REPUBLIQUE DU CAMEROUN
Paix – Travail – Patrie
****
UNIVERSITE DE YAOUNDE I
FACULTE DES SCIENCES
DEPARTEMENT DE D'INFORMATIQUE
****
Centre de Recherche et de Formation Doctorale en Sciences, Technologies et Géosciences
Laboratoire d'Informatique et Applications

REPUBLIC OF CAMEROUN
Peace – Work – Fatherland
****
UNIVERSITY OF YAOUNDE I
FACULTY OF SCIENCE
DEPARTMENT OF OF COMPUTER SCIENCE
****
POSTGRADUATE SCHOOL OF SCIENCE, TECHNOLOGY AND GEOSCIENCE
LABORATORY OF COMPUTER SCIENCE AND APPLICATIONS

CONTRIBUTION TO COMMUNITY DISCOVERY IN COMPLEX NETWORKS CONTRIBUTION À LA DÉTECTION DES COMMUNAUTÉS DANS LES RÉSEAUX COMPLEXES

PH'D THESIS
A dissertation submitted in fulfillment of the requirements for the Degree of Doctor of Phylosophy in Computer Science

Par : **GAMGNE DOMGUE Félicité**

Sous la direction de
NDOUNDAM René
Associate Professor
TSOPZE Norbert
Senior Lecturer

Année Académique : 2021

RÉPUBLIQUE DU CAMEROUN
PAIX-TRAVAIL-PATRIE
****
MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR
****
UNIVERSITÉ DE YAOUNDÉ I
****
CENTRE DE RECHERCHE ET DE FORMATION DOCTORALE EN SCIENCES, TE*CHNOLOG*IES ET G*EOSCIENCES*
****

REPUBLIC OF CAMEROON
PEACE-WORK-FATHERLAND
****
MINISTRY OF HIGHER EDUCATION
****
THE UNIVERSITY OF YAOUNDE I
****
POSTGRADUATE SCHOOL OF SCIENCE, TECHNOLOGY AND GEOSCIENCES
****

DÉPARTEMENT D'INFORMATIQUE
DEPARTMENT OF COMPUTER SCIENCE

ATTESTATION DE CORRECTION DE LA THESE DE DOCTORAT / Ph.D

Nous soussignés, **TCHUENTE Maurice, Pr., BOUETOU BOUETOU Thomas, Pr.,** membres du jury de la thèse de Doctorat / Ph.D présentée par **Mme GAMGNE DOMGUE Félicité, Matricule 06U820**, intitulée : **« CONTRIBUTION TO COMMUNITY DISCOVERY IN COMPLEX NETWORKS »** et soutenue le **09/04/2021**, en vue de l'obtention du diplôme de **Doctorat / Ph.D en Informatique**, attestons que toutes les corrections demandées par le jury de soutenance en vue de l'amélioration de ce travail, ont été effectuées.

En foi de quoi la présente attestation lui est délivrée pour servir et valoir ce que de droit.

Président	**Examinateur**
TCHUENTE Maurice, Pr., UYI	**BOUETOU BOUETOU Thomas, Pr., UYI**

REPUBLIC OF CAMEROON
Peace - Work - Fatherland
****

REPUBLIQUE DU CAMEROUN
Paix - Travail – Patrie
****

UNIVERSITY OF YAOUNDE I
Université de Yaoundé I
FACULTY OF SCIENCE
COMPUTER SCIENCE DEPARTMENT
****

POSTGRADUATE SCHOOL OF SCIENCE, TECHNOLOGY AND GEOSCIENCE
LABORATORY OF COMPUTER SCIENCE AND APPLICATIONS
Centre de Recherche et de Formation Doctorale en Sciences, Technologies et Géosciences
Laboratoire d'Informatique et Applications

CONTRIBUTION TO COMMUNITY DISCOVERY IN COMPLEX NETWORKS

CONTRIBUTION À LA DÉTECTION DES COMMUNAUTÉS DANS LES RÉSEAUX COMPLEXES

PH'D THESIS

A dissertation submitted in fulfillment of the requirements for the Degree of
Doctor of Phylosophy in Computer Science

By :
GAMGNE DOMGUE Félicité
Registration N° : 06U820

Thesis Co-directed by:
TSOPZE Norbert, Senior Lecturer, University of Yaounde I
NDOUNDAM René, Associate Professor, University of Yaounde I

2021

Contribution to Community discovery in Complex networks

PhD thesis

Gamgne Domgue Félicité

A thesis submitted in fulfillment of the requirements for the Degree of Doctor of Phylosophy in Computer Science in the Department of Computer Science at University of Yaounde I.

Supervisor : **Pr. René NDOUNDAM**
Director : **Dr. Norbert TSOPZE**

Dedications

To my loveable girl Sarah Ethelle Bengo Tagne.

Acknowledgements.

I begin this section with a sincere thank you to Dr. Norbert Tsopze, who has had the patience and kindness to coach me since Master 2. He is the very symbol of encouragement, because he has always been available, very understanding and very patient with me. In the same way, I would like to thank the director of research, Professor René Ndoundam, who believed in me and never ceased to contribute to smoothing my path. He remains a source of inspiration for a career as a researcher. I would like to thank these two men, whose success is well known, for their support which made this thesis manuscript possible.

I would like to thank the administrative and teaching staff of the Computer Science Department of the University of Yaounde I who have held and advised me since my arrival at the University in the computer science field and have been able to inscribe in my mind, the premises of this thesis.

I could not continue without thanking the coordinators of the ERMIT mobility project for the mobility grant of one year at the University Cheikh Anta Diop of Dakar (UCAD) in Senegal and the administrative and teaching staff of UCAD whose responsiveness has always made it possible for my stay in Senegal and my financing to go very well. I would also like to thank Bandjoun's community for its financial support during a part oh my thesis.

My gratitude goes to all my family for believing in me and always giving me a psychological comfort. I thank each and every one of you. I thank each and every person i met in my life who has in someway contributed to who i am today. I cannot end these thanks without mentioning all the exceptional people i have had to met during conferences, projects or simply with whom i have talked. A special thanks to my adorable children Ange Nael, Basile Le Grand, Jeannice Ornelle and Sarah Ethelle, who were able to encourage me effectively by putting up with my distances, my short and long absences, my silences, and also my anger, by their joy, their tears, their laughter and their complaints, which boosted me to put an end to this work.

I would like to conclude this section by thanking Him who is the creator of all works, the protector of all creatures; thank God my Creator.

Abstract

Complex networks are the sets made up of a large number of entities interconnected by links. They can be found in several areas: biology, transport, online social networks, agriculture, etc. Many recent applications handle huge volumes of personal or public data resulting from complex networks. They are modeled by graphs in which nodes represent entities and edges model the links between them. These entities generally tend to group themselves into communities, based on certain criteria of similarity or connectivity, and this is a very current research problematic called "community detection". A plethora of community detection methods have been implemented. However, many of them consider that communities should be dense and therefore do not take into account the interest that might bind entities within a community. Nevertheless, when interest is taken into account, it is based on semantic. In spite of its usefulness in the interpretation of data, semantic has the main drawback that the network should be known in advance before being exploited. This consideration is not trivial given the immense size of complex networks. Thus, a fundamental aspect remains to be considered, namely the interest based on topology which does not require a prior knowledge of the entire network.

The work of research presented in this manuscript addresses directed, attributed and multidimensional graphs and proposes methods for detecting communities of interest. These methods rely on the topology and properties of real networks to extract significant communities of interest depending on the context.

Thus, we propose in the first contribution, a triad-based method for detecting communities of interest in oriented networks, using a seed-centric approach. Indeed, triads constitute a more significant elementary topological structure than structures centered around an actor and a diad, because it offers more configurations. Hence, we define a similarity measure allowing to implement the interest of the incoming links with regard to the outgoing ones, with the result that the communities obtained are dense in triads. This density reflects the idea that nodes of the same community adhere to the strong opinion of the previously identified nodes of interest. The second contribution proposes a hybrid community detection method based on the optimization of a novel quality function, the hybrid modularity. This method is applied to attributed networks to extract communities of interest that are topologically similar and homogeneous in their attributes. In this respect, we propose the hybrid modularity which is a composite modularity combining Newman's classical modularity and a modularity based on the attributes and orientation of the links through the previous similarity measure. Through this hybrid method, link density is not guaranteed, but the interest in topological equivalence and attribute homogeneity is ensured.

Finally, for the case of multidimensional graphs modeling more types of interactions between two entities, we propose in the third contribution a method for identifying communities whose interest is modeled by the level of activity of a node in a dimension. These methods are based on machine learning techniques. Our algorithms, implemented on examples of context graphs, confirm their relevance by extracting groups of more homogeneous entities by common topological features.

Key-words: Community detection, complex networks, community of interest, triads, dimension relevance.

Résumé

Les réseaux complexes sont des ensembles constitués d'un grand nombre d'entités interconnectées par des liens. Ils ont en effet eu un essor important en biologie, transports, réseaux sociaux en ligne, etc. Pami celles-ci, de nombreuses applications récentes traitent d'immenses volumes de données personnelles ou publiques qui en découlent. Les réseaux complexes sont modélisés par des graphes dans lesquels les noeuds représentent les entités et les arêtes entre les noeuds représentent les liens entre ces entités. Ces entités ont généralement tendance à se regrouper en communautés, en fonction de certains critères de similarité ou de connectivité. Ceci constitue une problématique d'actualité appelée "détection des communautés". Une pléthore de méthodes de détection de communautés ont été implémentées. Toutefois, plusieurs d'entre elles considèrent que les communautés devraient être denses et par conséquent ne tiennent pas compte de l'intérêt des entités d'une même communauté. Lorsque l'intérêt est pris en compte, il est basé sur la sémantique. La sémantique possède la limite principale que le réseau devrait être préalablement connu pour être exploité. Ainsi, un aspect fondamental reste à considérer, à savoir l'intérêt basé sur la topologie qui n'exige pas une connaissance a priori de l'entièreté du réseau.

Les travaux de recherche présentés dans ce manuscrit exploitent les graphes orientés, attribués et multidimensionnels et proposent des nouvelles méthodes de détection des communautés d'intérêt. De ce fait, nous proposons dans la première contribution, une méthode de détection de communautés d'intéret dans les réseaux orientés, basée sur les triades, à travers une approche centrée-graine. En effet, la triade constitue une structure topologique élémentaire plus significative que les structures centrées autour de l'acteur et de la diade, car elle offre plus de configurations. Nous définissons ainsi une mesure de similarité permettant d'implémenter l'importance des liens entrants par rapport à ceux sortants. Ainsi les communautés obtenues sont denses en triades. Cette densité traduit l'idée selon laquelle les noeuds de la même communauté adhèrent à l'opinion forte des noeuds d'intérêt préalablement identifiés.

La deuxième contribution propose une méthode hybride d'optimisation d'une nouvelle fonction de qualité, la modularité hybride. Celle-ci intègre la modularité classique de Newman et une modularité basée sur les attributs et l'orientation des liens à travers la précedente mesure de similarité. Cette méthode est appliquée aux graphes attribués et permet d'identifier des communautés d'intérêt dont les entités sont topologiquement similaires et homogènes par leurs attributs. A travers cette méthode hybride, la densité des liens n'est pas garantie, mais l'intérêt relatif à l'équivalence topologique et à l'homogénéité des attributs est assurée.

Enfin, pour le cas des graphes multidimensionnels modelisant plusieurs types d'interactions entre deux entités, nous proposons dans la troisième contribution une méthode d'identification de communautés dont l'intérêt est modélisé par le niveau d'activité d'un noeud dans une dimension. Ces méthodes sont basées sur des techniques d'apprentissage automatique. Nos algorithmes mis en oeuvre sur des exemples de graphes du contexte, confirment leur pertinence en extrayant des groupes d'entités plus homogènes par des caractéristiques topologiques communes.

Mots-clés: Détection des communautés, réseaux complexes, communauté d'intérêt, triades, pertinence de la dimension.

Contents

List of Figures

List of Tables

Abbreviations

CDLPA Constrained Directed Label Propagation Algorithm. 42

CNA Complex Network analysis. vi, 23

CPM Clique Percolation Method. 50

LART Locally Adaptive Random Transitions. 49

LPA Label Propagation Algorithm. 36

MDLPA Multi Dimensional Label Propagation Algorithm. 49

SNA Social Network analysis. 21

TF-IDF Term Frequency - Inverted Document Frequency. 44

UCAD comm**U**nity dis**C**overy method in **A**ttributed-based multi**D**imensional networks. 6

CHAPTER 1

INTRODUCTION

1.1 Context

With the proliferation of social media and mobile applications, users are constantly interacting, sharing documents, images/videos and messages, etc. These interactions can be modeled by a complex system. A *complex system* is a system possessing some emergent properties, due to the interactions of its constituting objects [17, 114]. It can be encountered in many different domains such as biology, physics, computer science, sociology, etc. [3, 114]

Network modeling consists in representing such systems through complex networks [3, 17, 114] using nodes to represent the objects and links for their interactions. Thus, complex systems broaden the understanding of the topological real-world networks' properties, such as small-world effect, Scale-free, Homophily and Community structure, as described in Chapter 2. Likewise they help in the analysis of semantics and functioning of the systems of interest. In its most basic form, a complex network contains only nodes and links; it can then be qualified of *plain network*. However, one can introduce a richer information in this model, depending on the considered system, modeling needs and constraints, which makes it a very flexible tool. Thereby, the network can be *directed* [114] if the relations between objects are asymmetric. It can be represented by multiple dimensions, where each dimension represents one type of relationship between nodes, leading to *multidimensional* network [138]; it also can be an *attributed* or *assigned* network [114] when some attributes are added to nodes or links in order to a better description of the model, etc. Before highlighting the goals and the contributions of this thesis, we will present the main problem encountered in complex networks analysis directions.

1.2 Problem

Data quanta spread throughout Social Networks reflecting the way people interact with each other. Discussion forums, video platforms, sharing networks, so many platforms offering rich content can be explored. The aim is therefore to analyze large amounts of social data from these several distributed sources. This analysis involves many applications of data mining, machine learning, etc. and the use of a variety of tools. Among them, the task of community detection in complex networks remains of great interest to the community of scientific researchers. With the spread of mobile applications and the growing diversity of information on the web, the people forming a community attach more value to a shared area of interest or expertise. They constitute a community of interest. However, the communities of interest detection methods are in general based either on the link density or the semantic contents. They do not conserve the directionality of links in directed networks, and they do not consider the level of activity of a node in a multidimensional network. Yet, these informations seem to be relevant for more cohesion involving more interest within the nodes of the community. Thus, **the problem addressed in this thesis focuses on community of interest discovery in complex networks**. It is centered around the two sub-problems as stated in the sections below.

1.2.1 Disregarding the impact of incoming links

According to Girvan [116], a community corresponds to a set of nodes that are densely connected to each other and weakly connected to the other nodes of the network. This definition is interesting for some types of graphs like undirected ones; like this many community detection algorithms implemented for directed networks simply ignore the directionality during the clustering step [16, 116, 124, 127] while other technics transform the directed graph into an undirected weighted one [124, 132, 135], either unipartite or bipartite, and then algorithms for undirected graph clustering problem can be applied to them. These simplistic technics are not satisfactory because the underlying semantic is not retained. For example, in a food web network, according to them, the community structure will be corporated of predator species with their preys. This reflexion is not quite right, since preys must be clustered together and predators together. Moreover, another problem lies in disregarding the importance of in-centric nodes [99]. Indeed, several community detection methods are based on a quality function optimization. The most widespread is modularity [116, 118] that has been extended for directed graphs [121]. However, these methods do not address the impact of a community's incoming links as they do not realize that there is an added value in taking into account nodes with a high degree of their incoming links.

1.2.2 Ignoring the level of nodes'activity

Increasingly, in many real-life situations, entities within a network interact with each other in a variety of ways. Therefore, they can be interconnected by several types of relationships. In a co-authorship network for instance, if we connect two authors by the papers they write together, it is clear to see that each venue, taken as a type of relationship, provides its edges among the authors. Each type of relationship constitutes a *dimension* [14] also called *layer* [113]. Such networks are modeled by so-called *multidimensional networks* [14,138]. In this regard, several existing methods have dealt with the detection of communities in multidimensional networks [5, 15, 23, 104, 113]. In general, they assume the existence of a community on each dimension of the entire multidimensional network. However, they remain limited to apparent communities since they ignore the interest one node has for a particular dimension.

Indeed, a node which is active in several relationships is not involved at a same degree. In this vein, Nicosia [120] argued that an active node on one of these relationships can remain inactive on the rest of the dimensions. This level of activity implies the level of the dimension relevance. As highlighted in [36] the new interest focuses on the question not of how to detect communities, but on what kind of communities are we interested in detecting. Which approach is appropriate for the identification of communities whose entities have common centers of interest based on their relevant interactions/dimensions?

The following paragraph sets out the purpose of taking into account the interest of entities to be together, on the one hand, and on the other hand, the implemented methodology to achieve this solution.

1.3 Thesis goal and methodology

The thesis presented in this manuscript focuses its study on the uncovering of *communities of interest* being a set of entities interacting within a complex network and acquiring or exchanging information related to a shared area of expertise or activity. Because real networks are increasingly enriched by relevant informations on the interactions between entities, we focus on directed and multidimensional networks. According to these two types of studied graphs, the interest is based on both the topological and relational properties of links respectively. Indeed, for directed networks, the community detection approach stresses on the directionality of *in-links* (area of expertise), while for multidimensional networks, the community discovering method deals with the relevant dimensions (area of activity) to build its communities of interest. The following main goals arise:

1. Our first aim is to propose a method for extracting communities in directed graphs, based on the consideration of the incoming links to the nodes of interest, using triads. Triads are structures based on the homophily property of terrain graphs [86]. Therefore, interest-based similarity, such as in social network analysis, exhibits the idea that two entities going inward a third named as their common friend, have a higher probability of belonging to the same community. Indeed, the incoming link reflects the semantics of adhesion to the same idea as the node of interest, hence the notion of triad for directed graphs. The underlying goal behind it lies in revealing the *in-centric* nodes' importance. Furthermore, when these directed networks are assigned, we propose a method that simultaneously takes into account the directionality of edges and attributes of nodes to extract communities of interest;

2. Our second objective is to propose a method for community discovery in multidimensional graphs that includes the neighborhood quality and consequently the nodes interest based on their involvement level in their interactions. The interest is expressed by the relevance dimension-based similarity of nodes. The dimension relevance is assessed by the *neighborhood stability* of a node in that dimension, being dimensions in which the node owns more stable neighbors. The implied purpose is to show that a node's membership to a community depends on its level of activity in the dimensions included in that community, i.e. to establish that relevant dimensions are profitable for the community of interest extraction.

The methodology used to achieve these outcomes is described below: For the first goal related to community detection in directed networks:

- Define a similarity measure for kernel nodes' extraction,
- Extract the kernels by taking into account the interest principle based on the triads,
- Build communities centered around these kernels.

As far as the second aim for discovering communities in multidimensional networks is concerned, the implemented protocol is described below:

- Define a new centrality measure based on the stability of a node's neighborhood,
- Extract relevant dimensions of nodes based on the stability centrality measure ,
- Construct an *assigned monodimensional* network based on relevant dimensions,
- Extract communities from the monodimensional network.

The following sections consecutively present the contribution and the plan of this thesis.

1.4 Contributions

In order to tackle the research problem defined in this thesis, and to propose solutions to overcome the limits of existing approaches, we propose new methods and measures for discovering communities of interest, considering the topological and relational properties of links. This consideration highlights the common interest of the identified community nodes. The main purpose is to define a new way of looking at community of interest, different from the one discussed in the literature which focuses on semantics through ontologies [32]. This method is mainly limited by the fact that despite all works done on validation, they are still subject to discussion as knowledge not only evolves but also there is no evidence that ontology always captures all the knowledge in the field. In order to consider real network features, we will be able to reuse and/or adapt existing topology-based solutions to uncover communities of interest in our context (directed and multidimensional graphs). As a result, the three most prominent contributions are listed below, according to the type of graphs of the context. For directed graphs, the two contributions are described in Subsections 1.4.1 and 1.4.2 and for multidimensional networks, there is one contribution illustrated in subsection 1.4.3.

1.4.1 Heuristic for Community detection on directed networks.

The first heuristic is related to community detection in directed networks. It allows to detect communities densely linked by triads, since communities' members are centered around kernels, being structures consisting in dense triads. To detect communities, we first define *Kernel degree*, a similarity measure based on both triads and Jaccard index, to measure the strength of the kernel vertices' similarity. Afterwards, the kernels reflecting the nodes of interest sets are extracted. Then we define *NCI* (Node Community Index), a merging measure of non-kernel nodes to kernels, in order to detect communities of interest consisting in triad-based densely nodes. Finally, we merge non-kernel nodes to kernel for which the NCI measure is maximized. This contribution has been the subject of 4 publications, namely one paper in an international journal [46], and 2 papers in international conferences [45, 47] and one paper in national conference [48].

1.4.2 Novel quality function.

In order to take into account both relational and topological information, we propose a "modularity hybrid" quality function. It is a combination of 3 types of information: relational information based on link connectivity, topological information based on link directionality, and information based on node attributes. The modularity hybrid includes an hybrid similarity that investigates the topological aspect by applying the Kernel degree similarity measure implemented in the first contribution. This similarity measure contains informations on attributes and directionality and is joined to structural information to transform the directed attributed graph into a weighted one. Then, the resulting graph is applied to an hierarchical agglomerative algorithm to extract the communities qualified as more meaningful. This contribution has been the subject of one publication in CARI 2018, an international conference [49].

1.4.3 Heuristic for Community discovery multidimensional networks.

This contribution focuses on the implementation of comm**U**nity dis**C**overy method in **A**ttributed-based multi**D**imensional networks (UCAD) for community discovery in multidimensional networks. We use some topological graph properties to define a novel centrality called *stability*, needed for computing relevant dimensions. Then, we extract relevant dimensions of nodes based on the stability centrality measure. Afterwards, we enrich the attributes of nodes by their relevant dimensions. A dimension aggregation approach is then used to design a monodimensional attributed network. Finally, through a modified version of an hierarchical agglomerative method, we extract communities. This contribution was the subject of an accepted paper in CARI 2020 [50]. In addition, one submitted paper in an international revue is currently under revision.

1.5 Organization

This thesis is organized in 6 chapters: The *second* one is dedicated to explaining the fundamental concepts about complex networks and community structure. We also describe in detail some topological measures of centrality and illustrate communities.

The *third chapter* proposes an overview on community detection methods. We first describe different detection methods for directed network, before tackling attributed based approaches and finally community discovery methods in multidimensional networks.

The following chapters focus on the community detection problem itself. More precisely, in *fourth chapter* describes our heuristic for detecting communities in directed networks. We first give some preliminary definitions and new similarity metric dealing with directionality, in order to understand the proposed in-seed-centric scheme based on directed triads. Then we concentrate on the validation of the similarity measure on attributed networks. We indicate how to use that measure through both illustration on a small example of food network based on prey-predator relationships and application on a directed attributed network.

The *fifth chapter* describes a new community discovering method in multidimensional graphs. In this chapter, we focus on the extraction of relevant dimensions through the computation of node centrality. This centrality is subject to a new measure called stability, allowing to extract a posteriori communities that not only have a more stable neighborhood but also whose nodes are generally influenced by the same types of relationships, which defines the same center of interest. After this, we describe our multi-community detection framework, giving the details for each step.

Finally, in the *sixth chapter*, we summarize and criticize our work, propose some leads to solve the existing limitations, and identify our major perspectives.

CHAPTER 2

Generalities on complex networks analysis

The world around us can be seen as a set of interactions between elements. These interactions can take place at the microscopic level, such as those between proteins that are constantly at work in our human body, or on the contrary at a macroscopic scale like the gravitational interactions between astronomical objects. These set of interactions, known as complex networks [114, 149], are studied in many scientific fields: sociology, physics, economics, biology [87, 151], etc. They are frequently called *terrain* graphs, because they are used to model a real "life" situation. As mentioned in the introduction in Chapter 1, several real networks consider either asymmetrical relations between entities (for example: a citation network) or several types of relations between entities (for example: a co-author network). These two examples take into account the orientation and dimensionality of the links, giving rise respectively to directed and multidimensional networks. It appears that complex networks constitute a powerful modeling tool, able to represent most real-world systems. This chapter introduces the definitions and notations of the main concepts handled in this thesis. It is organized in 4 sections: the first section describes the graph theory concepts frequently used and necessary for a good understanding of this thesis. The second overviews some main properties of complex networks. The third one illustrate their modeling structures. The last section describes some applications and available tools for complex networks analysis. Finally we investigate on the focal notion of this study, namely communities in complex networks.

2.1 Basic concepts.

As we pointed out in the introduction, complex networks are modeled by *graphs*. Therefore, we can use graph theory to analyze them, as described in [21, 63]. In this perspective, we provide the terminology and typology of topological properties of graph theory for an understanding of the

Table 2.1: Table of notations

Notation	Description
$G = (V, E)$	A single graph with a set of vertices V and a set of edges E
$G =$	A multigraph with a set of vertices V , a set of edges E, and a set of dimensions D
$A = A^{(1)}, ..., A^{(k)}$	A multidimensional network of $K = \|D\|$ dimensions
n	The number of nodes of G, i.e. $\|V\|$
m	The number of edges of G, i.e. $\|E\|$
$m(v, c)$	The number of edges incident to v in the community c
k_v	Degree of node v disregarding the dimensions
k_v^l	Degree of node v in dimension l
$k_v^K = \sum_K^{l=1} k_v^l$	Overlapping degree of a node v across all K dimensions [120]
E^l	Set of edges in the dimension l
E_c^l	Edge number of the community c in dimension l
V^l	Set of nodes of the dimension l
V_c^l	Number of nodes of the community c in dimension l
D_v	Subspace of relevant dimensions of the node v
Γ_v^l	Neighborhood of a node v in a given dimension l
$\|C\|$	Size of a community in the partition
$cut(c)$	Number of links between the nodes of community c and the other nodes of the network
A_{ij}	Entry of the adjacency matrix which represents the existence or not of edge between nodes i and j
$\Delta(i, c)$	Number of triangles that vertex i closes with vertices in c
$ver\delta(i, c)$	Number of vertices of c that form at least one triangle with i
$\delta(c_i, c_j)$	The Kronecker function equal 1 if $c_i = c_j$ (i.e., if nodes i and j belong to the same community) and 0 otherwise.
$simA$	Attribute similarity function, whose value depend on the type of attributes
$C_{sjr} = 0, \omega$	Value indicating the absence (0) or presence (ω) of interconnection between dimensions
γ_s	A resolution parameter on dimension s
μ	A normalization factor

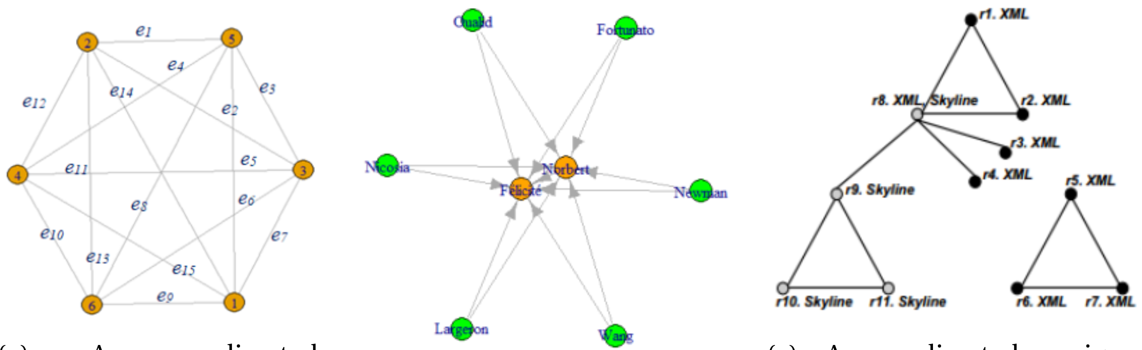
rest of this manuscript. Table 2.1 introduces several notations that will be used in the rest of the manuscript.

2.1.1 Graphs

This section first describes different typology of graphs, according to the interaction type in the corresponding network. When there is only one relationship type among entities, networks are represented by single graphs. Otherwise, they are multidimensional graphs. The second part outlines the key concepts and measures used throughout the manuscript.

Single graphs: weighted, assigned, undirected, directed graphs Single graphs constitute the basic representation of networks consisting in one type of relationship among its entities. They are also called *unidimensional*, monodimensional, *one-dimensional* graphs. For the sake of simplicity, we use the term *graph* to refer to a single graph. Figure 2.1 shows three examples of graph types.

Definition 2.1.1. *(**Graph**). A graph G is a pair (V, E) consisting of a set V of objects and a set E of edges, disjoint from V, together with an incidence function G which associates to each edge e of G a pair of vertices u and v (not necessarily distinct) of vertices of G, such that $\phi_G(e) = uv$.*



(a) An undirected weighted graph.

(b) A Directed graph.

(c) An undirected assigned graph.

Figure 2.1: Examples of single graphs.

The objects are called the *vertices* or *nodes* and the *edges* model the relationships between objects. For an *undirected* graph, an unordered pair of nodes that specify an edge joining these two nodes are said to form a *link*. Thus, the graph is said to be *undirected* if the edges are non-oriented. We also speak of *bidirectional* or *symmetrical* graphs. In other words, a link between nodes v_i and v_j indicates a relationship in both directions. This is the case for relationships of friendships, group membership, etc. Figure 2.1a shows a toy example of undirected graph.

For more flexibility, some additional information can be added to the simple graph, such as the direction and weight of the links, as well as attributes describing the nodes. Thus, the graph is *directed* if the edges are ordered to represent the asymmetry of the relationship between two vertices. The edge expressing ordered pair of nodes is called an *arc*. This is the case for relationships of parent-child, predator-prey, master-slave, etc. For the sake of simplicity, we will use the term edge to simply design a link between two nodes in the graph, leaving out the direction. Figure 2.1b shows an illustration of directed graph.

In addition, a *weighted* graph is one in which each link is assigned a positive numerical value, called a *weight*. This value expresses for example, the distance between two vertices or the density of their interactions. Figure 2.1a shows an example of weighted graph.

Definition 2.1.2. ***(Neighbor)***. *The simplistic definition of a vertex neighbors indicates those of nodes immediately connected to him in an undirected graph. The set of these neighbors refers to the neighborhood concept.*

This concept changes in two ways, in directed graphs: *In-neighbors* are those considered as sources of the edges pointing in to v_i while *Out-neighbors* are those of vertices considered as target of edges pointing out of v_i.

$$\Gamma_i = \{v_j : e_{ij} \in E\} \tag{2.1}$$

Definition 2.1.3. ***(In(Out)-Neighborhood)***. *The In(Out)-Neighborhood for a vertex v_i corresponds to the set of its predecessors(successors), or to the set of its immediately connected in(out)-neighbors as formally described above.*

Let Γ_i^{in} be the in-neighborhood vertices set of vertex v_i and Γ_i^{out} be the out-neighborhood vertices set of vertex v_i.

$$\Gamma_i^{in} = \{v_j : e_{ji} \in E\} \tag{2.2}$$

$$\Gamma_i^{out} = \{v_j : e_{ij} \in E\} \tag{2.3}$$

Definition 2.1.4. ***(Degree)***. *Let $v \in V$ be a node in a graph G. The degree of v is the number of nodes connected (with an edge) to the node v. It is the neighborhood cardinality of v.*

$$Degree(v) = |\{(u, v) \in Es.t.u \in V\}| \tag{2.4}$$

This concept can be dismembered into *in-degree* and *out-degree* in directed graph, where *in-Degree* being the number of incoming edges to the node and *out-Degree* being the number of out-going edges from the node.

Definition 2.1.5. ***(Path)***. *A path is a sequence of links which connects nodes, with the number of links representing the path's length.*

A shortest path between two nodes has the shortest length.

Definition 2.1.6. ***(Geodesic)***.*A geodesic is the shortest path between two nodes.*

Definition 2.1.7. ***(Distance)***. *A distance is the length of a geodesic.*

Definition 2.1.8. ***(Diameter).*** *The diameter of a connected graph refers to the largest possible distance among all the geodesics, between any two nodes.*

$$Diameter = max_{u,v} Geodesic(u, v) \tag{2.5}$$

Definition 2.1.9. ***(Triad).*** *A triad is a sub-graphs of three nodes involving at least two links between them.*

When there are three links among the nodes, it is called a *triangle* or *closed triad*, otherwise, it is an *opened triad.* Triads are considered as wedges, i.e paths of length 2 in undirected networks [86]. Directed networks have six opened triads as observed in Figure 2.2 below.

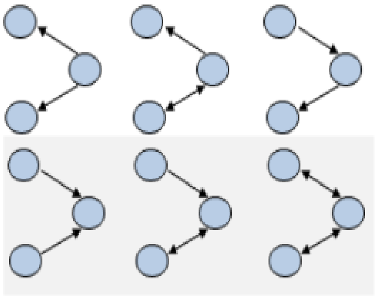


Figure 2.2: Opened triads in directed graphs

In most real networks, in addition to relational information, each node has a set of information describing it, called attributes. The corresponding graph is called *attributed* or *assigned* graph. An attributed graph is denoted as $G = (V, E; W)$, where V is the set of nodes, E is set of edges, and W is the set of attributes associated to the nodes in V for describing their features. Each vertex v_i is described by a real attribute vector $di = (w_1(v_i), ..., w_j(v_i), ..., w_m(v_i))$ where $w_j(v_i)$ is the attribute value of vertex v_i on attribute w_j. For instance, in a co-author network, a vertex represents an author and an edge represents the coauthor relationship between two authors. In addition, there are an author ID and primary topic(s) associated with each author. The research topic is considered as an attribute to describe the vertex property. Figure 2.1c shows an illustrating example of a coauthor graph with node assigned attributes namely *skyline* and *XML.* Attributes are classified into 3 categories: discrete/binary attributes also called categorical attributes, continuous or numerical attributes, and textual attributes [37].

