

Multipartite Encodings of Complex Networks

Matthieu Latapy¹ Thi Ha Duong Phan² Christophe Crespelle¹ Thanh Qui Nguyen³

Abstract

It appeared recently that most real-world complex networks have some properties in common which make them very different from models previously used. As a consequence, an intense activity is nowadays devoted to the definition of models which capture these properties. Among the most promising ones, it has been proposed to encode complex networks into bipartite graphs. However, this encoding has severe limitations. We explore here the possibility to go beyond these limitations by introducing a multipartite encoding of complex networks. Several definitions are possible, each leading to a new and interesting combinatorial object. The most immediate one produces infinite encodings, which is unsatisfactory. By refining it, we obtain a definition for which we prove convergence. In between, we propose a definition for which no convergence proof is available yet.

1 Introduction

It appeared recently [21, 1, 20, 9] that most real-world complex networks (like the internet topology, data exchanges, web graphs, social networks, or biological networks) have some non-trivial properties in common. In particular, they have a very low density, low average distance and diameter, an heterogeneous degree distribution, and a high local density (usually captured by the clustering coefficient [21])

Complex network models used until then (mainly random, complete or regular graphs, and ad hoc models) do not fit these properties. Though, it has been proved that they have a strong impact on various phenomena of interest like network robustness, spreading phenomena, and protocol or algorithm design [2, 7, 6, 8, 19, 18, 11, 14]. As a consequence, much effort has been done to design models able to capture the properties above, while still being formally tractable.

Random⁴ graphs with given numbers of vertices and edges [10, 5] fit the density and distance properties, but they have homogeneous degree distributions and low local density. Random graphs with prescribed distributions [16, 17] and the preferential attachment model [3] fit the same requirement, with the degree distribution in addition, but they still have a low local density. As these models are very simple, formally and computationnaly tractable, and rather intuitive, there is nowadays a wide consensus on using them.

However, when one wants to capture the high local density in addition to previous properties, there is no clear solution. In particular, we are unable to construct a random graph with prescribed degree distribution and local density. As a consequence, many proposals have been made, e.g. [21, 9, 13, 12], each with its own advantages and drawbacks.

The bipartite model

Among the most promising approaches, [12, 13] propose to model complex networks as bipartite graphs.

¹LIP6, CNRS and Université Pierre et Marie Curie (UPMC - Paris 6), 104 avenue du Prsident KENNEDY, 75016 Paris, France – Firstname.Lastname@lip6.fr

²Institute of Mathematics, 18 Hoang Quoc Viet, Hanoi, Vietnam and LIAFA, Universite Paris 7, 2 place Jussieu, 75005 Paris, France – phanhaduong@math.ac.vn, phan@liafa.jussieu.fr

³University of CanTho, 1, Ly Tu Trong, Quan Ninh Kieu, CanTho, Vietnam – ntquy2004@yahoo.com

⁴In all the paper, *random* means uniformly chosen in a given class.

This model relies on the notion of *projection*: given a bipartite graph $B = (\perp, \top, E)$ with $E \subseteq \perp \times \top$, the \perp -projection of B is the graph $B_{\perp} = (\perp, E_{\perp})$ where E_{\perp} is the set of pairs of vertices in \perp which have a neighbour in common in B . See Figure 1.

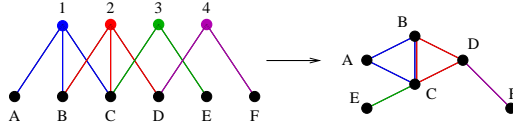


Figure 1: Left: a bipartite graph. Right: its projection, in which two vertices are linked together if they have (at least) a neighbour in common in the bipartite graph.

Any graph may be viewed as the projection of some bipartite graphs. Given a graph $G = (V, E)$, [12, 13] propose to consider the bipartite graph $B(G) = (V, \mathcal{K}(G), E')$ where $\mathcal{K}(G)$ is the set of maximal cliques of G and each vertex v in V is linked to the clique(s) to which it belongs: $E' = \{vc \mid c \in \mathcal{K}(G), v \in c\}$. The bipartite graph $B(G)$ is also called the *clique incidence graph* of G . Clearly, G is the projection of $B(G)$.

The key point is that $B(G)$ captures key information on G into its two degree distributions [12, 13]. Therefore, the model consists in generating a random bipartite graph with prescribed degree distributions, and then to use the projection as a model of G . The obtained graphs fit all the complex network properties listed above, including heterogeneous degree distribution and high local density. Moreover, their random nature make them suitable for formal approaches, see for instance [13, 4].

Limitations of the bipartite model

However, the bipartite model suffers from severe limitations. In particular, it does not capture overlap between cliques, which is prevalent in practice.

Indeed, consider a real-world complex network and its bipartite decomposition obtained with the above scheme. The neighbourhoods of vertices in this bipartite graph generally have significant intersections, as evidenced in [12, 15]: cliques strongly overlap and vertices belong to many cliques in common.

However, when one generates a random bipartite graph with prescribed degree distributions, the obtained bipartite graph have much smaller neighbourhood intersections, almost always limited to at most one vertex (under reasonable assumptions on the degree distributions). In other words, the generated graph (the projection of this random bipartite graph) consists in considering random sets of vertices (with size of prescribed distribution) which we all link together; the probability of choosing several vertices in common between two such random sets tends to zero when the graph grows (as long as the size of the sets is reasonable).

As a consequence, the bipartite model fails in capturing the overlapping nature of cliques and neighbourhoods in complex networks. This leads in particular to graphs which have many more edges than the original ones (two cliques of size d lead to $d \cdot (d - 1)$ edges in the model graph, while the overlap between cliques make this number much smaller in the original graph).

