Network decomposition into fixed points of degree peeling

James Abello, François Queyroi

Noname manuscript No.
(will be inserted by the editor)

Network Decomposition into Fixed Points of Degree Peeling

James Abello · François Queyroi

the date of receipt and acceptance should be inserted later

Abstract Degree peeling is used to study complex networks. It is a decomposition of the network into vertex groups of increasing minimum degree. However, the peeling value of a vertex is non-local in this context since it relies on the number of connections the vertex has to groups above it. We explore a different way to decompose a network into edge layers such that the *local* peeling value of the vertices on each layer does not depend on their non-local connections with the other layers. This corresponds to the decomposition of a graph into subgraphs that are invariant with respect to degree peeling, i.e. they are fixed points.
We introduce a general method to partition the edges of an arbitrary graph into fixed points of degree peeling, called the iterative-edge-core decomposition. Information from this decomposition is used to formulate a novel notion of vertex diversity based on Shannon's entropy. We illustrate the usefulness of this decomposition on a variety of social networks including weighted graphs. Our method can be used as a preprocessing step for community detection and graph visualization.

Keywords degree peeling · graph decompositions · fixed points · influence ranking

CR Subject Classification E.1 Graphs and Networks · H.2.8 Data mining · H.3.3 Clustering

James Abello
DIMACS Center, Rutgers University, Piscataway, NJ, USA
E-mail: abello@dimacs.rutgers.edu

François Queyroi
1- Sorbonne Universités, UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France
2- CNRS, UMR 7606, LIP6, F-75005, Paris, France
E-mail: francois.queyroi@lip6.fr

1 Introduction

The peeling value of a vertex v is the minimum value k at which v is removed from the network during the iterative removal of vertices of degree lower or equal to k [27]. In social networks, maximal induced subgraphs with peeling value at least k may be interpreted as some form of equilibrium for "a model of user engagement". In this scenario, "each player incurs a cost to remain engaged but derives a benefit proportional to the number of engaged neighbors" [11]. The peeling value was studied for random graphs [24] generated with the Erdös-Rényi model [16]. The maximum peeling value of a graph (also called *degeneracy*) relates to other graph theoretical measures such as the coloring number [29]) and arboricity. In [14], a peeling ordering of the vertices is used to improve the running time of an algorithm for the maximal cliques problem. Degree peeling or concepts related to it are useful in network analysis. It has been used to evaluate the relevance of communities in co-authorship networks [17]. The authors proposed a reformulation of peeling that takes into account edge weights. Some graph decompositions based on degree peeling have been used in [4] and [10] as an aid to provide *layered* visualizations of graphs. Some aspects of internet topology [12] have been addressed also in this context.

One of the interesting aspects of degree peeling is the unravelling of a network hierarchy. This hierarchy is obtained by partitioning the vertices of the network into groups according to their peeling value (in increasing order). The group with highest peeling value is called the core of the graph. The unique group that a vertex belongs to depends not only on the number of connections it has to vertices in its group but also on its con-

nections to vertices in upper groups.

Contribution: In this work we exploit the inherent locality of vertex peeling to efficiently obtain not only a partition of the vertex set but more importantly a partition of the edge set of any network. The algorithm's complexity is $\mathcal{O}(k|E|)$ where k is the maximum peeling value and $|E|$ is the number of edges.
The obtained edge partition, called here the *iterative edge core decomposition*, provides simultaneously distant and close readings of a network. It can be used to examine a network at different levels of granularity without loosing sight of the underlying vertex partition determined by the peel values. Each subset of edges, in the iterative edge core decomposition, defines a subgraph all of whose vertices have "local peeling value" = minimum subgraph degree. Equivalently, these subgraphs are *fixed points of degree peeling* (see Figure 2 for an example). We call each of them an *edge layer*. Since a vertex can be shared among different layers we use this information to record a vertex peeling profile. This profile is an indicator of the vertex role in the network. Its Shannon's entropy measures the degree of involvement in the different edge layer structures determined by peeling. We exemplify our findings on a sample of social networks.

The rest of the paper is organized as follows. Notational conventions and the basic concepts used are presented in Section 2. It also illustrates some of the main characteristics of graphs that are fixed points of degree peeling. In Section 3, we introduce the iterative-edge-core decomposition, its main properties, and an efficient algorithm to compute it. Section 4 indicates how to use the edge core decomposition to filter and analyse a network at different scales and it proposes a measure of vertex diversity based on Shannon's entropy. Applications of the proposed edge decomposition on a sample of social networks is the main subject of Section 5. We close with a discussion of possible future research directions in Section 6.

2 Peeling Values and Fixed Points

We provide here the notations and definitions used in this paper. In particular we introduce the concept of *fixed point of degree peeling* graphs. We use the term *network* interchangeably with *graph*. We concentrate on undirected graphs even though peeling based concepts are generalizable. We will discuss the case of weighted graphs later on.

In this section, we use the co-appearance network of *Les Misérables* [20] to illustrate the different concepts used (see Figure 1). The vertices correspond to characters of the novel of Victor Hugo and an edge connects two characters if they are found together in at least one chapter.

We denote by G an undirected graph with vertex set $V(G)$ and edge set $E(G)$. A partition of $V(G)$ is called a *vertex decomposition*. Similarly a partition of $E(G)$ is called an *edge decomposition*. The degree of a vertex u in G and the minimum degree are denoted by $d_G(u)$ and $d^-(G)$ respectively. The subgraph induced by a subset of vertices S is $G[S]$. For a given subset of edges $L \subseteq E$, the *layer* of G determined by L is the subgraph $G(L) = (V', L)$ where $V' = \{u \in V(G), \exists (u, v) \in L\}$.

Definition 1 (Peeling Value) The *peeling value* of a vertex $u \in V(G)$ denoted $peel_G(u)$ is the largest $i \in [1, d_G(u)]$ such that u belongs to a subgraph of G of minimum degree i. The *peeling value* of an edge $e \in E(G)$ denoted $peel_G(e)$ is the minimum peeling value of its endpoints.

In Figure 1, the peeling value of each character is mapped to the vertex color. For example, the main character "Valjean" has a peeling value of 8. The maximum peeling value of this network is 9 (red vertices).

For RAM resident graphs, the peeling value of all vertices can be computed efficiently in $\mathcal{O}(|E(G)|)$ [8]. For graphs that do not fit in RAM, an I/O efficient external-memory algorithm that computes an approximation to the peeling values has been recently proposed by [19] .

The *peeling value of* G, denoted $peel(G)$, is the maximum peel value of all its vertices. The peeling value of G is also called the *degeneracy* of G [21]. For a graph of peeling value k, its vertices can be ordered in a sequence $(v_1, \ldots, v_n)$ called the Erdős-Hajnal sequence [15] such that there are at most k edges going from v_i to $(v_{i+1}, \ldots, v_n)$. An easy but fundamental property of peel values is that they are a local manifestation of a global graph connectivity phenomenon. The following result states this precisely.

Theorem 1 ***(Peeling Value Locality [23])*** *A vertex $u \in V$ has at least $peel_G(u)$ neighbours with a peeling value at least $peel_G(u)$ and at most $peel_G(u)$ neighbours with a peeling value at least $peel_G(u) + 1$.*

The authors of the previous theorem exploit these local relations between the peeling value of a vertex and the peeling values of its neighbours to compute peeling values by a distributed algorithm.