In above mentioned networks, there are one type of interactions between entities. Yet, many connections may reside between any two nodes, either to reflect different kinds of relationships, or to connect nodes by different values of the same type of tie. This representation is expressed by multidimensional graphs.

Multidimensional graphs In general, the authors [14] use a multigraph to model a multidimensional network M and its properties. Formally, a matrix representation of M is: $A = \{A^{(1)}, A^{(2)}, ..., A^{(K)}\}$, where $A(i)$ denotes adjacency matrix of interactions among actors in the i^{th} dimension satisfying, such that $A^{(i)} \in \mathbb{R}_{+}^{n\times n}, A^{(i)} = (A^{(i)})^T, i = 1,2,...,K$. n is the total number of nodes involved in the multidimensional network. Figure 2.3 shows an example of multidimensional graph with interslice coupling being interactions across dimensions, and intraslice links, being interactions within the dimension. There are many formalizations of a multigraph. The simplest and most widely implemented formalization is as follows [14]:

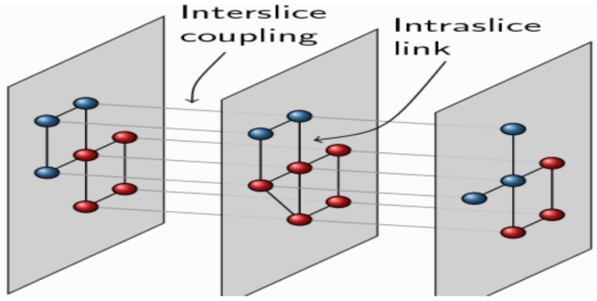


Figure 2.3: Example of multidimensional graph

Definition 2.1.10. ***(Multigraph)****. A multigraph is a triplet $G =< V, E, D >$ where V is the set of nodes, E is the set of edges, D is the set of dimensions, such as $V^d \subseteq V$; $E^d \subseteq V \times V$. $\forall d \in (1,...,K)$, the triple (u, v, d) describes an edge of E^d where $u, v \in V$ are nodes tied in d, with $d \in D$.*

Authors in [85] give another formalization of a multigraph as following: A multigraph G_M is a quadruplet $G_M = (V_M, E_M, V, L)$, where V be a set of nodes as defined in a graph, and D be the set of the types of relationships between pairs of nodes. They define $V_M \subseteq V \times D$ as the subset that contains only the node-dimension combinations such that a node-dimension tuple $(v, d) \in V_M$ if and only if v is present in dimension d. $E_M \subseteq V_M \times V_M$ is the subset of edges between node-dimensions. For the sake of simplicity, in our model in Chapter 5, we only consider undirected multigraphs and since we do not consider node labels and edge weights, hereafter we use the triplet-based definition of multigraph as defined in Definition 2.1.10.

Many of the concepts defined for single graphs can be extended to multidimensional graphs [14]. In [11], the degree of a node across K dimensions called *overlapping degree* is defined as following:

Definition 2.1.11. ***(Overlapping degree)***. *The overlapping degree of a node v is the number of connexions or edges to v across different dimensions.*

The function O_{Degree} is defined as:

$$O_{Degree}(v,D) = |(u,v,d) \in E, s.t. u \in V \wedge d \in D|$$

In multidimensional networks the degree of a node and the number of nodes adjacent to it are no longer related, since there may be more than one edge between any two nodes. In order to capture this difference and distinguish the neighbor concept in Definition 2.1.2 for single networks, we add an "s" for multidimensional cases as defined in the following:

Definition 2.1.12. ***(Neighbors)***. *Neighbors is the set of all the nodes directly reachable from node v by edges labeled with dimensions belonging to D.*

$$Neighbors(v,D) = \{u \in V | \exists (u,v,d) \in E \wedge d \in D\}$$

One key aspect of multidimensional network analysis is to understand how important a particular dimension is over the others for the connectivity of a node, i.e. what happens to the connectivity of the node if we remove that dimension. To assess that dimension behavior, a new concept, namely *Dimension Relevance,* is defined as follows.

Definition 2.1.13. ***(Dimension Relevance)***. *Dimension relevance computes the ratio between the number of neighbors of a node v connected by edges belonging to a specific set of dimensions in D_v and the total number of its neighbors.*

$$DR(v,D_v) = \frac{|Neighbors(v,D_v)|}{|Neighbors(v,D)|} \tag{2.6}$$

2.1.2 Topological measures

In the literature, real-world networks have been characterized by their non-trivial topological properties. We addressed some of them in the introduction, and we will describe them in Section 2.2. Authors used topological measures to quantify these properties. Consult [65, 95, 114] for detailed explanation of each measure. In this subsection, we describe some of these measures in further details. We first focus on those used later in this thesis and then describe similarity and dissimilarity measures, that are important since they are used by data mining techniques, such as clustering [140].

Global measures. Global metrics summarize the structure of a given network in a simple way. Some of them assess the size of the network while others focus on the organization of the network.

Definition 2.1.14. ***(Density)***. *The density of a graph is equal to the proportion of existing links compared to the total number of possible links.*

Indeed, the maximal number of edges in the graph (or sub-graph) with n vertices equals $\frac{n(n-1)}{2}$ in the case of undirected graphs and $n(n-1)$ on the contrary. Thus the directed graph density is $\delta = \frac{m}{n(n-1)}$ while the undirected graph density is $\delta = \frac{m}{\frac{n(n-1)}{2}}$, where m is the existing number of links. When the number of links takes into account the triads in the graph, we talk about triad density.

Definition 2.1.15. ***(Triad density)***. *The Triad Density of a graph (or sub-graph) is a ratio that conceals difference between real number of triads in that graph and maximal possible number of triads in the whole graph.*

Formally, it is defined as follow:

$$\delta_{\Delta} = \frac{|\Delta|}{\binom{3}{n}} \tag{2.7}$$

where the numerator expresses the number of triads from the graph, and the denominator denotes that combination value equals to $\frac{n!}{3!(n-3)!} = \frac{1}{6}(n(n-1)(n-2))$. $\delta_{\Delta} = 0$ if vertices are isolated or if $n < 3$. Otherwise $\delta_{\Delta} = 1$ if the graph is complete, i.e there is bidirectional edge between every pair of vertices.

We investigate in this thesis the edges*'in-direction.* Thus, the *in-neighborhood* cardinality of vertex v_i being its in-degree valuation, or the number of vertices in its *in-neighbourhood*, will be the one taken into account.

Local measures focus on describing the situation of a network objects, compared to the other objects, whether it is a node or a link. They can be interpreted as measures of centrality or accessibility. There are several types of centralities [20, 79], but we described only those that are useful for our work. The first centrality to be introduced is the degree centrality. It defines the importance of a node by the number of links it has. From this centrality, the chances of an individual (represented by a node) to become infected by a virus or influence the nodes around it can be estimated.

Definition 2.1.16. ***(Degree centrality)***. *The degree centrality of a node refers to the number of its direct neighbors, meaning the number of its incident links.*

It is formally defined as in Equation 2.1.4. In a weighted graph, this degree centrality can be weighted through the links'values. This degree is then called *weighted degree*. This measure takes into account the intensity of communication between the elements of the network. However, it removes the topological aspect of the network. Let W be the matrix of weights in which w_{uv} represents the link weight between node u and node v. The weighted degree of a node u is defined by the sum of the weights of all incident links at u :

$$wDegree(u) = \sum_{\forall \in V, u \neq v} (w_{uv}) \tag{2.8}$$

As one would expect information to travel via the shortest path called geodesic, a class of centrality metrics was introduced to evaluate the importance of nodes in relation to this notion. One of the centrality based on geodesic was introduced by [12] and is named the *Closeness* centrality.

Definition 2.1.17. ***(Closeness centrality)***. *Closeness for a node u is the sum of the inverse of distances between u and all the other nodes.*

Formally, the definition of this centrality for a node u is:

$$Closeness(u) = \sum_{v \neq u} \frac{1}{d(u,v)} \tag{2.9}$$

where $d(u,v)$ represents the distance between u and v. Another centrality based on the shortest path was introduced by Freeman and is called the Betweenness Centrality [64]. This centrality measures a different notion of importance than that of Closeness centrality. It measures the extent to which a node tends to be on the shortest paths between other nodes.

Definition 2.1.18. ***(Betweenness centrality)***. *Betweenness is the number of shortest paths between all vertex pairs that run along the node.*

Formally, the betweenness centrality of a node u is defined as follows:

$$Betweenness(u) = \sum_{j=1}^{n} \sum_{k=1}^{n} \frac{(g_{jk}(u))}{g_{jk}} \tag{2.10}$$

such that $g_{jk}(u)$ is the total number of geodesic between nodes j and k crossing node u, and g_{jk} is the total number of shortest paths between nodes j and k.

Newman [117] remains critical towards the betweenness centrality since, according to him, flows in a network do not necessarily follow the shortest or most efficient path. He therefore proposes that random-walk betweenness be taken into account, and acknowledges the existence of multiple existing measures to this effect.

These measures of centrality allow nodal evaluation, given that they indicate the importance of an entity in the network. For global evaluation, similarity measures as described in the following sub-section are used to assess the similarity of a set of entities.

Similarity, distance measures. Some measures are used to assess the objects'behavior in networks. Some of them are based either on similarity or distance. Unlike similarity measures, distance measures have some well-known properties: Positivity, Symmetry and triangle inequality [140]. If $d(x, y)$ is the distance between two points x and y, then the abovementioned properties hold.

- *Positivity* assumes that $d(x, y) \geq 0$ for all $x \neq y$ and that $d(x, y) = 0$ only if $x = y$;
- *Symmetry* assumes that $d(x, y) = d(y, x)$ for all $x \neq y$;
- *Triangle inequality* assumes that $d(x, y) \leq d(x, z) + d(z, y)$ for all x, y and z.

Definition 2.1.19. ***(Common neighbors - CN)****. This similarity computes the intersection of finite sample sets, being the number of common neighbors many nodes have together.*

$$sim_{CN}(x, y) = |\mathit{\Gamma}_x \cap \mathit{\Gamma}_y| \tag{2.11}$$

Definition 2.1.20. ***(Jaccard index)****. Also called Jaccard coefficient, it measures similarity between finite sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets.*

It allows to assess the similarity and diversity of two sample sets. If the sets refers to neighbors of nodes x and y respectively, the measure is defined as follows:

$$sim_{Jacc}(x, y) = \frac{|\mathit{\Gamma}_x \cap \mathit{\Gamma}_y|}{|\mathit{\Gamma}_x \cup \mathit{\Gamma}_y|} \tag{2.12}$$

Depending on the link direction, one can have either the *in*- or the *out*- Jaccard index, as defined below:

$$sim_{Jacc}(x,y)^{in} = \frac{|\Gamma_x^{in} \cap \Gamma_y^{in}|}{|\Gamma_x^{in} \cup \Gamma_y^{in}|} \tag{2.13}$$

$$sim_{Jacc}(x,y)^{out} = \frac{|\Gamma_x^{out} \cap \Gamma_y^{out}|}{|\Gamma_x^{out} \cup \Gamma_y^{out}|} \tag{2.14}$$

Unlike the Jaccard coefficient, Jaccard distance measures dissimilarity between sample sets.

Definition 2.1.21. ***(Jaccard distance).*** *It is complementary to the Jaccard coefficient and is obtained by subtracting the Jaccard coefficient from* 1*, or, equivalently, by dividing the difference of the sizes of the union and the intersection of two sets by the size of the union.*

Formally, Jaccard distance is defined :

$$dis_{Jacc}(x,y) = 1 - sim_{Jacc}(x,y) = \frac{|\Gamma_x \cup \Gamma_y| - |\Gamma_x \cap \Gamma_y|}{|\Gamma_x \cup \Gamma_y|} \tag{2.15}$$

Definition 2.1.22. ***(Euclidean distance).*** *The Euclidean distance between points x and y is the length of the line segment connecting them.*

$$Euc(x,y) = \sqrt{\sum_d (w_x^d - w_y^d)^2} \tag{2.16}$$

Definition 2.1.23. ***(Cosine similarity).*** *Cosine similarity is the measure of the cosine of the angle between points x and y. If x and y are two document vectors, cosine similarity is the most widely used metric.*

$$Cos(s,y) = \frac{x.y}{\parallel x \parallel \parallel y \parallel} \tag{2.17}$$

One can deduce the cosine distance between x and y as following:

$$dis_{Cos}(s,y) = 1 - \frac{x.y}{\parallel x \parallel \parallel y \parallel} \tag{2.18}$$

2.2 Complex networks properties

Complex networks have been receiving increasing attention by the scientific community, also due to the availability of massive network data from diverse domains. In their basic form, they are structured by nodes tied by links. In some cases, richer informations added are useful for better interpretation or modeling needs. Whatever the need, these networks can be classified into two groups: monodimensional networks and multidimensional networks. The former case presents the description of a single type of interaction between the entities of the network. Otherwise, the network is said to be multidimensional. Multidimensional networks best describe the variety of real interactions between individuals. In such networks, many connections may reside between any two nodes, either to reflect different kinds of relationships, or to connect nodes by different values of the same type of tie.

A complex network is a set made of a large number of entities interconnected with links. Social network would be a major example of a complex network where entities are individuals and links are relationship (friendship, message passing or other) between these individuals. Research on the complex systems modeling reveals the existence of common properties found in real-world networks, frequently called *terrain* graphs [77], because they are used to model real "real-life" situations, unlike random graphs which do not. Among these properties, one could include the following:

2.2.1 Small-world effect

This concept has been initially studied by Watts et al. [152]. It states that the average distance between two nodes logarithmically increases with the size of the network (number of nodes) [114]. It expresses the fact that terrain graphs often have very small diameters [147].

2.2.2 Scale-free heterogeneity

This property is based on the heterogeneity of node degree. The heterogeneity of node degree is characterized by the fact that there are (usually a small number of) nodes with higher degree (called hub nodes) compared to other nodes with smaller degree. In several cases the tail of this distribution can be described as a power law with good approximation [4,8]. Formally, $P(k) \sim k^{(\gamma)}$ where $P(k)$ denotes the probability of a node having k neighbors and γ the power law exponent. In [9], this property is proposed under the form of the *preferential attachment* principle stating that in a growing network, the new nodes are more likely to get connected to popular existing nodes. As shown in Figure 2.4, there are few hubs centralizing structures of distinct vertex shape. The figure represents a collaborative graph of scientists from different disciplines (represented by

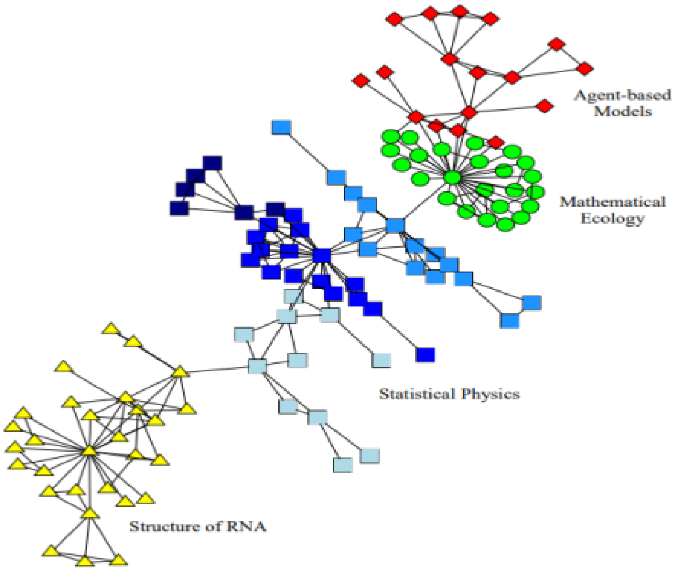


Figure 2.4: An example of a Collaborative Social Network between scientists [68]

different shapes) of the Santa Fe Institute in the United States [68]. The network contains 271 nodes representing the scientists of the institute during the two years 1999 and 2000. A link is put between two authors if they are co-authors of at least one research paper during these two years. Each color represents a community as described in subsection 2.2.4 below.

2.2.3 Transitivity or Clustering

Transitivity means the presence of a heightened number of triangles in the network. This property states that the probability that two nodes having at least one common neighbor are linked, is much greater than the probability of linkage between two randomly chosen nodes. Thus, complex networks contain high local clustering coefficient. For example, in social networks one generally observes that a person's friends tend to collaborate with each other [9]. The clustering coefficient is given by the following formula: $CC = \sum \frac{3\times\#\Delta}{\#\wedge}$, where $\#\Delta$ is the number of triangles in the graph, and $\#\wedge$ the number of triads.

2.2.4 Community structure

Authors in [9, 57] have shown that real networks are not random graphs because they are highly heterogeneous, revealing a high level of sequence and structure. Moreover, the edge distribution is not only global, but also locally non-homogeneous, with high aggregations in special groups of vertices and low aggregations between these groups. This real-world networks' feature is called

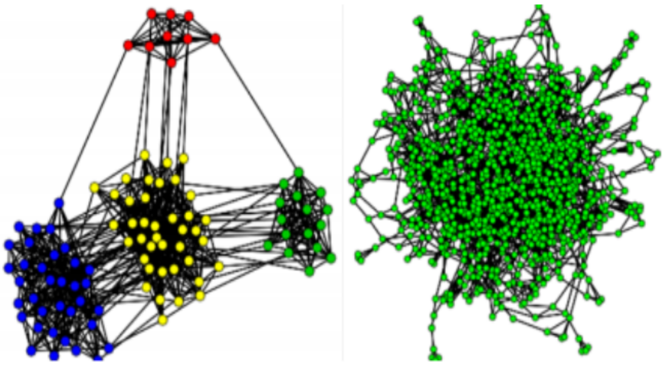


Figure 2.5: Networks with(left) and without (right) community structure

community structure. It is one of the most investigated properties of major importance, as introduced by Girvan and Newman [68]. A graph has a community structure if it is characterized by groups of nodes that have a higher density of edges within them, and a lower density of edges between other groups. This concept is more investigated in Section 2.4.3. Figure 2.5 shows a toy example of network with and without communities.

Increasingly, the community detection subject is attracting more and more attention of the community of scientific researchers. Therefore, there have been hundreds of studies on this topic over the last few years [60, 110]. This inevitably growing interest is explained by the diverse and multiple uses of the concept of community in many fields. In these different fields, objects can be social entities interacting with each other.

We argue in this thesis that a proper exploitation of these properties coupled with an elaborate use of some analysis measures can enhance the relevance of community's nodes interest.

2.3 Complex networks modeling

2.3.1 Monodimensional networks

A social network is a set of social entities such as individuals, close to each other through a relationship or a common interest. Hence, social networks form a basic example of complex networks. In their basic form, they describe entities' interactions of the same nature. They are referred to as *monodimensional* (*unidimensional*, or *one-dimensional*) networks. In order to better understand the behavior of individuals or groups of individuals in a social network, Social Network analysis (SNA) aims to define measures that capture the existing interactions between individuals in the network. Depending on the symmetrical or asymmetrical nature of these interactions, we refer to undirected or directed networks respectively. Figure 2.6 shows a toy example of monodimensional network of relations among Renaissance Florentine families [26]. In this network, there is only one type of relationship, expressing no matter what kind of relation (disregarding marriage or business relations) between its members. Node color depends on the “degree”categories: 3 colors are used to encode the simple pattern (grey color) very important, important (blue color), less important(red color).

When there are several types of interactions between entities, the network can be structured on dimensions, hence the term multidimensional network as described in the following paragraph.

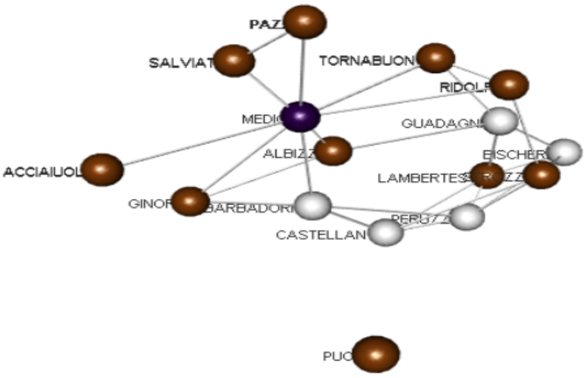


Figure 2.6: A monodimensional network

2.3.2 From complex networks to multidimensional networks

In the world as we know it, we can see a large number of interactions and relations among information sources, events, people, or items, giving birth to complex networks. With social media, people can connect to each other more conveniently than ever. In some social networking sites, entities other than human beings can also be involved. For instance, in YouTube, a user can upload a video and another user can tag it. In other words, the users, videos, and tags are knit together in the same network. This description expresses the data representation through a complex network, in which the "actors" are not homogeneous. Furthermore, examining activities of users, we can observe different interaction networks between the same set of actors. Take Facebook as an example. A user can become a friend of another user; he can also subscribe to another user. The existence of different relations suggests that the interactions between actors are not homogeneous but rather heterogenous [139].

When the network has multiple types of interactions (relations) between the same type of users, it becomes a multidimensional network. *Multidimensional networks* [14] are also designated as *multiplex networks* [113], *multilayer networks* [38], *multislice networks* [113], *edge-colored multigraphs* [85], or *network of networks* [109]. Here, the former designations mainly stress the same behaviors or roles for nodes in all networks. Moreover, there exist only tiny differences for these nodes in different dimensions. On the contrary, the latter designation, namely network of networks, mainly stresses the difference for these nodes in cross dimensions. Kivelä et al. [85] present network types that can be represented using their general multilayer-network framework.

When the number of interaction types between entities is d, the multidimensional network stands for a *d-dimensional* network. Figure 2.7 shows a two-dimensional social network depicting the network in Figure 2.6 taking into account the two types of interactions between its members, namely business and marriage ties. In general, there are two views of multidimensional networks: independent and interdependent multidimensional networks [14, 85]. Independent networks involve actors that are distinct from one dimension to another. They stand for a stacking of single networks (See Figures 2.8a and 2.8b). Interdependent networks involve actors that should be a fixed one across several dimensions. They stress the same behaviors for nodes in these dimensions as shown in Figure 2.8c. It shows configurations on multidimensional networks: namely in (a) an example of a multidimensional network (i.e. an interconnected network, a network of networks etc.), in (b) a representation of the same multidimensional network using another formalism (Node names are the same from the original network) and in (c) an alternative representation of the same multidimensional network in our considered formalism [85]. The type of graph used in this thesis only takes into account the "independent" aspect. The use of the aforementioned networks for the representation of complex systems provides a more thorough and efficient analysis of the processed data. The following paragraph describes the steps of data analysis.

2.4 Complex Network analysis (CNA)

A notable feature of the last two decades is the daily use of Web 2.0, which has become a veritable social media seedbed enabling users to interact, share, group together, collaborate, etc. From the content of these social media, interconnected and well-organized structures can be extracted. These structures are often referred to as "online social networks" to distinguish them from social networks as they are traditionally handled in the social sciences [72]. They are generally formed on the basis of either virtual/digital or physical interactions between individuals. Thus, one can

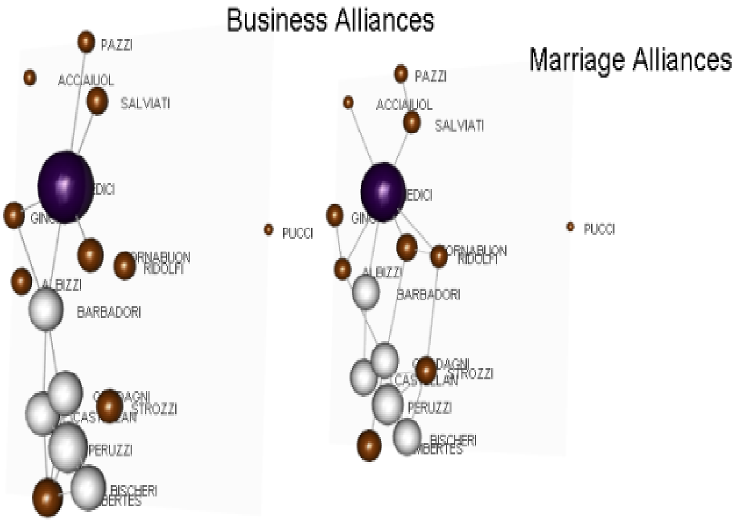


Figure 2.7: Florentine multidimensional network - visualization performed using muxviz package [39]

have networks of individuals, networks of web pages, networks of enterprises, networks of products/services, etc. These different examples reveal the complex nature of social networks, their omnipresence in our lives, but also the interest they present as a subject of study. The current problem is no longer how to collect and store data from social networks but rather how to use them in a relevant way for a value-added analysis.

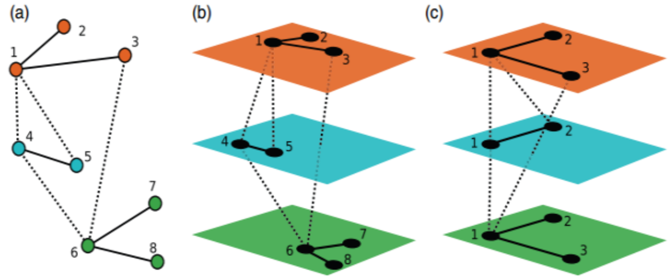


Figure 2.8: Multidimensional networks configuration

This section presents in the following paragraphs, the different research issues on data analysis in complex networks, communities and the tools for their analysis, and the basic concepts used in the rest of the manuscript.

2.4.1 Typology of Complex network analysis

The analysis of a complex network is often achieved by the following four steps: data collection, data processing, data analysis, and visualization of the network extracted from the data. The analysis phase consists in exploring the structure of the network by means of graph theory techniques or statistical modeling. The analysis is often carried out to understand or explain the structure of the network. It allows identifying, in particular, the nodes' social position, the network density, the network diameter, the level of connectivity, the impact of the social structure on the behavior of individuals, etc. These questions are addressed by significant research issues as described in [140], namely:

- Link prediction which consists in determining the presence at a time $t+1$ of a link non-existing at an earlier time t;
- Sentiment analysis consisting in the interpretation/prediction of users'opinions or emotions within text data;
- Community detection: the goal here is to identify groups of similar entities;
- Detection of influencer i.e. to determine the actors who can be the leaders of a particular trend;
- Information spreading which reflects how information is disseminated ;
- Categorization of nodes consisting in determining the state of an individual (healthy or infected in epidemiology);

The detection of communities is the focal point of these works. A multitude of detection approaches have been proposed in the literature, as described in Chapter 3. In order to detect communities in a social network, there are several typology for complex networks' description, as shown in Figure 2.9, which can be grouped into three key factors: the **scale** of analysis, the analysis **time**, and the **heterogeneity** of objects.

Scale criteria refers to the scope of the analysis; it can be structured in two levels: The local level or microscopic scale which focuses on a minority of network entities when computing social ranks or their categorization within the structure, and the global level or macroscopic scale studying the whole network, without specifying a subgroup on which to initiate the analysis. The *time* factor indicates the period of the network analysis, based on the assumption that the structure of a network is constructed and evolves over time rather than being considered to have been created instantaneously. Thus, the structure can be analyzed in the past (retrospective analysis), present (current-time analysis) or future (prospective analysis). Note that this chronological analysis requires a historical overview of the structure's changing history. The *heterogeneity* of objects expresses heterogeneous networks [139]. Accordingly, heterogeneous networks can be categorized in two different types: multi-mode network involving heterogeneous actors/entities with the same type of interactions between them (each mode represents one type of entities) and multidimensional network consisting in multiple types of interactions between the same type of users (each dimension represents one type of interaction). In this thesis, we are interested in community detection on the local level scale, by investigating the study at a specific time (static-based methods), together with the consideration of the interaction-based heterogeneity. In general, communities are obtained and visualized by means of Social Network Analysis tools. The following paragraph describes some of them.

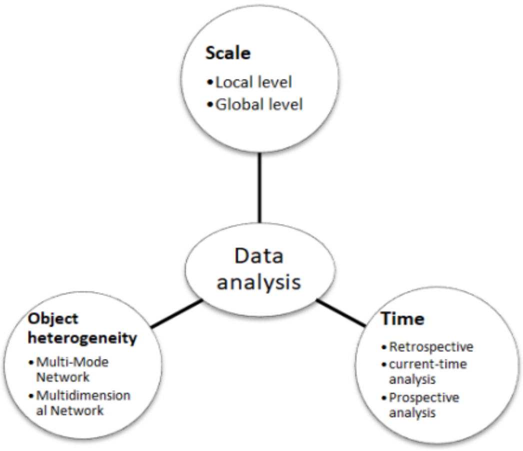


Figure 2.9: Different typologies of complex network design

Table 2.2: Comparison of some tools for complex network analysis

Tool and date of pubication	Category	Licence	OS	Implementation	Multidimensional network
Gephi 2009	Software	Free	Windows, Linux, MacOS	Java	
Statnet 2019	Library	Free	Windows, Linux, MacOS	R	
NetMiner 2010	Software	Private	Windows	Java	
Igraph 2006	Package	Free	Windows, Linux, MacOS	C, R, Python	✓
NetworkX 2019	Package	Free	Windows, Linux	Python	✓
Cytoscape 2016	Software	Free	Windows, Linux, MacOS	C++	✓
MuxViz 2014	Software	Free	Windows, Linux, MacOS	R	✓
MultinetX 2017	Package	Free	Windows, Linux	Python	✓
Multinet 2018	Package	Free	Windows	R	✓
Pymnet 2018	Library	Free	Windows	Python	✓
Py3Plex 2019	Library	Free	Windows, Linux	Python	✓

2.4.2 Complex Network Analysis Tools

With increasing amounts of data that lead to large social networks consisting of different node and edge types, there is an increasing need for versatile visualization and analysis software. A multitude of tools are used to analyze social networks. In this perspective, these tools allow the data loading, their analysis and the structure visualization. However, it should be noted that some tools only allow either visualization or analysis of social networks. In addition, these tools can be software or libraries. In general, software is easier to utilize as it provides a graphical and interactive interface for a broader user audience. Libraries require programming language skills and offer more opportunities for feature extensions. For example, the analyst can add functions in accordance with its needs.

Table 2.2 shows some of the tools used for CNA comparing them according to a grid designed around five criteria. The first comparison criterion indicates the category of the tool, i.e. whether it is software or a library/package. The second criterion refers to the type of the tool's license: free or private. The third criterion focuses on operating systems supporting the tool. The fourth criterion gives an idea of the programming language(s) of the tool implementation and therefore which can be used to extend the functionality of the tool. The last criterion specifies whether the tool has features allowing the analysis of multidimensional networks or not. Remember that there are many other criteria for comparing these tools. Our choices are mainly guided by the context of this thesis also dealing with multidimensional networks and that both performance and scalability of the tools are of great importance for complex applications such as community detection. Furthermore, it is necessary to underline that combining several tools is possible when a single tool does not provide satisfactory of the expected results.

A very common task in the analysis of terrain graphs, which has generated a prolific literature over the last twenty years, is the detection of communities [60, 62]. This involves finding sets of elements in a graph that interact more specifically with each other than with the rest of the graph, thus forming so-called *communities*. Section 2.4.3 outlines the notion of community.

2.4.3 Communities in social networks

As mentioned in Section 2.2.4, community structure is one of the most studied network properties. In this section we will remember some basic concepts of community definition in social networks.

There are several different terms referring to communities, like *modules*, *clusters* or *cohesive subgroups*. The notion of “Community”is a concept not unanimously accepted by the scientific literature, as it depends on some constraints together with the network structure under study and needs. A variety of considerations on the notion of community emerge from this [6, 41, 60, 114], according to the semantics or structural analysis criteria. Moreover, Malliaros [110] classifies communities into two groups, according to their topology: density-based communities and pattern-based communities. To generalize these insights, authors in [36] give a meta-definition of community as a set of entities that share some closely correlated actions with the other entities of the set.

From this generic definition, whatever the type of information or needs dealt with in the network, there are three things that need to be specified: the notion of connectivity, the notion of similarity among the nodes of the network, and the notion of interest/influence around which the nodes are centered.

1. Connectivity : this criteria expresses both the density-based and pattern-based clusters. The density-based clusters definition states the set of nodes that are strongly connected with each other while weakly connected with other nodes in the network [118] while the pattern-based clusters definition focuses on groups of nodes that go beyond edge density consideration. As we will describe shortly, an example of this category is the case of flow circulation, where information moves across nodes in the same community most quickly.

2. Similarity: this feature is based on a set of similar objects [153]. The connectivity links are ignored and only the node attributes are considered. Here, members of these communities being clusters, may know or care little about others. An example of such community is the community that contains mainly agricultural sector operators interested in "rice" planting.

3. Interest: Interest-based clusters are groups of nodes interested in the same types of information. The type of information may be found in the link direction as well as in the nodes' attributes. Interest-based clusters can be included in the two above community concepts, depending on the needs or the topological graph properties of the nodes. An example of such community concerns those authors attending the same venue, namely CRI’17, on the same research topic, namely Social network analysis. In the rest of this thesis, if not stated explicitly, this is the definition of community that is assumed. Examples of such communities will be presented in the following chapter.

With regard to the three axes that we have just elaborated, we can deduce the plural and non-exhaustive approaches of community discovery in social networks. Before addressing the chapter on the description of some existing community detection methods, several terms need to be explained in order to make it easier for the reader to understand both the manuscript and the proposed methods.