Our contribution

A natural direction to try to solve this problem consists in iterating the decomposition process underlying the bipartite model: one may encode any bipartite graph $B = (\perp, \top, E)$ into a tripartite one $T = (\perp, \top, C, E')$ where C is the set of bicliques of B and E' contains the edges

between any biclique c in C and all the vertices of B which belong to c . This approach is very general, and leads to multipartite decompositions of graphs. However, several variants may be considered, each with its own advantages. The aim of this paper is to explore some of them.

We first give a few notations, useful in the whole paper, including the definition of a fundamental notion, the *factorisation*, which will play a key role in the following. We then consider the most immediate generalisation of the bipartite decomposition (Section 2). We first discuss it informally in order to give an intuition to the reader. We then formally define it and show that it leads to infinite decompositions in some cases. We propose a more restricted version in Section 3, which seems to converge but for which the question remains open. Finally, we propose another restricted version in Section 4 for which we prove that the decomposition scheme always terminates. Both definitions lead to new interesting combinatorial objects, with subtle and complex properties, which we believe are worth of study in themselves.

Notations

All graphs considered here are finite, undirected and simple (no loops and no multiple edges). A graph G having vertex set V and edge set E will be denoted by $G = (V, E)$. We also denote by $V(G)$ the vertex set of G . The edge between vertices x and y will be indifferently denoted by xy or yx .

A k -partite graph G is a graph whose vertex set is partitioned into k parts, with edges between vertices of different parts only (a bipartite graph is a 2-partite graph, a tripartite graph a 3-partite graph, etc): $G = (V_0, \dots, V_{k-1}, E)$ with $E \subseteq \{uv \mid u \in V_i, v \in V_j, i \neq j\}$. The vertices of V_i , for any i , are called the i -th level of G , and the vertices of V_{k-1} are called its *upper vertices*.

$\mathcal{K}(G)$ denotes the set of maximal cliques of a graph G , and $N^G(x)$ the neighbourhood of a vertex x in G . When $G = (V_0, \dots, V_{k-1}, E)$ is k -partite, we denote by $N_i^G(x)$, where $0 \leq i \leq k-1$, the set of neighbours of x at level i : $N_i^G(x) = N^G(x) \cap V_i$. When the graph referred to is clear from the context, we omit it in the exponent.

Recall that $B(G)$ denotes the clique incidence graph of $G = (V, E)$, *i.e.* its bipartite decomposition: $B(G) = (\mathcal{K}(G), V, E')$ where $E' = \{vc \mid c \in \mathcal{K}(G), v \in c\}$.

In all the paper, an operation will play a key role. We therefore give a name to it here and define it generally.

Definition 1 (factorisation) *Given a k -partite graph $G = (V_0, \dots, V_{k-1}, E)$ with $k \geq 2$ and a set V'_k of subsets of $V(G)$, we define the factorisation of G with respect to V'_k as the $(k+1)$ -partite graph $G' = (V_0, \dots, V_k, (E \setminus E_-) \cup E_+)$ where:*

- V_k is the set of maximal (with respect to inclusion) elements of V'_k ,
- $E_- = \{yz \mid \exists X \in V'_k, y \in X \cap V_{k-1} \text{ and } z \in X \setminus V_{k-1}\}$, and
- $E_+ = \{Xy \mid X \in V'_k \text{ and } y \in X\}$.

When $V_k \neq \emptyset$, the factorisation is said to be effective.

2 Weak factor series

As explained before, our goal is to improve the bipartite model of [12, 13] in order to be able to encode non-trivial clique overlaps, that is overlaps whose cardinality is at least two. Indeed, the major drawback of the bipartite model is that when it is randomly generated, clique overlaps are

trivial with high probability. Since these overlaps in the graph result from the neighbourhood overlaps of the upper vertices, the purpose of the new model we propose is to encode the graph into a multipartite one by recursively eliminating all non-trivial overlaps of the neighbourhoods of upper vertices. We first describe this process informally, then give its formal definition and exhibit an example for which it does not terminate.

Neighbourhood overlaps of the upper vertices in a bipartite graph $B = (V_0, V_1, E)$ may be encoded as follows. For any maximal biclique C that involves at least two upper vertices and two other vertices, we introduce a new vertex x in a new level V_2 , add all edges between x and the elements of C , and delete all the edges of C , as depicted on Figure 2. We obtain this way a tripartite graph $T = (V_0, V_1, V_2, E')$ which encodes B : by the reverse operation, which we call *projection*, one may obtain B from T .

The reason why one would take the maximal bicliques is simply to try to encode all neighbourhood overlaps using a reduced number of new vertices. Notice that there are other ways to reduce even more the number of new vertices created, for example by taking a biclique cover of the edge set of B . This is however out of the scope of this paper.

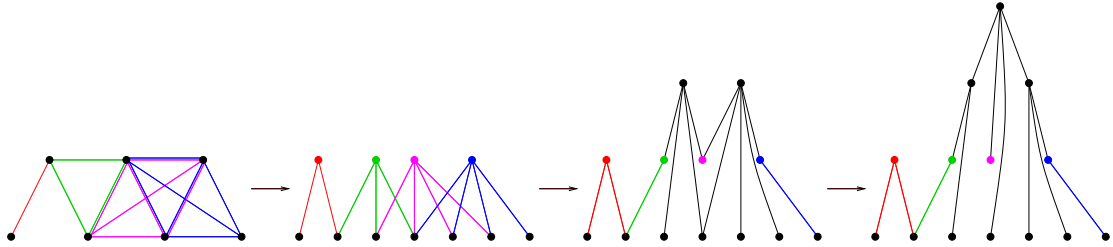


Figure 2: Example of multipartite decomposition of a graph. From left to right: the original graph; its bipartite decomposition; its tripartite decomposition; and its quadripartite decomposition, in which there is no non-trivial neighbourhood overlap anymore. In this case, the decomposition process terminates.