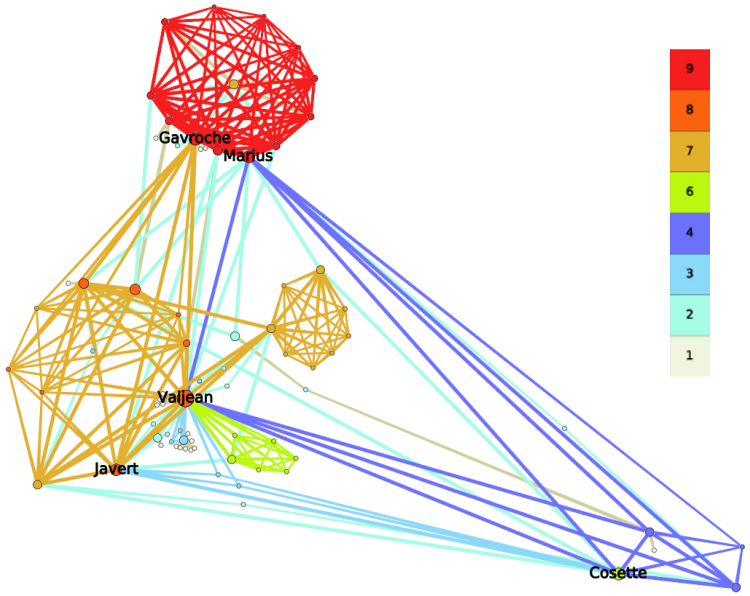


Fig. 1 The network *Les Misérables*. The peeling value is color coded. The edge coloring corresponds to our new edge decomposition (see Section 3). The 5 most diverse characters (see Section 4) are explicitly labelled.

Definition 2 (Peel Decomposition) The *vertex peel decomposition* of a graph G is the partition induced by the peeling values of the vertices of G.

Definition 3 (Graph Core) The core of G, $core(G)$, is the maximal subset of vertices of G whose peeling value is maximum, i.e. equal to the peeling value of G.

In Figure 1, the vertex peel decomposition is color coded by assigning the same color to all characters (vertices) with the same peeling value. This vertex partition contains 8 groups. The core of this network corresponds to the group of red characters.

Definition 4 (Local Peeling Values) Let $\mathcal{P}$ be a partition of $V(G)$. The *local peeling value* of a vertex $u \in G$ is $peel_{\mathcal{P}}(u) = peel_{G[P(u)]}(u)$ where $P(u)$ is the set in $\mathcal{P}$ that contains u. Similarly, if $\mathcal{L}$ is a partition of $E(G)$, the *local peeling value* of an edge $e \in E(G)$ is $peel_{\mathcal{P}}(e) = peel_{G(L(e))}(e)$ where $L(e)$ is the set in $\mathcal{L}$ that contains e.

Definition 5 (Fixed Point) A graph F is a fixed point of degree peeling k if $core(F) = V(F)$ and the peeling value of F is k. Equivalently, a graph F is a *fixed point of degree peeling* if the vertex peel decomposition of F has only one class and its peeling value is equal to its minimum degree.

Note that if F is a fixed point of degree peeling, the local peeling values of elements in F do not depend on elements with higher local peeling values. Our quest is therefore to partition the edge set of a graph G into a union of fixed points of degree peeling. Among all possible edge partitions of G into fixed points of degree peeling, the one we propose is maximal (a precise definition of maximality is given in Section 3).
For the network *Les Misérables*, the edge coloring in Figure 1 corresponds to this decomposition. Each set of edges with the same color forms a layer of the network and this layer is a fixed point of degree peeling. The subgraph determined by the brown edges corresponds for example to a fixed point of peeling value 7.

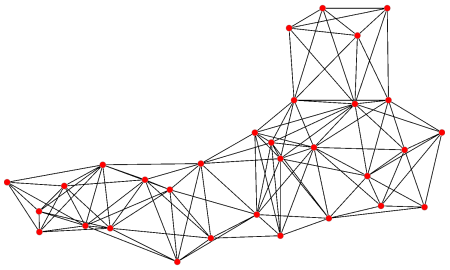


Fig. 2 An example of a random geometric subgraph in FP_5.

We denote by FP_k the class of graphs that are fixed points of degree peeling k. They are also called *strongly k-degenerate graphs* in [9]. FP_k includes well-known classes of graphs. For example, the class FP_1 corresponds to forests (without isolated vertices), cliques of size n are in FP_{n-1}, k-regular graphs are in FP_k, and one can easily exhibit less obvious graph classes (see Figure 2). For fixed points $F \in FP_k$, the peel value lo-

cality property captured by theorem 1 can be re-stated as: "a vertex $u \in V(F)$ has at least k neighbours of peeling value k". The size of the maximum clique in $F \in FP_k$ is bounded above by $k+1$. Bounds on the minimum and maximum number of edges of F are given in the following proposition.

Proposition 1 *($E(F)$ **Size**) If $F \in FP_k$ a fixed point of degree peeling with n vertices then*

$$\frac{kn}{2} \leq |E(F)| \leq kn - \binom{k+1}{2} \tag{1}$$

The lower bound of inequality (1) is the number of edges in a k-regular graph with n vertices. The upper bound is the number of edges in edge-maximal FP_k graphs with n vertices *i.e.* graphs such that an edge can not be added between two independent vertices without increasing the maximum peeling value [9]. Graphs generated using the Barabási-Albert model [6] model with a clique of size k as seed are examples of edge maximal FP_k graphs. More generally, the construction of any "edge-maximal" FP_k graph goes as follows: from a clique of size k iteratively add $(n-k)$ vertices linked to exactly k vertices. This property indicates that the average degree of an FP_k graph with n vertices is αk with $1 \leq \alpha \leq 2$. Any k-connected subgraph or connected component of an FP_k graph is a fixed point with peeling value k.

3 Decomposition into Fixed points of Degree peeling

In this section we present two different decompositions of a graph into fixed points of degree peeling. In principle, a peeling based vertex decomposition into fixed points can be obtained by first partitioning the vertex set into groups, according to their peeling values, and then recursively applying the peel decomposition to the subgraphs induced by each set in the partition. Since the peeling value can not increase one can stop the recursion when the peeling value remains the same. In other words, the recursion will end when fixed points are reached (see an example in Figure 3(a)). This divisive decomposition is just one possible partition into fixed points.

Observe however that the partition may not be maximal, in the sense that some of the obtained fixed points could be merged to obtain fixed points of higher peeling value. Notice that the same idea could be used to partition the edges. The resulting decomposition can also be non-maximal. Among all possible vertex or edge decompositions into fixed points of degree peeling, the two we propose respect the following *maximality* property.

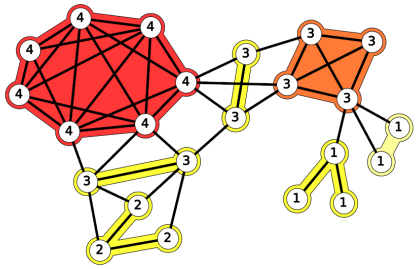
Definition 6 (Maximal FP decomposition) For a graph G, a vertex or edge decomposition $\mathcal{P}$ into fixed points is said to be *maximal* iff for any subgraph G' of G that is FP_k one of the following two conditions is met

i. $k < \max_{e \in G'} peel_{\mathcal{P}}(e)$
ii. $\forall e \in G'$, $peel_{\mathcal{P}}(e) = k$

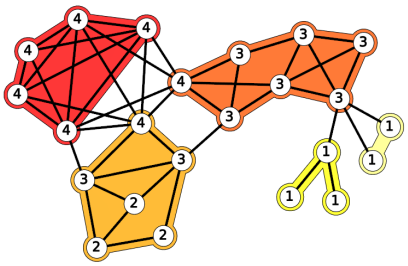
The maximality of an FP decomposition implies that the merging of layers in the decomposition will always result in a subgraph with peeling value equal to the maximum peeling value of its layers. If $\mathcal{P}$ is a maximal vertex (resp. edge) decomposition then there is no subset of vertices (resp. edges) whose local peel values can be increased without lowering the local peel values of some others vertices (resp. edges). As an illustration, in Figure 3(b), we can find a larger FP_4 by adding to the red set the two vertices labelled 4 that are not in the set. This vertex decomposition is therefore not maximal. In Figure 3(c) the vertex decomposition is maximal. For example, one could increase the local peel value of the two vertices in the fixed point of peeling value 0 to form a tree with their neighbour labelled 3. However, the local peeling value of this vertex will decrease from 3 to 1.