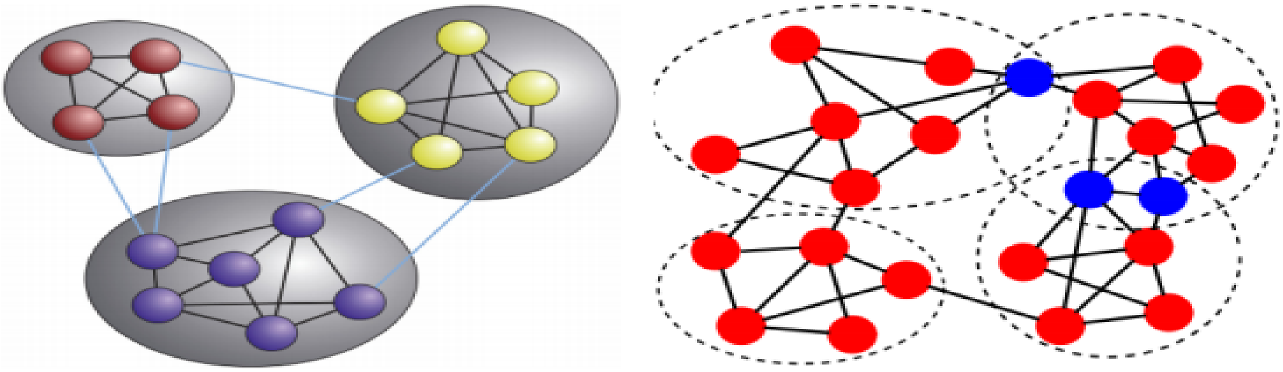
Definition 2.4.1. (Node of interest). Also called "seed", or "ego", a node of interest is any influential node that we are interested in at a given situation.

Definition 2.4.2. (Semantic definition: Kernel). A kernel is a set of nodes of interest. [150].

For Wang [150], a kernel called community kernel, is considered as a set of influential nodes inside a group. Each member of a community kernel has more connections to/from the kernel than a vertex outside the kernel does. As shown in Figure 2.1b, nodes with orange color form a kernel. Community kernel consists then in nodes centralizing information. Fortunato [60] gives three different families of definitions: Global, local and interest-based.

Definition 2.4.3. (Global communities). Global communities are sub-graphs whose nodes possess remarkable properties relatively to the rest of the network.

Definition 2.4.4. (Local Communities). A local community is based on the exploitation of an information concerning a node and its close neighborhood, and neglecting the rest of the network.



(a) Disjoined communities.

(b) Overlapping communities.

Figure 2.10: Examples of partitions and covers.

Definition 2.4.5. (Semantic definition: Community of interest). A Community of interest is a community of people who share a common interest or passion [59]. In other words, it is a set of individual that share the same subject of interest/passion.

In the context of online social networks (OSN), these people exchange ideas and thoughts about the given passion, but may know (or care) little about each other outside of this area. An example of such a community is the set of fans of "The Beattles". This definition is based on the interaction semantics [32]. Because semantic based definition drawback relies difficulty in inferring new knowledge, in our research, we consider the definitions of communities of interest based on the topology, according to the context graphs, namely directed graphs on the one hand (see Definition 30), and multidimensional graphs on the other hand (Definition 36), as described in our contribution sections in Chapters 4 and 5 respectively. **Overlapping vs disjoined communities** . A community is an overlapping community if a portion of its nodes simultaneously belongs to other communities while a disjointed community has exclusive nodes.

For example, in social networks actors may be part of different communities: work, family, friends, and so on. All these communities will share a common member, and usually more since a work colleague can also be a friend outside the working environment. Figure 2.10 shows an example of both disjoined and overlapping communities' partitions. In Subfigure 2.10b, blue nodes are share by three communities. **Cover vs partition** . A cover C is division of a graph into overlapping (or fuzzy) communities. A cover C is a set of overlapping communities c_i such that $\cup c_i = V$. A Partition P of a graph is a cover whose communities are disjointed: $\forall c_1, c_2 \in C,\ c_1 \neq c_2 \Rightarrow c_1 \cap c_2 = \emptyset$. In other words, partitions are structures whose standard definition forbids multiple memberships or vertices [60, 62]. Throughout the document we will use the expression "partition" to indicate the result of a community detection algorithm without regard to the overlapping nature of communities. But when necessary, we will specify that the result is a cover.

Naturally, several alternate definitions on this concept also exist. However, community detection-related works are often focused on designing new methods to detect communities, and less on what a community exactly is. For this reason, the notion of community is very frequently implicitly defined as the result of the considered community detection method. The next chapter overviews some community detection methods.

2.5 Conclusion

In this chapter, we have outlined the background to complex network analysis. Specifically, we have reviewed some concepts of graph theory necessary to well understand the remainder of this thesis, as well as a few related metrics such as the centrality and similarity of entities in a network.

In addition, we presented the complex network properties as well as the models used for representing them, and finally we discussed the notion of community which constitutes the focus of this thesis. In the next chapter, we present the state of the art in the area of community detection in both monodimensional and multidimensional networks. We also focus more specifically on detection approaches based on the optimization of quality functions.

CHAPTER 3

Community discovery methods and applications

3.1 Introduction

We now focus on the notion of community detection, which was introduced in the previous chapter. As aforementioned, a key characteristic of terrain graphs is the presence of community structure. The concept of communities is complex and no universal definition is recognized [60]. Because of this popularity, hundreds of different algorithms [7, 62, 110] were developed for the task consisting in identifying the community structure of a network, an operation generally called *community detection*. Indeed, community detection is used in several fields, as described in Section 3.5.2 below. In this chapter, we describe the state of the art in the field of community detection and present some of its applications. The objective of this study is to clearly define the research context in order to highlight the contribution of our work with respect to the existing literature. More specifically, section 3.2 presents some quality functions, section 3.3 describes the methods of community detection in single graphs, section 3.4 deals with multidimensional graphs. Finally, Section 3.5 presents different ways to evaluate the methods and their applications in terrain networks.

3.2 Quality functions

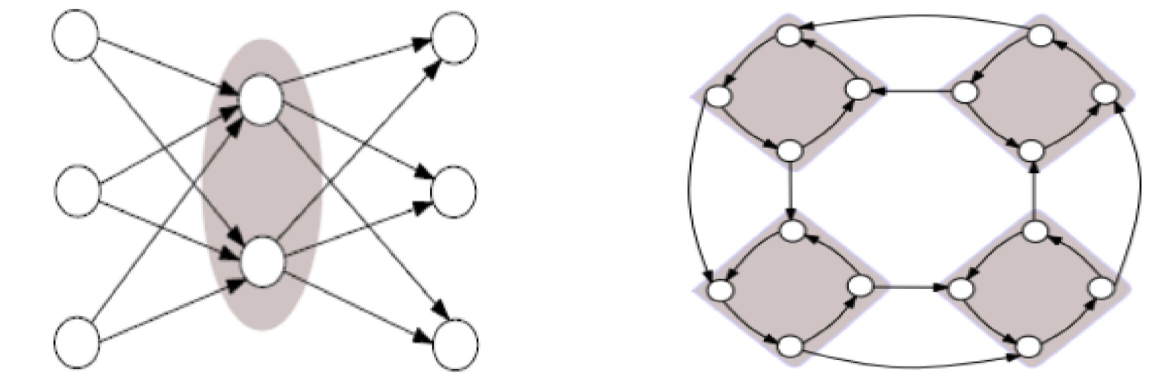
Remember that the resulting structure of the community detection is called cover or partition, depending on the overlapping nature of the nodes. To ensure the validity of the partitions or covers, a function called *quality function* is sometimes optimized [116]. The estimation of the value for this quality function is either done directly during the optimization, because it is used as the objective function to optimize, or indirectly because it is used a posteriori to evaluate the mined community structure. Yang and Leskovec [154] have defined four characteristics that they consider desirable in the expected communities.

- The internal density: the nodes within the community are very connected to each other.
- Separability: the community has different characteristics from its surroundings. For example, nodes within the community have more neighbors inside than outside the community.
- Internal cohesion: the characteristics of the community are more robust to the deletion of nodes or edges. For example, it is necessary to remove many of the edges of a community so that it is no longer connected.
- Triadic closure: for u, v, w the nodes of the community, if $(u,v) \in E$ and $(v,w) \in E$ then, generally $(u,w) \in E$.

These characteristics are not completely independent. For example, high internal density is often correlated with high triadic closure. The Topological equivalence is a good illustration of this reliable correlation. There are two main definitions of topological equivalence for vertices: *structural equivalence* [105], in which vertices are equivalent if they have the same neighbors, as shown in Figure 3.3a; regular equivalence [55], in which vertices of a class have similar connection patterns to vertices of the other classes, as shown in Figure 3.3b. However, these features sometimes reveal a conflict. Indeed, method X may produce communities that are denser than method Y, but Y produces communities with more similar entities. This is why the community detection is often perceived as a multi-objective optimization problem. Nevertheless, it is possible to describe the intuitive notion of communities based on interest, as well as its detection task, which is at the heart of the thesis and will be taken up again, detailed, formalized and discussed in the following Chapters. In this section, we describe the quality functions investigated throughout this thesis, to assess a community detection method and outline the steps of quality function optimization.

Definition 3.2.1. (Quality function). Given a partition P. A quality function is an application $q(P) \longrightarrow \mathbb{R}$ that quantifies the aforementioned characteristics on a partition in order to obtain a Numeric result.

Quality functions are presented in the table 3.1 below. There are three levels of quality function estimation: *Microscopic, mesoscopic* and *macroscopic* levels. Microscopic level consists in



(a) Pattern based on structural equivalence.

(b) Pattern based on regular equivalence.

Figure 3.1: Different types of structural patterns.

functions defined at the **node level**. Such functions compute a quality for all nodes of the graph. They take three parameters as input: the node v, the community c in which it is located, and the partition P.

Clustering coefficient [152] . The Clustering Coefficient of a node is the probability that two of its neighbors randomly selected in the same community are also neighbors. This function measuring the triadic closure refers also to a terrain graph property as stated in subsection 2.2.3.

Permanence [30] . The permanence is a metric of communities' separability. It measures the connectivity of a node within its community by involving the Clustering coefficient. This function is weighted in such a way that a node has a lower permanence if it is relatively highly connected to another community in particular. Thus Permanence measures both separability and triadic closure.

Mesoscopic level consists in quality functions that are defined at the **community level**, i.e. they calculate a quality for each community. In this case, they take as input a cluster c of the partition P.

Conductance [73, 134] . The conductance Φ is defined for the community c as the ratio between the number of links having one end in the community and the minimum between the number of links inside the community and the number of links outside it. The community detection method we suggest in chapter 4 presents a variant of this function applied at the microscopic level i.e. at the nodal level. The value of $\Phi(S)$ ranges from 0 (when the community is good since it is a connected component) to 1 (the community is bad since it has no internal links, or has an infinite number of external links).

Weighted community Clustering [125] . The Weighted Community Clustering (WCC) for a community c computes the density of triangles indicating the nodes in c close to each other. It can be extended for the whole partition through an average function as described in Equation 3.1.

$$Q(P) = \frac{\sum_i f(P_i)}{|P|} \tag{3.1}$$

Multidimensional density [104] . This metric measures the density of links in multidimensional networks, through an extension of the community density in single networks [2].

In order to evaluate the community-level functions $f()$ over the whole partition P, the quality average of the communities that make it up could be determined, as shown in Equation 3.1.

Macroscopic level functions compute the whole partition validity. It includes many parameters based on various features of the partition.

Modularity [118] . Modularity is a function detecting the ratio between intra- and inter-community number of edges. It is one of the basic objective metric about the quality of a particular division into clusters for a network and is widely studied in many works (one of them is a greedy optimization able to scale up to networks with billions of edges) [16] and has been successively extended, according to the type of directed [121], attributed [37], multidimensional graphs [113], as described in the following corresponding paragraphs.

The modularity has been extended in order to consider characteristics of directed, attributed and multidimensional networks respectively.

Directed modularity [96, 121] . Directed modularity is similar to modularity, but the configuration model is modified, as discriminates incoming and outgoing degrees of the nodes.

Modularity attribute [37] . Unlike Newman's modularity [118] which does not include the attribute similarity between nodes, the "modularity attribute" Q_{Attr} of Dang and Viennet [37] include similarity based on nodes'attributes. The measure used to compute the similarity depends on the type of attributes, as described in Subsection 3.3.3

Table 3.1: Quality functions

Level	Name	Functions
Node Level	Clustering coefficient	$qCC(P,c,v)=\frac{2\lvert u,v,w\in c/((u,v);(v,w);(u,w))\in E^3\rvert}{\lvert c\rvert(\lvert c\rvert-1)}$
	Permanence	$qPerm(P,c,v)=$
Community Level	Conductance	$\Phi=\frac{cut(c)}{min(\sum_{v\in\bar{c}}k_v,\sum_{u\in c}k_u)}$
	Multidimensional Density	$Density_{Multi}(c)=\frac{\sum_{l=1}^{\lvert D\rvert}E_c^l-\sum_{l=1}^{\lvert D\rvert}min(E_c^l)}{\sum_{l=1}^{\lvert D\rvert}max(E_c^l)-\sum_{l=1}^{\lvert D\rvert}min(E_c^l)}$
	Weighting Community Clustering for a community	$Wcc(c)=\frac{1}{\lvert c\rvert}\sum_{i\in c}(\frac{\Delta(i,c)}{\Delta(i,V)}.\frac{ver\Delta(i,V)}{\lvert c\{\{i\}\rvert+ver(i,V)})$
Partition Level	Weighting Community Clustering for a partition	$WCC(P)=\frac{1}{\lvert V\rvert}\sum_{j=1}^{\lvert P\rvert}(\lvert C_j\rvert.Wcc(C_j))$
	Modularity	$Q=\frac{1}{2m}\sum_{ij}(A_{ij}-\frac{k_ik_j}{2m})\delta(c_i,c_j)$
	Directed modularity	$Q_d=\frac{1}{2m}\sum_{ij}(A_{ij}-\frac{k_i^{out}k_j^{in}}{2m})\delta(c_i,c_j)$
	Multislice modularity	$Q_{multislice}=\frac{1}{2\mu}\sum_{ijsr}[(A_{ijs}-\gamma_s\frac{k_i^sk_j^s}{2\lvert E^s\rvert})\delta_{sr}+\delta_{ij}C_{jsr}]\delta(\varrho_{is},\varrho_j)$
	LPA-based objective function	$F=\frac{1}{2}\sum_{u,v\in V}A_{v,u}\delta(l_v,l_vu)$
	Multi LPA-based objective function	$F_{multi}=\frac{1}{2}\sum_{u,v\in V}\sum_{d\in D}A_{v,u}^{(d)}\delta(l_v,l_u)$

Multislice modularity. Mucha et al. [113] derived the generalized modularity, a metric to assess the quality of a given partition into multidimensional communities.

LPA-based objective function. Barber and Clark [10] defined a function for an undirected monodimensional network G, based on the Label Propagation Algorithm (LPA).

To support the multidimensional setting, one straightforward way to redefine the LPA-based objective function as F_{multi} is to sum over all within-community edges irrespective of their dimensions.

3.2.1 Steps for quality function optimization

The optimization process of a quality function consists in three stages. Hence, all the optimization heuristics follow the same process but may vary in their implementation of the three phases. In the following, we summarize these three steps and explore some possible variants.

1. The first phase is community initialization. It consists in finding the set of nodes, called "seed", which represents the initial composition of the community. The seed may consist

of one or more nodes. The choice of the seed depends on the nature of the communities we want to identify [78]. For example, if we want to detect an ego-centric community, we can initialize the seed with the node of interest [106]. On the other hand, if we already have an idea about the initial status of the community and we would like to expand it, it is more useful to consider that the seed includes a group of nodes. This variant is used in [150]. In addition, if we are interested in diverse forms of a community, it is recommended to use random seed initialization. Thus, each time we build a community, we will have a different composition. This is the variant chosen in [94]. The contribution in Chapter 4 on community detection in oriented graphs uses this principle of community initialization by kernel nodes

2. The second phase is the setting of a stopping condition for the optimization process. The most intuitive stopping condition is to run until all the neighbors of the node of interest are visited [73]. It is also possible, as demonstrated by Guimera [69], to consider that the optimization of the quality score is bounded by a given threshold. They show that the optimal number of communities that maximizes modularity is closed to $\sqrt{n}$. This threshold is held in Chapter 5 to assess the uncovered covers

3. The third phase consists in a quality function optimization. Indeed, optimization refers to minimizing [31] or maximizing [106] the quality score. The goal of this phase is to reorder the graph in order to obtain a partition that offers a higher quality score. This process is repeated until the stop condition is verified.

In general, community of interest detection techniques refer to a classification of network nodes that are more densely connected than others, in order to construct related classes of users with the same characteristics with respect to a measure of similarity referring to common interests. Thus, their objective is to create a group of vertices, taking into account the relationships between the vertices in the graph and their attributes, so that the communities are composed of vertices which respect the definition of the community previously established.

Deterministic vs non-deterministic algorithm. A non-deterministic community detection algorithm is an algorithm that detects different communities by running it repeatedly on the same dataset, unlike the deterministic algorithm that detects identical communities.

3.3 Community detection in single networks

The objective of community detection in graphs, or in complex networks, is to create a partition of vertices, taking into account the relationships and features that exist between vertices in the graph, so that the communities are composed of strongly connected vertices [60,114]. In addition, Mehdi et al. [7] presented a deepen recent overview of community detection methods. We will present the most addressed methods in this thesis, focusing on those that optimize a partition quality criterion, notably modularity, since our proposals use this criterion, in section 2.3.2.

3.3.1 Community detection in undirected networks

This section deals with approaches that just take into account the methodological principles without dwelling on the internal information of the network. There is a wide variety of approaches to community identification. Kanawati [77] classifies them in four groups of non-restrictive approaches:

- Group-centered approaches where nodes are grouped into communities based on common topological properties;
- Network-centric approaches where the overall structure of the network is examined to separate the graph into communities;
- Propagation-centric approaches which often apply a procedure for the community structure to emerge by exchanging messages between neighboring nodes;
- Seed-centric approaches where the community structure is built around a set of knowledgably selected nodes.

Group-centered approaches. The principle is to restrict the definition of a community to that of a group of nodes that share some topological features. Clique percolation [41] is the most obvious example of method which assimilates a community to a maximal clique in the graph (a clique is a complete subgraph). However, the problem of maximal clique estimation is that it is an NP-difficult problem [19], which makes it difficult to consider its use in the context of very large graphs. Moreover, terrain graphs are mainly sparse. Therefore such structures are often very minority in these graphs. In contrast, dense groups of nodes may be used as seeds for community detection.

Moreover, $k-core$ (a $k-core$ is a maximum connected subgraph in which each node's degree is greater than or equal to k) is another concept of community that authors in [123] explore to identify communities.

Network-centric approaches. Most of the approaches proposed in the literature are based on a scheme of computation taking into account the global connection of the graph. Authors in [60, 142] synthesized them in three families of approaches:

Traditional methods: Traditional Methods consisting in the optimal partitioning into k "clusters" of the graphs representing the Social Networks, k being set as input, include graph partitioning [81], partitional clustering [107] and Spectral clustering [51]. Their solution look for partitions often of the same size. This constraint being too restrictive and difficult to find in real situations, it has been loosened in order to investigate communities but without having to specify the exact size.

Hierarchical methods: The most applied heuristics are based on the principle of hierarchical classification. Two opposing approaches are widely experimented: **Agglomerative** (or bottom-up) approaches, which start from vertices as separate clusters (singletons), and merge two communities at each iteration, to ends up with the graph as a unique cluster. The communities to be merged are those that promise maximum modularity. *Louvain* [16, 124] are some examples of these approaches. **Divisive** (or top-down) approaches, such as Edge Betweenness [68, 116, 126], start from the whole graph as a cluster. At each iteration, an attempt is made to split the cluster in two in order to maximize modularity.

Optimization-based approaches: The problem of community detection can be reduced to the clustering of the nodes in the network, in terms of an optimization problem of a predefined objective function. Quality functions described in Table 3.1 are optimized by these family of approaches [73, 125, 134]. The most widely addressed function is the modularity [115], whose maximization is a difficult NP-hard problem [24]. So other alternatives, such as the Louvain algorithm [16], are based on greedy techniques to provide suitable solutions in computing time.

Propagation centric approaches. These approaches are based on an information propagation within the communities. Therefore, they explore the density-based property of intra-community links. Indeed, because of the higher relative density of communities and weak inter-community links, it is reasonable to assume that a signal emitted by a node and retransmitted by its neighbors is more likely to remain in the community of the source node than to propagate to other communities. *Walktrap* [124] is based on the probability that a random walker will reach the other nodes of the network in k time steps. *Label propagation* [127] present another example of algorithms based on label propagation techniques.

Seed centric approaches. Seed-Centric approaches for Community Detection in Complex Networks generally follows these principal steps [78]:

1. Seed computation which consists in identifying core or influential nodes;
2. Seed local community computation enlarges seed to built their local communities;
3. Community computation out from the set of local communities from step 2.

Different seed expansion strategies are also proposed, as described in [77]. In many algorithms, the heuristics developed for the identification of local communities apply this family of approaches. Ngomnang *et al.* [119] consider that the starting node is at the boundary of a community, unlike the other approaches which do not guarantee the coverage of all nodes of a graph in the resulting community structure. In [76], a more original approach is proposed where, after seed detection, each node in the graph (seed or outside the seed) computes a community membership preference vector for each seed. This community membership of the nodes is the result of a local choice process involving the node and its direct neighbors. The following methods incorporate one or more of the aforementioned features, since they consider additional information from both structural and semantic networks, that are not considered by the methods in this section.

3.3.2 Community detection in directed networks

Finding clusters in directed networks is a challenging task with several important applications in a wide range of domains. Malliaros *et al.* [110] give a taxonomy on the directed network clustering approaches depending on the way directed edges are treated. They classify them in four main categories: Naive graph transformation approach, transformations maintaining directionality, Extending clustering objective functions and methodologies to directed networks, alternative approaches.

Naive graph transformation approach. As Santo Fortunato stated that developing methods of community detection for directed graphs is a hard task [60], a common approach is to ignore the direction of the link and run the algorithms designed for undirected networks, largely due to no other better options. Thus the potentially useful information of the edge directions is discarded and the meaningful communities are also missed. The algorithms based on this approach are : *Walktrap* [124], *Edge Betweenness* [68], *Label Propagation* [127] and *Louvain* [16]. Indeed, when a directed network is set as input, the results are the same as well as the network was undirected.

Transformations maintaining directionality. This category concerns simple schemes that convert a directed graph into either unipartite weighted network through symmetrization techniques [84, 132] or bipartite [70, 157, 159], this enabling to utilize the richness and complexity of existing methods to find communities in undirected graphs. However, the basic question behind such approaches remains the same: *How to consider the common interest of the nodes on the principle of the edges' directionality?* Thus, to try to deal with the problem without transforming the original graph structure, several methodologies would be to extend quality functions and tools, to the case of directed networks, as described in the following paragraph.

Approaches based on the extension of objective functions and methodologies. These approaches focus on the extension of tools and measures developed for undirected case. The ones based on the optimization of the so-called *directed modularity* [96, 121], the *directed clustering coefficient* [34] and the objective function of weighted cuts in directed graphs [112]. In the following, we address the limits of the modularity, whose optimization steps have been described in Section 3.2.1, since we use it through this thesis.

- Resolution Limit: It has been shown in [61, 92] that the size of each community depends on the number of nodes in the network. It is then difficult to detect small communities, even well separated. To make up for this limit, Gautier and Lancichinetti [88, 91] proposed to inform some parameters about either the number of communities or whether the method should extract small communities or not. However, these parameters can greatly affect the accuracy of an approach if the values provided by the user are incorrect since there is no knowledge to the network.
- No edge directionality discrimination: As shown in [84, 99], modularity ignores the impact of incoming-link degree of nodes. Indeed, it does not implement the idea that an edge from

a low out-degree but high in-degree node to an opposite case node should be considered of a bigger value. In order to make good use of the directionality, a recent heuristic based on Constrained Directed Label Propagation Algorithm (CDLPA) is proposed in [99]. The authors consider the balance growth of communities through an improvement of LPA for directed networks. CDLPA is effective for datasets with a monotonous degree distribution of nodes, and overcomes the imbalance growth of communities limit. Indeed, it assumes that communities must have a similar capacity of nodes; therefore it constrains the membership of a node towards a community to which it is not strongly connected.

- Instability: The high modularity score is obtained even in random networks [80]. Indeed, the assumption behind the modularity is that a random network is not supposed to have community structure. The current community structure is then compared with a *null model*, leading to many possible realizations from the null model.

Alternative approaches follow different and diverse methodological principles. Some of them based on various probabilistic models in [156] have been proposed for community detection. Among them, stochastic block models are probably the most successful ones in terms of capturing meaningful communities, producing good performance, and offering probabilistic interpretations. However, the method discontinues in practice when the number of iterations goes beyond 20, and results become insignificant. To make up for this failure and investigate topological properties to assess the pattern-based communities of interest, some authors [71, 150] explore two structures respectively: “triads”and “kernels” like described below.

A kernel standing for the community seed, as described in Subsection 3.3.1, is considered as a set of influential nodes inside a group. It seems to be information centralizing nodes. Some methods explored the problem of detecting community kernels, in order to either exhibit different influence and different behavior of nodes inside a structure for easily interpreting the common interest of nodes or uncover the hidden community structure in large social networks. Wang et al. [150] identify those influential members, the kernels, to detect the communities and propose efficient algorithms for finding community kernels; however in their method named GREEDY, they proposed to extract community kernels (group of nodes of interest) then their auxiliary communities (non-kernel nodes), by a non-deterministic algorithm. Wang models its community kernels associated to their auxiliary communities by an unbalanced weakly bipartite (UWB) structure as shown in Figure 3.2. The UWB structure consists in two disjoined graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ such that: $d_{21} > d_{11} > d_{22} >> d_{12}$, where $d_{ij} = |E(V_i, V_j)|/|V_j|$, with $E(V_i, V_j) =$

$(u, v) \in E | u \in V_i, v \in V_j$ and (u, v) an ordered edge.

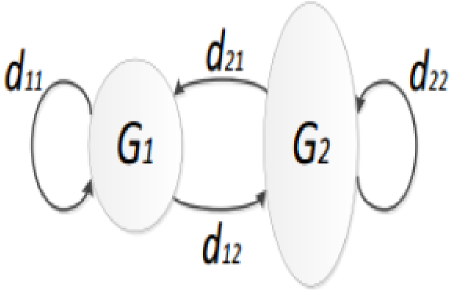


Figure 3.2: An Unbalanced weakly bipartite structure

Wang's method [150], in addition to its speed and simplicity, integrates both community separability and triadic closure feature of partitions. However, this method sets the size of kernels, and therefore the size the communities and their number. It proceeds by a random choice of node to initiate the kernel. By this way, results are arbitrary and not efficient for directed graphs as confirmed by Seifi [133], because of the random node choice and the difficulty for a better parameterization. This constitutes a considerable drawback. In fact, providing accurate values input parameters, including the number of communities, requires a priori knowledge of the network to be analyzed. Whereas, in practice, such knowledge is not always available.

3.3.3 Community detection in attributed networks

Ding [43] studies the type of connections that exist between entities in a community and describes two types of connections: social connections that are often real relationships in networks such as friendship, communication or collaborative relationships, and similaritybased connections that are derivative connections. Based on these types of connections, he distinguished topology-based methods [33, 68] and interest-based methods [141]. In the first category, the identified communities consisting in densely connected nodes but differing interests, while in the second category, communities with coherent interests but rather disconnected nodes may result. In order to deal both with link density and common interests, Zhou *et al.* [160] introduced clustering in attribute vector graphs where entities are described by numerical vectors. The partitioning of the graphs is based on the similarity of the attributes so that nodes with the same attribute values are grouped into a single partition. As illustrated in Chapter 2, there are three types of attributes: Discrete, continuous and textual attributes. If the attributes are discrete, a commonly used similarity measure is based on the simple matching criterion. The similarity between two nodes in an attributed graph is determined by examining each of the d attributes and counting the number of attribute values they have in common. For continuous attributes, the most commonly used metric is based on the Euclidean distance as defined in Equation 2.1.22 in Chapter 2. If the attributes are textual, the authors need to transform them into numeric values. They represent a text document through a bag of words. Each word is represented as a separate variable having numeric weight. The most popular weighting schema is Term Frequency - Inverted Document Frequency (TF-IDF) [131]. Each document is then represented as a vector of weight. To measure the similarity between two document vectors, cosine similarity is the most widely used metric.

Bothorel [22] studied clustering methods in assigned graphs and classified them into three families, according to their methodological principles :

Attributes based clustering approaches. In this family of approaches, attribute based clustering method first exploits attributes by graph or node enrichment and compute similarity or distance measures over nodes' attributes [56, 160], then apply a clustering technique to detect communities. According to the **SA-Cluster** method [160], the unified random walk distance is applied to an augmented graph. Then, a simple graph clustering algorithm [116] that optimizes modularity for weighted graphs is applied for clustering the whole vertices of V . Moreover, according to ANCA method [56], after characterizing each node by its relationship with preselected seeds, authors compute a similarity between nodes and apply unsupervised learning techniques to generate attribute and topological communities.

Relational based clustering approaches. In the relational based clustering model, structural properties are considered first through either a neighborhood similarity. Li in [98] proposed a hierarchical clustering by filtering process of cores (kernels) based on structural information, then merging them by their attributes similarity. Dang and Viennet [37] studied two approaches. $SAC1$, the first one, applies Louvain's detection method [16] to partition the graph into k groups, then apply the "modularity attribute"maximization involving attribute-based similarity $SimA(v_i, v_j)$, to evaluate the more positive gain by moving of nodes. In the second approach $SAC2$, the author constructs a KNN (k nearest neighbour) directed graph through an attribute similarity, then apply Louvain's method.

Methods exploring together attributes and relationships. These methods belong to the category of semi-hybrid approaches since they investigate together attributes and structure. Combe *et al.* in [35] propose the **I-Louvain** algorithm which uses the inertia based modularity combined with the Newman's modularity. More recently, a multiobjective evolutionary algorithm based on structural and attribute similarities (MOEA-SA) is first proposed to solve the attributed graph-clustering problems by Li et al. [101]. Two objectives termed as modularity Q (see Modularity in Table 3.1) and attribute similarity (SA) are used to be maximized in the algorithm. SA is defined as a criterion to measure the quality of attribute similarity of nodes inside clusters of G.

3.3.4 Summary of detection algorithms in monodimensional graphs

In this section, we provide a critical overview of existing solutions based on networks with one type of relationships within entities namely monodimensional networks. An overview Table 3.2 cumulates some of the algorithms presented above in four criteria.

The first criterion indicates the particular used strategy. This criterion is assessed on the basis of three sub-criteria, namely, the type of approach, its advantages and disadvantages. With the second criterion, it is possible to know which types of networks are handled by the algorithm. We considered two types of networks: directed and attributed networks. The third criterion gives an overview about the types of communities covered: their nature and their overlapping aspect. The last criterion presents information about the behaviour of the algorithm. Three sub-criteria are used to evaluate this criterion, namely determinism, complexity and prior knowledge on the number of communities. The first sub-criterion focuses on the deterministic behaviour of the algorithm. In other words, does the algorithm always return the same output community for the same input data? The second sub-criterion gives an idea about the speed of the algorithms. The third sub-criterion gives information on whether the algorithm predefines the number of communities to be detected or not. While reading Table 3.2, we note that each approach has advantages and weaknesses. We also observe that no algorithm covers all the defined criteria simultaneously and that none of these approaches is deterministic.

The table is divided into two panels: In contrast to the top panel, the bottom panel concerns the methods applied to the assigned graphs. A main drawback has been observed: the disregard of the importance of the incoming degrees of the nodes that would have a higher value in the community detection.

When reading the top panel, we observe that only one method is concerned by the communities of interest based on the authority of the seed nodes. The methods implemented by these authors have the advantage of taking into account the nodes of interest called kernels, by extracting as kernels those of influential nodes with a similar dense neighborhood. Precisely, the GREEDY algorithm extracts kernels on the basis of a random choice of the initial node from which to carry out the kernel construction process. The partition is therefore arbitrary depending on the choice of this node. Moreover, the size of the kernel is given as a parameter. If this parameter is misinformed, then the output is still insignificant.

Furthermore, very few methods deal with both directed and attributed networks, as presented in the bottom panel. We identified only one method that does, namely SAC1. It has the advantage of producing more meaningful communities. Indeed, it adapts to both network topology and a variety of attribute types simultaneously. Although this method combines structural (topological) and semantic (attribute) information, it has a fundamental limitation, namely the lack of information on edge directionality. In view of its approach based on the optimization of modularity, it does not distinguish the importance of the incoming degrees of a node from its outgoing degrees, as expected by the modularity. In addition, the centrality of authority in oriented graphs stipulates that authoritative nodes have more important incoming than outgoing degrees.

Finally, the algorithms we have presented in this section have been designed only to deal with monodimensional networks. As a result, they are not able to interpret the multiple types of interactions between the entities, which are the object of many types of real-world networks, namely multidimensional networks. In the following section, we present some methods for detecting communities in multidimensional networks.

3.4 Multidimensional community discovery methods

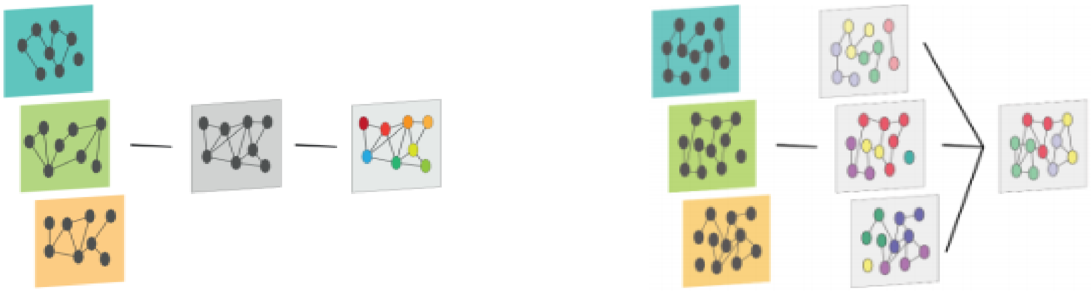
This section presents a short study on some multidimensional community discovery methods. It precisely gives an overview of their operating principles and organization mode according to their computational accuracy and efficiency. Hmimida [74] classifies them into two main groups: those for transforming into a unidimensional community detection problem and those for generalizing existing methods to deal directly with multidimensional networks.