This process, which we call a *factorising step*, may be repeated on the tripartite graph T obtained, see Figure 2. Since the bipartite graph induced by the vertices of $V_0 \cup V_1$ in $T = (V_0, V_1, V_2, E)$ does not contain any non-trivial overlaps of the neighbourhoods of the vertices of V_1 , one can now focus on factorising neighbourhood overlaps between vertices of V_2 . This may be done by repeating the previous process on the bipartite graph $B' = (V_0 \cup V_1, V_2, E')$, where E' is the subset of edges of T between level V_2 and the rest of the graph: $E' = E \cap (V_0 \cup V_1 \times V_2)$. This step results in a quadripartite graph which has no non-trivial neighbourhood overlaps in the first three levels. This factorising process can be iteratively applied in an attempt to remove all neighbourhood overlaps. Then, the key question is to know whether the process terminates or not.

We will now formally define the factorising process and show that it may result in a infinite sequence of graphs. In the following sections, we will restrict the definition of the factorising step in order to always obtain a finite representation of the graph.

Definition 2 (V_k^\bullet and weak factor graph) Given a k -partite graph $G = (V_0, \dots, V_{k-1}, E)$ with $k \geq 2$, we define the set V_k^\bullet as:

$$V_k^\bullet = \{\{x_1, \dots, x_l\} \cup \bigcap_{1 \leq i \leq l} N(x_i) \mid l \geq 2, \forall i \in [1, l], x_i \in V_{k-1} \text{ and } \bigcap_{1 \leq i \leq l} N(x_i) \geq 2\}.$$

The weak factor graph G^\bullet of G is the factorisation of G with respect to V_k^\bullet .

As in the bipartite case, the weak factorisation admits a converse operation, called *projection*, that allows to retrieve the original graph from its weak factor graph. To that purpose, 1) add the edges between any pair of vertices x, y such that $x \in V_{k-1}$, $y \in V_i$ for some $i \in \llbracket 0, k-2 \rrbracket$ and x and y have a common neighbour in V_k ; and 2) delete the vertices of V_{k-1} along with the edges defining their neighbourhood. This shows that the factor graph of G , as well as its iterated factorisations, is an encoding of G . Note that this projection operation is a direct multipartite generalisation of the bipartite one. In particular, when applied to the clique incidence graph of a graph G , it results in G itself.

The *weak factor series* defined below is the series of graphs produced by recursively repeating the weak factorisation step.

Definition 3 (weak factor series $WFS(G)$) *The weak factor series of a graph G is the series of graphs $WFS(G) = (G_i)_{i \geq 1}$ in which $G_1 = B(G)$ is the clique incidence graph of G and, for all $i \geq 1$, G_{i+1} is the weak factor graph of G_i : $G_{i+1} = G_i^\bullet$. If for some $i \geq 1$ the weak factor operation is not effective then we say that the series is finite.*

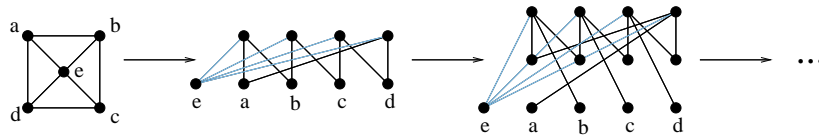


Figure 3: An example graph for which the weak factorising process is infinite. From left to right: the original graph G , its bipartite decomposition $B(G)$, and its tripartite decomposition $B(G)^\bullet$. The shaded edges are the ones involving vertex e , which play a special role: all the vertices of the upper level of the decompositions are linked to e . The structure of the tripartite decomposition is very similar to the one of the bipartite decomposition, revealing that the process will not terminate.

Figure 2 gives an illustration for this definition. In this case, the weak factor series is finite. However, this is not true in general; see Figure 3. Intuitively, this is due to the fact that an edge of the bipartite model may be copied an infinite number of times in the upper levels of the multipartite graphs obtained along the process, always leading to new factorisations and new copies of the referred edge. The aim of the next sections is to avoid this case by giving more restrictive definitions.

3 Factor series

In the previous section, we have introduced weak factor series which appear to be the most immediate extension of bipartite decompositions of graphs. However, weak factor series are not necessarily finite, which is an important drawback as one does not know when to stop the decomposition. In this section, we introduce a slightly more restricted definition for which we know no infinite instance. However, we have no proof that it necessarily gives finite series, which remains an open question.

Definition 4 (V_k° and factor graph) *Given a k -partite graph $G = (V_0, \dots, V_{k-1}, E)$ with $k \geq 2$, we define the set V_k° as:*

$$V_k^\circ = \{X \in V_k^\bullet \text{ such that } |\bigcap_{y \in X \cap V_{k-1}} N_{k-2}(y)| \geq 2\}.$$

The factor graph G° of G is the factorisation of G with respect to V_k° .

We restricted the weak factor definition by considering only sets $X \in V_k^\bullet$ such that the vertices of $X \cap V_{k-1}$ have at least two common neighbours at level $k-2$. The reason is that, in this way, an edge cannot contribute to create vertices in more than one factorising step. An edge can still be copied many times in the successive upper levels but only the edges between levels $k-1$ and $k-2$ can create new vertices on level k . Indeed, with the new definition, so that set $X \in V_k^\bullet$ can give rise to a new vertex at level k , it is necessary, but also sufficient, that the vertices of $X \cap V_{k-1}$ have at least two common neighbours at level $k-2$. This restriction also plays a key role in the convergence proof of the *clean factor series*, defined in next section. That is why we think it may be possible that it is sufficient to guarantee the convergence of the factor series, but we could not prove it with this sole hypothesis.

Definition 5 (factor series $\mathcal{FS}(G)$) *The factor series of a graph G is the series of graphs $\mathcal{FS}(G) = (G_i)_{i \geq 1}$ in which $G_1 = B(G)$ is the clique incidence graph of G and, for all $i \geq 1$, G_{i+1} is the factor graph of G_i : $G_{i+1} = G_i^\circ$. If for some i the factor operation is not effective then we say that the series is finite.*

Open question 1 *Is there a graph G such that its factor series is infinite?*

4 Clean factor series

In the two previous sections, we studied two multipartite decompositions of graphs. The first one is very natural but it does not lead to finite objects. The second one remains very general but we were unable to prove that it leads to finite object. As a first step towards this goal, we introduce here a more restricted definition for which we prove that the decomposition is finite. This new combinatorial object has many interesting features, and we consider it worth of study in itself. It may also help in designing a proof for the definition of previous section, or lead to the construction of a counter-example.