It is worthwhile to mention that maximal vertex decompositions have the property that any vertex with local peeling value k_i has at most $(k_j - 1)$ connections to a fixed point P_j of peeling value $k_j > k_i$. Similarly, in maximal edge decompositions, any edge in a layer with peel value k_i have at least one of its endpoints with at most $(k_j - 1)$ connections to vertices in a layer with peel value $k_j > k_i$. This is a local peeling edge analogue to the local peeling vertex property stated in Theorem 1.

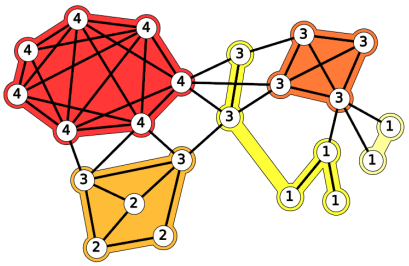
A simple and efficient approach to obtain a maximal partition into fixed points is to iteratively remove the core vertices and all its connections or alternatively remove just the edges with both end points on the core. In the first case we obtain a *vertex partition* into maximal fixed points and in the second case we obtain a novel *edge partition* into fixed points.
In both cases, we are iteratively peeling vertices or edges in the core starting with the initial graph core. We refer to both of these methods as "backward peeling". Backward vertex peeling produces what we call an "iterative vertex core decomposition" and backward edge peeling produces our desired "iterative edge core decomposition". These methods are formally stated below as Algorithm 1 and 2. Their complexity is $\mathcal{O}(k|E|)$ where k is the maximum peeling value of G. Their correctness follows directly from the first principle properties



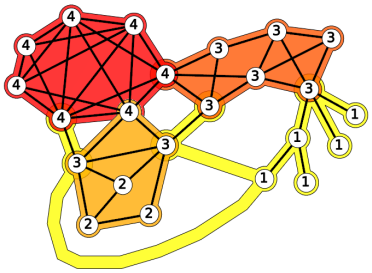
(a) Recursive peeling vertex decomposition (non-maximal).



(b) Another non-maximal vertex decomposition.



(c) Iterative Vertex Core decomposition



(d) Iterative Edge Core decomposition

Fig. 3 Four different decompositions of a graph into fixed points of degree peeling. The induced subgraphs formed by taking all vertices in a hull is a fixed point of degree peeling. Pale yellow represents the lowest peeling value 0 and red represents the highest peeling peeling value 4. Vertices are labelled according to their peeling value. The decomposition a) is not maximal since the subgraph induced by the union of the yellow sets has a peeling value of 2. b) is not maximal since two vertices labelled 4 have 4 connections to a FP_4 fixed point.

of the peel values stated in Section 2. Since the main focus of this paper is the edge partition into fixed points, we discuss further the properties of the iterative edge core decomposition although similar statements can be proved for the iterative vertex core decomposition. Figure 3(c) and 3(d) illustrate the differences between the iterative vertex core decomposition and the iterative edge core decomposition. We present the generalisation of the later method to weighted graphs in Section 3.3.

Algorithm 1: Iterative Vertex Core decomposition of G.

Input: $G = (V, E)$
Output: $\mathcal{C} = (C_1, \ldots, C_l)$, each C_i is a fixed point.

```
G' ← G;
C ← ∅;
while V(G') > 0 do
    C ← C ∪ {core(G')};
    G' ← G'[V(G') − core(G')];
end
return C;
```

3.1 Iterative Vertex Core Decomposition via Backward Peeling

Algorithm 1 computes a vertex decomposition of G into fixed points of degree peeling. It relies on the fact that, for any graph G, $core(G)$ is a fixed point of degree peeling. After the removal of $core(G)$, the peeling value of the remaining vertices will directly drop if they were connected to the core. This operation can affect other vertices due to the iterative computation of peeling values. This means that in each obtained fixed point F of peeling value k, all vertices in F have a local peeling value lower or equal to their global peeling value in G.

Notice that this iterative vertex core decomposition discards the connections between the different groups in the graph. This is one of the main reasons we introduce the following iterative edge core decomposition.

3.2 Iterative Edge Core Decomposition via Backward Peeling

The iterative edge core decomposition (see Algorithm 2) assigns to each edge a value that corresponds to the peeling value of its endpoints at the first time they belong to the core. In the example given in Figure 3(d), removing the edges within the red hull leaves most of the vertices in the core isolated. Three of them have connections to the rest of the graph. The leftmost one has actually enough connections to be part of the next

core (of peeling value 3) but after that it is also isolated. The idea here is that all the vertices that belong to a fixed point will not be similar, in the sense that some of them can actually be part of other fixed points.

Algorithm 2: Iterative Edge Core decomposition of G.

Input: $G = (V, E)$
Output: $\mathcal{L} = (L_1, \ldots, L_p)$, each L_i is a fixed point.
$G' \leftarrow G$;
$\mathcal{L} \leftarrow \emptyset$;
while $E(G') > 0$ **do**
  $A = \{(u, v) \in E(G'), u \in core(G') \wedge v \in core(G')\}$
  $\mathcal{L} \leftarrow \mathcal{L} \cup \{A\}$;
  $E(G') \leftarrow E(G') - A$;
end
return $\mathcal{L}$;

The maximum number of iterations of Algorithm 2 is bounded by k. Indeed, removing the edges of the core reduces the peeling value by at least 1. The size of the resulting partition is therefore at most k. The peeling value of vertices is computed at each iteration and this operation can be done in $\mathcal{O}(|E(G)|)$. This decomposition is maximal according to Definition 6 (see Theorem 2).

Theorem 2 *For a non-empty graph G, the edge decomposition $\mathcal{L}$ computed by Algorithm 2 is maximal (see Definition 6).*

Proof Let $\mathcal{L} = (L_1, L_2, \ldots, L_p)$ denote the ordered edge decomposition computed by Algorithm 2. If $p = 1$ then that decomposition is maximal since $L_1 = G[core(G)]$ is a fixed point. Indeed, there is not subgraph of L_1 of higher peel value than the peel value of L_1 and every edge in L_1 have the same peel value.

Assume the result is true for edge decompositions $\mathcal{L}_k$ in $k \geq 1$ layers. We will argue the maximality for edge decompositions with $k + 1$ layers. If we assume now that $\mathcal{L}_{k+1}$ is not maximal according to Definition 6 then it must exist a subgraph $G' \in FP_j$ such that $j \geq max_{e \in E(G')} peel_{\mathcal{L}_{k+1}}(e)$ and $peel_{\mathcal{L}_{k+1}}(e) \neq j$ for some $e \in E(G')$.

We argue first that $E(G') \cap G[core(G)] = \emptyset$. Indeed, the subgraph $G[core(G)]$ has the maximum peeling value therefore we would have $j = peel(G)$. However, it does not exist an edge $e \in G[core(G)]$ with $peel_{\mathcal{L}_{k+1}}(e) < j$. Moreover, since $core(G)$ is the maximal subgraph of edges of maximum peel value j, for every $e \in (G - G[core(G)]$, $peel_{\mathcal{L}_{k+1}}(e) < j$. So $E(G') \cap G[core(G)] = \emptyset$ therefore the subgraph G' is actually a subgraph of $(G - G[core(G)])$. The edge decomposition of $(G - G[core(G)])$ has strictly less layers than the edge decomposition of G because the core of G is non-empty. Therefore, the edge decomposition of $(G - G[core(G)])$ produced by the algorithm is maximal. This edge decomposition plus $G[core(G)]$ is the maximal edge decomposition of G produced by the algorithm.