3.4.1 Transformation into a unidimensional community detection problem

This family of approaches aims to carry out specific tasks on dimensions before proceeding to community extraction from the generated task result. They differ by the type of extracted infor-

Table 3.2: Summary of some community detection algorithms in single networks

Method	Approach			Network Type		Community nature		Algorithm behavior		
	Category	Advantages	Drawbacks	Directed	Attributed	Detection Nature	Overlap ?	Deterministic	Complexity	Predefined community number ?
Klymko [86]	Group centered	Communities are denser	NP-Difficult	✓	Ö	Structural	No	No	-	No
CPM [41]				✓	×	Structural	Yes	No	$O(exp(n))$	No
Louvain [16]	Quality function optimization	No limit resolution, fast	Order of node selection affects the computation time	×	×	Structural	No	No	$O(nlog(n))$	No
Edge betweeness [118]			Instability in communities	×	×	Structural	No	No	$O(n^3))$	Yes
Nicosia [121]		Consideration of Edge directionality	Do not discriminate in- or out- degree of links	✓	×	Structural	Yes	No	$O(\|C\| * n^2)$	Yes
Greedy [150]	Seed-centric	The interest focuses on influent nodes of the community	It ignores the link information between non-kernels and kernel members - The number and the size of communities is known	✓	×	Structural	No	No	$O(m+n)$	Yes
Ngomnang *et al.* [75]		Seed could be located at the boundary of the communities	Are not applied to a broad type of networks	×	×	Structural	Yes	No	$O(n^2)$	No
CDLPA [99]	Propagation based approach	Balance growth of communities	Not effective for networks with power law distribution of nodes'degree	✓	×	Structural	Yes	No	-	No
Walktrap [124]		Whatever the length of the random walk, the nodes spaced at this length are similar.	Require the length of the random walk	×	×	Structural	Yes	No	$O(n^2 log(n))$	No
LPA [127]		Linear time complexity	If the network is directed, it ignores the edges'direction	×	×	Structural	Yes	No	$O(n))$	No
ILouvain [35]	Quality function optimization	Uses numerical attributes	Instability limit of the modularity	Ö	✓	Structural and semantic	Yes	No	-	No
SAC [37]		Scalable - considers several types of attributes	Higher complexity - It does not discriminate in- from out- degrees of nodes.	✓	✓	Structural and semantic	No	No	$O(n^2)$	Yes
MOEA-SA [101]		Takes into account link information via a hybrid link representation	A lot of informations have to be given in input parameters	Ö	✓	Structural and semantic	No	No	$O(n^2)$	Yes
CESNA [155]		Scalable - Find relevant node attributes	Only categorical attributes are implemented	Ö	✓	Structural and semantic	Yes	No	$O(n)$	Yes
SA-CLuster [155]		Very good balance between structural and attribute similarities	Downgrade the intracluster cohesiveness	Ö	✓	Structural and semantic	No	No	$O(n^3)$	Yes
ANCA [56]	Seed-centric approach	Provides meaningful communities	Random choice of seeds	Ö	✓	Structural and semantic	No	No	-	Yes



(a) Process of dimension aggregation based approach.

(b) Integration process of the multidimensional community structures.

Figure 3.3: Process of multidimensional community discovery

mation from each dimension.

Aggregation based approaches. This group of methods generally considers each dimension of the multidimensional network individually. Thereafter, by applying a *dimension aggregation scheme* [27], one constructs a *weighted network flatten*. Different weight computation techniques can be applied, namely frequential aggregation, binary aggregation, similarity aggregation and linear combination aggregation [13,27]. Traditional community detection algorithms based on weighted links can then be applied on the flattened network (see Figure 3.3a).

Meta clustering based methods. These approaches apply simple community detection algorithms on each dimension to extract partitions. Thereafter, proceed through the ensemble-clustering technique [15, 93] on the whole partitions computed from each dimension of the multidimensional network, to generate the final partition of the system. Figure 3.3b illustrates this process.

They depend on some parameters defining the number of nodes needed in the overlap for it to be considered as a community. The parameters are critical for the algorithm since they determine the number of communities to return.

Structural features based approaches. The third group of this family of approaches aims to extract the structural features from the multidimensional network by constructing for each dimension a *utility matrices*. Afterwards, they unify them to generate an *aggregated multidimensional utility matrix* [53,143,144]. The major disadvantage of these approaches lies in the need to specify the number of communities targeted (in terms of the use of the k-means algorithm [52]).

These approaches have the advantage of preprocessing the multigraph into a mathematical structure that promotes the simple use of classical community detection methods. However, they suffer from some limitations: sensitivity to non-relevant dimensions, loss of information induced by dimension compression, and their dependence on traditional methods for detecting communities in one-dimensional graphs. In order to consider some structural features in the communities inherent in a multidimensional network, some approaches explore the dimensions simultaneously. Thus, several methods for extending traditional community detection approaches in simple networks and quality functions for multidimensional networks, emerged.

3.4.2 Generalization of unidimensional-oriented algorithms to multidimensional networks

This family of approaches focuses on the simultaneous exploration of dimensions. They allow classical techniques (those from monodimensional networks) to deal directly with multidimensional networks.

Quality function optimization based approaches. This family of approaches focuses on the optimization of some metrics to form the hierarchical structure for multidimensional networks. More precisely, Mucha *et al.* [113] derived a generalized modularity (GM-Louvain), namely Multislice modularity, a metric to assess the quality of a given partition into multidimensional communities, as defined in Table 3.1. Liu *et al.* [104] proposed a method based on an extension of the community density in single networks [2]. They concluded that the denser a community in multidimensional networks is, the larger the outcome of this equation will be.

Random walk based approaches. This family devoted oneself to the extension of the classical methods based on random walk in monodimensional networks, namely Infomap [129] and WalkTrap [124] algorithms. Specially De Domenico *et al.* [38] introduced Multiplex Infomap (Multimap), and Kuncheva et al. [89] developed Locally Adaptive Random Transitions (LART).

Propagation based Approaches. Methods focus on the label propagation principle [5, 23]. A multidimensional version of the classic LPA [10], namely Multi Dimensional Label Propagation Algorithm (MDLPA) was intoduced in [23]. This method is based on an iterative process that is inspired by the principle of label propagation. Its major disadvantage lies in the technique of determining the relevant dimensions, since it uses the degree centrality of a node to improve F_{multi} objective function (see Table 3.1). The implication is that maximizing that objective function based on these relevant dimensions does not necessarily guarantee an optimal partitioning. In fact, if the node has a constant degree centrality over all the dimensions, the *global maximum*[1] to avoid will always be reached.

Topological organization based approaches. Several approaches of discovery communities interpret a community as the combination of a set of sub-graphs that share nodes between them. Tehrani *et al.* [146] developed Mul-CPM which is an extension of the popular Clique Percolation Method (CPM) [41] to multidimensional networks. The author rethought the basic concepts on which the original CPM is based, including cliques and clique adjacency, to handle the presence of multiple types of ties and extract overlapping communities.

Despite the fact that most of these approaches leverage all structure information across dimensions, their major limitation is that they are strongly parameterizable. Specifically, some variables are fulfilled at the input of the algorithm, such as the number and size of communities to be uncovered, the random walk length and the rate of relaxation. These parameters could greatly affect the accuracy of an approach if the values provided by the user are incorrect. Moreover, the considerably large and increasingly growing size of complex networks lead to a waste of time and resources caused by implementing partitions based on global knowledge of network.

To overcome these limitations, some scholars adopted local community detection schemes.

Similarity based approach. This approach yields communities based on similarity of nodes during their clustering process. Hmimida *et al.* [74] introduced mux-LICOD, a multidimensional version of the LICOD approach [78]. This method is based on a seed-centric approach, using the Jaccard coefficient as similarity measure to obtain real leaders. Recently, Li *et al.* in [100] implemented M-ALCD (Multi-Layer Attribute and Local Community Detection), an attribute-based discovering method, based on the attribute similarity between a node and its neighbors in the corresponding structure. The method pre-defines a maximum number of communities required as the condition for terminating the algorithm.

In most of interconnected systems, there is a correlation between entities due to the fact that information travels not only among vertices of the same dimension, but also between pairs of dimensions. Thus, intuitively, the activity of a node may evolve from one dimension to another.

[1] Maximizing F_{multi} does not necessarily guarantee optimal network partitioning.

However, most of the above-mentioned approaches do not take into account either the correlation or the activity of the nodes in multiple dimensions. To overcome these limits, some studies have already been done, as described in the following section.

3.4.3 Overview on multidimensional network- based community detection algorithms

Table 3.3 presents a n overview of some usual community discovery algorithms identified. Some of the comparison criteria are of importance to better situate the method proposed in chapter 5. The first criterion is the implemented technique or strategy of detection. The second criterion is Information nature referring to the consideration or not of attributes. In the third criterion, the overlapping nature of communities is stated. We indicate in the fourth criterion whether the method is parameterizable or not. The last column specifies the fact that methods consider the activity level across dimensions, through Dimension relevance criterion. The function $f(M)$ in the complexity column means that the complexity of the algorithm is unstable, as it depends on another method M to which it is applied to determine partitions. The Guangyao's method [161] was applied on three algorithms, namely Louvain [40], OSLOM [91] and Infomap [129]. Thus, M could be one of them in this case.

According to the table 3.3, the majority of the methods only take into account structural information. A recent work [100] includes nodes attributes to identify more semantic communities. It also observed that all of the methods are parameterizable. This reflects the importance of providing certain parameters by the user in order to perform results. So, the quality of the parameters affects the quality of the outcomes. Therefore, if the parameters are wrongly supplied, the results will not be satisfactory. It is therefore interesting to have a prior knowledge of the network.

Table 3.4 presents the advantages and drawbacks of the above methods listed in Section 3.3. Specifically, since the advantages are varied, the focus is on the drawbacks, which are grouped into four criteria. The first outlines the methods that set the number and size of communities. The second criterion presents methods that are insensitive to relevant dimensions because they do not deal with the level of a node's activity in a dimension. In the third criterion, the performance limit of the methods is mentioned, i.e. methods with high complexity. The last criterion concerns the instability of the method due to three parameters, namely the pre-established order of dimension assessment, the various existing methods applied on the method, and the initialization of some

Table 3.3: Summary of some multidimensional community discovery algorithms

Algorithm	Technique	Information Type	Overlap	Param.	Dimension Relevance	Complexity	Year	Ref.
M-ALCD	Similarity maximization during community expansion	Attribute	No	Yes	No	$O(K\|C\|^2)$	2019	[100]
Mul-CPM	Clique Percolation method	Structure	Yes	Yes	No	NP-hard problem	2018	[146]
Liu W. *et al.* (2018)	Optimization based on density metric	Structure	Yes	Yes	No	$O(K \times mlog_3 n)$	2018	[104]
MDLPA	LPA function objective optimization	Structure	No	Yes	Yes	(mn)	2017	[23]
LART	Random walk	Structure	Yes	Yes	No	(mn^2)	2015	[89]
Mux-LICOD	Seed centric	Structure	Yes	Yes	No	$O(m+nlog(n))$	2015	[74]
Multimap	Random walk	Structure	Yes	Yes	Yes	$O(mlog^2 n)$	2015	[38]
MultiMOGA	LPA Local search on neighbors	Structure	No	Yes	No	$O(m+dn^2)$	2014	[5]
Guangyao *et al.* (2014)	Dimension integration by a Unified	Structure	No	Yes	Yes	$f(M)$	2014	[161]
ABACUS	Meta clustering based on frequent itemset mining	Structure	Yes	Yes	Yes	$O(2n)$	2013	[15]
Carchiolo V. *et al.* (2011)	Optimization of the objective function based on modularity	Structure	No	Yes	No	$O(nlog(n))$	2011	[29]
Aggregation	Optimization of the objective function based on modularity	Structure	No	Yes	No	$f(M)$	2005	[27].

parameters.

3.5 Evaluation techniques and applications

3.5.1 Evaluating community detection methods

Estimating the performance of a method consists in checking whether expected communities are obtained or not. Given a community detection method, how can we attest that it performs well? And how to compare two different methods that eventually optimize two different objectives functions? These are the main questions addressed by evaluation techniques for community detection. According to the knowledge about the ground truth, there are two kind of community detection methods' assessment.

Evaluation with ground truth. When the ground truth is known, the evaluation consists in comparing how well the algorithm recovers the known communities. Fortunato [62] divides measures in three categories, based on *pair counting, cluster matching* and *information theory*. Among many others [60], we choose to describe the following pair counting measures, depending on the one used in this thesis: the Jaccard Index, the Mutual Information and its variants. Pair counting consists in computing the number of pairs of vertices which are classified in the same (different) clusters in the two partitions [62]. Let us consider $S = (s_1, s_2, ..., s_{cs})$ and $T = (t_1, t_2, ..., t_{ct})$ being two partitions of the network G with cs and ct clusters respectively. Jaccard index as defined in Chapter 2 determines the similarity between two clusters s_c and t_c as following:

$$Jacc(s_c, t_c) = \frac{|s_c \cap t_c|}{|s_c \cup t_c|} \tag{3.2}$$

The *mutual Information (MI)* [6] is a measure allowing to compare two partitions by quantifying their common information. The mutual information of two partitions S and T is given by:

$$MI(S,T) = \sum_{i=1}^{|S|}\sum_{j=1}^{|T|} P(i,j) log \frac{P(i,j)}{P(i)P(j)} \tag{3.3}$$

Where $P(i) = \frac{|s_i|}{N}$ is the probability of a node in the community s_i in the first partition, $P(j) = \frac{|t_j|}{N}$ is the probability that a node in the community t_j in the second partition and $P(i,j) = \frac{|t_i \cap t_j|}{N}$ the joint probability. Since the values of the mutual information are not in ranging between 0 (different partitioning) and 1 (identical partitioning), Lancichinetti et al. [91] normalized them as follows:

$$NMI(S,T) = \frac{2MI(S,T)}{H(S)+H(T)} \tag{3.4}$$

where $H(S)$ is the entropy of S.

Evaluation without ground truth. In some situations, the ground truth is unknown. Therefore, comparing two algorithms consists in comparing their quality function values. The widely and popular measure is the modularity objective function and its variants, as defined in Section 3.2. A community structure is neither always present nor easy to detect. It is therefore possible for graphs with ground truth to check the presence of the community structure before applying the quality measures. This topic is the subject of some research in [80] and [28], using the notion of *consensus* to decide whether or not a network has a community structure. More precisely, different partitions are found in the network by using many executions of a non deterministic algorithm and the frequent nodes of the same communities form the consensus (also called community cores). It has been shown in [28] that in a network without community structure, community cores are trivial, either containing all the nodes of the graph or one node each.

3.5.2 Applications

In recent years, many researchers devoted attention to the problem of community detection. The interest of this detection is multiple, and we can highlight as examples the following applications.

Recommendation: Identifying communities of customers with similar interests in online sales and purchasing channels between customers and products, such as *Amazon11* , enables the development of more effective recommender systems [128], in order to better respond to customer needs and improve market opportunities. In addition, the identification of agricultural practitioners having a preference for a specific type of crop, in order to promote the exchange of experiences through recommendations of suitable treatment products and soil types, to increase yields while overcoming climate change.

Table 3.4: Summary table of advantages and limits of some multidimensional methods

Algorithm	Benefits	Limits
ABACUS [15]	Scalable, Successfully identify a high-resolution partitioning	The size and number of communities to be identified is known
Carchialo [29]	Scalable, multi-resolution and naturally gives a hierarchical decomposition of the network	
PMM & SC-ML [53, 144]	Meaningfulness results : Flexibility in combining structural features	
M-ALCD [100]	Scalable and efficient because it is based on attributes and interaction of nodes	
Aggregation [27]	Easy and simple to apply	Sensitivity to non-relevant dimensions , Instability
Mux-LICOD [74]	Better performance	
GM-Louvain	Adaptable to a broader class of networks	Higher complexity - Low performance because the correlation depends on topological features which could be complex to compute if the graph is large-scale
LART [89]	The length of the random walk promotes the discovery of communities across all dimensions	
MDLPA [23]	Considers the dimension relevance	Instability due to the undesirable global maximum
Multimap [38]	Accuracy of results as it reveals smaller modules with more overlap	Instability due to its nondeterminism
MultiMOGA [5]	Flexible of use: possibility of specifying the order of exploration of dimensions	Not efficient: it assumes the existence of a partition on each dimension; Instability due to the pre-established order of dimension assessment
Guangyao *et al.* [161]	Simply implementation; more tools and techniques to use on the unified matrix	Unstable because of variables existing methods applied on it

Link prediction: This task involves identifying new interactions between members of a Social Network that are likely to occur in the close future [102]. Thus, information on future interactions can be extracted from the network topology, in particular, the community structure in order to predict these relationships and anticipate containment measures or new habits to be taken.

Spread of epidemics: It is crucial to understand how an epidemic expands once it has occurred in order to control it. We are recently witnessing the Covid-19 pandemic, a pandemic of an infectious emerging disease called coronavirus 2019 (Covid-19). Individuals back from a trip are possibly in direct contact with their close surroundings. Some criteria related to distance induced by the means of communication in addition to geographical distances are incorporated in recent models of epidemic spread. Some of these models have shown that community structure is a key factor in the behavior of network percolation processes such as the spread of an epidemic. Indeed, the structure of the community can both impose and inhibit the processes of dissemination [118].

Information spreading: A major feature of Social Networks is the dissemination of information, such as rumors, stories and opinions. Indeed, the processes of this dissemination are now affected by the community structure [103]. More specifically, the detection of dynamic communities related to hot topics, linking content designers and disseminators, allows a rapid dissemination of information.

Prediction of cellular functions: In biology, Protein-protein Interaction Networks (PINs) are characterized by a noticeable modular organization that reflects the functional associations between proteins. Thus, a group of proteins that collaborate on the same cell function correspond to communities. The detection of these communities and the analysis of PINs are thereby a precious tool for functional prediction.

Detection of terrorist organizations : Understanding the hierarchies within criminal organizations and discovering the members who play a central role is necessary to support law enforcement agencies. In this context, the study of criminal networks using communication tracks from telephone call recordings reveals the underlying community structure that will be exploited by forensic investigators [58].

3.6 Conclusion

Throughout this chapter, the concepts of community detection methods have been explored, including the definition of quality functions together with a description of detection approaches for both mono- and multidimensional graphs. Particular emphasis was placed on the applications of this crucial area of research, which is the detection of communities in complex graphs. Indeed, it has been a matter of reviewing the main solutions for detecting communities in undirected, directed, attributed and multidimensional graphs. The study of the existing situation allowed us to identify five shortcomings that, if taken into account, will improve the communities of interest discovery from complex networks.

Firstly, algorithms for detecting communities in directed and attributed graphs generally focus on the topological (link density) and semantic (attribute similarities) characteristics of the network by studying, for instance, triangles, homophily, centrality or distance measures and/or similarities between nodes, without really taking into account the directionality of links. Indeed, this topological modeling disregards the triad-based seed-centric interest between network elements, because it does not implement the idea that an edge from a low out-degree but high in-degree node to an opposite case node should be considered of a bigger value, as described in subsection 3.3.2. Secondly, the existing detection methods are for the most part non-deterministic. This non-deterministic behavior is caused by this variation of methods or the random choice of some parameters. As a result, there is a glaring instability in the obtained partitions. Thirdly, all methods are highly parameterizable, as they require certain parameters to be filled in, in advance, which when misinformed, lead to incorrect results. In fact, providing accurate values input parameters, including the number of communities, requires a priori knowledge of the network to be analyzed. However, in practice, such knowledge is not always available. The fourth shortcoming is that the interest in multidimensional graphs is not taken into account since only a few methods consider the relevance of dimensions in the clustering process. These focus on the degree of nodes that constitute a limit. Indeed the importance of a dimension for a node cannot be limited to its degree because if this node has the same degree in all dimensions, then all these dimensions would be relevant to it. This consideration thus constitutes a limitation to which it would be important to remedy.

After identifying these four shortcomings, the next chapters will therefore present the contributions of this thesis, aiming to overcome the limitations of existing approaches and propose effective solutions for the detection of communities in the context graphs of our study

CHAPTER 4

Community of interest detection in directed networks

4.1 Introduction

In Chapter 3, we presented some examples of partition quality functions, then we described in two sections, the community detection methods in simple graphs and in multidimensional graphs. Concerning the first group of methods, we have dislocated the description in three parts: detection methods in undirected graphs, in directed graphs and then in attributed graphs. We also proposed a method of community detection in directed networks in [44] that improves Edge Betweenness method [68]. It appears that several of these methods for directed graphs do not focus on the interest on incoming links of nodes. Yet incoming links are more important than outgoing ones, as stipulated by modularity [99, 110], a favorite function in the literature to assess the quality of a partition. Indeed, the link density on which existing methods predominantly have been focusing is limited by the link directionality, which gives a non-objective meaningfulness or interpretation to the resulting communities.

Among these approaches, some investigated the classification of vertices with attributes [160]. Others approaches [37] stressed on the combination of both relational and attribute data without taking into account the directionality of the edges. This can lead to an imbalance of the communities, since there is no consideration of the three types of informations simultaneously (both link density and directionality, as well as the attributes of the nodes).

In this chapter we present our first two contributions, one being the detection method of triad-based clusters in Section 4.2, and the other being a hybrid model of community detection in Section 4.3, designed to address the above weaknesses.

4.2 Community detection using triads

We study in [45, 48] two methods of detection of triad-based communities. In [45], we proposed a method for optimizing the kernel degree metric, initially defined in [47]. This method is executed in two steps: kernel extraction and community building. The kernel extraction step is the most critical part of the clustering process. Indeed it consists in the optimization of the Kernel degree metric, namely K_{uv} (see Equation 4.1 in Section 4.2.3 below). After determining v as the maximum incoming degree node, it is a question of computing the K_{uv} metric for all the other nodes u of the graph. Then after estimating the average K_{uv}, we proceed by a kernel improvement heuristic by choosing to put the node u in the kernel of v for which K_{uv} is higher than this average. After a certain pre-specified number of steps for which K_{uv} does not cross any more the process is interrupted. This method is an improvement of method in [50] because instead of going link by link to compare the kernel degree that optimizes the kernel in training, it computes the average of the kernel degrees obtained on the kernel in training to optimize it. This approach improves the complexity of the method described in [47] and is applicable to larger datasets.

One of the most obvious drawbacks of the method in [45] is the over propagation problem. The main reason behind the over propagation is the rapid and aggressive expansion of the core of some communities. The weaker in-degree nodes have little chance to grow. The extreme case of the over propagation is one giant community, dividing all the nodes into one class. However, comparing with other cases, the one giant community is not so bad. It at least notifies if the community detection failed in this attempt and needs another try. To overcome this drawback, we improved it in [48]. The method proposed there builds kernels no longer on the basis of the optimization of the Kernel degree function, but on the basis of the structural equivalence of the nodes. Thus, the nodes having the common neighborhood are eventually members of the same kernel. The kernel degree measure is used to determine the threshold for selecting nodes to be kernel members. Therefore, the threshold for creating these kernels is not set as a parameter anymore. It is derived from the standard deviation and interclass inertia based on the kernel degree. An extension of this method was elaborated in [46], thus constituting the subject of the first contribution addressed in Section 4.2.2.

As a reminder, we are interested in the detection of communities of interest in directed graphs. This method has several objectives: first of all, it should create communities of interest in directed networks based on topological features of the complex networks namely *homophily* [111] and *pref-*

erential attachment [9], and secondly, it should promote the impact of the incoming links in the communities through the definition of a measure of similarity namely the *kernel degree*, which expresses the interest based on the same information perception by a group of nodes. Thus, we focus on community of interest referring to entities centered round kernels. In other words, the interest reflects the fact that kernels enjoy communities' members confidence. The higher the value of kernel degree is raised, the more the nodes of the kernel are similar through the trust in the same kernel (same in-neighborhood). This measure is then used in the second proposal of this chapter, as investigated in Section 4.3, to take into account the attributes between the nodes of a community, in order to strengthen the cohesion between members of the same community.

We begin this section by defining the notion of community of interest based on triads. Stress that this concept is very important because it is the baseline for designing our solution.

4.2.1 Topological Community of interest definition

Triads were initially studied by Wasserman *et al.* [151] in social network analysis. They are considered as wedges, i.e paths of length 2 by Klymko [86] who focuses on the density in triangles for identifying communities. This method has the advantage of detecting denser communities. However, triangle density would be a restriction since the triangle is an elementary clique, and finding cliques in a graph has been proved to be an NP-complete problem [19]. To remedy to this complexity, algorithms based on Clique percolation [41] may work well for graphs characterized by a large number of cliques, like certain social networks, whereas it may give poor results otherwise. In directed graphs, the process of extracting communities should take into account either "in"or "out"directionality of the edges for meaningful interpretation. For example, in citation graphs, the incoming degree (also called the number of citations) is used to quantify the importance of scientific publications [66]. Therefore, it becomes interesting to specify those of nodes centered around *kernels* (set of influential nodes inside a group) according to *in-direction* of the edges. This in-direction reflects directed triads in directed networks. Moreover, Yang and Leskovec [154] show that more simple criteria such as conductance and enrollment rate in triads, in particular, often better characterize a community structure than modularity.

Then, in this chapter we propose a community of interest detection method based on topology, unlike the one in [32] based on ontology, which shows that directed triads enlarges the possibility to imply considerably in-degree than out-degree of nodes, as was expected the directed modularity function. Our method is based on the common neighborhood of nodes and identify groups of nodes dense in triads.

Considering a given directed graph $G = (V; E)$ with $n = |V|$ the number of vertices and $m = |E|$ the number of edges. An edge e_{ij} connects vertex v_i with vertex v_j. We now give the definition of the community of interest in directed networks, based on some topological features namely high clustering coefficient and structural equivalence. In this context of directed networks, the topological definition of community is as in the following:

Definition 4.2.1. ***(Community of interest)****. A community of interest in a directed graph refers to a structure in which nodes are both dense in terms of triads, but also similar since they point to the same kernel.*

We refer to *triad-based clusters* to express communities extracted from our model, because these communities express subgraph engendered by kernels and whose nodes are densely tied by triads.

4.2.2 Triad-based model

The Triad-based community detection method we propose in this section aims to generate communities of interest. The intuition behind this method is that the nodes of the same community follow a set of nodes of interest because they subscribe to their ideology or consider their opinions. The interest is expressed by the "in-direction "of edges towards the kernels. Hence the kernels have important incoming degrees. Indeed, the nodes of the same community are interested by the kernel with that they have a maximum of links which point to it. Thus the incoming direction of these links reflects the fact that they trust kernels. When an entity's in-degree is maximal, it indicates that the number of that entity/user's subscribers reflects his popularity. For instance, in a social network such as Facebook, there are subscribers who seldom publish, but are regularly followed by a large number of fans. Consider a citation network in which the nodes represent the authors and the links represent the relationship ‶ a quotes b ″. In such a network, the community of interest corresponds to the set of authors interested in a specific research topic. Such specific research topic is implicitly derived from the kernels which are effectively made up of the authors pioneers of the field. Hence, the fundamental metric on which our work is based relies on the *in-degree*, being the number of incoming edges to node. Therefore, our model, based on the seed-centric approach [78], uses the in-degree of nodes and their common neighborhood according to triads as shown in the grey bottom part of Figure 2.2. It is inspired by Wang's GREEDY method [150]. Our triad-based model keeps the same constructive idea of the communities, but relies on a kernel score criterion based on dense triads to build the kernels before migrating the non-kernel nodes to the kernels thereby forming the final communities. The concept of kernel related to nodes of interest [150] is outlined in Definition 4.2.2. It considers a kernel as set of nodes having more connections to/from the kernel than a vertex outside the kernel does. Unlike this view, the kernel proposed here focuses on the structural equivalence and limit the over propagation through a threshold as follows:

Definition 4.2.2. ***(Kernel)****. A kernel is a set of vertices with the same neighborhood, such that these neighbors expand gradually inward the kernel, according to a threshold σ.*

Formally, the kernel K fulfills the following properties:

- $K = \{v_1, ..., v_i, ..., v_{|K|}\}, \quad v_i \in V$
- $\forall v_i, v_j \in K \, , \Gamma_i^{in} \simeq \Gamma_j^{in},$
- $\forall i \neq j \backslash v_i, v_j \in K, K_{ij} > \sigma.$

The first property states that the nodes of the kernel are nodes of the graph. The second property specifies that the kernel nodes have almost the same neighborhood. In the third property, the kernel nodes are similar, because they are subject to peers whose kernel degree measure valuation is above a σ threshold.

We describe the triad-based community model through the Algorithm 1 below. The steps that it consists in are grouped into the following 3 main phases:

1. Kernel candidates' generation (from step 1 to step 3) at the end of which the dictionary structure $KDict$ contains these candidates;
2. Kernel extraction (step 4 and step 5) where t kernels formed by triads are extracted from the candidates;
3. Community computing process (Step 6) where kernels are extended by migration of the non-kernel nodes to kernels.

We will use Wang's network [150], shown in Figure 4.1, to illustrate the steps of the proposed method. It is an extract from Twitter and we named it Subtwitter in this thesis for sake of simplicity.

It contains 14 nodes and 32 edges. The notations are simplified by abbreviating the names of the entities as follows: Demi Moore (DM), Oprah Winfrey(OW), Al Gore (AG), Barack Obama (BO), Ashton Kutcher (AK).

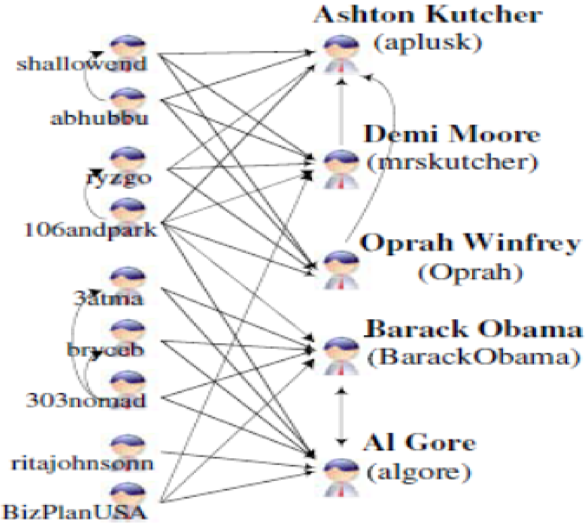


Figure 4.1: Subtwitter: Extract from Twitter social network used in [150]

Algorithm 1 The triad-based community model for community detection

```
Require: Directed graph $G = (V, E)$
Ensure: List of Communities $C = \{C^{(1)}, ..., C^{(t)}\}$
Step 1: Compute In-degree pruned Central List $CL$ according to the degree average of the graph. The list is in decreasing order of degree
Step 2: Compute Kernel Dictionary $KDict$ based on each distinct pair of $CL$ such as $KDict = [((v_i, v_j), K_{ij})]$
Step 3: Compute Interclass inertia vector $I$ according to $K_{ij}$ values of $KDict$
Step 4: Compute a threshold $\sigma$ being the standard deviation of the vector $I$
Step 5: Extraction of kernels as described in Algorithm 2 from Line 3 to Line 13.
Step 6: Community building through non-kernel nodes migration, as described in Algorithm 3
```

4.2.3 Step 1: Kernel candidates' generation

This step consists in generating the list of eligible nodes to eventually belong to kernels. It spreads out into three subtasks: extract the list of node in-degrees through computing a degree centrality list; then compute the values of the weights between pairs of nodes from the previous list, through computing kernel dictionary; finally grouping these couples according to their neighbors' similarity through the computation of an inter-class inertia vector.

In-degree centrality list computing. This step consists in determining a list of nodes sorted in the descending order of their in-degree; that list is called *Centrality List* (CL). So that those with maximal in-degree are more eligible than those with a low in-degree, since we focus on the preferential attachment property of real-world complex networks. Then, pruning from the list *in–pendant* and *in–isolated* vertices i.e. those of nodes with an *in–degree* below the *in–degree graph average*, as inspired by Steven L. and al. [90] who defined a pendant as vertex with a single neighbor which has degree 1. This filtering step improves performance and allows simplifying assumptions later when deciding whether to include a vertex into a kernel. For instance, in a citation network, an *in–pendant* or *in–isolated* vertex corresponds to an author whose the research area does not interest other researchers, so removing these nodes with an in-degree below 2 improves the processing speed and produces more cohesive communities of interest later. For illustration on the Subtwitter network, the CL contents is: $CL = ['AG', 'BO', 'AK', 'DM', 'OW']$ because they have an in-degree above 2, being average degree of the network.

***Kernel dictionary computing*.** This step consists in computing the strength of similarity of nodes that could be membership of the same kernel. To compute that strength, we defined a measure called *kernel degree*, which computes the score of kernel, in order to determine whether the ending nodes of every edge (v_i, v_j) such that $v_i, v_j \in CL$, will belong to the same kernel or not, as described in Definition 4.2.5 below. This measure consists in two metrics: The Neighborhood Overlap as defined in Definition 4.2.3 and the Triad Weight as defined in Definition 4.2.4.

Definition 4.2.3. ***(Neighborhood Overlap)***. *Given two vertices v_i and v_j. The neighborhood overlap NO_{ij} is a Jaccard Index variant [140], which consists in measuring neighborhood similarity of two vertices v_i and v_j so that they could belong to the same kernel.*

$$NO_{ij} = \frac{|\Gamma_j^{in} \cap \Gamma_i^{in}|}{|\Gamma_j^{in} \cup \Gamma_i^{in}| - \theta}$$

Unlike the Jaccard Index which does not consider the connectivity between the nodes because it just computes the common neighbors of 2 vertices v_i and v_j, Neighborhood Overlap integrates the fact that there could be or not an edge between v_i and v_j. That is why we use the θ parameter in the denominator to compare 2 similar kinds of neighbor sets. In fact, according to the numerator, one vertex can belong to the in-neighborhood of another, and vice versa. θ can take different values 0, 1 and 2, depending on the connectivity of v_i and v_j vertices.