Definition 6 (V_k^* and clean factor graph) *Given a k -partite graph $G = (V_0, \dots, V_{k-1}, E)$ with $k \geq 4$, we define the set V_k^* as:*

$$V_k^* = \{X \in V_k^\circ \mid \forall x, y \in X \cap V_{k-1}, \forall p \in \{0\} \cup \llbracket 2, k-3 \rrbracket, N_p(x) = N_p(y)\}.$$

The clean factor graph G^* of G is the factorisation of G with respect to V_k^* .

Definition 7 (clean factor series $\mathcal{CFS}(G)$) *The clean factor series of a graph G is the series of graphs $\mathcal{CFS}(G) = (G_i)_{i \geq 1}$ in which $G_1 = B(G)$ is the clique incidence graph of G , $G_2 = G_1^\circ$, $G_3 = G_2^\circ$ and, for all $i \geq 3$, G_{i+1} is the clean factor graph of G_i : $G_{i+1} = G_i^*$. If for some i the clean factor operation is not effective then we say that the series is finite.*

The rest of this section is devoted to proving the following theorem.

Theorem 1 *For any graph G , the clean factor series $(G_i)_{i \geq 1}$ is finite.*

Notation 1 *Let $(G_i)_{i \geq 1}$ be the clean factor series of G . For any $i \geq 1$, any $x \in V_i$ and any $j < i$, we denote by $V_j(x)$ the set $N_j^{G_i}(x)$ and by $V(x)$ the set $\bigcup_{0 \leq j < i} V_j(x)$.*

Remark: In the rest of the paper, when referring to Definition 6, it is worth keeping in mind that for $x \in V_{k-1}$ and $p \leq k-3$, the sets $N_p(x)$ and $N_p(y)$ used in the definition are precisely the sets $V_p(x)$ and $V_p(y)$.

Definition 8 We denote by \mathcal{O}' the set $\{O \subseteq V(G) \mid \exists k \geq 2, \exists C_1, \dots, C_k \in \mathcal{K}(G), (\forall j, l \in \llbracket 1, k \rrbracket, j \neq l \Rightarrow C_j \neq C_l) \text{ and } O = \bigcap_{1 \leq i \leq k} C_i\}$; and by \mathcal{O} the set $\{O \in \mathcal{O}' \mid |O| \geq 2\}$. For any $O \in \mathcal{O}'$, we denote by $K(O)$ the set $\{C \in \mathcal{K}(G) \mid O \subseteq C\}$. We also denote by \mathcal{C} the set $\{Y \subseteq \mathcal{K}(G) \mid \exists O \in \mathcal{O}', Y = K(O)\}$.

Proposition 1 For any $A, B \in V(G)$, $K(A) \cap K(B) = K(A \cup B)$. Conversely, if $A, B, C \in \mathcal{O}'$ and $K(A) \cap K(B) = K(C)$ then $A \cup B \subseteq C$.

Proof. Let $A, B \in V(G)$. The cliques in $K(B) \cap K(C)$ are exactly the cliques that contain both A and B , i.e. the cliques that contain $A \cup B$. Therefore $K(A) \cap K(B) = K(A \cup B)$.

Let $A, B, C \in \mathcal{O}'$ such that $K(A) \cap K(B) = K(C)$. Since $K(A) \cap K(B) = K(A \cup B)$, we have $K(A \cup B) = K(C)$. Then, $A \cup B \subseteq \bigcap_{y \in K(A \cup B)} V_0(y) = \bigcap_{y \in K(C)} V_0(y)$. On the other hand, since $C \in \mathcal{O}'$, $C = \bigcap_{y \in K(C)} V_0(y)$. It follows that $A \cup B \subseteq C$. \square

Notice that this implies that \mathcal{C} is closed under intersection. The fact that \mathcal{O}' is also closed under intersection is clear from the definition.

In all the G_i 's of the clean factor series, vertices at level 0 correspond to vertices of G , vertices at level 1 correspond to the maximal cliques of G , that is for any $y \in V_1$, $V_0(y) \in \mathcal{K}(G)$. That is the reason why in the following we do not distinguish between the elements of $\mathcal{K}(G)$ and those of V_1 . We will show that the vertices of V_2 correspond to the elements of \mathcal{O} . Indeed, $x \mapsto V_0(x)$ is a bijection from V_2 to \mathcal{O} . First, for any $x \in V_2$, by definition, $|V_0(x)| \geq 2$, then $V_0(x) = \bigcap_{y \in V_1(x)} V_0(y)$ belongs to \mathcal{O} . Let $O \in \mathcal{O}$. Let us show that $X = K(O) \cup \bigcap_{y \in K(O)} V_0(y)$ is a maximal element of V_2 . First note that $X \cap V_0 = O$ and then $|X \cap V_0| \geq 2$. Now, if you augment X with an element of $y \in V_1 \setminus K(O)$, since $y \notin K(O)$, $X \cap V_0$ will decrease. Thus X is maximal and there is a corresponding $x \in V_2$ such that $V_0(x) = O$. Furthermore, it is straightforward to see that the maximality of $V(x)$ implies that $V_1(x) = K(O)$. Which proves the uniqueness of the $x \in V_2$ such that $V_0(x) = O$.