The iterative edge core decomposition is maximal. Notice that this is only true for edge decompositions into fixed points with the same local peeling values that our decomposition. Therefore every maximal decomposition could be obtained by partitioning the edges of each layer into fixed points of the same peeling value. In other words, every maximal edge decomposition is a refinement of the iterative edge core decomposition.

3.3 Generalisation to weighted graphs

The concept of peeling value can be adapted for graphs whose edges are weighted [17]. Suppose we have a graph $G = (V, E, w)$ where w assigns a positive real value to every edge. We define the weighted degree of a vertex u denoted $d_w(u)$ as the sum of the weights of the edges adjacent to u. In this case, the weighted peel value of a vertex u, denoted $peel_w(u)$, is the maximum k in $[0, d_w(u)]$ such that u belongs to a subgraph of G with minimum weighted degree k.

The concepts detailed in section 2 and 3.2 are naturally extensible to weighted graphs. In particular, the Algorithm 2 can be modified to select the weighted core of the graph at each iteration. This leads to an edge partition of the graph into fixed points of weighted degree peeling.
Some of the fixed points properties we detailed earlier do not hold when taking edge weights into account. In particular, a fixed point of weighted peeling value k can contain only one edge with $w(e) = k$. This affects the complexity of Algorithm 2. Indeed, the degeneracy of a weighted graph is unbounded according to our definitions. However, for non-empty graph G, since the core of G contains at least one edge the number of layers in the final decomposition is at most $|E|$.

Notice that the proposed decomposition of $G = (V, E, w)$ will be the same as that of $G = (V, E)$ in the case that w assigns the same value α to each edge. Indeed, in this situation the peeling value of edges is simply multiplied by α. In practice (see experiments in Section 5.4 and 5.5), we observe that the weighted iterative edge core decomposition produces more layers than the unweighted decomposition.

4 Uses of the Iterative Edge Core Decomposition

In this section, we indicate how to use the edge core decomposition to filter and analyse a network at different scales. We also propose a measure of vertex diversity based on Shannon's entropy.

4.1 Network Analysis at Different Scales

The iterative edge decomposition focuses on local peel values and since it is maximal each edge gets assigned to its highest possible layer. Each layer locality is captured by the fact that it is a fixed point of degree peeling. The usual peel decomposition fails to incorporate locality since the vertices of peeling value k could very well form an independent set. The peel decomposition tends to produce more layers than our iterative edge core decomposition.

The differences between the two decompositions can be expressed using the following metaphor: The network hierarchy according to peeling can be viewed as a terrain. In this case, a "*plateau*" (or tableland) could be seen as an intermediate level *i.e.* an almost flat area that would be qualified as top if we discard everything above. In a network, this would correspond to subset of vertices with different peeling values but high local peeling values.
The k-cores decomposition will go through this landscape from the top to the bottom meter per meter. The process will reach every mountains tops but it will also go through everything in between. Some post-process are required to differentiate a plateau or a mountain top from any other part of the mountain at the same height. The iterative edge core decomposition follows a top-down approach. Its computation jumps from the overall maximum to subsequent levels of local "plateaux". They would therefore be easier to identify at the end. Note however, that this process will discard an important information which is the height of every part of the mountains (the peeling value of the vertices). For this reason, we shall keep track of this information during the analysis.

4.2 Network Filtering and Community Structure

Peeling values can be used to filter out vertices with few connections. In the iterative edge core decomposition, vertices of peeling value k are present in layers with peeling value at most k. The lowest layer L_p may contain vertices of peeling value 1 but also their sparse connections between vertices from layer p or above. As an illustration, consider a network formed by two cliques of different sizes linked by an edge. This bridge edge will fall into the lowest layer of the decomposition. Some layers can be filtered out to make the community structure (if any) more apparent. Even though there is no universally accepted notion of communities, the class of fixed points may include some "patterns" that are intuitively accepted as communities in some other works [26, 22]. This is illustrated on a real-world example in Section 5.2. However, it is worth to notice that our method is not directly aimed at finding the community structure of a network.

By associating with each vertex the first time that it appears in the iterative edge core decomposition, we can keep track of the proportion of recently added vertices to any layer. Sudden proportion changes between consecutive levels are an indicator of a possible community structure. Notice that the iterative edge core decomposition is more likely to detect overlapping communities than the vertex peel decomposition since edge partitioning inherently allows vertex overlaps between communities [3].

4.3 Assessing Vertices Diversity

For a given vertex u, the edges adjacent to u in G can be partitioned into different classes given by the iterative edge core decomposition. We associate with each vertex a *profile vector* containing its peeling information according to the iterative edge core decomposition.

Definition 7 For a graph G and its iterative edge core decomposition $\mathcal{L} = (L_1, \ldots, L_p)$, the *profile* of a vertex $u \in V(G)$ denoted $profile(u)$ is a sequence of integers $(l_1, \ldots, l_p)$ where each l_i indicates the number of edges adjacent to u that are part of the layer L_i.

Notice that the number of times a vertex u is detected as part of a layer, in the iterative edge core decomposition, corresponds to the number of non-zero entries in the vector $profile(u)$. The sum of the entries in the profile vector of u is equal to the degree of u. Vertex profiles are used next to assess the *diversity* of a vertex *i.e.* how the degree of a vertex is evenly distributed across the layers.

Definition 8 (Vector Majorization [5], Shannon's Entropy [28] and Vertex Diversity) For a vector u in R^k, let $p(u) = (p_1(u), p_2(u), \ldots, p_k(u))$ denote the vector obtained by sorting the entries of u from largest to smallest. A vector v in R^k is said to be majorized

by a vector u in R^k iff for $1 \leq l < k$, $\sum_{j=1}^{l} p_j(v) \leq \sum_{j=1}^{l} p_j(u)$ and $\sum_{j=1}^{k} p_j(v) = \sum_{j=1}^{k} p_j(u)$.

Let $H(profile(u))$ be the Shannon's entropy of the profiles normalized by the degree.

$$H(profile(u)) = -\sum_{i=1}^{p} \frac{l_i}{d_G(u)} \log_2 \left(\frac{l_i}{d_G(u)} \right) \quad (2)$$

For two vertices u and v, if $profile(v)$ is majorized by $profile(u)$ then $H(profile(v)) \geq H(profile(u))$ since $H(.)$ is a Schur-concave function. Therefore, we can rank vertices using the entropy of their profiles. We call this measure vertex diversity. Namely, the *diversity of vertex* u is $H(profile(u))$. The diversity of a vertex does not solely depends on its peeling value or its highest layer in the iterative edge core decomposition. A vertex that is not part of the core of the graph can still have a bigger diversity than vertices from the core.

This measure can also be defined in the case of the weighted edge core decomposition introduced in Section 3.3. In this case, the diversity denoted H_w captures how the total weight of the edges adjacent to a vertex is uniformly spread among the different layers.

5 Network Samples

We illustrate the application of the iterative edge core decomposition on five networks with different characteristics. The last two can be viewed as weighted networks. We use node-link diagrams generated using Tulip software [1].In most cases, the peeling values of the vertices are color coded. The color of an edge correspond to the peeling value of its layer in the iterative edge core decomposition. Both vertices and edge values use the same color scale. The diversity of a vertex (see Eq. 2) is encoded by the size of the vertex.
We also provide 3D *z-ordered* visualizations of the networks by mapping into the z-axis the edge core decomposition numbers. This method is inspired by [10].