- $\theta = 0$ if $(v_i, v_j) \notin E$ and $(v_j, v_i) \notin E$
- $\theta = 1$ if $(v_i, v_j) \in E$ and $(v_j, v_i) \notin E$
- $\theta = 2$ if $(v_i, v_j) \in E$ and $(v_j, v_i) \in E$

Definition 4.2.4. ***(Triad Weight)****. The Triad Weight TW_{ij} of any edge e_{ij} in graph G can reflects the percentage of directed wedges linking the nodes v_i and v_j. It computes the ratio between the number of triads crossing both v_i and v_j and the number of triads in which v_j is involved such that:*

$$TW_{ij} = \frac{|\Delta_{ij}|}{|\Delta_j|}$$

where $|\Delta_{ij}|$ represents number of triads crossing both v_i and v_j according to the scheme presented in the bottom of the Figure 2.2 and $|\Delta_j|$ to represents the number of triads in which v_j is involved. Let us assume that v_j is the target node of the edge e_{ij}. The idea behind the kernel degree measure is that kernel nodes must have high affinity with each other, based on their common neighborhood. For this reason, we combine the two neighborhood metrics above mentioned to develop the proposed new similarity measure, as defined in Definition 4.2.5.

At first glance, the triad weight and the neighborhood overlap simultaneously compute the common neighborhood of nodes v_i and v_j. Nevertheless it would be necessary to mention the difference between the two concepts. While the triad weight considers the directionality of edges on the numerator through the directed triads, the neighborhood overlap considers it on the denominator because it favors situations in which kernel nodes are not tied. These considerations of directionality reinforce the strength of nodes' similarity. We have shown in Section 4.2.6 that the exclusive consideration of one or the other (either triad weight or neighborhood overlap) does not favor the high kernel score. Their combination leads to scalable results. Indeed, our empiric tests on metric taken separately show the superiority of *Kernel Degree* on various networks, as evaluated in section 4.2.6. Thus, we define Kernel degree in the following way:

Definition 4.2.5. ***(Kernel Degree)****. Intuitively, Kernel degree measures the strength or the score of the kernel vertex similarity.*

Its value between a pair of vertices v_i and v_j is evaluated using Equation 4.1.

$$K_{ij} = TW_{ij} \times NO_{ij} \tag{4.1}$$

In Equation 4.1, the first term is based on triads, and promotes the *Triad Weight* in a kernel; Given two vertices v_i and v_j, a standard way to compute the percentage of triads they form together is to compute the ratio between the total number of triads in which the pair of vertices is included (numerator) and the total number of triads in which vertex v_j is contained (denominator). The second term promotes the *Neighborhood overlap* of v_i and v_j vertices. The values of kernel degree are represented by a kernel dictionary named ($KDict$), whose items are structured through the following format: ($key\_dict; value\_dict$). key_dict is any *unordered* pair of nodes from CL pruned list, and $value\_{d}ict$ is the corresponding kernel degree K_{ij} of these pairs. Formally, $KDict = [((v_i, v_j), K_{ij})]$.

For illustration, $KDict$ = *[((‘DM’, ‘OW’),* 1.6*), ((‘AG’, ‘BO’),* 0.595 *), ((‘AK’, ‘OW’),* 0.32 *), ((‘AK’, ‘DM’),*0.267*), ((‘BO’, ‘DM’),* 0.0635 *), ((‘AG ’, ‘DM’),* 0.057 *), ((‘BO’, ‘OW’),* 0.0158 *), ((‘AG’, ‘OW’),* 0.0143 *), ((‘BO’, ‘AK’),* 0.013 *), ((‘AG’, ‘AK’),* 0.012 *)].*

Let us remember that a *kernel* in this thesis is a set of nodes owning a common central in-degree overlapping neighborhood. This task of extracting kernels focuses on determining those of nodes more eligible to belong to kernel via interclass inertia.

Interclass inertia computation. Given that the clustering main goal is to form homogeneous groups, the measure used here to divide objects into two groups, those eligible to belong to a kernel and those not eligible is *Inter-class Inertia.* A list I of the inter-class inertia values is computed on basis of $KDict$ dictionary. Indeed, high interclass inertia values indicate that objects tend to be more dissimilar, and consequently should belong to distinct groups. To delimit node in two groups, we compare values from Inter-class Inertia List to a computed *Standard Deviation* σ on I. This way, vertex pairs (i, j) of $KDict$ whose Inter-class Inertia value is larger than σ are more eligible to belong to kernels. The Inter-class Inertia between 2 sub-groups G_1 and G_2 is expressed as:

$$I(G_1, G_2) = |G_1|(\mu_1 - \mu)^2 + |G_2|(\mu_2 - \mu)^2 \tag{4.2}$$

$|G_1|$ and $|G_2|$ are respectively the number of edges in groups G_1 and G_2. μ_1, μ_2, and μ are respectively the average *Kernel Degree* for G_1, G_2 and G. The Subtwitter Network in Figure 4.1 presents distinct groups G_1 and G_2 respectively as the following, and the corresponding Inter-class Inertia of *KDict* as : for G_1 = *{(DM, OW)} and* G_2 = *{(AG, BO), (AK, OW), (AK, DM), (BO, DM), (AG, DM), (BO, OW), (AG, OW), (BO, AK), (AG, AK)}*, the Inter-class Inertia for these groups is 1.987. Then, the following pair of nodes in $KDict$ list moves from G_2 to G_1, and their contents become: G_1 = *{(DM, OW), (AG, BO)} and* G_2 = *{(AK, OW), (AK, DM), (BO, DM), (AG, DM), (BO, OW), (AG, OW), (BO, AK), (AG, AK)}*, and the Inter-class Inertia for these groups is 1.705. We change the G_1 and G_2 contents and so on. The interclass inertia vector is progressively computed and its contents are presented as follows: I = [1.987, 1.705, 1.359, 1.162, 0.844, 0.627, 0.439, 0.297, 0.186, 0.131]. Afterwards, a threshold helpful for kernel extraction process is computed, as detailed in the next section.

4.2.4 Step 2: Kernel extraction approach

This section firstly presents the methodological principle followed by the properties fulfilled by kernels, and finally it describes the threshold on which the kernels are structured.

Methodological principle. The phase begins by initiating kernels with distinct pair of vertices possessing the highest corresponding Inter-class Inertia, through the mileage of the $KDict$ dictionary. Given that an initiating vertex r of a kernel t between the initiating pair of vertices $\{r, u\}$. If a vertex p in $KDict$ is coupled to another one q with whom the Kernel Degree K_{pq} is lower than its Kernel Degree K_{pr} with the initiating kernel vertex r, p immediately migrates to that kernel t. So the kernel t will be made of $\{r, u, p\}$. Then those already belonging to the kernel will not be treated in the future steps. The vertices belonging to the kernel own almost the same neighbors. The approach proposed here makes use of a new concept Kernel Degree K_{ij} as defined in Definition 4.2.3, that measures the strength of a kernel according to a threshold. This concept is based on the triadic membership to emphasize the semantic proximity that ties kernel members conducting to efficient centralization of information over the network.

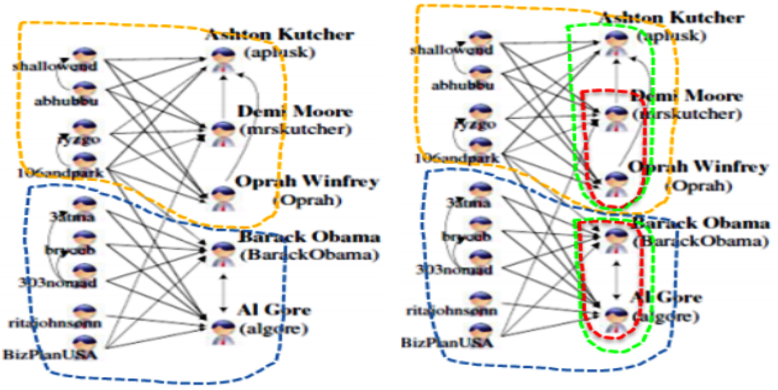
Kernel properties. We require that the kernel fulfills the following properties:

1. Every kernel contains distinct pair of vertices with inter-class inertia upper than a threshold.
2. The kernel vertices have higher *Kernel Degree* values, proportionally to the degree distribution of the graph.
3. Given an initiating pair (i, j) and a border vertex k in a kernel, the neighborhood overlap cardinality of (i, j) must be higher than the neighborhood overlap cardinality of any neighbor t of (i, j, k). Formally, Given $\forall (i, j) \backslash i, j \in CL$, and $k \in K$, $\mid \Gamma_{i,j} \cap \Gamma_k \mid \geq \mid \Gamma_{i,j} \cap \Gamma_t \mid$, where $\Gamma_{i,j} = \Gamma_i \cap \Gamma_j$.

Standard deviation σ. To compute Kernels, we focus on a threshold, which is the standard deviation from interclass inertia list I. Unlike the well-known meaning of the standard deviation, we observe during the experimental phase that the higher the standard deviation σ computed from a set nodes, the more likely they possess an almost common neighborhood. As a matter of fact, as illustrated through the Table 4.3, a lower standard deviation indicates that these vertices have a quasi-null common-neighborhood cardinality. Because of the power-law degree distribution in real-life networks, very little nodes get a high in-degree widely above the in-degree average. We make the assumption that according to [97], there tend to be a few "hub"vertices with a very high degree and great number of vertices with a much lower degree. In the case of directed graphs, the concept of hub vertices depend on the in-degree or the out-degree value. This study stresses on in-degree vertices, meaning that they receive more information from the other vertices than "non-hub"vertices. The standard deviation is expressed as:

$$\sigma = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(x_i^2) - \mu^2} \tag{4.3}$$

where $\mu = \frac{1}{n}\Sigma_{i=1}^{n} x_i$ indicates s_i average (or mean), and x_i indicates every element of the interclass inertia array. A kernel is initially made of a pair of vertices, and expands progressively by adding vertices which are in couple with kernel members, whose the corresponding *Kernel Degree* value is above σ. This leading to an expansion of the starting kernel. As shown in Figure 4.2b, initial kernels are surrounded of red dashed lines, and grow progressively (see green dashed lines in Figure 4.2b). When a node already belongs to a kernel, it is omitted later in the list of eligible nodes, because discovered communities are disjointed. We make use of a denoted Key variable which could be any pair/couple of vertices of $KDict$. In fact, each eligible key is integrated into a new kernel, after confirming its non-existence anywhere in the list of kernel vertices. This merging step improves performance and allows simplifying assumptions later when deciding whether to choose the favorite kernel by a *non-kernel* vertex (the vertex not belonging to a kernel).



(a) Output by Wang's algorithm

(b) Output by our triad-based algorithm

Figure 4.2: An illustration of outputs from the Subtwitter network in Figure 4.1

The implementation for kernel is presented in Algorithm 2, which extracts a list of kernels named $ListK$, from the overall nodes of the graph. $standard\_deviation(I)$ is the function computing the standard deviation from the inter class inertia vector I. $inkey$ represents a boolean array of distinct nodes from $KDict'$ reflecting whether they are in a kernel or not. $neighbor(e)$ returns the other member of the pair of nodes defined by key in $KDict$, orderless.

Algorithm 2 Kernel extraction

```
Require: Directed graph $G = (V, E)$
Require: $I$ inter-class inertia vector //corresponding to the vector $I$ in the explanation above.
Ensure: Structured-by-key Kernels set called $ListK$
Initialization : $\sigma \leftarrow standard\_deviation(I)$ , $ListK \leftarrow \emptyset$;
$\forall$ $distinct\ e \in KDict.key\_dict$, $inkey[e] = False$
$KDict' \leftarrow KDict$ such as $KDict.value\_dict > \sigma$
while $\exists e \in KDict'.key\_dict / inkey[e] = False$ do
    if $inkey[neighbor(e)] = False$ then
        $Key \leftarrow (e, neighbor(e))$
        $ListK \leftarrow ListK \cup Key$
    else
        $K \leftarrow K \cup \{e\} / neighbor(e) \in K$ and $K \in ListK$
    end if
    $inkey[e] \leftarrow True$
    $inkey[neighbor(e)] \leftarrow True$
end while
return$ListK$
```

The standard deviation value for that network is $\sigma = 0.62$. It is the threshold on which kernels are to be built. The model computes the first kernel K_1 initialized by nodes '*DM*'and '*OW*'for which the associated inertia in I is $1.987 \geq 0.62$; thereafter, K_1 is extended by the node AK because AK is in the couple with the other nodes already assigned to kernels (See $KDict$ in the above **Step 1**), with corresponding inertia of 1.359, 1.162 (See I list in Interclass inertia paragraph above); the second kernel K_2 is initialized by '*AG* 'and '*BO* 'for which the associated inertia in I is $1.705 \geq 0.62$. The process is repeated on the other i values in I for which $I[i] \geq \sigma$; and if the corresponding $KDict[i]$ pair nodes are already keys or associated values of keys, they are just omitted. Figure 4.2b shows in green dashed lines the kernels.

4.2.5 Step 3: Community computing process

After extracting kernels, the other nodes not into the kernels, called *non-kernels* vertices, remain. The process of generating *global communities* (communities containing both kernels and non-kernels vertices) is an iterative optimization process of a function named *Node community Index*(*NCI*) (see Definition 4.2.6). It consists in migrating *non-kernels* vertices to the kernel with whom they have a maximal number of links as shown in Equation 4.5. NCI is based on the number of connection each non-kernel vertex owns with the kernel.

Definition 4.2.6. ***(Node community Index)***. *NCI is a node membership score defined for a node x to belong to the kernel K as the ratio between the number of outgoing links from x pointing to K and the minimum between the number of outgoing edges from x and the number of nodes in the kernel*

K; then NCI is defined as $NCI : V \times K \longrightarrow \mathbb{R}_+$. Formally, NCI is defined in Equation 4.4 below.

$$NCI(x,K) = \frac{m_{out}(x,K)}{min(m_{out}(x), n_K)} \tag{4.4}$$

This measure corresponds to an extension of the conductance measure defined in Section 3.2 in Chapter 3. Indeed, in the NCI formula, the community is restricted to a unique node, unlike the community considered in the conductance formula. NCI consists in determining the membership of a vertex x, depending on three parameters: $m_{out}(x,K)$ is the number of *outgoing* edges from x pointing to a kernel K, $m_{out}(x)$ is the total number of *outgoing* edges from x or its out-neighborhood cardinality, and n_K is the number of vertices in the kernel K.

A vertex x migrates to kernel K^* if:

$$K^* = \underset{K_l \in ListK}{\text{argmax}}(NCI(x,K_l)) \tag{4.5}$$

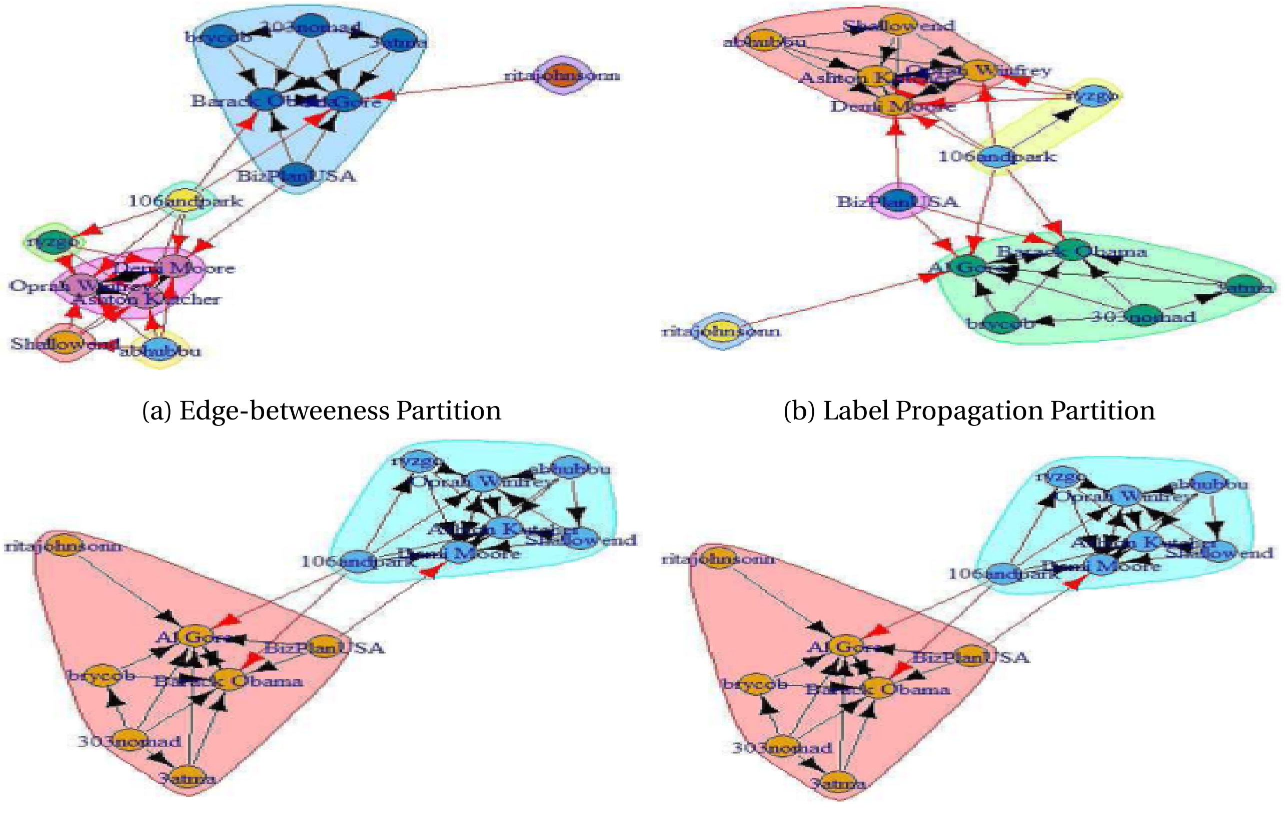
where $ListK$ is a set of extracted kernels.

The pseudo-code of this migration approach is described in the following algorithm 3. It presents in Line 1 the initialization of Communities named G_i by their corresponding kernel K_i computed in the preceding kernel extraction step (see Section 4.2.4). From Line 2 to Line 6, the method computes for each non-kernel node its Node Community Index (NCI) and puts it in the kernel (or growing community) whose NCI is maximal. In Line 7, results which are global communities (communities not growing, but definitely computed) are produced.

Algorithm 3 Algorithm for non-kernels vertices migration

```
Require: Communities Kernels $ListK = \{K_1, K_2, ..., K_t\}$
Require: $NonKernelSet = \{G.nodes \backslash \cup K_i\}$ //nodes $x$ of $G$ not belonging to any $K_i$
Ensure: Global Communities $G_K = \{G_1, G_2, ..., G_t\}$
 1: $\forall i \in \{1, ..., t\}$ , $G_i \longleftarrow K_i$
 2: for $x \in$ NonKernelSet do
 3:     Compute $NCI(x, G_i)$ for each $G_i$
 4:     $G^* \longleftarrow argmax(NCI(x, G_i))$
 5:     $G^* \longleftarrow G^* \cup \{x\}$
 6: end for
 7: return$G_K$
```

Non-kernel vertices for the Figure 4.1 Network are listed below: *shallowend, abhubbu,ryzgo, 106andpark, 3atma, brycob, 303nomad, ritajohnsonn, BizPlanUSA.* Global communities detected by Triad-based approach are shown in Figure 4.3d below. The number of communities is visibly 2: We see in Figure 4.2a that Wang extracts the same partition as well as our model. Nevertheless, Wang sets the number of communities to detect. In other words, if its input on the number of communities was 1, his result would had been different from ours. Results on this illustration network are shown in Figure 4.3 according to 4 methods namely Walktrap [124], Louvain [16], Edge-betweenness [68], Label propagation [127] and Triad-based method.



(a) Edge-betweeness Partition

(b) Label Propagation Partition

(c) Walktrap and Louvain Partition

(d) Triad-based partition

Figure 4.3: Visualization of the partitions obtained from Subtwitter Network in Figure 4.1, by the different algorithms, using R 3.5.1 package

Complexity analysis: In view of the size of G with n the number of vertices and m the number of edges, the complexity is assessed according to each phase.

The first phase of constructing candidate kernels is assessed in 3 ways as shown in Section 4.2.2: Step 1 in Algorithm 1 computes a Centrality list CL. Assume that the length of CL is $p = n - k$. The complexity of this sorted degree-based centrality list CL is $(p)log(p)$. Step 2 in Algorithm 1 computes the kernel dictionary $KDict$. Its computation is assessed considering the right and left sides of the kernel degree measure K_{ij} : Given n_i and n_j the number of neighbors of nodes v_i and v_j respectively. The left side namely triad weight is assessed as follows: the numerator is the intersection of neighbors of nodes v_i and v_j. So the numerator complexity is $O(n_i + n_j)$. The denominator is $O(n)$, in the worst case. This worst case is reached when v_j get all of the other nodes $(n-1)$ as neighbors. For p elements of CL, we will have $O(pn)$. The right side namely Jaccard index variant, possesses a complexity of $O(n_i + n_j)$. Thus the complexity of the sorted Kernel dictionary computation is $O(pn + nlogn)$. Step 3 computes the interclass inertia vector. Its complexity is $O(p^2)$. So the first phase of kernel candidate's generation is $pn + (p)log(p)$ or $n^2 + (n)log(n)$ in the worst case.

The second phase of kernel extraction namely Step 5 in Algorithm 1, is assessed as follows: given that KDict is pruned considering the threshold, and that its remaining elements are copied in $KDict$', let us assume that the size of $KDict$' is s, the number of distinct element; thus, to obtain kernels, we compare one element of $KDict$' to the other, so the complexity is $O(s^2)$. In the worst case when the number of nodes involved in pairs of $KDict$' is n, the complexity of Kernel extraction is $O(n^2)$.

The third phase based on migration of non-kernel nodes to kernels in order to constitute final communities (Step 6) is assessed as follows: Suppose that t is the number of kernels and L the number of non-kernel nodes. So the complexity will be $O(Lt)$. In the worst case, we have $(n-2)$ non kernel nodes with one kernel. Thus, complexity in the worst case is $O(n)$.

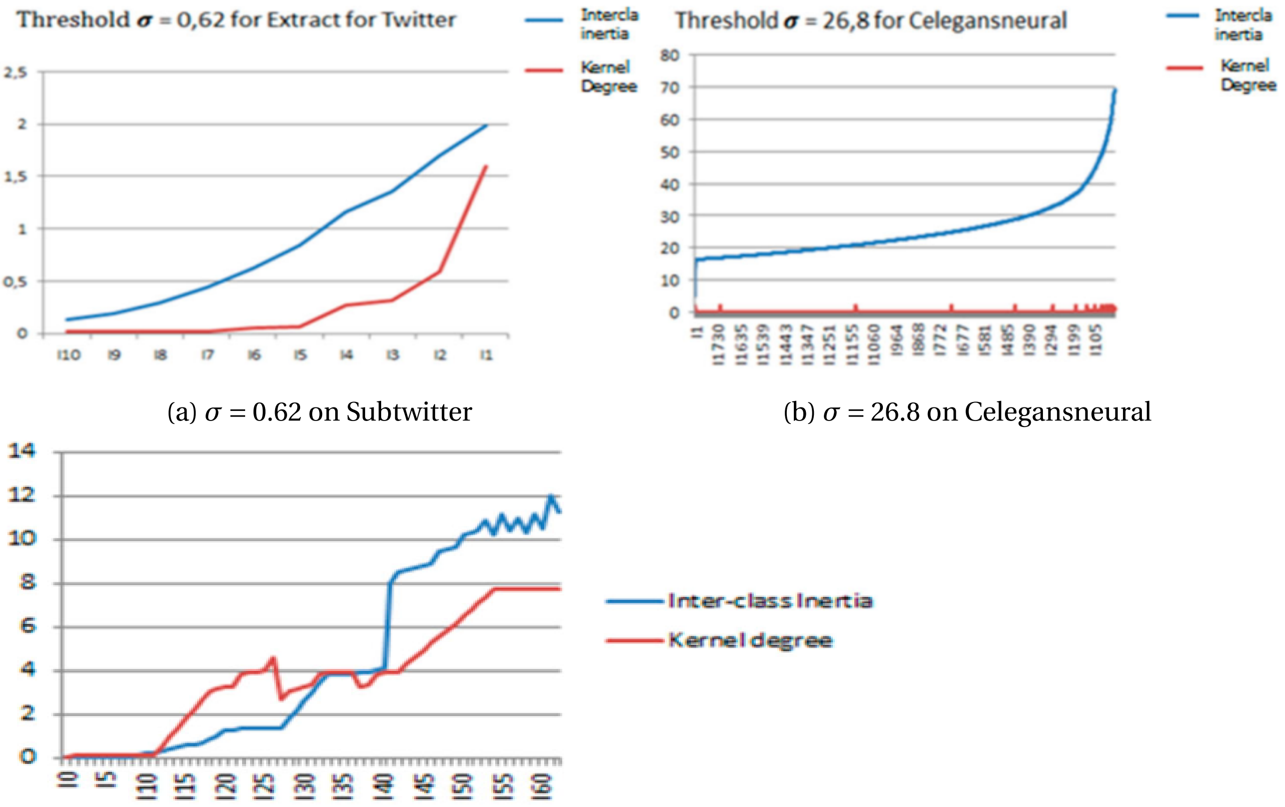
The global complexity of the proposed model is $O(n^2 + nlogn)$.

4.2.6 Empirical evaluation and experiments

In this section, we show experiment results. We assess a variety of models on three main tasks: Triad density of the partition, modularity evaluation and the number of communities. In order to evaluate kernels, the study of the *kernel degree* measure as shown in Paragraph 1 below will be made on the illustration on Subtwitter network as shown in Figure 4.4, and tested through some criteria as described below; and the experiments will not focus on *Kernel Degree* metric, but on three criteria as described in Paragraph 2 namely partition *Triad Density* referenced by σ_Δ defined through the formula 2.7, partition quality through *directed modularity* Q_d defined in Table 3.1 and the number of communities each partition of experimented datasets get.

Datasets

In the following experiments, we use a neural network called Celegansneural, a blog network namely Polblogs, and two paper citation networks namely Citeseer and Cora. Information about



(a) $\sigma = 0.62$ on Subtwitter

(b) $\sigma = 26.8$ on Celegansneural

(c) $\sigma = 3.9$ on Polblogs

Figure 4.4: Standard deviation distribution

Table 4.1: Characteristics of the test graphs

Networks	Nodes	edges	Comm
Celegansneural	297	2,345	5
Polblogs	1,490	19,090	-
Citeseer	3,327	4,732	-
Cora	2,708	5,429	-

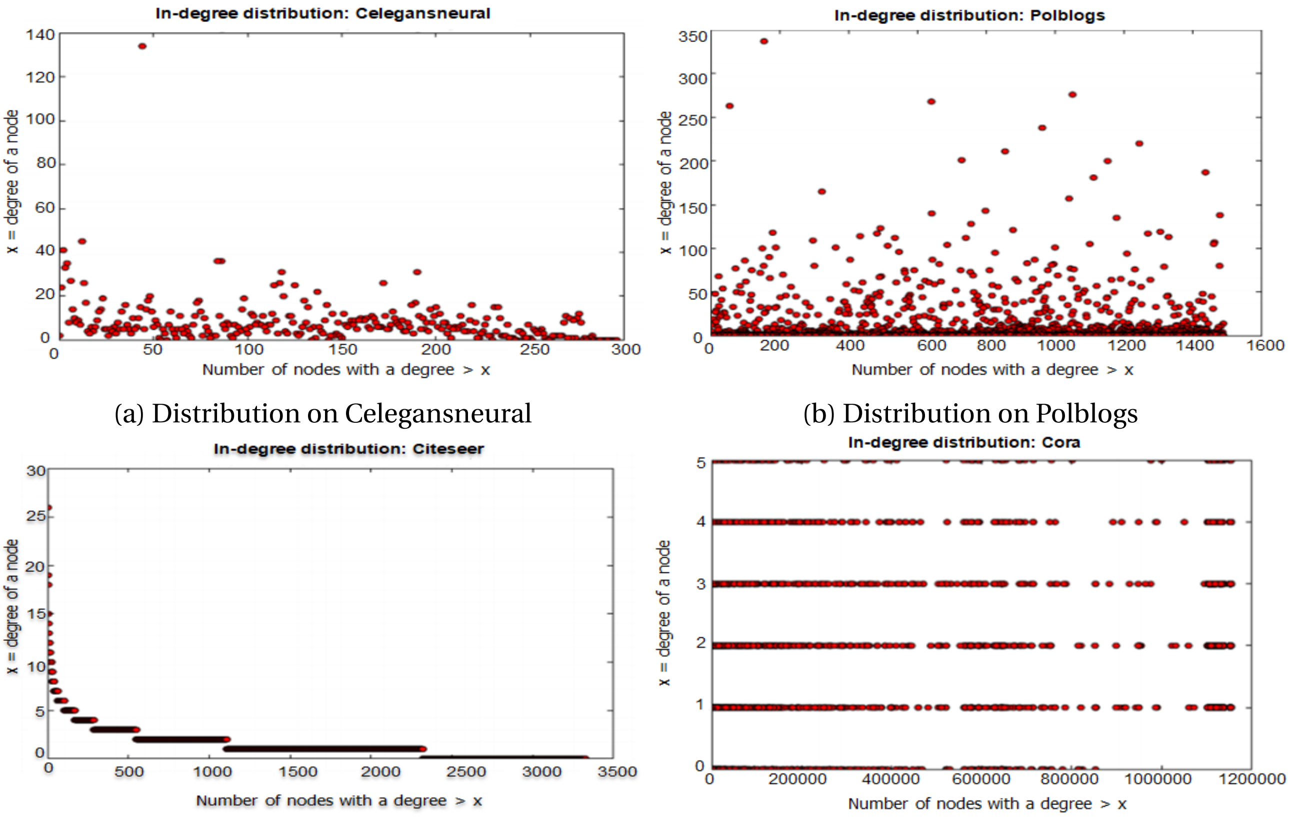
each network can be found in Table 4.1. The columns Nodes, Edges and Comm refer to the number of nodes, the number of edges and the number of communities expected, respectively. Only the Celegansneural network possesses an expected number of communities [86]. The other datasets, as denoted by the character "-", do not have a ground truth on the number of communities.

Celegansneural network. This is a weighted, directed network representing the neural network of Celegansneural. The weighted parameter is not taken into account in this work. There are 297 nodes and $2,345$ links. This dataset possesses 5 communities as obtained by Tianbao [156].

Political Blog Network. This is a directed and unconnected network of hyperlinks between a set of weblogs about US politics [1]. In this network, there is a total of $1,490$ nodes and $19,090$ links. Seeing that the new approach is based on connected networks, the largest connected subgraph with the highest number of links and nodes is the one taken into account throughout the execution of the approach.

Paper Citation Networks. We use the Cora and the Citeseer paper citation networks processed by Getoor et al. [67]. There are 2,708 nodes connected by 5,429 links in Cora network, while 3,327 nodes and 4,732 links in Citeseer network.

The phenomenon described by these datasets follows a power-law in-degree distribution except the in-degree distribution in Cora network. The scatter plots for in-degree valuation of nodes are presented in Figure 4.5. In fact, a small number of vertices possess a high in-degree value, implying that a small amount of nodes have high quasi-common neighborhood cardinality, while larger nodes have less common neighbors. Yet, the indegree in Cora dataset follows a rather uniform distribution with in-degree not larger than 5. We suspect such a distribution is due to the small scale of the Cora dataset which leads to many references, and therefore in-links, inside the dataset.



(a) Distribution on Celegansneural

(b) Distribution on Polblogs

(c) Distribution on Citeseer

(d) Distribution on Cora

Figure 4.5: In-degree distribution on dataset nodes

The goal of experiments is to demonstrate the influence of in-links emphasized by the method, as the numbers of authors quoting an article favors to delimit a topic area among a pioneer area (the node of interest or the node of interest set). In other words, our goal is to evaluate if our new Kernel Degree based metric yields the link semantic of communities in directed networks, in ac-

Table 4.2: Using metric Comparison.

Metric	**Figure 4.1 Network**		**Celegansneural**	
	#Comm	$TriadDens$	#Comm	$TriadDens$
Kernel-degree	2	0.64	5	0.711
Neighborhood Overlap	2	0.64	91	0.20
Triad Weight	2	0.64	73	0.254

Table 4.3: σ choice evaluation.

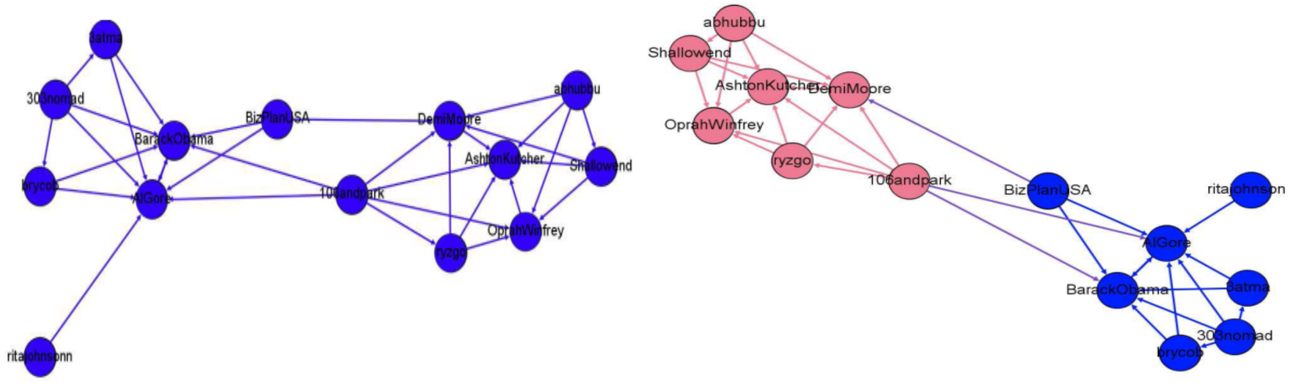
Inertia Criteria	Figure 4.1 Network		Celegansneural	
	#Comm	$TriadDens$	#Comm	$TriadDens$
$I[e_{ij}] > \sigma$	2	0.6428	5	0.711
$I[e_{ij}] < \sigma$	1	0.417	103	0.065

cordance with triad-based community definition. The empirical evaluation of the new approach, to show its performance, is compared to some of the state-of-the-art methods: *Walktrap* [124], *Louvain* [16], *Edge-betweenness* [68], *Label propagation* [127].