Definition 9 Let G be a graph and let $(G_i)_{i \geq 1}$ be its clean factor series. The characterising sequence $S(x) = (O_1(x), \dots, O_{k-1}(x))$ of a vertex $x \in V_k$, with $k \geq 2$, is defined by:

- $O_1(x) = V_0(x)$
- $\forall j \in \llbracket 2, k-1 \rrbracket, O_j(x)$ is the unique element⁵ of \mathcal{O}' such that $K(O_j(x)) = \bigcap_{y \in V_j(x)} V_1(y)$.

Remark: For any $x \in V_k$, with $k \geq 2$, $K(O_{k-1}(x)) = V_1(x)$.

Note that O_j is properly defined. Indeed, since \mathcal{C} is closed under intersection, a simple recursion would show that for all $i \geq 3$ and for all $y \in V_i$, $V_1(y) = \bigcap_{z \in V_{i-1}} V_1(z) \in \mathcal{C}$.

Theorem 2 Let G be a graph and $(G_i)_{i \geq 2}$ its clean factor series. We then have the following properties:

1. $\forall k \geq 2, \forall x \in V_k, O_1(x) \in \mathcal{O}$ and $O_1(x) \subsetneq \dots \subsetneq O_{k-1}(x)$
2. $\forall k \geq 2, \forall x, y \in V_k, x \neq y \Rightarrow S(x) \neq S(y)$,

⁵By convention, $O_j(x) = V(G)$ when $\bigcap_{y \in V_j(x)} V_1(y) = \emptyset$.

3. $\forall k \geq 2, \forall (O_1, \dots, O_{k-1}) \in \mathcal{O}^{k-1}, O_1 \subsetneq \dots \subsetneq O_{k-1} \Rightarrow \exists x \in V_k, S(x) = (O_1, \dots, O_{k-1})$.

Proof. The proof is by recursion on k . We denote by $H_1(k)$ the claim that $\forall x \in V_k, O_1(x) \subsetneq \dots \subsetneq O_{k-1}(x)$; by $H_2(k)$ the claim that $\forall x, y \in V_k, S(x) = S(y) \Rightarrow x = y$; and by $H_3(k)$ the claim that $\forall (O_1, \dots, O_{k-1}) \in \mathcal{O}^{k-1}, O_1 \subsetneq \dots \subsetneq O_{k-1} \Rightarrow \exists x \in V_k, S(x) = (O_1, \dots, O_{k-1})$.

It is straightforward that $H_1(2), H_2(2)$ and $H_3(2)$ are true.

For $k \geq 3$, we will prove an additional property. For $i \geq 3, 2 \leq j < i$ and $x \in V_i$, we denote by $S_j(x)$ the set $\{y \in V_j \mid O_{j-1}(x) \subseteq O_{j-1}(y) \subseteq O_j(x) \text{ and } (O_1(y), \dots, O_{j-2}(y)) = (O_1(x), \dots, O_{j-2}(x))\}$. Then we denote by $H_4(k)$ the claim that $\forall i \leq k, \forall 2 \leq j < i, \forall x \in V_i, V_j(x) = S_j(x)$. We will prove that $H_1(3), H_2(3), H_3(3)$ and $H_4(3)$ are true.

Since $S_j(x)$ is defined only for $x \in V_i$ with $i \geq 3$ and $2 \leq j < i$, $H_4(3)$ is equivalent to: $\forall x \in V_3, V_2(x) = \{y \in V_2 \mid O_1(x) \subseteq O_1(y) \subseteq O_2(x)\}$. Let $x \in V_3$; we denote by $V_2(x)$ the set $\{a_1, \dots, a_l\}$. Clearly, for any $i \in \llbracket 1, l \rrbracket, \bigcap_{1 \leq i \leq l} V_0(a_i) \subseteq V_0(x) \subseteq \bigcup_{1 \leq i \leq l} V_0(a_i)$. By definition, $O_1(x) = V_0(x) = \bigcap_{1 \leq i \leq l} V_0(a_i)$, and $K(O_2(x)) = \bigcap_{1 \leq i \leq l} V_1(a_i) = \bigcap_{1 \leq i \leq l} K(O_1(a_i))$. Then, from Proposition 1, $\bigcup_{1 \leq i \leq l} O_1(a_i) \subseteq O_2(x)$. Consequently, for any $i \in \llbracket 1, l \rrbracket, O_1(x) \subseteq O_1(a_i) \subseteq O_2(x)$. That is $V_2(x) \subseteq S_2(x)$. Conversely, we show that for any $y \in S_2(x), V_0(x) \subseteq V_0(y)$ and $V_1(x) \subseteq V_1(y)$. First, we have $V_0(x) = O_1(x) \subseteq O_1(y) = V_0(y)$. Since $O_1(y) \subseteq O_2(x)$, we have $K(O_2(x)) \subseteq K(O_1(y)) = V_1(y)$. Since, by definition, $K(O_2(x)) = V_1(x)$ (see Remark 4) we obtain $V_1(x) \subseteq V_1(y)$. Since x is maximal in V_3° , this implies that $y \in V_2(x)$. This shows that $V_2(x) = S_2(x)$, and so $H_4(3)$ is true.

Since $V_2(x) = S_2(x)$ and $|V_2(x)| \geq 2$, it follows that $O_1(x) \subsetneq O_2(x)$. Therefore, $H_1(3)$ is true.

As, for any $z \in V_3, V_2(z) = S_2(z)$, for any $x, y \in V_3, (O_1(x), O_2(x)) = (O_1(y), O_2(y))$ implies $x = y$, and so $H_2(3)$ holds.