5.1 A Co-Appearance Network: *Les Misérables*

We start with the co-appearance network of the novel *Les Misérables* (see Section 2). The network contains 77 vertices and 254 edges. The peel decomposition contains 8 groups. The core of the graph (red vertices) corresponds to the "revolutionary student club" appearing during the Paris uprising in the novel. While some of them are very important in the novel (like "Marius"), most of them are not and the reason they have the maximum peeling value is because of the size of the group. We can differentiate them by looking at their connections to vertices of lower peeling value and this is exactly what the iterative edge core provides (see Figure 1).

Table 1 Statistics for the iterative edge core decomposition of the Manufacture network.

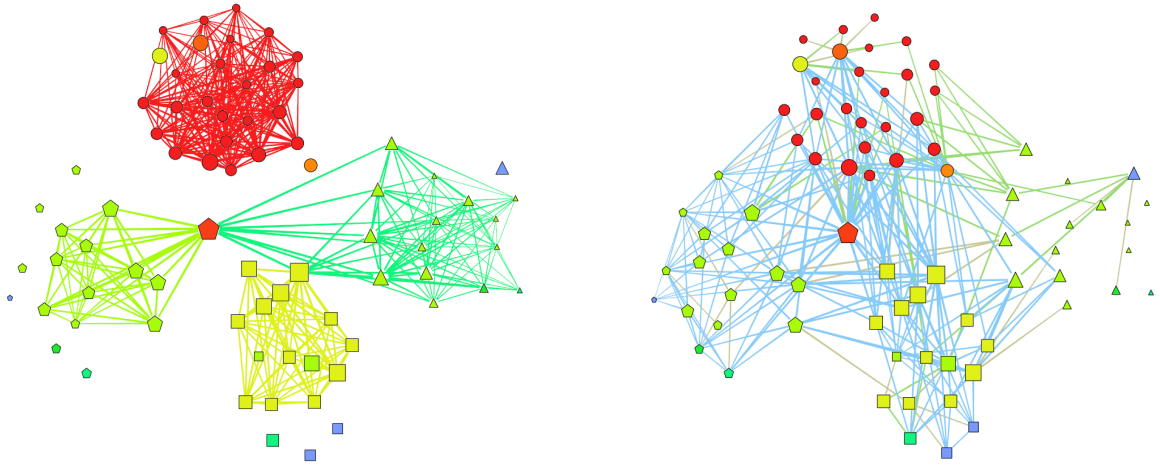
Layers	# Vertices	% New vertices	Clust. Coef.
16	24	100 %	0.89
11	12	100 %	1
10	11	100 %	1
8	17	88 %	0.80
5	45	31 %	0.23
2	28	4 %	0.03
1	38	0 %	0

The second layer of the iterative edge core decomposition (brown edges) contains characters such as the Thenardier family. Their son, "Gavroche", was part of the core but he actually has enough connections to be grouped with them at this level. Each layer seems to correspond to a community in the novel. For example, the blue edges layer determines a subgraph with 6 vertices, its density is maximum as fixed point of peeling value 4 according to Property 1. It corresponds to the group formed by Marius along with members of his family, his fiance "Cosette" and the tutor of Cosette, "Valjean".

The labels displayed in Figure 1 correspond to the names of the five most *diverse* characters: Valjean, Gavroche, Cosette, Marius, and Javert. Notice that Cosette which is a main character in the novel appears here even if her peeling value is relatively low when compared to the others. However, in the iterative edge core decomposition, she is well connected in the layers she belongs to.

5.2 A Social Interaction Network: The Manufacture Network

The example we now discuss is an intra-organizational network [13] where the vertices represent 77 employees and an edge links two employees when one of them indicates that the other provides him useful information (at least somewhat infrequently). The employees work in four different locations: Paris, Frankfurt, Warsaw and Geneva (see Figure 4).



(a) Highest four layers. (b) Lowest three layers.

Fig. 4 Results of the iterative edge core decomposition for the Manufacture network. The graph is drawn using a force-direct layout algorithm. The positions between a) and b) are preserved. The shape of the vertices corresponds to the different locations: Frankfurt (circles), Paris (squares), Geneva (pentagons) and Warsaw (triangles). The edges of the graph are separated here according to the decomposition. The union of a) and b) gives the complete network.

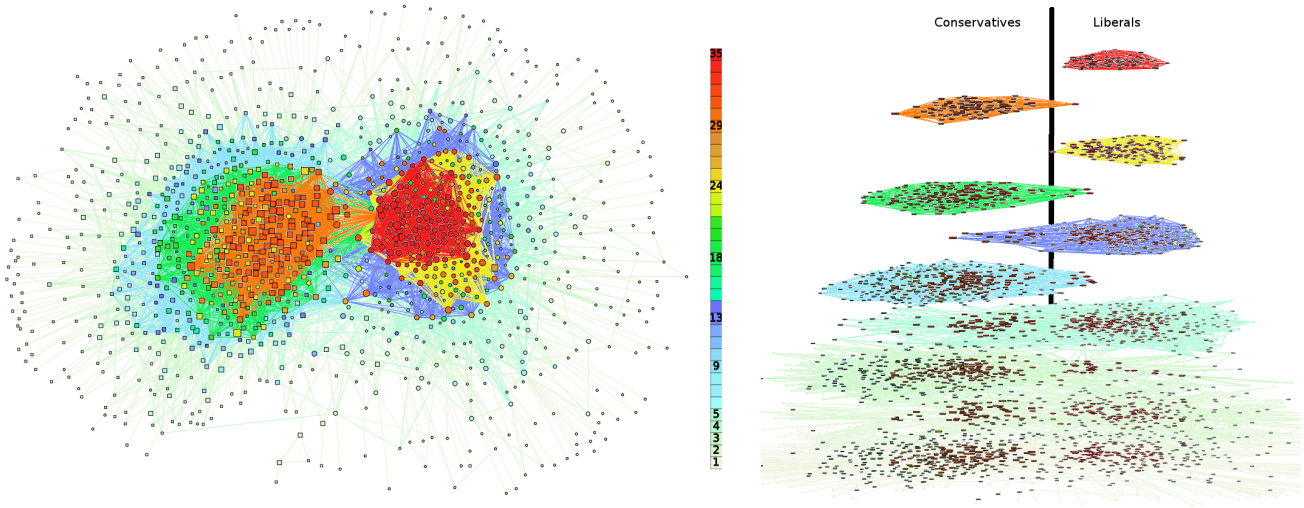
This example illustrates the usefulness of our decomposition for graph filtering (see Section 4). Observe a transition between the layers 8 and 5 in terms of their proportion of new vertices found and their average clustering coefficient (see Table 1). The first four layers (see Figure 4(a)) separate the core of the communities induced by the locations even if they have different global peeling value. Two vertices in the center have also enough connections to be part of the layer of peeling value 8 that groups vertices from Warsaw. Notice that those vertices have also a high diversity. If we assume the community structure is given by the locations, our decomposition is able to extract "communities cores" since they have different connectivity characteristics. Notice also that they form dense subgraphs. The vertices that are isolated in those layers correspond to vertices whose number of connections is too low or is too split between the different locations.

The last three layers (see Figure 4(b)) also bring relevant information. The blue edges determine a fixed point of peeling value 5. This subgraph contains a substantial number of vertices from higher layers. It suggests that even without the connections between people from the same location, the graph structure still allows the diffusion of information in the company. Notice that most of the employees from Warsaw do not belong to this subgraph. This suggests that connectivity is "stronger" between people working in Frankfurt, Paris and Geneva.