Paragraph 1: Kernel degree metric and threshold evaluation **Kernel degree metric evaluation.** To appreciate the powerfulness of the *Kernel Degree* formula, let us consider two networks namely Subtwitter Network and Celegansneural network for better results' visualization. *Kernel Degree* computes the similarity strength between kernel vertices; in other words, it determines the kernel power. Both *Triad Weight* (Definition 4.2.4) and *Neighborhood Overlap* (Definition 4.2.3) are associated to reinforce this similarity, because, when taken separately, the expected results are not obtained, as presented in the Table 4.2. In fact, for the Subtwitter Network, results are the same regardless of the criteria (2 communities with the same triad density and same modularity). But for the Celegansneural network, using separately Neighborhood Overlap or Triad Weight leads to results (91 and 73 communities respectively) far from expected one as demonstrated by Klymko and Tianbao [86, 156] who detect 5 communities. Furthermore, taken separately, they lead to a computation of weak values of triad density, contrary to the new composite kernel degree metric which computes a better triad density of 0.711, close to the triad density value of 0.78 obtained by Klymko.

Threshold σ evaluation. As far as the threshold σ is concerned, the empirical experiments show that when taking descent values of the interclass inertia, meaning those less than σ, expected results are not obtained. For illustration, as seen from the Table 4.3, our approach performs the best in both datasets. Figure 4.6 illustrates the comparison of these both σ considerations. For sake of simplicity, we assume that $I[e_{ij}] = I_{ij}$. Then, in the first case ($I_{ij} < \sigma$ as shown in subfigure

4.6a), the Subtwitter Network just contains 1 community with a low triad density of 0.417 and Celegansneural 103 communities with 0.065 triad density value. On the contrary, the Subtwitter Network, for the second case ($I_{ij} > \sigma$) contains 2 communities, as shown in subfigure 4.6b, with a triad density of 0.6428, and Celegansneural 5 communities with a high value of triad density equals to 0.711. This result means that the Subtwitter Network partition is not well structured for the first case. Higher inter-class inertia values indicate better kernel based-triad structures and therefore, finding vertices with similar neighbours whose inter-class inertia values are upper than threshold provides a method for extracting the underlying kernel structure. The Figure 4.4 shows the analysis made on the idea that the more the inter-class inertia is upper than a threshold σ, the more the kernel degree values are large, meaning better triad-based structures.



(a) The first case ($I_{ij} < \sigma$) produces one community

(b) The second case ($I_{ij} > \sigma$) produces two communities

Figure 4.6: Graphical visualization with Gephi tool, on Subtwitter Network, for the threshold evaluation

Paragraph 2: Performance on Community Detection The community detection performances for different models on the four datasets are given in Table 4.4. Notably, the Subtwitter Network obviously contains 2 communities in all meanings of the term. This constitutes a ground truth, as shown on the Figure 4.6b. We first illustrate the performance of triad-based approach based on the Subtwitter and Celegansneural networks results, as shown in the following: To illustrate the results of our approach, based on Subtwitter Network, Table 4.4 shows some results and compares them to triad-based approach. We observe that as well as the Triad-based approach, both Walktrap and Louvain detect 2 communities with the same high values of triad density and modularity. Label and Edge Betweenness methods compute respectively 5 and 7 communities with lowest triad density and modularity values. Visibly, our approach extracts expected structures better than some other methods, as pointed up in Figure 4.6b. Celegansneural network is also used to illustrate the new approach methodology and its hidden idea because it possesses a ground truth

Table 4.4: Community detection performance where the best performances are in bold.

Datasets	Methods	TriadDens	Modularity	#of Communities
Twitter network	Edge-Betweenness	0.0857	0.187	7
	Walktrap	**0.6428**	**0.410**	2
	Label Propagation	0.34	0.306	5
	Louvain	**0.6428**	0.395	2
	Kernel Approach	**0.6428**	**0.410**	2
Celegans Neural	Edge-Betweenness	0.0004	0.081	194
	Walktrap	0.0458	0.363	21
	Label Propagation	0.0135	0.0027	29
	Louvain	0.608	0.379	6
	Triad-based Approach	**0.711**	**0.393**	5
Polblogs	Edge-Betweenness	0.0064	0.1872	55
	Walktrap	**0.67**	**0.4302**	12
	Label Propagation	0.0026	0.386	244
	Louvain	0.0085	0.427	274
	Triad-based Approach	**0.5732**	**0.429**	34
Citeseer	Edge-Betweenness	0.0	0.5344	738
	Walktrap	0.0	0.811	593
	Label Propagation	0.0	0.491	842
	Louvain	0.079	0.886	466
	Triad-based Approach	**0.407**	**0.8907**	121
Cora	Edge-Betweenness	0.0516	0.3999	1028
	Walktrap	0.2131	0.756	265
	Label Propagation	0.2801	0.6565	133
	Louvain	**0.313**	**0.808**	100
	Triad-based Approach	0.0853	0.212	1107
	CDLPA	-	0.6042	-

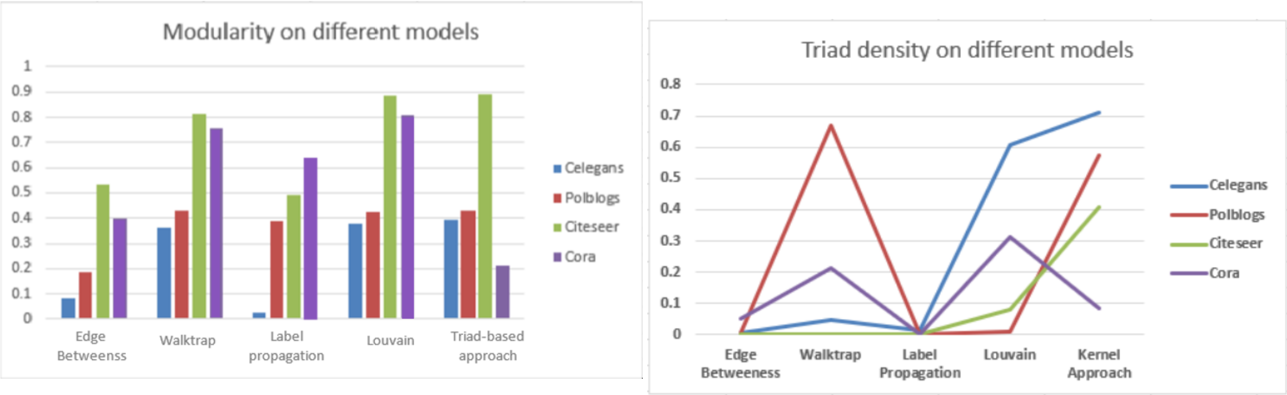
result [86]. With this dataset, both the expected number of communities and triad density metrics are evaluated.

Triad density δ_Δ and Modularity Evaluation. For the Celegans dataset, Table 4.4 shows that Triad-based method improves triad density $\delta_\Delta = 0.711$, close to 0.78, being Klymko's triad density [86]; likewise, the higher modularity (see 0.393) proves its performance on the partition quality. However, Edge Betweenness algorithm produces the weakest triad density $\delta_\Delta(0.0004)$ while Label propagation obtains the weakest modularity (0.0027), since it favors the over propagation and giant communities, meaning that its communities could be scarcely dense.

For the polblogs network, Triad-based approach methods slowly performs in all of the criteria: it improves triad density to $\delta_\Delta = 0.5732$ and modularity value of 0.429 as shown in Table 4.4. This result indicates that models of "what is a growing-community"are somehow in agreement with the notion of Kernel degree measure; moreover, as confirmed by Yang's [154] assertion, triads are more effective in contributing to community structures than modularity. Walktrap performs the best since it considers unconnected partitions, indicating that it captures the so-called “outliers”, which are anomalous nodes (belonging to none of the communities). Meanwhile, Louvain method results are not so interesting with a δ_{Δ} of 0.0085. This result could be due to the fact that Louvain’s method stresses on the modularity optimization. Indeed, this measure does not implement the higher consideration of nodes with higher incoming edges and weak outcoming edges than the opposite. Since Label propagation method operates by moving nodes from one community to another according to its common neighbors’ label, it computes the weakest δ_{Δ} (0.0026) because Polblogs is a non-connected network.

In Table 4.4, through results presented for Citeseer dataset, Triad-based approach improves values of modularity and triad density. As shown in Figure 4.7a, the Citeseer dataset has good community structures based on link density, because its application to all the methods allows for greater values of modularity. Triad density criteria is underlined in Figure 4.7b. There, triad density has a good exponential progression on this dataset, expressing that for Triad-based approach, triad density is upper, contrary to its value on the other datasets and other approaches. This result underwrites that this kind of network with a power-law distribution is characterized by better triad density, one of the main criteria of the Kernel approach. As shown in figure 4.5c, more than 80% of the nodes have a degree between 1 and 3 and the remaining nodes have a degree between 4 and 26. Since the majority of the nodes have such a low degree, it means that the method will produce few kernels, and therefore few communities. In other words, the resulting structure will have more followers than leaders, more citations than articles containing them. This behavior reflects the reality insofar as for 2 articles, one could have about fifty articles in the bibliography. According to figure 2.2 which presents the triads considered in our approach, we can deduce that it is quite normal that the proposed method produces this high value of triad density, compared to the low values obtained by the other methods. Moreover, the value of modularity obtained by our approach is not very far from Louvain because of link density that the latter takes into account.

The null triad density values for the other methods in the table 4.4 illustrates better type of scorpus that our method performs on. In fact, contrary to the other datasets, Citeseer follows the deepest power law distribution, because it possesses a hub node (node with a higher degree distant from the other nodes degrees), as presented in subfigure 4.5c. Tsourakakis [148] confirms the plausibility of these results by its argumentation that low degree nodes form fewer triangles than higher degree nodes; and according to Durak [54], citation networks are dominated by heterogeneous triangles; like this, triads are included into triangle. So results on Citeseer, a citation network type, seem to be valid in regard of both precedent demonstrations.



(a) Link Density (modularity) as a criteria of network type

(b) Triad Density as a metric of partition evaluation

Figure 4.7: Global measures evaluation

For Cora network, results shown in Table 4.4 indicate that the new scheme is based on power-law distribution in datasets. Indeed, since Cora follows a uniform in-degree distribution as shown in Figure 4.5, our kernel approach produces weak results; CDLPA improves LPA results, since it overcomes the imbalance growth of communities; Louvain's method performs the best. This result is due to the fact that it is based on density of links disregarding the benefit of the node in-degree.

Summarily, these weak results for Louvain method compared with the proposed triadbased approach on the overall of datasets indicate that it focuses solely on link density in the community without no interest of the topology or in-link based semantic of triads into the communities. The triad-based approach performs the best in all the cases except on the triad density for Cora network. These results also illustrate that most of the time, it is beneficial to use both triad weight and neighborhood overlap measures simultaneously, establishing Kernel Degree formula, to enhance the similarity kernel vertices in a directed network.

Number of communities. According to results on Celegansneural as shown in Table 4.4, Triad-based method confirms the 5 communities detected by [86]. Since Edge Betweenness algorithm focuses on links between nodes by searching the central edge (geodesic) meaning the short path linking two communities, it detects 194 communities for this dataset, instead of 5, as expected by ground truth of Klymko [86].

Results on Polblogs are slowly claimed for Triad-based method. As mentioned above, Polblogs's nodes do not follow a power law degree distribution. In addition, the network is non-connected, meaning that there are isolated nodes into that network. Since Triad-based method focuses on common neighborhood and is applied to connected graphs, it has to cover the connected component of the whole network. We think that this slight result on the number of communities is due to these considerations on the networks. As far as Label Propagation method is concerned, a node moves from one community to another if, its neighbors share the same label. Hence, for the polblogs network, it computes the high number of communities (namely 244 communities) because Polblogs is a non-connected network. Walktrap produces few communities (namely 12). Indeed, it is also based on the principle of common neighborhood. Since the neighborhood implies triad density [157], they produce the smallest community number, more close to Triad-based method number, compared to the other values of the community number.

An efficient report made from Table 4.4 is that the more the number of communities is low the more triad density and modularity values are great. Indeed, the proposed approach shows that the number of communities depends on the depth of the power law distribution. The deeper this distribution is (case of Citeseer, Celegans), the fewer communities there are. These results show that taking into account the edge directionality through the triads together with the density of links, yields more cohesive structures. However, the informations based on node attributes are not taken into account. The work in the following section proposes a solution to overcome this limitation.

4.3 Towards a hybrid model of communities detection

As already stated, we are studying the detection of communities in directed graphs. The previous section proposed the triad-based method, based on both topological and relational informations. For more cohesive or semantic communities, it seems interesting to integrate semantic information based on nodes attributes. The method proposed in [49] thus constitutes the essence of this contribution.

In this section, we propose a hybrid technique dealing with the attributes of nodes, together with the structure in a directed graph. Indeed, we define a hybrid similarity measure which includes node attribute informations along with the network structure and edges' directionality. Then by application of a hierarchical agglomerative clustering technique namely Louvain [16], we evaluate its performance and results on a dataset with ground truth by showing that with attributes joined to vertices, it is possible to extract meaningful clusters. To make the difference with the communities uncovered from the previous method described in Section 4.2, we called *hybrid clusters* those of the communities identified by the hybrid model. This section describes first of all in Subsection 4.3.1 the properties to be fulfilled by hybrid clusters, before setting out the semi-hybrid approach intended to contextualize the proposed hybrid approach. Afterwards, the hybrid model proposed to detect hybrid clusters is investigated in Subsection 4.3.2.

4.3.1 Clustering Graph models

A major difference between structure graph clustering and traditional data clustering is that, structure graph clustering measures vertex closeness based on connectivity (e.g., the number of possible paths between two vertices) and structural similarity (e.g., the number of common neighbors of two vertices); while data clustering measures distance mainly based on attribute similarity (e.g., Euclidian distance between two attribute vectors). Remember that approaches for attributed graph clustering handle both structure and vertex attributes, and differ by their manner of combining the structure and attributes of nodes, but they largely ignore edge directionality. The consideration of directionality is included into properties, as described in the following section.

Hybrid clusters properties

In attributed networks, the clustering task should take into account both structure network and attribute information by achieving a good balance between the following two properties : *(i)* vertices within one cluster are closed to each other in terms of " structure", meaning that vertices are arranged according to a specific pattern, while vertices across clusters are not patterned (Figure 3.3 shows pattern-based structures); *(ii)* vertices within one cluster are more similar by their attributes than vertices from different clusters that could have quite different attribute values. In this work, we consider that the partitioning process focuses on both a *patterned* structure based on triads and node attributes. In others words, our structure concept includes not only connectivity, but also link directionality according to the in-degree of nodes. The approach consists in detecting a partition of k clusters c_i, from the set of nodes V such that :

1. $c_i \cap c_j \neq \emptyset \forall i \neq j$ and $\cup_i c_i = |V|$;
2. Hybrid clusters are patterned according to the structural equivalence;
3. The similarity between nodes takes into account three criteria : the connectivity, the node attribute and the edge directionality;
4. Vertices within clusters are homogeneous, while the vertices in different clusters are heterogeneous according to their attributes.

Semi Hybrid clustering

As described in Chapter 3, there are three sights of clustering in attributed networks. The first approaches refer to the M_a model since they first exploit attributes by graph enrichment through a node attribute similarity function. SA-Cluster [160] is an example method of this model consisting in augmenting the initial graph, thereafter, a random walk distance is applied to the augmented graph. The second approaches refer to the M_r model since they consider relational-based properties first [37, 98, 137]. An example of this model is the hierarchical clustering of Li *et al.* [98], which consists in its first phase in detecting community seeds with the relational information; thereafter, the final communities are built under constraints defined by the attributes. This leads to merging the seeds on the base of their attributes'similarity. The third approaches refer to semi-hybrid SH_{ar} model [35, 101] since they combine both relational information and attributes similarity of nodes. A typical instance of semi-hybrid technique is Combe's model [35] based on a similarity measure defined as a linear combination function as in Equation 4.6

$$disG(v_i, v_j) = \alpha.disT(v_i, v_j) + \beta.disS(v_i, v_j) \tag{4.6}$$

where $disT$ and $disS$ denote a distance measure for attribute data and geodesic distance for structure data respectively.

Semi-hybrid methods are significant, but they do not integrate the edges directionality. A straightforward way to integrate link directionality is to combine relational, attribute and directionality similarities by adding another factor to the Equation 4.6 as described in the section 4.3.2 below.

4.3.2 Hybrid clustering model

This section presents the hybrid community detection method for directed attributed graphs which exploits a similarity based on the attributes and directionality informations, jointly with the Newman modularity Q_{Newman}. To avoid confusion to the semi-hybrid measure previously mentioned (not taking into account link direction), we add the *hybrid* similarity based on both nodes' attributes and edges' directionality, named $SimH$, as defined in Equation 4.11. In the following paragraphs, we first presents measures optimized by the method and secondly we describe the hybrid method.

Hybrid similarity measure Before clustering phase, some similarity measures must be determine. Assume that the nodes of the network are described by attributes.

Since the proposed hybrid model involves the structure, the node attributes and edge directionality, we have to formally describe each of these informations.

Let $simS(v_i, v_j)$ be the structure based information as shown in Equation 4.7

$$simS(v_i, v_j) = \frac{1}{2m}.(A_{ij} - \frac{k_i k_j}{2m}) \tag{4.7}$$

sim refers to the link strength between nodes v_i and v_j. It is included into the Newman's well-known modularity [118] function as defined in Section 3.2 in Chapter 3. Newman's modularity based on the density of links through $simS(v_i, v_j)$ can be written as:

$$Q_{newman} = \sum_{l=1}^{|P|} \sum_{v_i, v_j \in l} simS(v_i, v_j) \tag{4.8}$$

Concerning node attributes, let $simA(v_i, v_j)$ be the attribute-based similarity. If nodes are associated to categorical attributes, the attribute-based similarity is based on the euclidean distance as described in the formula 4.9 below.

$$simA(v_i, v_j) = \frac{1}{1 + \sqrt{\sum_{k \in T} (x_i^k - x_j^k)^2}} \tag{4.9}$$

where x_i^k is the value of the attribute k associated to the vertex v_i.

For the information concerning the directionality of edges, let $simR(v_i, v_j)$ be the similarity based on edges' directionality, corresponding to Kernel degree measure, as defined in one of our previous works [47], such that:

$$simR = \frac{|\Delta_{ij}|}{|\Delta_j|} . \frac{|\Gamma_j^{in} \cap \Gamma_i^{in}|}{|\Gamma_j^{in} \cup \Gamma_i^{in}| - \theta} \tag{4.10}$$

Although there is a modularity for directed graphs [121], it does not take into account the importance of the incoming degrees compared to the outgoing degrees of nodes. Indeed, directed modularity does not implement the idea that an edge from a low outdegree but high in-degree node to an opposite case node should be considered of a higher value. We define an hybrid similarity

($simH$) in Equation 4.11, based on node attribute (see Equation 4.9) and edge Directionality Similarity (see Equation 4.10), then we propose a "hybrid modularity" Q_{hyb} of a partition in Equation 4.12.

$$simH(v_i, v_j) = \omega.simA(v_i, v_j) + (1-\omega).simR(v_i, v_j) \tag{4.11}$$

$$Q_{hyb}(v_i, v_j) = \sum_{l=1}^{|P|} \sum_{v_i, v_j \in l} simH(v_i, v_j) \tag{4.12}$$

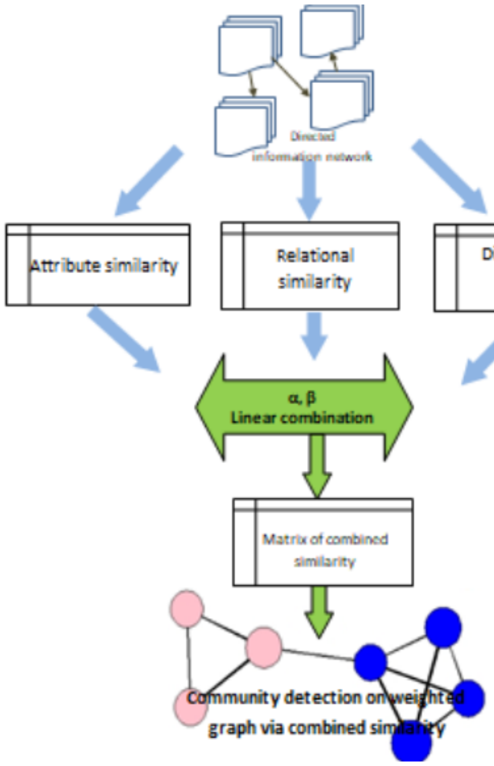

Directed information network
Attribute similarity
Relational similarity
Directionality similarity
α, β
Linear combination
Matrix of combined similarity
Community detection on weighted graph via combined similarity

Figure 4.8: Progress of the hybrid model

Q_{hyb} is used for extending the modularity composite of Dang and Viennet [37]. We named that extended modularity as *Global modularity* Q_G which combines simultaneously 3 informations data namely: structural information through $simS$, edge directionality through $simR$ and attribute of nodes through $simA$.

Hybrid method description

As stated above, a direct application of our measure $simH$ is the community detection in complex networks represented by an attributed directed graph $G = (V, E, W)$ where V is a set of vertices, E is a set of edges and where each vertex $v \in V$ is described by attributes $w_i \in W$.

Our method, called **Hybrid-Louvain**, is based on the exploration principle of the Louvain method. Since Louvain method does not include the attribute similarity between nodes together with directionality of edges, Hybrid-Louvain consists in determining hybrid clusters by optimizing the global criterion Q_G defined in Equation 4.13, to find answer of the following question: *Whether meaningful communities be detected by dealing with direction of the edges?* Pseudo code of this model is shown in Algorithm 4 and its architecture is shown in Figure 4.8.

$$Q_G(P) = \alpha.Q_{newman}(P) + \beta.Q_{hyb}(P) \tag{4.13}$$

where α and β are weighting factors that enable to give more importance to the structural, attribute or directionality of edges. $\alpha + \beta = 1$. The next step is to find an approximate optimization of Q_G (direct optimization is a NP-hard problem [24]). We follow an approach directly inspired by the Louvain algorithm [16]. The algorithm starts with each node belonging to a separated community. A node is then chosen randomly. The algorithm tries to move this node from its current community. If a positive gain is found, the node is then placed to the community with the maximum gain. Otherwise, it stays in its original community. This step is applied repeatedly until no more improvement is achieved. When moving node x to community C, the composite modularity gain is calculated as:

$$\Delta Q = \alpha.\Delta Q_{newman} + (1-\alpha).\Delta Q_{hyb} \tag{4.14}$$

with

$\Delta Q_{newman} = \sum_{v_i,v_j \in C\cup x} simS(v_i,v_j) - \sum_{v_i,v_j \in C} simS(v_i,v_j) \quad = \frac{1}{2m}(\sum_{v_i \in C} A_{v_i,x} - \frac{k_x}{2m}\sum_{v_i \in C} k_i)$ and

$\Delta Q_{hyb} = \sum_{v_i,v_j \in C\cup x} simH(v_i,v_j) - \sum_{v_i,v_j \in C} simH(v_i,v_j) \quad = \sum_{v_i \in C} simH(x,v_i)$

Algorithm 4 Algorithm for hybrid clusters identification

```
Require: An attributed directed network $G = (V, E, W)$, Similarity matrix
Ensure: Partition of hybrid clusters
Phase 1: Initialize each node to a separated community;
repeat
  for $i \in V$ do
    for $j \in V$ do
      Remove $i$ from its community, place to $j$'s community
      Compute the composite modularity gain $_G$
    end for
    Choose $j$ with maximum positive gain (if exists) and move $i$ to $j$'s community
    Otherwise $i$ stays in its community
  end for
until No further improvement in $Q_G$
Phase 2: Each community is considered as new hypernode
  Compute $Q_G$ through summing up the similarity of their members
  Reapply Phase 1.
```

The first phase is completed when there is no more positive gain by moving of nodes. Following Louvain, we can reapply this phase by grouping the nodes in the same communities to a new community-node. The weights between new nodes are given by the sum of the weight of the links between nodes in the corresponding communities. To determine the attribute similarity between two communities, we choose the majority attribute.

4.3.3 Illustration of the hybrid model

To our knowledge, there is no referenced benchmark with attributes information handling edge directionality. Thus we used an illustration network namely **Food web** where a vertex represents a specie and edge the relationship between prey and predator. Also, each vertex is described by the attributes according to the *mode of reproduction* (viviparous, oviparous, Asexual) and to the *mode of nutrition* (carnivorous, herbivorous, producers/- consumers, vegetables etc.). An example[1] of food web chain network is shown in Figure 4.9. We assume a small ground truth dataset based on this network as shown in Table 4.5. These assumptions help to study the behavior of methods, according to the consideration of each type of information namely structural, attributes or directionality.

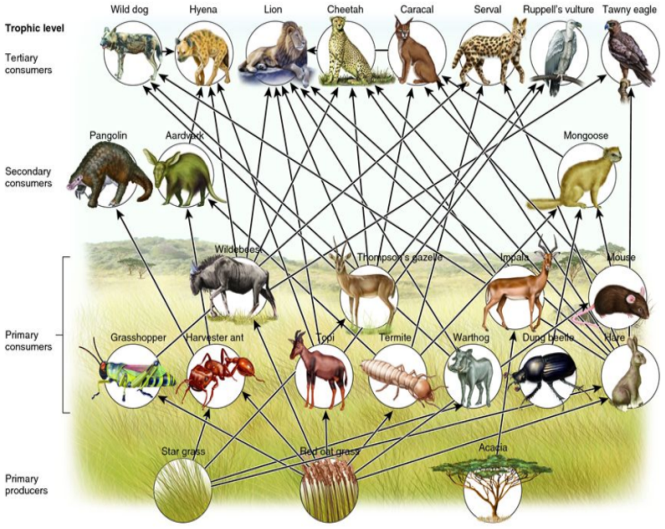


Figure 4.9: Food web illustration network with Diet sectors

Assumptions on food web network

Here we enumerate partitioning scenario and present expected results which are ground truth of food web network in Figure 4.9, since we focus on communities of interest. The interest depends on each type of informations (relational, attribute, directionality and combination of them). We

[1] https://www.pinterest.com/pin/241998179953934424/ image viewed on May 20, 2020

Table 4.5: Number of species by nutrition sector and mode of reproduction

Category	Diet mode	Mode of reproduction	Number
A	Carnivorous	Viviparous	8
B	Carnivorous	Oviparous	3
C	Herbivorous	Viviparous	7
D	Herbivorous	Oviparous	4
E	Vegetables	Asexual	3
Total			25

consider 5 subsets of vertices A, B, C, D, E describing species diet mode and by their reproduction mode, to be real meaningful cluster of Hybrid-Louvain. The Table 4.5 shows an illustration of properties of each animal :

- Relational (connectivity) : 2 sectors according to their consumer or producer status. Th first is based on tertiary, secondary and primary consumers and the second one on primary producers. Indeed, there are three consumers sectors with dense edges among species, namely primary consumers corresponding to vegetarian animals, secondary consumers that are carnivorous, and tertiary consumers, those eating species of the others sectors. The second component of this partition is based on primary producers which do not have any prey. So the ground truth partition P_s is structured by two communities of species belonging to each sector. $P_s = \{A \cup B \cup C \cup D, E\}$.

- Attribute : 3 clusters in which species are grouped by their mode of reproduction, either viviparous or oviparous or asexual. The ground truth partition is formally defined as $P_a = \{A \cup C, B \cup D, E\}$.

- Directionality (Neighborhood) : 3 clusters in which species are grouped by their diet mode, either carnivorous, herbivorous or oxygen(nutriments for vegetables diet). Since the information based on directionality focuses on in-direction of edges, the primary producers sector is separated because vegetables do not have any in-neighborhood. The ground truth partition is formally defined as $Pr = \{A \cup B, C \cup D, E\}$

- Hybrid informations : 5 clusters of species since we identify species by their both diet mode and mode of reproduction characteristics, then attributes, relational and directionality properties should be used. Like this, the resulting partition is $P_h = \{A, B, C, D, E\}$.

Illustration on Food web network.

Table 4.6: Food web results : NMI

Models	P_r	P_a	P_s	P_h
M_r	**0.753**	0.350	0.323	0.023
M_a	0.741	**0.842**	0.28	0.625
SH_{ar}	[0.012 – 0.018]	[0.205 – **0.441**]	[0.027 – 0.291]	[0.085 – 0.397]
H_s	[0.098 – 0.217]	[0.110 – 0.185]	[0.075 – 0.181]	[0.558 – **0.895**]

Given that this study focuses on directed attributed graphs which have not yet been investigated in detail, the illustration of Hybrid-Louvain consists in checking these assumptions described above, by evaluating stated models of Section 4.3.1 (M_a, M_r, SH_{ar}). We compare these 3 models with the hybrid model (H_s). The synthesis of results is shown in Table 4.6, according to the Normalized Mutual Information (NMI) measure [137]. Then clusters issued from the ground truth clustering transcripts the following partitions : the group of species by their diet mode (P_r), by their mode of reproduction (P_a), and by the both simultaneously (P_s).

- **Clustering according to textual attributes :**M_a **Model**. In this approach corresponding to the technique in Sect.4.3.1, the euclidean distance computed on the textual attributes helps to weight each edge. So, attributed graph becomes a weighted one; thereafter an unsupervised method is applied to the resulting graph. The method performs well when the ground truth partition is $P_a = \{A \cup C, B \cup D, E\}$ by a higher NMI value (0.842) than considering the partitions P_r or P_s.

- **Clustering according to structure** : M_r **Model**. This method firstly exploits the structure and secondly, with attributes handled, it detects communities so that the nodes in the same community are densely connected as well as homogeneous. The NMI value for the ground truth partition namely P_r is higher (0.753) than its value for the ground truth partition P_a and P_s. More specifically, $Pr = \{A \cup B, C \cup D, E\}$ produces a higher $NMI = 0.753$. This result demonstrates that a technique based on successively structure then attributes, performs well in case of detecting two clusters of species with a densely internal connectivity and common neighborhood, corresponding to diet mode. However, for the P_h partition, the lowest NMI value indicates that the density, the directionality, together with attributes are not simultaneously handled.

- **Semi-hybrid clustering** : SH_{ar} **Model**. As far as this method is concerned, it deals with both types of information simultaneously (structure and attributes) as studied by Combe [35] through a weighted distance function. In experiments, the NMI value fluctuates as a function of the weighting factors α and β. It changes its value according to the weighting factor

α. NMI is in the interval [0.205 -0.441] for the ground truth P_a and [0.028 - 0.291] for P_s. For the P_r ground truth partition, since the existing methods do not take into account the link directionality in the assigned graphs, the corresponding NMI has the lower value in $[0.012 - 0.018]$. The interval bounds correspond to the NMI values depending on α: when $\alpha = 0.75$, results concern the lower bounds and when $\alpha = 0.5$, the upper bounds are concerned. Remember that $\beta = 1 - \alpha$.

SH_{ar} Model performs the best for the ground truth P_a, meaning that textual attributes describe better the vertices similarity, but produces weak outcomes as proved by [35] for the overall results.

- **Hybrid clustering** : H_s **Model**. The objective of this hybrid based experiment consists in 2 ways. First it shows that the consideration of the textual attributes improves better the cluster semantics through the highest NMI values as presented in bold in the Table 4.6. Second it shows that combining simultaneously the three types of information which are link directionality, relational and attribute properties respectively, leads to the highest NMI for that expected partition $P_h = \{A, B, C, D, E\}$. Like this, it detects the five classifying species clusters by their diet and reproduction mode simultaneously with a NMI value of 0.895 when the weighting factors α and β both equal 0.5; NMI value decreases to 0.558 when the weighting factors α and β equal 0.9 and 0.1 respectively, meaning that the negligence of the hybrid similarity related to link directionality and node attributes property affects the result.

4.3.4 Experimental Study

In this section, we performed experiments to evaluate the performance of the hybrid approach on one real-world network namely *Political Blogs Dataset* [1]. It is a directed network of hyperlinks between weblogs on US politics. This dataset contains $1,490$ weblogs with $19,090$ hyperlinks between these weblogs. Each blog in the dataset has an attribute describing its political leaning as either *liberal* or *conservative*. We use both Density and Normalized mutual information (NMI) measures to evaluate the quality of clusters generated by different methods.

Evaluation on Polblogs dataset To assess the validity of the proposed hybrid model on this dataset, we use four methods according to the models implemented: SA-Cluster for the model considering first attributes then relational information (M_a), SAC1 and Li's model [97] based on the model considering first relational and then attributes informations (M_r) and Combe's model

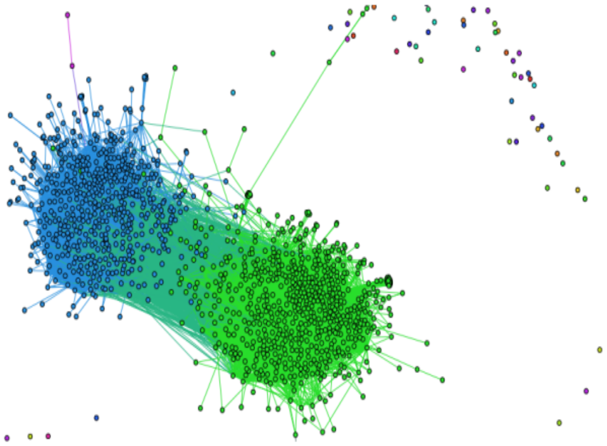


Figure 4.10: Political blogs partition by Hybrid-Louvain with 9 communities

(SH_{ar}) considering semi hybrid informations through the combination of both types of data. Given that Hybrid-Louvain is based on the global modularity being a linear combination of two objective functions, we considered for the sake of fairness the same weighting factors. Indeed, $\omega = 0.5$ in the hybrid similarity formula 4.11 and $\alpha = 0.5$ in the global modularity in formula 4.13, unless otherwise stated.