Let $O_1, O_2 \in \mathcal{O}$ such that $O_1 \subsetneq O_2$. Let $Y_2 = \{y \in V_2 \mid O_1 \subseteq O_1(y) \subseteq O_2\}$. Let $X = Y_2 \cup \bigcap_{y \in Y_2} V(y)$. Since $O_1 \subsetneq O_2, |X \cap V_2| \geq 2$. Then $X \in V_3^\circ$. Let us show that X is maximal in V_3° . From Definition 9 and Proposition 1, we have $\bigcap_{y \in X \cap V_2} V_1(y) = \bigcap_{O_1 \subseteq P \subseteq O_2} K(P) = K(O_2)$. Let $z \in V_2$ such that $O_1 \not\subseteq O_1(z)$ or $O_1(z) \not\subseteq O_2$. If $O_1(z) \not\subseteq O_2, K(O_2) = \bigcap_{y \in X \cap V_2} V_1(y) \not\subseteq K(O_1(z)) = V_1(z)$. Then, adding z to $X \cap V_2$ would decrease $\bigcap_{y \in X \cap V_2} V_1(y)$. We have $\bigcap_{y \in X \cap V_2} V_0(y) = \bigcap_{y \in X \cap V_2} O_1(y) = O_1$. If $O_1 \not\subseteq O_1(z)$, then adding z to $X \cap V_2$ would decrease $\bigcap_{y \in X \cap V_2} V_0(y)$. Consequently, X is maximal and vertex $x \in V_3$ corresponding to X has the desired characterising sequence (O_1, O_2) . Finally, $H_3(3)$ is true.

Now, let us suppose that $k \geq 4$ and that $H_1(k-1), H_2(k-1), H_3(k-1)$ and $H_4(k-1)$ are true.

From recursion hypothesis $H_4(k-1), H_4(k)$ is true for all $i \leq k-1$. Let $x \in V_k$. We denote by $V_{k-1}(x)$ the set $\{a_1, \dots, a_l\}$. Let $i, j \in \llbracket 1, l \rrbracket$ and $p \in \{0\} \cup \llbracket 2, k-3 \rrbracket$. From Definition 6 of the clean factor graph, we have that $V_p(a_i) = V_p(a_j) = V_p(x)$. It follows that $H_4(k)$ is also true for $i = k$ and $2 \leq j \leq k-3$. Then we just have to prove that $V_{k-2}(x) = S_{k-2}(x)$ and $V_{k-1}(x) = S_{k-1}(x)$.

Let us denote by S' the set $\{y \in V_{k-2} \mid O_{k-3}(x) \subseteq O_{k-3}(y) \subseteq \bigcap_{1 \leq i \leq l} O_{k-2}(a_i) \text{ and } (O_1(y), \dots, O_{k-4}(y)) = (O_1(x), \dots, O_{k-4}(x))\}$ and by S'' the set $\{y \in V_{k-1} \mid \bigcap_{1 \leq i \leq l} O_{k-2}(a_i) \subseteq O_{k-2}(y) \subseteq O_{k-1}(x) \text{ and } (O_1(y), \dots, O_{k-3}(y)) = (O_1(x), \dots, O_{k-3}(x))\}$. We will show that $S' = V_{k-2}(x)$ and $S'' = V_{k-1}(x)$.

By definition of the clean factor graph, $V_{k-2}(x) = \bigcap_{1 \leq i \leq l} V_{k-2}(a_i)$. From recursion hypothesis $H_4(k-1)$, we get that $\forall i \in \llbracket 1, l \rrbracket, V_{k-2}(a_i) = \{y \in V_{k-2} \mid O_{k-3}(a_i) \subseteq O_{k-3}(y) \subseteq O_{k-2}(a_i) \text{ and } (O_1(y), \dots, O_{k-4}(y)) = (O_1(a_i), \dots, O_{k-4}(a_i))\}$. From Definitions 6 and 9, we have $O_p(x) = O_p(a_i)$ for any $i \in \llbracket 1, l \rrbracket$ and $p \in \llbracket 1, k-3 \rrbracket$. Consequently, $V_{k-2}(x) = \bigcap_{1 \leq i \leq l} V_{k-2}(a_i) =$

$\{y \in V_{k-2} \mid O_{k-3}(x) \subseteq O_{k-3}(y) \subseteq \bigcap_{1 \leq i \leq l} O_{k-2}(a_i) \text{ and } (O_1(y), \dots, O_{k-4}(y)) = (O_1(x), \dots, O_{k-4}(x))\} = S'$.

We now show that $V_{k-1}(x) \subseteq S''$. Again, let $i, j \in \llbracket 1, l \rrbracket$ and $p \in \{0\} \cup \llbracket 2, k-3 \rrbracket$. Since $V_p(a_i) = V_p(a_j) = V_p(x)$, it follows that $O_1(a_i) = O_1(a_j)$ and for $p \neq 0$, $O_p(a_i) = O_p(a_j)$. Then, for any $i \in \llbracket 1, l \rrbracket$, we have $(O_1(a_i), \dots, O_{k-3}(a_i)) = (O_1(x), \dots, O_{k-3}(x))$. By definition, $K(O_{k-1}(x)) = V_1(x) = \bigcap_{1 \leq i \leq l} V_1(a_i)$, and $V_1(a_i) = K(O_{k-2}(a_i))$. It follows that for any $i \in \llbracket 1, l \rrbracket$, $K(O_{k-1}(x)) \subseteq K(O_{k-2}(a_i))$ and $O_{k-2}(a_i) \subseteq O_{k-1}(x)$. Thus, we have $\bigcap_{1 \leq i \leq l} O_{k-2}(a_i) \subseteq O_{k-2}(a_i) \subseteq O_{k-1}(x)$. Finally, $V_{k-1}(x) \subseteq S''$.