5.3 The Political Blogosphere Network

This network represents the undirected links between political blogs before the 2004 US election [2]. The 1222 blogs are divided into two groups: liberals or conservatives (see Figure 5). The peel decomposition contains 35 clusters (the maximum peeling value in the network). The core of the network contains liberals blogs, vertices of peeling value between 32 and 34 are other liberal blogs connected to the core. A substantial change happens when looking at vertices with peeling value 31 since a large proportion of them are conservative blogs. Subsequent groups in the peel decomposition contain both liberal and conservatives blogs.

On the other hand, a quite different picture emerges using the 10 layers of the iterative edge core decomposition of this network (see Figure 5(a)). Indeed, either liberal or conservative blogs are over-represented in the layers detected (see the statistics provided in Table 2). A *3D z-oderered* visualization (see Figure 5(b)) reveals this phenomenon. Notice however, that an edge layer can contain blogs from the "opposite" side. This suggests that the local peeling values of the blogs mostly come from blogs with the same affiliation. Observe that the average clustering coefficient of the layers decreases heavily. This can be explained by the fact that each layer contains an important proportion of vertices already detected in the previous layers.



(a) Plain view obtained using a force-direct algorithm.

(b) 3D z-ordered view.

Fig. 5 Results of the iterative edge core decomposition for the Political Blogs network. The graph is drawn using a force-direct layout algorithm. The shape of the vertices corresponds to the politic orientation: liberal (circle) or conservative (square). The labels on the color scale indicate the peeling values of the edge layers found.

Table 2 Statistics for the iterative edge core decomposition of the Political Blogs network.

Layer	**# Blogs**	**% Liberals**	**% New Blogs**	**Clustering Coefficient**
33	66	100 %	100 %	0.72
29	95	4 %	99 %	0.5
24	120	99 %	68 %	0.26
18	186	5 %	65 %	0.14
13	206	93 %	44 %	0.09
9	315	6 %	55 %	0.05
5	383	77 %	42 %	0.03
3	382	2 %	35 %	0.02
2	373	64 %	34 %	0.02
1	706	45 %	25 %	0

Table 3 Statistics for the first ten diverse cities in the airport network.

City	H_w (Rank)	deg_w	Occurrences$_w$	First layer$_w$	H (Rank)	deg	Occurrences	First layer
Paris	3.4 (1)	3925	22	370	2.92 (2)	368	9	36
Sao Paulo	3.37 (2)	1167	16	170	2.73 (6)	108	7	22
London	3.28 (3)	4635	22	370	2.82 (4)	407	9	36
Papeete	3.27 (4)	339	11	42	1.52 (122)	36	3	4
Tokyo	3.25 (5)	1529	13	273	2.53 (10)	125	7	22
Jeddah	3.18 (6)	1407	16	170	2.92 (1)	144	8	22
Stockholm	3.15 (7)	1781	16	370	2.4 (17)	172	7	36
Santiago de Chile	3.14 (8)	723	14	138	2.33 (22)	72	6	14
Frankfurt	3.11 (9)	3530	17	370	2.82 (3)	331	9	36
New York	3.08 (10)	2318	13	370	2.8 (5)	199	9	36

5.4 The Air Transport Network

The Air transport network is an undirected graph where each vertex represents the airports of a city and edges represent a direct flight from one city to the other [25]. The network contains 1490 nodes and 12353 edges. The size of the peel decomposition is equal to the maximum

peeling value in this network which is 36. However, the iterative edge core decomposition provides a partition of the edges into 12 layers (see Figure 6(a)). European cities appear in many layers although the third and sixth layers (of peeling value 17 and 8) contain a majority of Asian cities. The fact that the local peeling value is high between those airports is difficult to spot using the peel decomposition.

In this case, we notice two interesting aspects when looking at the results of the iterative edge core decomposition. The backward peeling procedure leaves a few cities isolated at each step. As a case in point, the network remaining after the removal of all edges of local peeling value greater than 4 contains a giant connected component with 1380 airports.
A second interesting aspect is observed when looking at airports diversity (see Table 3). The most diverse airport is Jeddah in Saudi Arabia. This may because this city has a distinctive role being very close to the Islam's holy city of Mecca. The next most diverse airports are: Paris, Frankfurt, London, New York City, Sao Paulo and Beijing.

We illustrate here the generalisation of our method for the analysis of edge-weighted networks. We use the natural extension of the iterative edge core decomposition described in section 3.3. As edge weight, we take the rounded logarithmic transform of the number of passengers. This is done to consider the magnitude of a link, the distribution of the number of passengers is indeed heavily skewed even if the data contain only the connections taken by at least 500 people in the year 2000.
This weighted edge core decomposition contains 46 layers. Looking at Figure 6 we can see that the upper edge layers of the weighted and unweighted decompositions are very similar. This similarity indicates that the distribution of the weights and the topology of the network are intricate. It can be explained by the fact that the density of airline connections in an area is somehow correlated to the traffic within this area.

We also computed the diversity of each city taking the weights into account. The top 10 cities are shown in Table 3. Some big cities are still present in this list (Paris, London, Frankfurt, New York). The city of Jeddah is also here even if its rank is lower. The diversities of Sao Paulo, Santiago de Chile and Papeete (French Polynesia) are surprisingly high. The case of Papeete illustrates well the information gained by using the weights of the connection. The airport is actually highly central and the other airports in this area (that is composed by a large amount of small islands) have only few connections. The high diversity of this airport suggests however that its impact in the region is comparable to its impact outside.

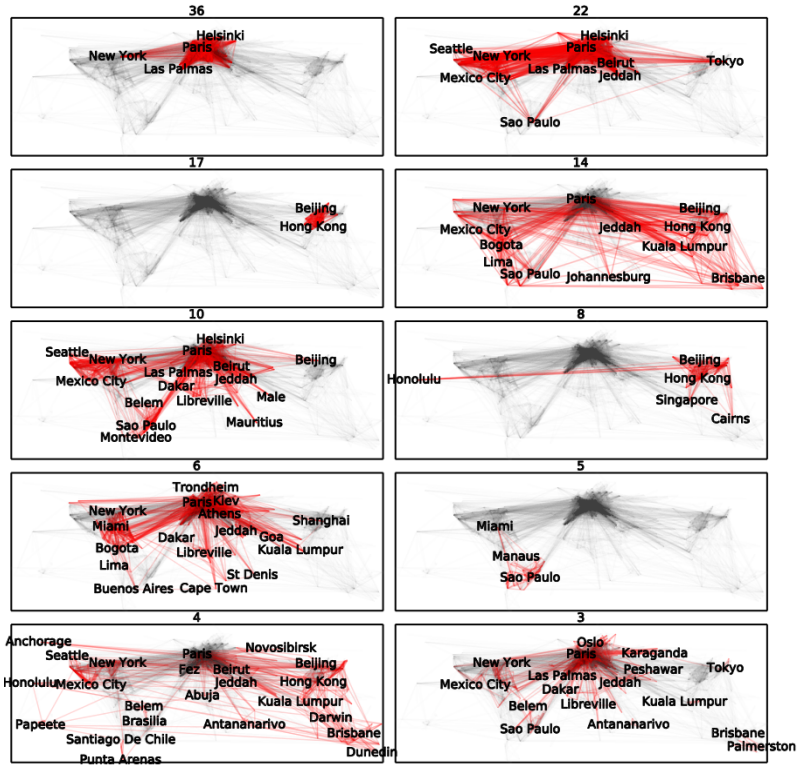
Table 4 Statistics for the first ten diverse authors in the collaboration network.