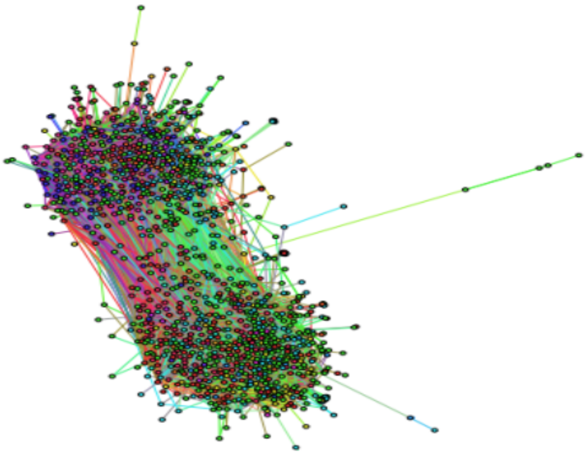


Figure 4.11: Political blogs partition with weighting factor not null on directionality

Table 4.7: Polblog results : NMI

Models	SA-Cluster	SAC1	Li's Model	Hybrid-Louvain
M_r	0.350	0.153	0.323	**0.578**

To assess the quality of these methods, we compare the number of communities and the density. When a truth-ground is unknown, the NMI can be used for comparison of 2 partitions. The validity through the NMI measure considers the SH_{ar} as basic model. Thus, the partitions obtained by the M_r, M_a and H_s models are compared to the partition obtained by the SH_{ar} model, and the results are reported in the table 4.7. We notice that Hybrid-Louvain presents a higher NMI

value of 0.578 than the one obtained by the other approaches. Figure 4.13 compares Density between the five methods on Political Blogs when the number of clusters is set to $k = 3,5,7,9$. The density values of the hybrid model are in general slow because Political Blogs is a non-connected network. Since this model consider as well as the relational information based on the neighborhood of nodes, it seems fair for this type of dataset. For SA-cluster and SAC1 methods, the density values are high. This demonstrates that attributes are relevant for community discovery. The density values by Li and Combe's models are close. When $k = 3$, they remain around 0.6. When $k = 5,7$, they range between 0.2 and 0.4. This demonstrates that both methods can find the same partitions according to a specific number of communities. On the other hand we observe that Combe's model presents a density decreasing when k increases.

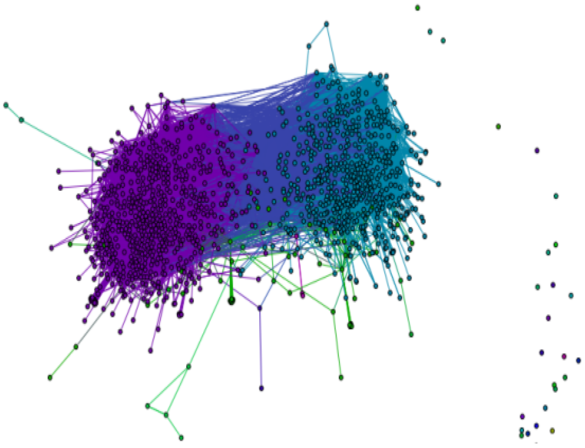


Figure 4.12: Political blogs partition by Hybrid-Louvain with 3 communities

The density of the partitions obtained by the hybrid approach is in general (when $k = 5,7,9$) higher than that of Combe, according to the diagram in Figure 4.13, because Combe takes into account only relational and attribute information. Yet our approach takes into account the orientation of the links. And since this direction is based on the triad-based topology that is included in the structural information, then referring to density, the triad density is included in the structural density. Hence the density of the partitions is greater than that of Combe.

The table 4.8 shows some results on the number of communities and the density of partitions obtained by Hybrid-Louvain, according to the information not covered (whose weighting factor is null) in the objective function criterion.

In the first case, when the weighting factor associated to the structure is null, Hybrid-Louvain detects 277 low density communities, which shows that the structural characteristics are still relevant for the communities [116] and therefore that the structural aspect is the basic element of the notion of communities as defined by Newman [116]. In the second case, 161 communities were obtained when the weighting factor associated to the attributes in the formula 4.11 is null. Thus partition density is higher than the previous case ignoring structural information. Indeed, Hybrid-Louvain approach is thus assimilated to a model based solely on structure information which includes topology based on directionality of edges.

In the third case, when the weighting factor associated to the directionality of the links is null, the hybrid approach becomes semi-hybrid as well as SAC1 and Combe's methods and obtains 89 communities with a density close to the one of SAC1, namely 0.241. When the edge directionality information is considered alone, a non-structured partition is observed as shown in Figure 4.11. On the other hand, we obtain 270 communities and a partition density of 0.018 when the weighting factors on both the attributes and the directionality are null, i.e. when only the relational aspect of the structure is considered. This result shows that directionality alone is not enough to obtain significant communities.

In Figure 4.12, there are three communities identified by different colors. The green nodes belong to the third community, since they are singleton and consequently do not belong to the connected component of the network. Likewise, they do not change their initial position because they do not improve the global modularity criterion. Figure 4.10 shows 9 communities, two of which are located in the related component, and the others are made up of the singletons nodes. This shows that some singletons nodes, by their attributes, can belong to related communities.

The observation after experimentation shows that these results together with communities in Figures 4.12 and 4.10 strengthen the interpretation according to that high density does not inevitably denote good separation of communities, unlike the observation made by Dang and Viennet in [37]. In the other hand, trough results in Table 4.8, density is correlated with the size of the partition; in fact, the greater the number of communities, the less dense the partition.

Table 4.8: Polblog results: Null Weighting factor consideration for Hybrid-Louvain

Weighting factor (WF)	Number of communities	Density
WF on Structure	277	0.017
WF on Attributes	161	0.115
WF on Directionality	89	0.211

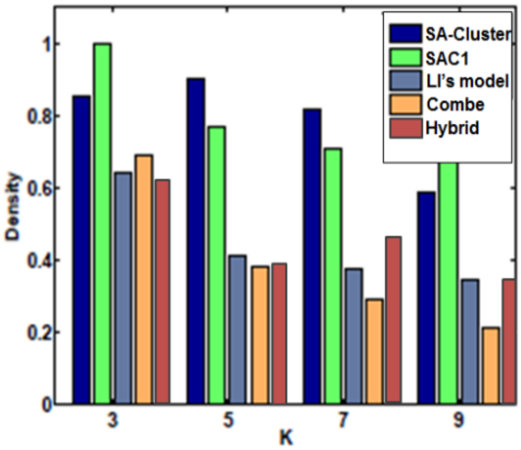


Figure 4.13: Density comparison on Political Blogs

4.4 Conclusion

In this chapter, we presented two contributions. In the first part, we presented a seed-centric approach of community detection in directed graphs, based on triads. The objective was to identify communities of interest based on triads, by showing the importance of triads in the detection of more cohesive communities, overcoming the modularity drawback related to edge directionality discrimination between in-degree and out-degree of the nodes. Indeed, real-world networks as addressed above, have the fundamental feature of high clustering coefficient degree. This property reflects the attachment of small degree nodes to *popular* / *hub* nodes, therefore it becomes important to take it into account. To this end, we defined a new similarity metric between potential kernel nodes to compute kernel scores, in order to select the effective nodes to be part of the kernels. Then after obtaining the kernels, it was a question of making migrate the other nodes of the network to the kernels for which they are most connected, via the implementation of a membership measure named NCI. This triad-based approach has shown its performance by the high results of the metrics utilized. In particular, the communities obtained have a density of more impotent triads.

Furthermore, the second contribution consisted in proposing a hybrid model for the detection of hybrid communities whose the interest is based on common features namely homogeneous nodes with the same pattern. It allowed to show the validity of the preceding kernel degree measure, included in the proposed composite "hybrid modularity" objective function. The latter integrates three types of information, namely relational information, i.e. link density, attribute-based information, and finally information based on the edge directionality. Through this function, we obtained, after application of a hierarchical clustering method, namely Louvain's method, more communities of interest depending on the type of information taken into account. We apply this method to Political blogs network, showing that there is an interest in using each of the types of information handled in the context.

The graphs used in this work do not take into account the multiple relationships that can exist between two nodes of a network, as it is the case of multidimensional networks. This limitation is discussed in the next chapter dealing with the community of interest discovery in multidimensional networks.

CHAPTER 5

Community discovery in multidimensional networks

5.1 Introduction

The detection of community called *multi-community*, to refer to community identified from multidimensional networks, has been the subject of many studies [82, 83], as described in Chapter 3. Many of them assume the existence of a community on each dimension of the entire multidimensional network [5, 144]. Therefore, it can lead to a substantial loss of information about the actual organization of the modeled system since the same importance is not always given to the different types of interactions between entities [120]. Thus, one aspect of such studies has been disregarded so far: the level of activity of a node in anyone of the dimensions. Indeed, an active node on one dimension can remain inactive on the rest of the dimensions [120]. This aspect has been mainly addressed in [18, 23] through the identification of the customized set of dimensions of interest for each vertex, namely the relevant dimensions. The problem is that the subspace of relevant dimensions of these methods takes into account the quantity of the neighborhood at the expense of the neighborhood quality in the clustering process. However, neighborhood quality could lead to a more meaningful communities discovery, because in practical applications, communities whose members possess common or similar interest (meaning neighbors), have a great promotion on intelligent information retrieval, marketing management and other information management domains [130]. This chapter deals with the community of interest discovery in multidimensional networks, based on the level of activity of entities in a dimension. The method presented in this chapter uses a proposed new measure of centrality, which is called stability. This measure has been the subject of a publication [50]. We start by defining in the section 5.2 the notion of community of interest in multidimensional graphs as well as the problem of detection of these communities of interest addressed in this thesis. In Section 5.3, we define the concepts used in the proposed method to solve the problem of communities of interest discovery in multidimensional graphs that we describe in section 5.4. Finally, in the section 5.5, we present an experimental study on the existing methods compared to the proposed one.

5.2 Community of interest discovery problem

One key aspect of multidimensional network analysis is to understand how important a particular dimension is over the others for the connectivity of a node. This importance of a dimension is a property of multidimensional networks that derives from node centrality metrics, with respect to the analysis of the within and across-dimension relations in the network [120].

In this section, we first present the characteristics of the multidimensional network we deal with in this study (Section 5.2.1). Thereafter, we describe the multi-community of interest problem addressed in this research (Section 5.2.2).

5.2.1 Multidimensional network model

Hereafter, we consider undirected and connected multidimensional networks, meaning those in which it exists at least one path that connects any pair of nodes from any dimension as shown in Figure 5.1. This figure illustrates a connected multidimensional network with a path designed as edges in red color, linking X and Y nodes of the first and third dimensions respectively. The union of the sets of nodes in all dimensions is not null. Formally, $\bigcup_{d \in D} V^d = V$ and $\forall d \in D, \exists d'\ s.t.$ $V^d \cap V^{d'=\emptyset}$. In contrast if $\exists r \in D$ such as $\forall s \in D,\ s \neq r$, and if $V^r \bigcap V^s = \emptyset$, then the network is not connected; consequently it does not correspond to the case of multidimensional network model treated in this work.

Methods based on the importance of dimension differ from the other approaches in their ability to support the relevance of dimensions in the process of community discovery. Some of them have been identified in the state of the art. More precisely, the aforementioned Multimap [38] and ABACUS [15] approaches (see Table 3.3 in Chapter 3) work in different manners. The former aims to minimize a modified version of the Infomap map equation [129] and determines relevant dimensions by using the coding scheme of the map equation to grant unique names to important structures of the network. The latter firstly extracts communities per dimension, then transforms the set of communities into a formal context. Afterward, frequent closed itemset is mined. Finally, relevant dimensions are determined, based on a support threshold fixed as an input parameter. In addition, Oualid *et al.* [23] is inspired by the LPA method [10] which requires initial labeling of each node of the network and assigns to a node the label carried by the majority of its neighbors. Likewise, the authors used the concept of dimension relevance as defined by Berlingerio [14], and assigned to each neighbor v of a node u, a weight standing for the relevant dimensions connecting u to its neighbors v. Remind that structural equivalence and homophilia are topological properties inducing communities of interest in social networks, which is a typical example of real complex networks. Since our work focuses on topological communities of interest, we have been interested in the last method, namely MDLPA [23], which is also based on the neighbourhood of nodes. Moreover, a node can be part of a community but not directly connected to the others: it is an outlier.

In the two paragraphs below, we describe the two aspects to be taken into consideration: accounting for the stable neighbourhood, and accounting for outliers.

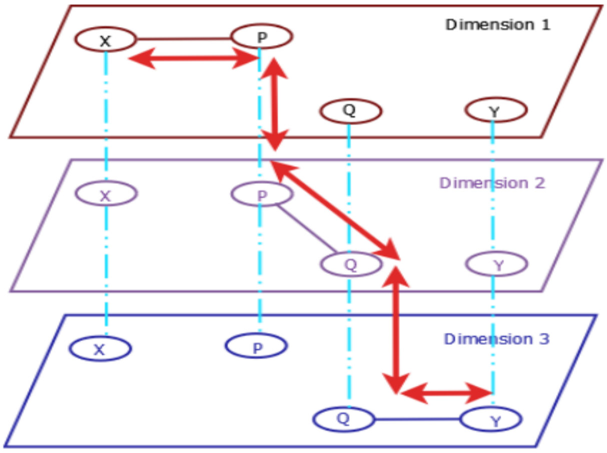


Figure 5.1: Connected multidimensional network.

Stable neighborhood

In MDLPA approach, the subspace of relevant dimensions $RD(v)$ of a node v is based on the degree of v, since it is computed through its fraction of direct neighbors, following edges belonging exclusively to dimensions $RD(v)$ [14]. In spite of the fact that MDLPA produces better cohesive communities according to its relevant dimensions, this method has the main drawback of focusing on the number of neighbors of the node and not on the quality of the neighborhood. This consideration stipulates that if a node has the same maximum degree in a subset of dimensions, then the dimensions concerned by this subset are relevant to it. Such a consideration seems, however, too restrictive, since, if this subset is equal to all the dimensions of the entire multidimensional network, then all the dimensions of the network would be relevant for him. This means that the node has the same degree of activity on all dimensions. However, in practice, this node would have a preference of one dimension in which it would have more trusted friends than in another. This dimension would correspond for example to the dimension in which its friends are stable. This indicates the trust and therefore the interest that binds it to this dimension.

Non-exclusiveness principle: outlier's inclusion

Existing methods only manage exclusiveness, as they consider a node to belong to a community if there is a direct connection between that node and other nodes of the same relevant dimensions. However, an individual of an *irrelevant* dimension (see Section 5.3.3) would be of vital importance to another individual of the community. So ignoring this crucial node in a community would be a mistake. We refer to this behavior as the principle of the community's *non-exclusiveness*, which is also addressed in the proposed approach. Figure 5.2 presents a toy example of a multi-community taking into account that non-exclusiveness property. Nodes are described by their relevant dimensions being either A, B or D and the crucial node T expresses the outlier since its relevant dimension (D) defers from relevant dimension of the community (A,B). To better illustrate this principle of *non-exclusiveness*, consider the example of a business *2-dimensional network* including two dimensions: medical and teaching staff relationships. It consists in two friends described by their dimension medical and teaching respectively. The one from teaching staff, when ill, reaches the hospital and is kindly received by his long-time friend who is from the medical staff. We see that they are not directly tied and belong to different relevant dimensions (teaching and medical staff respectively), but could be the subject of the same community later, namely the community of major friends, including some teachers and the doctor.

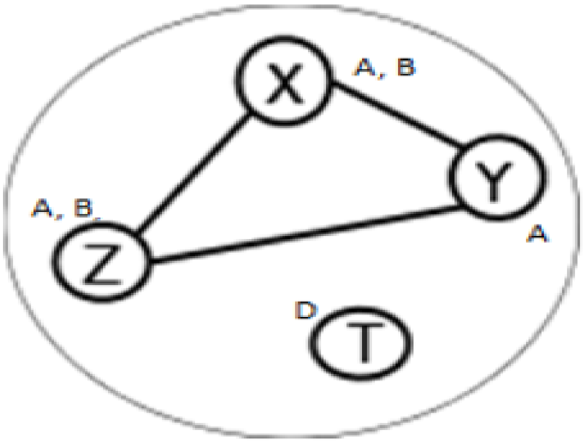


Figure 5.2: A multi-community with non-exclusiveness principle on node T

5.2.2 Multi-community of interest problem

As highlighted in [36], the new challenge focuses on the question not of how to detect communities, but on what kind of communities are we interested in detecting. The question "*what is a community in multidimensional networks?*"is an embarrassed one. Intuitively, each dimension can have its communities, and these communities may vary from one dimension to another. As the behaviors of communities differ from one another from different points of views, the definition of a community in multidimensional networks should then depend on the particular task [104]. Remember that we focus on communities of interest detection in complex networks in this research. Authors in [36] gave a meta definition of community as *a set of entities that share some closely correlated actions with the other entities of the set.* Here we consider similar relevant dimensions as a particular and very important kind of action. This kind of action refers to an interest involving nodes of the same multi-community. Thus, when a set of nodes are grouped according to the level of interest they give to a specific or set of specific dimensions, we attend to communities of interest in multidimensional networks. We now give some following suitable definitions.

Definition 5.2.1. ***(Multi-Community of Interest)****. A community of interest in multidimensional networks called Multi-Community of interest is a set of nodes which have a similar center of interest through their relevant dimensions.*

Our objective is to identify covers denoting partitions of overlapping multi-communities, whose nodes are not only densely tied, but also more active in similar dimensions encountered in the multi-community, from the multigraph G, where a node $v \in V$ can belong simultaneously to several multi-communities $C_l = (V_l, D_l)$, $l = 1...L$, such that L is the number of multi-communities (previously unknown). In this work we distinguish three subsets, namely $RD(u)$, D_l and DF_l such that $RD(u) \subseteq D$ and $D_l \subseteq DF_l \subseteq D$. To illustrate, consider Figure 5.2 standing for a multi-community l where $D_l = \{A, B\}$ and $DF_l = \{A, B, D\}$. Our approach has the advantage of taking into account the "outliers", which are nodes with a relevant dimension different from that of the community (Node T).

A multi-community is said to be **densely relevant** as the majority of its nodes are both densely linked and described by D_l subspace of relevant dimensions. More precisely, the proposed approach aims to discover *densely relevant* multi-communities, since it is based on both the structural information and the similarity of the relevant dimensions. The former takes into account the density of links between nodes in the same community. The latter consists in enriching the attribute vector of each node (which can initially be empty in the case of a non-attributed multigraph) with its relevant dimensions. One can add relevant dimensions to the set of node attributes for multi-community discovery method in attribute networks. For the sake of simplicity, we reserve for future work this part of the problem, and we focus only on the description of nodes by their relevant dimensions. Figure 5.2 and the right sides of Figures 5.5(b) and 5.6(b) below show toy examples of our multi-community scheme, with densely-relevant nodes.

The following properties must be satisfied:

- The multi-community $C_l = (V_l, D_l)$, $l = 1...L$ is a non-empty subset of the network G;
- In each multi-community $C_l = (V_l, D_l)$, the nodes in V_l must be more densely relevant across all dimensions of D_l, than elsewhere in the multigraph. In other words, not only must the density of internal links in C_l, be higher than with nodes external to C_l, but also most of C_l's nodes must have attribute values of more similar relevant dimensions than with nodes external to C_l;
- It is possible to have a node u described by irrelevant dimension attributes and belonging to C_l;
- The subsets of dimensions D_l can be overlapping and may have different cardinalities;
- The multi-communities can be overlapping and of varying sizes.

Definition 5.2.2. ***(Multi-community discovery problem in attribute-based multidimensional network).*** *Given a multigraph G, find a cover $P(G)$ of densely relevant multi-communities that maximizes the dimension-based modularity objective function Q_{dim}.*

$$P(G) = \underset{P_i}{\text{argmax}}(Q_{dim}(P_i(G))) \tag{5.1}$$

where $P_i(G)$ is the i^{th} cover among a set of those obtained in the iterative process and Q_{dim} the modularity dimension, defined in Section 5.3.4.

5.3 Basic concepts

Some basic concepts are defined in this section for better understanding of the newly proposed approach. Given a multidimensional graph $G =< V, E, D >$. After its frequential aggregation, the G_{flat} resulting graph stands for a weighted flattened graph in which nodes are described by their relevant dimensions and edges are weighted through the frequential weighting scheme. Formally we have:

$$G_{flat} =< V, E_{flat} >$$

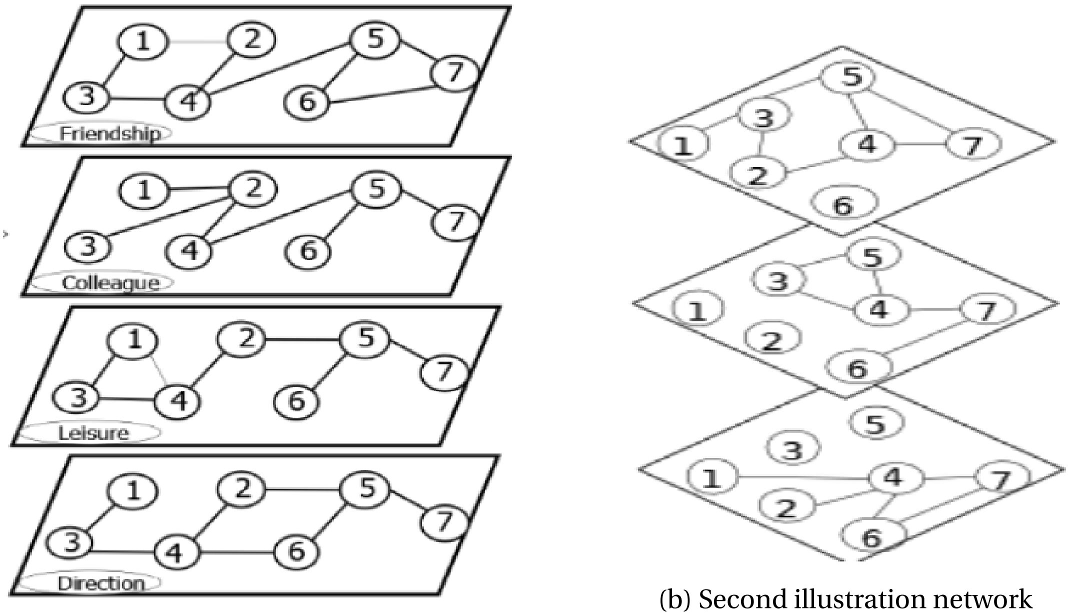
where E_{flat} is the weighted edges set. The weight w_{uv} of a link $(u, v) \in E_{flat}$ is computed by its redundancy in the different dimensions [14] such as: $w_{uv} =\| \{d / A^d_{uv} \neq 0\} \|$. Each node u is associated with an attributed vector of dimensions namely $RD(u) \in D$.

As the approach focuses on both the structural information and the similarity of relevant dimensions, some concepts are essential to be defined, mainly those related to the centrality and connectivity of nodes across dimensions. These centrality and connectivity metrics can reflect the significance that a node gives to a dimension. Unlike the wellknown measure of degree centrality, which considers the importance of a node with respect to its number of neighbors (degree), the new measure called *Stability* focuses on its common neighbors, as described in Subsection 5.3.2. This reflects the fact that the node is more comfortable or shares more similar behaviors with these neighbors than with others, so the dimensions in which it has a stable neighborhood are of interest to it. As studied in [120], the activity of a node in a dimension is often correlated with its activity in some other dimensions. To this vein, the new centrality metric depends on the inter-dimensional correlation, as described in Subsection 5.3.1. Figure 5.3 shows two multidimensional networks on which the illustrations of the proposed method will be carried out

5.3.1 Inter-dimensional correlation

The inter-dimensional correlation of a node u computes the proportion of the common neighborhood of this node between two dimensions p and q, through a *Jaccard index* similarity. Figure 5.4a shows inter-dimensional correlations, in blue color, for each node from the network in Figure 5.3a.

Definition 5.3.1. ***(Inter-dimensional correlation)****. The inter-dimensional correlation of a node u between two dimensions p and q is described by the function $Cor : V \times D \times D \longrightarrow [0, 1]$ as :.*



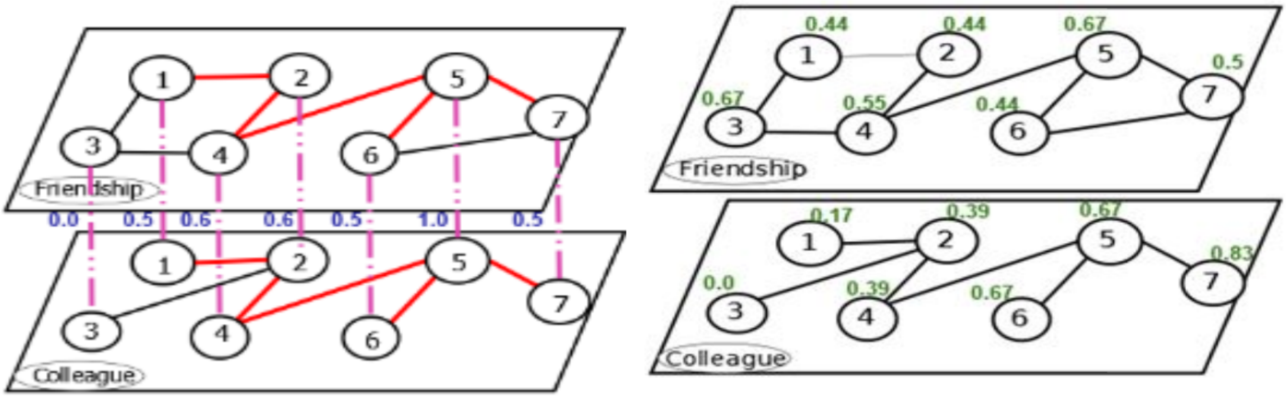
(a) First illustration network

(b) Second illustration network

Figure 5.3: Examples of multidimensional network

$$Cor(u,p,q) = \frac{|\Gamma_u^p \cap \Gamma_u^q|}{|\Gamma_u^p \cup \Gamma_u^q|} \tag{5.2}$$

Its values lie in the interval [0,1]. At 1, the neighborhood of a node u in both dimensions p and q is the same. This is the case of the labeled node (5) in Figure 5.4a. For better visualization, we illustrated the correlation between the two first dimensions of the network.



(a) Inter-dimensional Correlation illustration

(b) Stability of nodes in Friendship and Colleague dimensions respectively

Figure 5.4: Illustration of the Stability centrality from network of Figure 5.3a

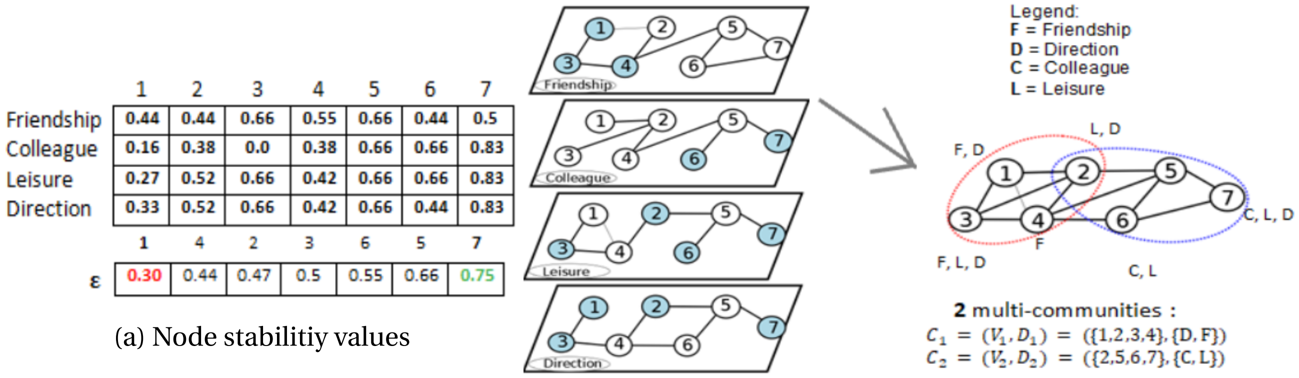
5.3.2 Stability centrality

The weight of the nodes have been the subject of some studies. According to graph theory, this weight corresponds to the sum of the weights of the edges incident to this node. It is a variant of the degree centrality. This centrality measure shows that the importance of a node depends on the number of communications it establishes with its neighborhood. It represents its activity level in a network. Extending this measure in multidimensional networks, Nicosia [120] demonstrated that the activity of a node in a particular dimension is very often correlated with its activity in another dimension. He considered the centrality degree as a measure of the node activity in a dimension. However, the number of neighbors, being the *quantitative* neighborhood aspect, seems to be meaningless when studying behavior of entities in a context of correlated dimensions, since it only favors the variety of received information. Then, it becomes necessary to maintain the stable behavior of a node, considering the *qualitative* neighborhood aspect, in order to maintain trust among its community membership. The idea behind this centrality metric is that a stable node is the more important because its stable neighbors rely on it. Likewise, the more the neighbors are the same, the more the friendship relationship is reliable. Thus, its inter-dimensional correlation is used and helps to compute the proportion of the common neighborhood of this node between two dimensions p and q, as defined through Definition 5.3.3. Figure 5.4 shows an illustration of stability centrality computation.

Definition 5.3.2. ***(The stability centrality of a node in a dimension)***. *Stability centrality of node u in dimension q measures its common neighborhood between q and the other dimensions. The function Stability* : $V \times D \longrightarrow [0,1]$ *is defined as:*

$$Stability(u,q) = \frac{1}{|D|-1} \sum_{p=0,p\neq q}^{|D|-1} Cor(u,p,q) \tag{5.3}$$

where $|D|$ denotes the number of dimensions. Stability takes its value in $[0,1]$. We refer to *disassortative stability* when its neighborhood is totally different in all dimensions; stability tends to be null. Otherwise, it is the *assortative stability*; it tends to its maximal value 1. In this work, the node with the lowest disassortative stability is unstable and the one with the highest assortative stability is the most stable over the network. As shown in Figures 5.5a and 5.6a, node 1 possesses a disassortative stability, unlike node 7 which gets an assortative stability. Figure 5.4b indicates the node stability centrality in green color, of each node for the two displayed dimensions in network of Figure 5.3a



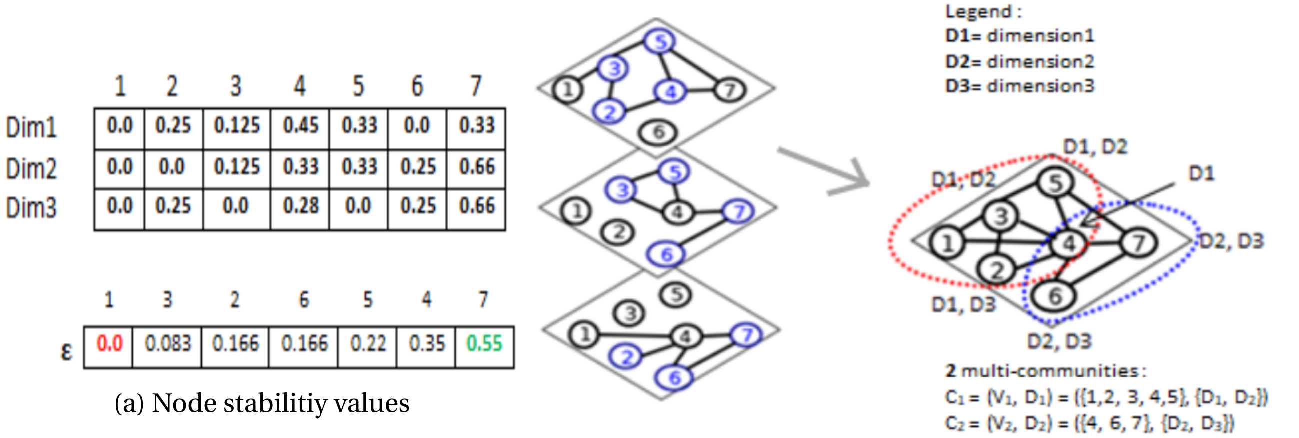
	1	2	3	4	5	6	7
Friendship	0.44	0.44	0.66	0.55	0.66	0.44	0.5
Colleague	0.16	0.38	0.0	0.38	0.66	0.66	0.83
Leisure	0.27	0.52	0.66	0.42	0.66	0.66	0.83
Direction	0.33	0.52	0.66	0.42	0.66	0.44	0.83

	1	4	2	3	6	5	7
ε	0.30	0.44	0.47	0.5	0.55	0.66	0.75

(a) Node stabilitiy values

(b) Multi-community computation

Figure 5.5: Process of relevant dimensions extraction from stability to multi-community for the example in Figure 5.3a



	1	2	3	4	5	6	7
Dim1	0.0	0.25	0.125	0.45	0.33	0.0	0.33
Dim2	0.0	0.0	0.125	0.33	0.33	0.25	0.66
Dim3	0.0	0.25	0.0	0.28	0.0	0.25	0.66

	1	3	2	6	5	4	7
ε	0.0	0.083	0.166	0.166	0.22	0.35	0.55

(a) Node stabilitiy values

(b) Multi-community computation

Figure 5.6: Process of relevant dimensions extraction from stability to multi-community for the example in Figure 5.3b

5.3.3 Relevant dimension

The concept of dimension relevance of a node addressed by Oualid [23] in their approach of community discovery, stresses on that in which the node has the most important exclusive degree as defined by Berlingerio [14]. Indeed, it computes the fraction of neighbors directly reachable from node u following edges belonging exclusively to subset of dimensions Dl as shown in Equation 5.4. It is a variation of Equation 2.1.13 described in Chapter 2. The resulting communities C_l of these authors [23] are disjointed and consist of nodes, densely connected through the subset D_l of relevant dimensions.

$$DR_{XOR}(u, D_l) = \frac{Neighbors(u, D_l)}{Neighbors(u, D)} \tag{5.4}$$

The authors focus only on the number of neighbors (i.e. degree) without stressing on information on the type of neighbors. Indeed, the type of neighbors that an individual possesses implies an underlying semantic. In this way, we give a new definition of the notion of dimension relevance that takes into account the stability centrality defined above (see Definition 5.3.3).