We now show the converse: $S'' \subseteq V_{k-1}(x)$. To do this, we show that any $y \in S''$ is such that $V_p(y) = V_p(x)$ for any $p \in \{0\} \cup \llbracket 2, k-3 \rrbracket$ and $V_{k-2}(x) \subseteq V_{k-2}(y)$ and $V_1(x) \subseteq V_1(y)$. From recursion hypothesis $H_4(k-1)$, we know that $V_0(y) = O_1(y) = O_1(x) = V_0(x)$ and that for any $p \in \llbracket 2, k-3 \rrbracket$, $V_p(y) = \{z \in V_p \mid O_{p-1}(y) \subseteq O_{p-1}(z) \subseteq O_p(y) \text{ and } (O_1(y), \dots, O_{p-2}(y)) = (O_1(x), \dots, O_{p-2}(x))\}$. Since for all $p \in \llbracket 2, k-3 \rrbracket$, $O_p(y) = O_p(x)$, it follows that $V_p(y) = V_p(x)$. Again from recursion hypothesis $H_4(k-1)$, we get $V_{k-2}(y) = \{z \in V_{k-2} \mid O_{k-3}(y) \subseteq O_{k-3}(z) \subseteq O_{k-2}(y) \text{ and } (O_1(z), \dots, O_{k-3}(z)) = (O_1(y), \dots, O_{k-3}(y))\}$. On the other hand, $V_{k-2}(x) = \{z \in V_{k-2} \mid O_{k-3}(x) \subseteq O_{k-3}(z) \subseteq \bigcap_{1 \leq i \leq l} O_{k-2}(a_i)\}$. From the definition of S'' we get $\bigcap_{1 \leq i \leq l} O_{k-2}(a_i) \subseteq O_{k-2}(y)$. Still from the definition of S'' , we have $O_{k-3}(y) = O_{k-3}(x)$. Together with $\bigcap_{1 \leq i \leq l} O_{k-2}(a_i) \subseteq O_{k-2}(y)$, it implies that $V_{k-2}(x) \subseteq V_{k-2}(y)$. In addition, since $O_{k-2}(y) \subseteq O_{k-1}(x)$, $K(O_{k-1}(x)) \subseteq K(O_{k-2}(y))$. And consequently $V_1(x) \subseteq V_1(y)$. Thus, by maximality of $V(x)$ (Definitions 1 and 6), $y \in S_{k-1}(x)$ necessarily belongs to $V_{k-1}(x)$. As the other inclusion has been shown before, we conclude that $V_{k-1}(x) = S''$.

Since for all $i \in \llbracket 1, l \rrbracket$, $O_{k-2}(a_i) \supseteq O_{k-3}(x)$, $\bigcap_{1 \leq i \leq l} O_{k-2}(a_i)$ contains at least two elements and belongs to \mathcal{O} . From recursion hypothesis H_3 , there exists $b \in V_{k-1}$ such that $O_{k-2}(b) = \bigcap_{1 \leq i \leq l} O_{k-2}(a_i)$ and $\forall j \in \llbracket 1, k-3 \rrbracket, O_j(b) = O_j(x)$. From recursion hypothesis, $V_{k-2}(b) = S_{k-2}(b) = S' = V_{k-2}(x)$. It follows that $O_{k-2}(x) = O_{k-2}(b) = \bigcap_{1 \leq i \leq l} O_{k-2}(a_i)$. As a consequence, $S_{k-2}(x) = S'$ and $S_{k-1}(x) = S''$, and $H_4(k)$ is true.

Since for any $i \in \llbracket 1, l \rrbracket$, $(O_1(a_i), \dots, O_{k-3}(a_i)) = (O_1(x), \dots, O_{k-3}(x))$, we have $O_1(x) \subsetneq \dots \subsetneq O_{k-3}(x)$. Since $V_{k-2}(x) = S_{k-2}(x)$ and $|V_{k-2}(x)| > 1$, necessarily $O_{k-3}(x) \subsetneq O_{k-2}(x)$. We also established that $V_{k-1}(x) = S_{k-1}(x)$. Since $|V_{k-1}(x)| > 1$, it follows that $O_{k-2}(x) \subsetneq O_{k-1}(x)$. Finally, $H_1(k)$ is true.

Let $x, x' \in V_k$ such that $S(x) = S(x')$. From $H_4(k)$, $V_{k-1}(x) = V_{k-1}(x')$ and as a consequence $x = x'$. Therefore $H_2(k)$ is true.

Let $(O_1, \dots, O_{k-1}) \in \mathcal{O}^{k-1}$ such that $O_1 \subsetneq \dots \subsetneq O_{k-1}$. From recursion hypothesis H_3 , for any $P \in \mathcal{O}$ such that $O_{k-3} \subsetneq P$, there exists $x_p \in V_{k-2}$ such that $S(x_p) = (O_1, \dots, O_{k-3}, P)$. We denote by Y_{k-1} the set $\{y \in V_{k-1} \mid S(y) = (O_1, \dots, O_{k-3}, P) \text{ with } O_{k-2} \subseteq P \subseteq O_{k-1}\}$. Let $X = Y_{k-1} \cup \bigcap_{y \in Y_{k-1}} V(y)$. Since $O_{k-2} \subsetneq O_{k-1}$, $|X \cap V_{k-1}| \geq 2$. Then $X \in V_k^*$. We will show that X is maximal in V_k^* . From Definitions 9 and 8, we have $\bigcap_{y \in X \cap V_{k-1}} V_1(y) = \bigcap_{O_{k-2} \subseteq P \subseteq O_{k-1}} K(P) = K(O_{k-1})$. Let $z \in V_{k-1}$ such that $S(z) = (O_1, \dots, O_{k-3}, P')$ with $O_{k-2} \not\subseteq P'$ or $P' \not\subseteq O_{k-1}$. If $P' \not\subseteq O_{k-1}$, $K(O_{k-1}) = \bigcap_{y \in X \cap V_{k-1}} V_1(y) \not\subseteq K(P') = V_1(z)$. Then, adding z to $X \cap V_{k-1}$ would decrease $\bigcap_{y \in X \cap V_{k-1}} V_1(y)$. Using recursion hypothesis $H_4(k-1)$, we obtain that $\bigcap_{y \in X \cap V_{k-1}} V_{k-2}(y) = \{z \in V_{k-2} \mid \bigcup_{y \in X \cap V_{k-1}} O_{k-3}(y) \subseteq O_{k-3}(z) \subseteq \bigcap_{y \in X \cap V_{k-1}} O_{k-2}(y) \text{ and } (O_1(y), \dots, O_{k-4}(y)) = (O_1, \dots, O_{k-4})\}$. Since $\bigcup_{y \in X \cap V_{k-1}} O_{k-3}(y) = O_{k-3}$ and $O_{k-3}(z) = O_{k-3}$, and since $\bigcap_{y \in X \cap V_{k-1}} O_{k-2}(y) = O_{k-2}$, it follows that if $O_{k-2} \not\subseteq P'$ then adding z to $X \cap V_{k-1}$ would decrease $V_{k-2}(x)$. Consequently, X is maximal and vertex $x \in V_k$ corresponding to X has the desired characterising sequence (O_1, \dots, O_{k-1}) . This shows that $H_3(k)$ is true, which ends the proof.