Name	H_w	deg_w	Occur_w	1st layer_w
R. E. Tarjan	3.29	151	12	21
K. Mehlhorn	3.21	233	12	32
D. T. Lee	3.11	138	10	23
M. Sharir	3.1	697	15	99
J. S. Vitter	3.1	138	11	38
J.-R. Sack	3.03	184	12	28
J. O'Rourke	3.01	222	11	38
J. Urrutia	3.01	209	11	38
R. Tamassia	2.99	388	12	56
D. P. Dobkin	2.96	163	10	38

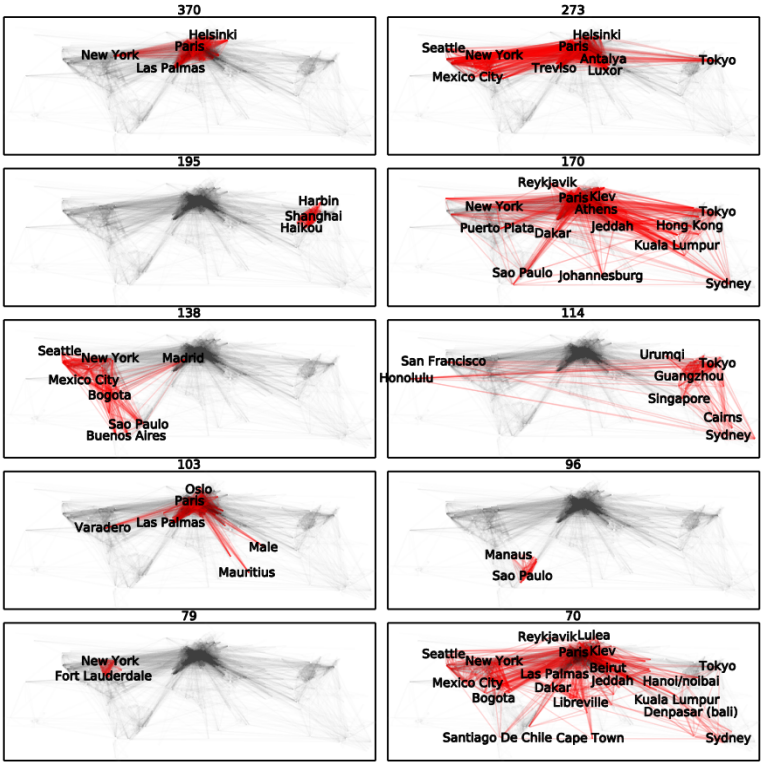
5.5 Co-authorship network

We now illustrate our iterative edge core decomposition in the case of a co-authorship network among computer scientists that have published in the field of computational geometry [7]. This network was extracted from the Computational Geometry Database bibliography. The edges in the network are weighted by the number of common published works (articles, books etc.). We focus here on the biggest connected component which contains a total of 3621 authors, 9461 relations and the mean weighted degree is 10.9.

The weighted edge core decomposition contains 34 layers while the vertex weighted peel decomposition has 57 vertex sets. A plain view of the decomposition as well as the z-ordered 3D view of all layers can be found in Figure 7. The highest layers may contain only few authors. For example, the second, fifth and sixth layers contain only one edge. Some layers exhibit some interesting common features. For instance, the seventh layer contains mostly Italian authors. The reader may be curious about commonalities in other layers. When looking at the weighted diversity of the authors (see Table 4), we observe a phenomenon similar to the airport network. The authors with the highest rank in diversity are not necessarily those with the highest peel value or who occur in the first layers. In this case, it is R. E. Tarjan which is ranked first.

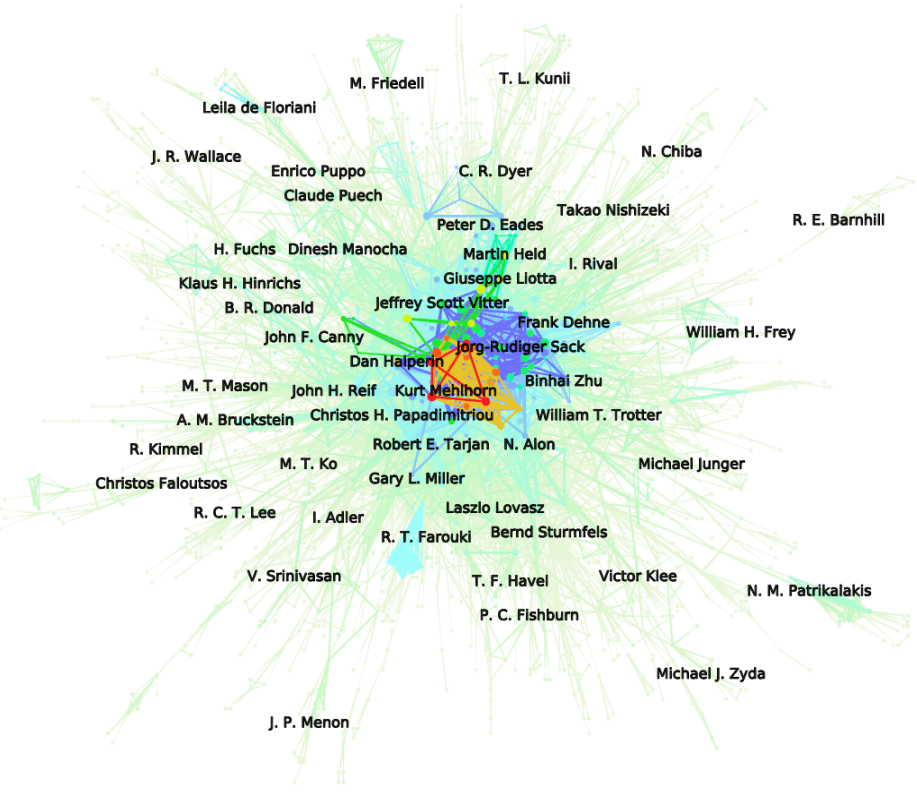


(a) Unweighted



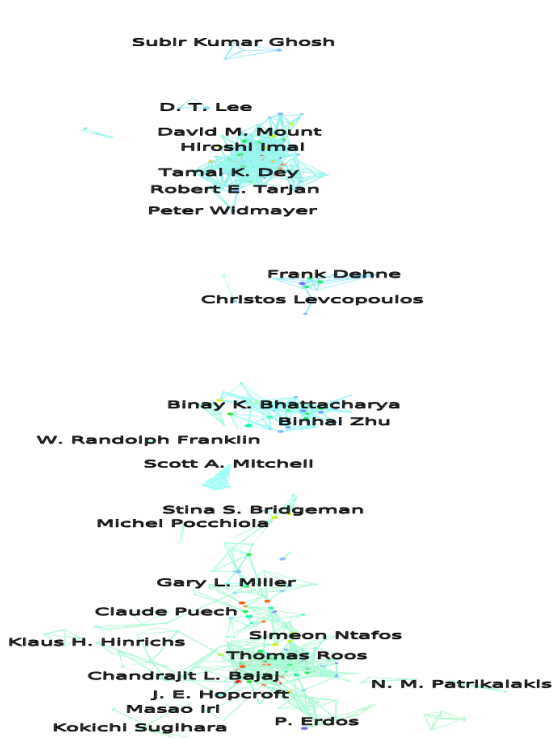
(b) Weighted

Fig. 6 Small multiples view of the first ten layers of (a) the iterative edge core decomposition and (b) the weighted decomposition of the Airport network. Each box contains a small view of the network and corresponds to an edge layer (red edges). The peeling value of the layer is indicated at the top of each box. The vertices coordinates correspond to the geographical positions to the airports.

