A.L.) dedicated to the tracking and assembly of detector parts. This versatile software found applications outside particle physics, in particular in industry as Enterprise Resource Programming (ERP) software and Business Process Management (BPM), and in accounting and finance. He is currently working on the use of emerging graph, semantic and structural abstraction techniques for data management and visualization in conjunction with techniques acquired in his previous work. This led to the development of the Collaboration Spotting software, a generic platform for visual analytics of complex datasets. The platform is being used to build various demonstrators including for compatibility and dependency relationships in software and metadata of an experiment at CERN, in scientometry with publication and patent information, pharmaco-analytics and neurosciences. Jean-Marie Le Goff holds a PhD in experimental particle physics and a DPhil in Computer Science. From 06/2003–06/2009, he has been Visiting Professor at the University of the West of England, Bristol, UK.

Stéphane Marchand-Maillet Professor in the Department of Computer Science at the University of Geneva, Switzerland, where he leads the Viper group. His research is directed towards large-scale, high-dimensional distributed machine learning and information mining and indexing, with applications to data modelling and prediction. He has authored, co-authored or edited a number of publications on these topics. He and his group are part of several national and European and international projects in the domain. He is Senior PC Member of the International Joint Conference on AI (IJCAI, one of the oldest established conferences in AI). He was general co-chair of the International Conference of the ACM-SIG on Information Retrieval in 2010 (ACM-SIGIR 2010) and general co-chair of the 16th IEEE Conference in Business Informatics in 2014 (IEEE-CBI 2014)