□

Theorem 1 is a corollary of Theorem 2. Indeed, Theorem 2 states that the characterising sequence $(O_1(x), \dots, O_{k-1}(x))$ of any node x at level k is such that $O_1(x) \subsetneq \dots \subsetneq O_{k-1}(x)$. The strict inclusions imply that the length of the characterising sequence, which is equal to $k - 1$, cannot exceed the height h of the inclusion order of elements of \mathcal{O} . Since $h \leq n - 1$, necessarily V_{n+1} is empty. It follows that the clean factor series is finite and stops at rank at most n .

5 Conclusion

In this paper, we explored the possibility to extend a bipartite decomposition of graphs proposed previously to model real-world complex networks. It consists in encoding high-level properties (like cliques, their overlaps, overlaps of overlaps, and so on) into multipartite graphs obtained through an iterative decomposition process. A key feature of these processes is their ability to produce *finite* encodings. We show that the most immediate approach does not have this feature; we introduce a more restricted version for which we are able to prove termination. In between, we consider another interesting decomposition for which the question remains open.

The multipartite decompositions which we introduce here are very rich combinatorial objects for which many questions arise. The termination property *on all instances* was our main concern here. For processes which do not always terminate, one may investigate the underlying reasons, and try to identify classes of graphs for which the process terminates. Another question of interest is the termination speed, as well as the size of the obtained encoding. If the encoding is compact enough, it may be interesting to use it to represent the graph in central memory and solve algorithmic problems directly on the multipartite decomposition, without projecting it. Conversely, as the decompositions we study here are NP-hard, one may introduce other decompositions easier to compute, while still encoding the original graph.

Finally, the use of multipartite decompositions as models of complex networks, in the spirit of the bipartite decomposition, asks for several questions. In this context, the key issue is to generate a random multipartite graph while preserving the properties of the original graph. To do so, one has to express the properties to preserve as functions of basic multipartite properties (like degrees, for instance) and to generate random multipartite graphs with these properties. This is a promising direction for complex network modelling, but much remains to be done.

Acknowledgements. We warmly thank Jean-Loup Guillaume, Stefanie Kosuch and Clémence Magnien for helpful discussions.

References

- [1] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74, 47, 2002.
- [2] R. Albert, H. Jeong, and A.-L. Barabási. Error and attack tolerance in complex networks. *Nature*, 406:378–382, 2000.
- [3] A.-L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286:509–512, 1999.
- [4] Etienne Birmelé. A scale-free graph model based on bipartite graphs. *Discrete Applied Mathematics*, 2008. To appear.
- [5] B. Bollobas. *Random Graphs*. Cambridge University Press, 2001.
- [6] D.S. Callaway, M.E.J Newman, S.H. Strogatz, and D.J. Watts. Network robustness and fragility: percolation on random graphs. *Physics Review Letters*, 85:5468–5471, 2000.
- [7] R. Cohen, K. Erez, D. Ben-Avraham, and S. Havlin. Resilience of the internet to random breakdown. *Physics Review Letters*, 85, 2000.
- [8] R. Cohen, K. Erez, D. Ben-Avraham, and S. Havlin. Breakdown of the internet under intentional attack. *Physics Review Letters*, 86:3682–3685, 2001.
- [9] S.N. Dorogovtsev and J.F.F. Mendes. Evolution of networks. *Advances in Physics*, 51, 2002.
- [10] P. Erdős and A. Rényi. On random graphs I. *Publications Mathematics Debrecen*, 6:290–297, 1959.
- [11] Philippe Gauron, Pierre Fraigniaud, and Matthieu Latapy. Combining the use of clustering and scale-free nature of user exchanges into a simple and efficient P2P system. In *Euro-Par*, 2005.
- [12] Jean-Loup Guillaume and Matthieu Latapy. Bipartite structure of *all* complex networks. *Information Processing Letters (IPL)*, 90(5):215–221, 2004.
- [13] Jean-Loup Guillaume and Matthieu Latapy. Bipartite graphs as models of complex networks. *Physica A*, 371:795–813, 2006.
- [14] Matthieu Latapy. Main-memory triangle computations for very large (sparse (power-law)) graphs. *Theoretical Computer Science (TCS)*, 407:458–473, 2008.
- [15] Matthieu Latapy, Clémence Magnien, and Nathalie Del Vecchio. Basic notions for the analysis of large two-mode networks. To appear in *Social Networks*, 2007.
- [16] M. Molloy and B. Reed. A critical point for random graphs with a given degree sequence. *Random Structures and Algorithms*, 1995.
- [17] M. Molloy and B. Reed. The size of the giant component of a random graph with a given degree sequence. *Combinatorics, Probabilities and Computation*, 1998.
- [18] M.E.J. Newman. The spread of epidemic disease on networks. *Physcs Review E*, 66, 2002.

- [19] R. Pastor-Satorras and A. Vespignani. Epidemic spreading in scale-free networks. *Physica Review Letters*, 86:3200–3203, 2001.
- [20] S.H. Strogatz. Exploring complex networks. *Nature*, 410, 2001.
- [21] D. Watts and S. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.