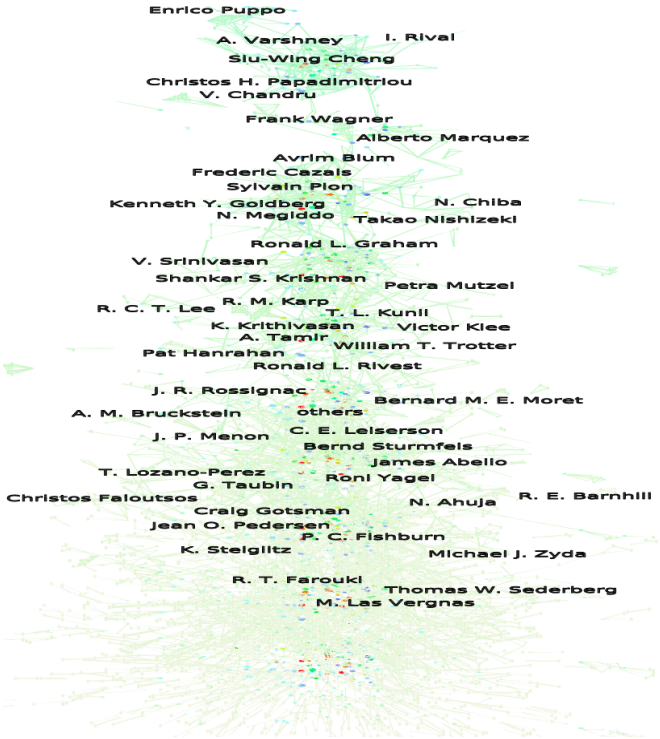


(a) Plain view obtained using force-directed algorithm

(b) First 12 layers (from peel 99 to 28)



(c) Next 11 layers (from peel 24 to 12)



(d) Last 11 layers (from peel 11 to 1)

Fig. 7 Results of the iterative edge core decomposition for the collaboration network in computational geometry. Only the names of the 10% most diverse authors are shown. The whole z-ordered 3D view is here split into three part (Figures 7(b), 7(c) and 7(d)).

6 Conclusions

We introduced an efficient graph edge partition into fixed points of degree peeling. Each layer in the decomposition has a unique local peel value. We used a procedure called "backward peeling" to produce a partition where the local peeling values are maximal. This partition is unique and the complexity of the algorithm is $\mathcal{O}(k|E|)$. The presented algorithms and techniques are generalized to weighted networks. Information from the decomposition allowed us to formulate a novel notion of vertex diversity with an associated measure based on Shannon's entropy. We illustrated graph filtering and analysis at different scales using 3D z-ordered node-link diagrams. We illustrated the usefulness of our decomposition on a variety of networks.

We are currently studying the mathematical properties of fixed points of degree peeling. which may be useful in the understanding of some fundamental graph streaming computations. The concept of graph peeling can be extended in different directions. For example, in directed graphs, one can peel the graph according to incoming degrees or outgoing degrees [18]. We also want to investigate the use of our decomposition as a preprocessing step for a variety of graph drawing algorithms and visualisations.

Acknowledgment

This work is supported by the Request and CODDDE (ref. ANR-13-CORD-0017) grants from the Agence Nationale de la Recherche, DIMACS and mgvis.com.

References

1. Tulip software. URL `http://www.tulip-software.org`
2. Adamic, L.A., Glance, N.: The political blogosphere and the 2004 us election: divided they blog. In: Proceedings of the 3rd international workshop on Link discovery, pp. 36–43. ACM (2005)
3. Ahn, Y.Y., Bagrow, J.P., Lehmann, S.: Link communities reveal multiscale complexity in networks. Nature **466**(7307), 761–764 (2010)
4. Alvarez-Hamelin, J., Dall Asta, L., Barrat, A., Vespignani, A.: Large scale networks fingerprinting and visualization using the k-core decomposition. Advances in neural information processing systems **18**, 41 (2006)
5. Arnold, B.C.: Majorization and the Lorenz order: a brief introduction, vol. 43. Springer-Verlag Berlin (1987)
6. Barabási, A., Albert, R.: Emergence of scaling in random networks. science **286**(5439), 509–512 (1999)
7. Batagelj, V., Zaversnik, M.: Collaboration network in computational geometry (2002). URL `http://vlado.fmf.uni-lj.si/pub/networks/data/collab/geom.htm`
8. Batagelj, V., Zaversnik, M.: An o (m) algorithm for cores decomposition of networks. arXiv preprint cs/0310049 (2003)
9. Bauer, R., Krug, M., Wagner, D.: Enumerating and generating labeled k-degenerate graphs. In: 7th Workshop on Analytic Algorithmics and Combinatorics (ANALCO), pp. 90–98 (2010)
10. Baur, M., Brandes, U., Gaertler, M., Wagner, D.: Drawing the as graph in 2.5 dimensions. In: Graph Drawing, pp. 43–48. Springer (2005)
11. Bhawalkar, K., Kleinberg, J., Lewi, K., Roughgarden, T., Sharma, A.: Preventing unraveling in social networks: the anchored k-core problem. Automata, Languages, and Programming pp. 440–451 (2012)
12. Carmi, S., Havlin, S., Kirkpatrick, S., Shavitt, Y., Shir, E.: A model of internet topology using k-shell decomposition. Proceedings of the National Academy of Sciences **104**(27), 11,150–11,154 (2007)
13. Cross, R.L., Parker, A.: The hidden power of social networks: Understanding how work really gets done in organizations. Harvard Business Press (2004)
14. Eppstein, D., Löffler, M., Strash, D.: Listing all maximal cliques in sparse graphs in near-optimal time. CoRR **abs/1006.5440** (2010)
15. Erdős, P., Hajnal, A.: On chromatic number of graphs and set-systems. Acta Mathematica Hungarica **17**(1), 61–99 (1966)
16. Erdös, P., Rényi, A.: On the evolution of random graphs. Publ. Math. Inst. Hungar. Acad. Sci **5**, 17–61 (1960)
17. Giatsidis, C., Thilikos, D., Vazirgiannis, M.: Evaluating cooperation in communities with the k-core structure. In: Advances in Social Networks Analysis and Mining (ASONAM), pp. 87–93. IEEE (2011)
18. Giatsidis, C., Thilikos, D.M., Vazirgiannis, M.: D-cores: measuring collaboration of directed graphs based on degeneracy. In: 2011 IEEE 11th International Conference on Data Mining (ICDM), pp. 201–210 (2011)
19. Goodrich, M.T., Pszona, P.: External-memory network analysis algorithms for naturally sparse graphs. CoRR **abs/1106.6336** (2011)
20. Knuth, D.E.: The Stanford GraphBase: a platform for combinatorial computing. AcM Press (1993)
21. Lick, D.R., White, A.T.: k-degenerate graphs. Canad. J. Math **22**, 1082–1096 (1970)
22. Mann, C.F., Matula, D.W., Olinick, E.V.: The use of sparsest cuts to reveal the hierarchical community structure of social networks. Social Networks **30**(3), 223–234 (2008)
23. Montresor, A., Pellegrini, F.D., Miorandi, D.: Distributed k-core decomposition. CoRR **abs/1103.5320** (2011)
24. Pittel, B., Spencer, J., Wormald, N., et al.: Sudden emergence of a giant k-core in a random graph. Journal of combinatorial theory. Series B **67**(1), 111–151 (1996)
25. Rozenblat, C., Melançon, G., Koenig, P.Y.: Continental integration in multilevel approach of world air transportation (2000-2004). Networks and Spatial Economics (2008)
26. Schaeffer, S.E.: Graph clustering. Computer Science Review **1**(1), 27–64 (2007)
27. Seidman, S.: Network structure and minimum degree. Social networks **5**(3), 269–287 (1983)
28. Shannon, C.E.: A mathematical theory of communication. ACM SIGMOBILE Mobile Computing and Communications Review **5**(1), 3–55 (2001)
29. Szekeres, G., Wilf, H.S.: An inequality for the chromatic number of a graph. Journal of Combinatorial Theory **4**(1), 1–3 (1968)