Definition 5.3.3. ***(Relevant Dimension)****. The relevant dimensions $RD(u)$ of a node u refers to those dimensions for which the node has a stability centrality greater than a certain threshold ε. It is described by the function $RD : V \longrightarrow D$ as :*

$$RD(u) = \{q \in D, s.t.Stability(u,q) > \varepsilon\} \tag{5.5}$$

The threshold ε is defined in the Section 5.4.1 and two illustrations for its computation are shown in Figures 5.5a and 5.6a. A node can have several relevant dimensions $RD(u)$ such as $RD(u)D$. When the node u has a stability centrality whose value is upper than a threshold ε, it is said that the node u is stable for the subset of dimensions $RD(u)$, or that the dimensions in the subset $RD(u)$ are relevant for the node u. Furthermore, a dimension d of a community C_l is irrelevant for a node v when it belongs to its set of relevant dimensions $RD(v)$ but does not belong to the set D_l of relevant dimensions of the community. Formally, the d dimension is irrelevant for v if $d \in RD(v)$ and $d \notin D_l$. The left sides of Figures 5.5a and 5.6a show a multidimensional network in which nodes in blue color express their relevant dimensions.

5.3.4 Dimension-based objective function

The proposed approach optimizes the following defined modularity dimension objective function Q_{dim} in Definition 5.3.4.

Definition 5.3.4. ***(Modularity dimension)****. The modularity dimension Q_{dim} is the difference between the expected number of links within a multi-community of a subspace of relevant dimensions and the number of links within a community retrieved from a graph with the same degree distribution as the actual graph.*

$$Q_{dim} = \sum_{C} \sum_{i,j \in C} (\alpha.Q + (1-\alpha).Sim_D) \tag{5.6}$$

where α is a weighting factor, $0 \leq \alpha \leq 1$, Q and Sim_D respectively denote Newman's modularity [118] and dimension-based attribute similarity function computed using Equation 5.7. Sim_D is a *Jaccard index variant* in G_{flat} based on relevant dimension attributes. This objective function describes the idea that node merging into a community maximizes simultaneously its intra-density of links and its similarity of nodes based on their relevant dimensions, according to the value of α.

$$Sim_D = \frac{1}{1+\sum_C \sum_{i,j \in C} (\frac{|RD(i) \cap RD(j)|}{|RD(i) \cup RD(j)|})} \tag{5.7}$$

5.4 Multi-community discovery algorithm: UCAD

The illustration workflow of UCAD is described through Equation 5.8, where ψ is a function that converts a multidimensional network G to an aggregated and attributed network Gf_{lat}, $BGLL_{new}$ is the algorithm for multi-community discovery based on stability centrality of nodes. We next give description of the workflow.

$$G \xrightarrow{\psi} G_{flat} \xrightarrow{BGLL_{new}} C_{flat} \longrightarrow Metrics \tag{5.8}$$

The general scheme of our proposed commUnity disCovery method in Attribute-based multiDimensional networks (UCAD), is divided into two main achievements:

1. Preprocessing step of network flattening by ψ
2. Multi-community discovery through BGLL algorithm[1] transformation by $BGLL_{new}$.

Algorithm 5 reports the pseudo code of the preprocessing step and Algorithm 6 reports the second step, $BGLL_{new}$.

5.4.1 Preprocessing step

The ψ preprocessing function consists in 2 sub-tasks: *(a)* the identification of the relevant dimensions subset $RD(u) \subseteq D$ for each node u; *(b)* aggregation of the graph based on relevant dimensions.

The first sub-task consists in 3 phases:

[1] Blondel, Guillaume, Lambiotte and Lefèvre known as Louvain algorithm [60]

1. Determination of the stability centrality of each node on each dimension (Line 4): it consists in average of inter-dimension correlation as defined in Equation 5.3.

2. Computation of the ε threshold as a mean (Line 5), denoting the intuitive idea that the information is more coherent when a node neighborhood is better stable over the majority of dimensions, as defined in Equation 5.9. The choice of the mean as threshold stems from the fact that values below that mean denote a low stability, and therefore, a more changeable node neighborhood.

3. Choice of the subspace of dimensions (Lines 7-9), for which the node's stability centrality is greater than ε (see Equation 5.5)

$$\varepsilon(u) = \frac{1}{|D|} \sum_{d=1}^{|D|} Stability(u,d) \tag{5.9}$$

The second sub-task consists in computing weights of edges in the attributed flattened graph $G_{flat} =< V, E_{flat} >$ (Lines 11-14). We use the *frequential* weighting scheme to label edges and each node is described or enriched by its relevant dimensions, in order to preserve most of the original multidimensional information residing in G, as degree, neighborhood, etc.

5.4.2 Algorithmic step of the $BGLL_{new}$ method

As described in the pseudo-code through Algorithm 6, this step allows to extend the BGLL algorithm [16] for aggregated graphs stressing on relevant dimension-based attributes. The advantage of this algorithm is its sensitivity to relevant dimensions and overlapping nature of clusters. The proposed approach highlights the topological features of real-world networks namely preferential attachment of nodes and geodesic average distance k. The first node to be inserted in the list is the node v with maximal degree. Then, it is followed by its non-allocated k-neighbors[2] in decreasing degree order. An iterative process inserts neighbors of nodes until the overall mileage of nodes of G_{flat}.

Thus, unlike the BGLL algorithm which initializes each cluster with a randomly selected node and migrates it to the cluster that maximizes modularity, our approach is based on a deterministic scheme that computes a set of seed nodes, and re-orders the remaining nodes for optimization of the new defined modularity. $BGLL_{new}$ uses these seeds called *sub-networks*, as starting clusters s_i. Indeed, the proposed method determines an initial Number of Sub-Networks (NSN) containing

[2] Neighbors of a node according to a distance k

Algorithm 5 ψ: Generation of the aggregated attributed-based graph

```
Require: multidimensional graph $G =< V, E, D >$
Ensure: Weighted attributed-based graph $G_{flat} =< V, E_{flat} >$
 1: for Each node $u \in V$ do
 2:     $Sum[u] \leftarrow 0$; $\psi \leftarrow 0$; $RD(u) \leftarrow \emptyset$
 3:     for Each dimension $d \in D$ do
 4:         $Sum[u] \leftarrow + = Stability(u, d)$
 5:         $\varepsilon \leftarrow + = Sum[u]/|D|$ #threshold computation
 6:     end for
 7:     if $Stability(u, d) > \varepsilon$ then
 8:         $RD(u) \leftarrow RD(u) \cup \{d\}$ # assign attributes standing for relevant dimensions
 9:     end if
10: end for
11: for each edge $(u, v, d) \in E$ do
12:     $E_{flat} \leftarrow E_{flat} \cup \{(u, v)\}$
13:     $w_{uv} \leftarrow w_{uv} + 1$ #Weights of each edge
14: end for
15: return$G_{flat}$
```

seed nodes taken in a defined order. Like this, we define a $Node_{List}$ (Line 1), referring to a list of nodes according to a degree-based selection order. At the beginning of the process, the minimal number of expected resulting sub-networks or communities equals to $\sqrt{n}$ (Line 2), as the modularity is used in the optimization process. Indeed, in [69], it has been proved for an important number of networks that the modularity is maximal when the number of communities reaches $\sqrt{n}$. In the experiments, the number of discovered communities also accounts for this value. Nodes in s_i are distanced by a maximum average path k based on geodesic (Line 3). The objective of determining the initial clusters is double. The former promotes the gain of modularity for small-scale communities. The latter helps to compute covers (overlapping communities) by initializing node membership to several communities simultaneously through *InitialComm* function (Line 4).

Following BGLL algorithm which operates in two phases, $BGLL\_new$ achieves two functions, that of modularity optimization adapted to the proposed "modularity dimension", which generates optimal communities via the *OptimalComm* function (Line 6), and that of agglomeration of these communities via the *Compress* method (Line 7). This *Compress* process consists in grouping the nodes in the same communities to a new community-node. The weights between new nodes are given by the sum of weights of the links between nodes in the corresponding communities [16]. To determine the dimension-based similarity between two communities, we use D_l, subspace of relevant dimensions in the corresponding l community.

Algorithm 6 ψ: Generation of the aggregated attributed-based graph

```
Require: Aggregated attributed-based graph $G_{flat} =< V, E_{flat} >$
Ensure: Densely relevant Cover $P$
 1: Compute $Node_{List}$ #List of nodes for selection, according to decreasing degree order
 2: for Each dimension $d \in D$ do
 3:     $MinSN \leftarrow \sqrt{n}$
 4:     $k \leftarrow aver^{dist}(G_{flat})$ #Compute the average distance of the flattened graph
 5:     $Clust_{init} \leftarrow InitialComm(Node_{List}, k)$ #Compute seed communities
 6:     repeat
 7:         $OptC \leftarrow optimalComm(Clust_{init})$ #Node merging function
 8:         $P \leftarrow Compress(OptC)$
 9:         $NSN \leftarrow |P|$
10:     until NSN > MinSN
11:     return $P$
```

5.4.3 Computational complexity analysis

Given $K = |D|$ dimensions from the multidimensional network $G =< V, E, D >$. we can analyze the time complexity for each step as follows:

Algorithm 5: As the method has to compute stability and subspace of relevance dimensions for each node in each dimension (Line 1 to 10), the time complexity is $O(nK)$). The last part (Line 11 to 14) assigns weight to edges of the aggregated graph, in $O(m)$. Finally, the time complexity of the preprocessing step is in $O(nK + m)$.

Algorithm 6: At the beginning, the algorithm extracts list of nodes in their decreasing order of degree (Line 1) in $O(n^2 + nlog(n))$ (n^2 for the degrees of n nodes and $nlogn$ for the sort). It computes the distance of the flattened aggregated graph (Line 3), using Dijkstra algorithm, in $O(m + nlog(n))$. The initial clusters also called seed communities from which to begin the community extraction process (Line 4) are computed in $O(n^2)$. The second part of this algorithm (Line 5 to 9) describe the core operation of the proposed method. *OptimalComm* function (Line 6) computes optimal communities. Each node migrates from its seed community to another one if the modularity is improved. In the better case, we will have a single cluster containing all the nodes, hence there is no merging. In the worst case, we will have as many clusters as there are nodes. The number of steps needed to verify whether the running modularity maximum can be improved or not should be $O(n^2)$. The Compress function (Line 7) aggregates nodes of the same subnetwork in a single node in $O(n^2)$. Since the process can at worst be iterated a significant number of times equal to $\sqrt{n}$, then the main part can perform in $O(n^{1/2}(n^2 + m^2))$ which can be reduced to $O(n^2)$. Thus, the complexity of the algorithm is $O(m + nlog(n) + n^2)$. In final, the preprocessing step being in $O(nK + m)$ and the $BGLL_{new}$ step being in $O(m + n^2)$, the global complexity of the overall UCAD

model would be $O(m+n^2\sqrt{n})$.

5.5 Experimental Evaluation

In the following, Section 5.5.1 summarizes the evaluation datasets, Section 5.5.2 introduces competing methods, Section 5.5.3 describes experimental settings and results are presented in Section 5.5.4.

5.5.1 Data

We used six real-world multidimensional network datasets, namely Florentine [122], AUCS [108], Biogrid [136], DBLP [18], Bankwiring [145] and Monastery [25]. Some of them are available in [42]. Florentine describes relations among 16 politically prominent families in the city of Florence around the year 1430 structured in two blocs: business ties and marriage alliances. AUCS, an attributed multidimensional network, models relationships between 61 employees of Aarhus University Computer Science department considering five different aspects: coworking, having lunch together, Facebook friendship, offline friendship, and coauthorship. As our method considers connected multidimensional networks (see Section 5.2.1), we used the subset of 52 nodes (out of 61) sampled in [23] because the others are disconnected from the network. Each node is described by two attributes: Workgroup and Grade. Biogrid is a protein-protein interaction network, where dimensions correspond to seven different types of interactions between proteins. In DBLP, nodes correspond to authors and dimensions represent the top-50 Computer Science conferences. Two authors are connected on a dimension if they co-authored at least two papers together in a particular conference. Bankwiring is the observational data on 14 employees from the bank wiring room. The interaction categories include 6 dimensions. For Monastery, Sampson recorded the social interactions between a group of monks while residing as a vision experimenter, and collected many sociometric rankings. Their views on the types of relationships among 18 monks were classified into 10 dimensions through 510 connections between them.

Table 5.1 summarizes main characteristics of our evaluation datasets. Node relations in all datasets are treated as symmetric and AUCS is the only dataset with the nodes' attributes. We denote with A_{deg} the average degree of a node considering multiple edges and with A_{dim} the average number of dimensions in which a node is present. OCN (OmniConnected Nodes) [14] computes the percentage of nodes that exist in all the dimensions of the network. Note that DBLP is the richest in term of dimensions, and that Biogrid and DBLP have nodes, in average involved in less than two dimensions (resp. 1.9 and 1.35). Node degree distribution in DBLP is low (3.8).

Table 5.1: Main characteristics of the multidimensional network datasets

Dataset	#Nodes	#Edges	#Dim	Density	A_{deg}	A$_{dim}$	OCN
Florentine	16	35	2	0.21	4.38	1.87	0.69
AUCS	61	620	5	0.114	20.33	3.67	0.18
Biogrid	8215	43366	7	$4.8e-4$	17.6	1.9	0.074
DBLP	83901	159302	50	$8.9e-5$	3.8	1.35	0.018
Bankwiring	14	110	6	0.054	15.43	4.43	0.29
Monastery	18	510	10	0.019	38.83	3.44	0.78

5.5.2 Competing methods

In order to demonstrate the efficiency of UCAD, five approaches considering the three core contributions of our community discovery algorithm were selected: approach based on relevance of dimensions (MDLPA[3] [23])to evaluate the level of activity of a node, those considering overlapping feature of the structure used from the R Multinet package [158] (ABACUS [15], Generalized Louvain (gLouvain) [113], Multiplex CPM (Mul-CPM) [146], Multimap [38]), and the overall abovementioned methods to evaluate the determinism property of our method.

5.5.3 Experimental settings

The performance evaluation criteria for the selected algorithms are classified into two categories of measures: Mesoscopic and macroscopic measurements. For the former, an evaluation based on community level is performed. We assessed the behavior of the proposed UCAD method in terms of: *(1)* size of extracted overlapping multi-communities, *(2)* Multi-Community Density: (MCD), *(3)* distribution of dimensions involved in each multi-community: Redundancy (ρ), *(4)* Number of triads in each multi-community: Triad Multi-Community density (TMC_{dens}), *(5)* Impact of relevance dimension: Relevance Multi-Community density (RMC_{dens}). *(6)* quality of the cover: multislice modularity $Q_{multislice}$ [113].

Note that the 5_{th} point based on relevant dimension evaluation could be tracked through other measures, by consideration of the subspace of relevant dimensions of the multi-community, D_l,

[3] Source code is available on https://github.com/BoutemineOualid/MDLPA-Algorithm

instead of the subspace of relevant dimensions found in the multi-community, DF_l as described in Section 5.2.2. Mesoscopic measures are described in the following:

- The redundancy as defined in [13], for a community l, calculates the ratio between the number of links existing in at least two dimensions $(\overline{\overline{P_l}})$ and the total number of links (P_l) in the community. Equation 5.10 describes it.

$$\rho_l = \sum_{(u,v)\in\bar{\bar{P}}_l} \frac{|\{d : \exists(u,v,d)\in E\}|}{K.|P_l|} \tag{5.10}$$

- $MCD(l)$ as defined in [15], for a community l, computes its proportion of links as shown in Equation 5.11.

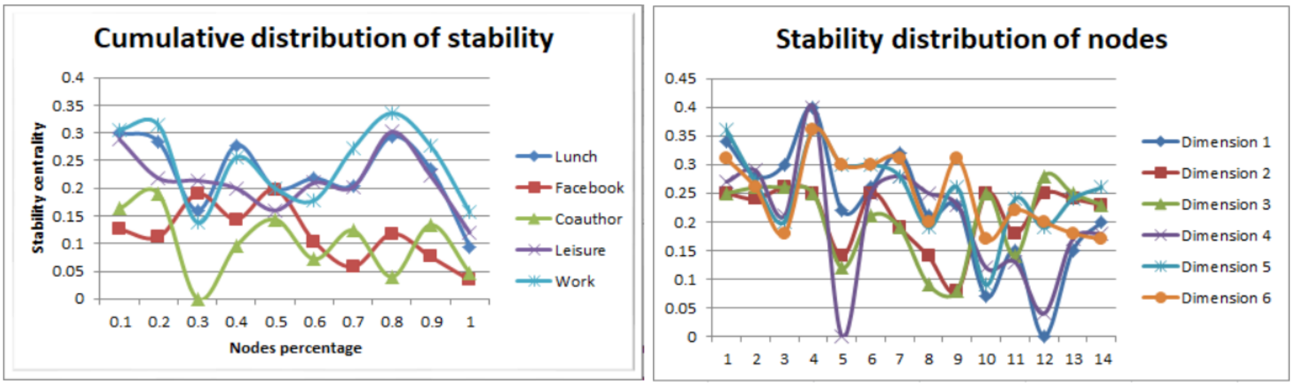
$$MCD(l) = \frac{2.|E_l|}{|DF_l|.|V_l|.(|V_l-1|)} \tag{5.11}$$

- The simple and easy to compute defined measure $TMC_{dens}(l)$ being the number of triads in a community l normalized by the maximum possible for that community, or in formula 5.12

$$TMC_{dens} = \frac{|\Delta_l|}{|DF_l|.\binom{3}{|V_l|}} \tag{5.12}$$

- For evaluating the proportion of relevant dimensions in a community l, we define another measure namely $RMC_{dens}(l)$ being the number of nodes in a community l belonging to a subspace of relevant dimensions D_l normalized by the maximum possible for that community, or in formula 5.13

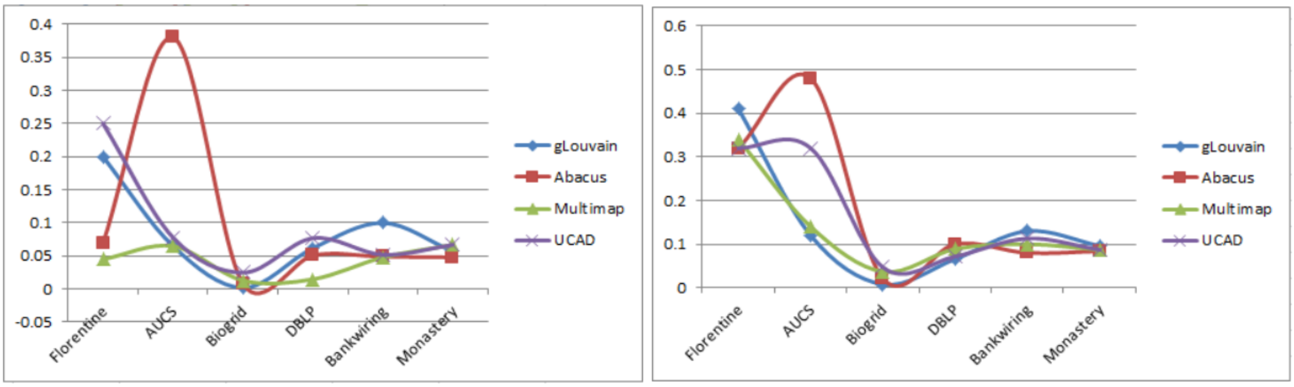
$$RMC_{dens} = \frac{|V_{D_l}|}{|DF_l|.|V_l|} \tag{5.13}$$



(a) Assortative behavior of nodes on AUCS

(b) Disassortative behavior of nodes on Bankwiring

Figure 5.7: Stability centrality on node behavior



(a) TMC: evaluation based on triads

(b) MCD: density of links in multicommunities

Figure 5.8: Mesoscopic measure's evaluation based on density, for $\alpha = 0.5$

Given that these measures apart the modularity are computed for each community, we compute the average to assess the overall cover $P(G)$. In the remainder, we refer to these metrics designed for the average of the cover as $\rho, MCD, TMC_{dens}, RMC_{dens}$ respectively.

5.5.4 Results

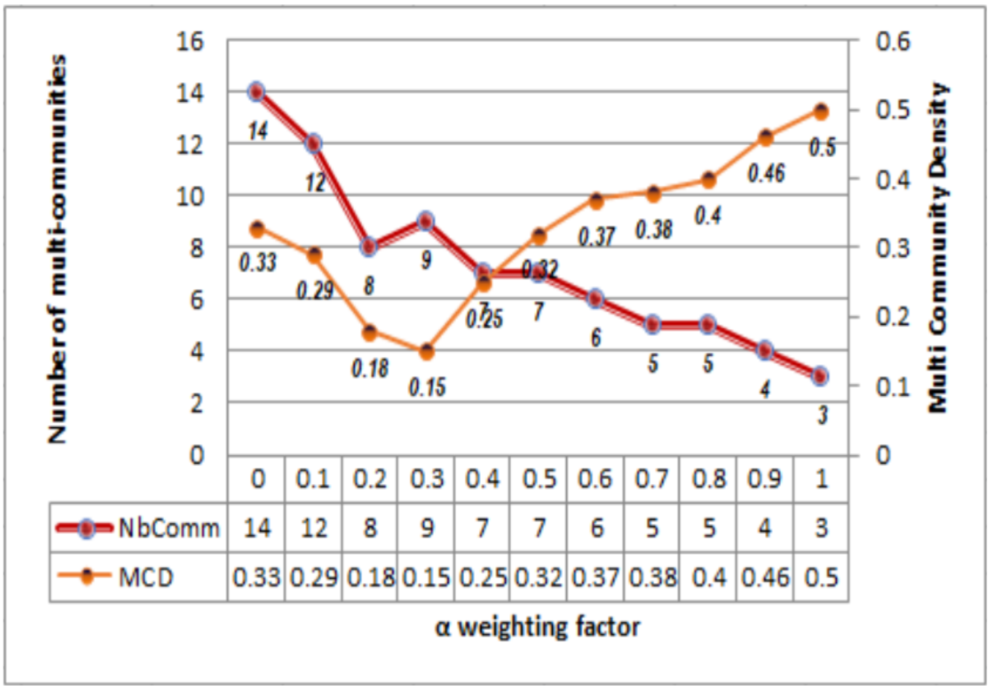
In this section, we present experimental results in three paragraphs as follow: The first one presents how the stability is correlated to the density of links. The second one stresses on the behavior of UCAD with respect to the communities identified. The third one compares the UCAD method with methods that reveal the overlapping nature of communities and that focus on dimension relevance.

Stability centrality evaluation

The stability centrality is assessed by comparing its values according to the density of links in the resulting structures. Through our analysis, it is reported that the stability centrality is correlated to the MCD metric.

Indeed, MCD computes the density of links between nodes of the same community. Since stability deals with the common neighborhood of a node across all dimensions, then intuitively that node with its stable neighbors would eventually be densely tied together in a multi-community.

According to Figure 5.9, with $\alpha = 1$, UCAD extracts 3 multi-communities with a value of MCD of 0.5. Since Figure 5.7a shows that the set of nodes is predominantly described by 3 dimensions (Lunch, Leisure and Work) among the 5 existing ones (which makes a ratio of $3/5 = 0.6$). Then it is quite acceptable that the nodes are mid-density. In fact, the value of MCD, i.e. 0.5, is closed to 0.6.



	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
NbComm	14	12	8	9	7	7	6	5	5	4	3
MCD	0.33	0.29	0.18	0.15	0.25	0.32	0.37	0.38	0.4	0.46	0.5

Figure 5.9: How α affects Relevant dimension-based attributes on AUCS.

Table 5.2: Number of Communities

Dataset($\sqrt{n}$)	gLouvain	ABACUS	Multimap	Mul-CPM	UCAD
Florentine (4)	5.4	6.2	3	4.5	3
AUCS (7.8)	5.3	55.4	4.8	36.1	7
Biogrid (90.6)	806.5	7759.5	141.6	327.5	86
DBLP (289.6)	6705.1	9506.8	411.5	1975.5	283
Bankwiring (3.7)	2	23.5	1.1	12.5	2
Monastery (4.24)	3.2	142.1	1	22	1

Table 5.3: Community size

Dataset	gLouvain	ABACUS	Multimap	Mul-CPM	UCAD
	μ vs σ	μ vs σ	μ vs σ	μ vs σ	μ vs σ
Florentine	3.5 ± 0.08	4.5 ± 0.05	4.8 ± 0.15	4.00 ± 0.09	5
AUCS	7.33 ± 0.04	5.1 ± 0.02	12.2 ± 0.41	6.5 ± 0.25	9.5
Biogrid	152.4 ± 90.19	387.8 ± 100.1	2546.4 ± 111.1	2081.4 ± 58.1	4710.5
DBLP	127.5 ± 50.8	670.1 ± 89.6	557.6 ± 100	1231.4 ± 79.1	8310.7
Bankwiring	7.9 ± 0.02	5.5 ± 0.1	14 ± 0.01	6.7 ± 0.31	14.4
Monastery	10.1 ± 1.5	9.4 ± 0.08	18 ± 0.02	9.5 ± 0.22	18

UCAD Evaluation

Here we assess two questions. **Q1**: Does our approach determine communities that are densely connected ? **Q2**: How does α affects dimension relevance of nodes? Let us answer **Q1** and **Q2**.

Q1: Density-based property of covers. This paragraph assesses both TMC_{dens} and MCD measures, as the proposed approach also considers density of links in their discovered structures.

Figure 5.8 shows measures based on density of links. Mul-CPM results are not displayed as the method computes largest values of these measures as well as it focuses on the CPM principle [41]. The first report is that values of MCD are larger than those of TMC. As shown in that Figure 5.8b, and according to the table 5.4, ABACUS has the highest values of these measures on AUCS. This may be explained by the fact that, this dataset is the densest because it has simultaneously both high density and Adeg (see in Table 5.1, 0.114 and 20.33 respectively) and it detects a large number of small communities each. More precisely, it finds an average of 55 communities with an average size of 5 nodes as shown below, in Tables 5.2 and 5.3 respectively. Nevertheless, for these two metrics, UCAD finds larger values than gLouvain and Multimap on the AUCS as well as it is initially assigned and whose attributes are enriched by relevant dimensions. As far as TMC_{dens} is concerned, this measure is based on the triadic closure[4]. Thus we compare the values of TMC from different methods, as shown in Figure 5.8a. UCAD produces a better improvement of TMCdens for the majority of datasets, unlike Bankwiring whose nodes have a dissortative behavior as shown in Figure 5.7b.

Q2: How α affects dimension relevance of nodes ? The Q_{dim} quality function in Equation 5.6 shows how to use to weight both similarities. As α can affect both density-based similarity and attribute-based similarity, it is easy to notice that the higher the α is, the greater the influence of density-based similarity is on attribute-based similarity, and vice versa. To explain this variation in weighting factor, AUCS dataset is of interest because it has additional attributes, such as "Workgroup" and "Grade". Figure 5.9 shows the evaluation results stressing on α variation. The left vertical axis stands for the community number in AUCS, whereas the right vertical axis represents the MultiCommunity Density (MCD). From this figure, we can see that, when $\alpha > 0.5$, the MCD became the largest one, and the number of communities decreased. Moreover, no matter how the weighting factor α changed, MCD value seemed to remain below 0.5. We think that the main reason why this phenomenon happens is that, despite the more higher average degree of nodes (see $A_{deg} = 20.33$ in Table 5.1), the main behaviors of how nodes have the same relevant dimensions are mainly preserved in each community.

Comparison with other community discovery methods

Note that the following metric results on UCAD method were obtained through the Q_{dim} optimization (see *Modularity dimension* in Definition 5.3.4), under the weighting factor $\alpha = 0.5$.

[4] A triad stands for one of the fundamental features of real networks, namely *homophily*. Homophily explains the tendency of individuals to associate and bond with similar others.

Size and number of multi-communities. The Table 5.2 and Table 5.3 show the average on the multi-community number, means (μ) and the standard deviation (σ) of the size of the communities, respectively, identified by ABACUS, gLouvain, Mul-CPM, and Multimap from an experimentation on 10 executions, in order to reduce the bias due to their non-deterministic behavior except UCAD which is deterministic. On average, UCAD yields the largest communities on all datasets except DBLP which has the weak density (see Table 5.1). A weak density means that the network is sparse and nodes are solely involved in all dimensions. UCAD and Multimap communities slightly prevail in size with respect to the other methods and produces the smallest number of communities as shown in Table 5.2. Throughout our analysis we found that the higher the number of communities is computed, the more likely they are fairly sized.

On Biogrid, UCAD discovers 86 overlapping communities with an average size of 4710.5. These values correspond both to the minimum and to the maximum among of values for the number and size of communities respectively, compared to other datasets. This result confirms the reality behavior of molecules (nodes) that, they are not necessarily associated with the same biological mechanism (because of the sparsely feature of the network through its weak density as shown in Table 5.1), however they often interact together (large average size of communities). This observation reflects the principle of non-exclusiveness as expressed by UCAD (refer to the bottom of Section 5.2.1). Indeed, for the "synthetic_genetic" dimension containing the smallest number of links, namely four links, the related nodes can be strongly involved in the disease transmission process.

To justify it, the figure 5.10 shows that the nodes Hyx, L and $Mus312$ of this dimension are included in almost 60 percent of the obtained communities, unlike the other approaches that include them in a low community rate. This is enough to show that neglecting a molecule in a protein interaction system could be just as fatal. It is also deduced that the size of the dimension does not affect the involvement of its related nodes in community discovery.

On Florentine, all methods tend to identify quite small communities, which can proceed from the nature of the node relationship i.e. binary dimension values, either "Marriage"or "Business". The same behavior is observed on the other datasets. We realized through analysis based on these datasets that the higher the number of dimensions of the multigraph, the more likely communities are larger. UCAD exactly uncovers 7 multicommunities from AUCS, as well as presumed, standing for the number of workgroups of employees.

The evaluation of the number of communities obtained by UCAD was also done in accordance

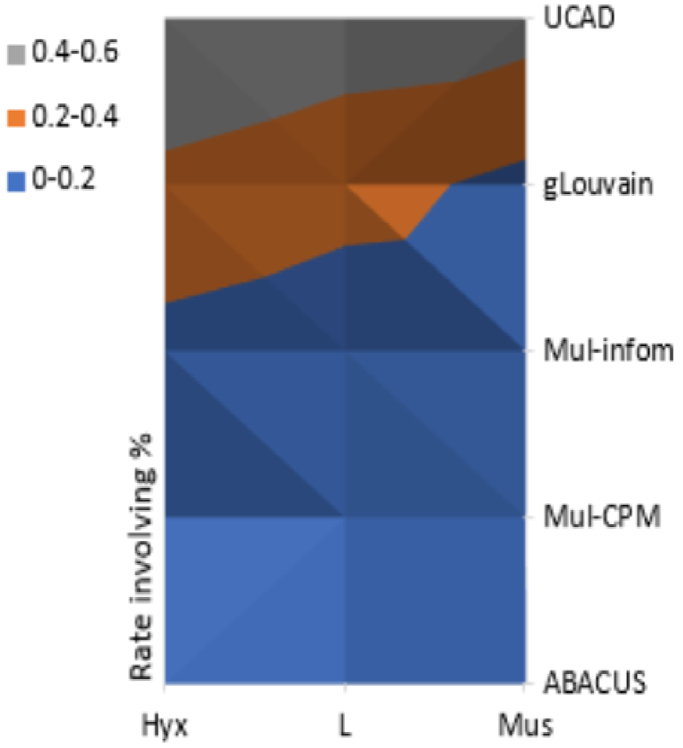


Figure 5.10: Involving rate of 3 nodes of Biogrid in the overall communities.

with the principle of Guimera [69] which shows that the optimal number of communities that maximizes modularity is closed to ($\sqrt{n}$). According to this principle, we compare UCAD to the gLouvain method because they are both approaches based on the optimization of modularity. The Table 5.2 shows that for Florentine, AUCS, Biogrid and DBLP datasets, UCAD results in bold are more closed to their corresponding value ($\sqrt{n}$) given in brackets in the “Dataset”column of this table. However, gLouvain produces the more closed community number in Bankwiring and Monastery datasets. In fact, Bankwiring presents an disassortative behavior of nodes (see Figure 5.7b); so it becomes difficult to compute relevant dimensions of multi-communities by UCAD. On the other hand, Monastery is densest so, since gLouvain focuses on density of links optimization through its modularity function, it performs well in this dataset.

Redundancy ρ evaluation. Since UCAD stresses on inter-dimensional correlation involving several dimensions simultaneously, we make use of this metric as it captures the phenomenon for which a set of nodes that constitute a community in a dimension tend to constitute a community also in other dimensions.

Figure 5.11 reports the results for compared algorithms, evaluated with the redundancy ρ. As we can see, UCAD slightly prevails on AUCS, Biogrid, DBLP and Bankwiring datasets. These values strengthen the idea that the resulting communities have nodes centered around a common interest, i.e. the same relevant dimensions. We also notice that since Florentine dataset only has two

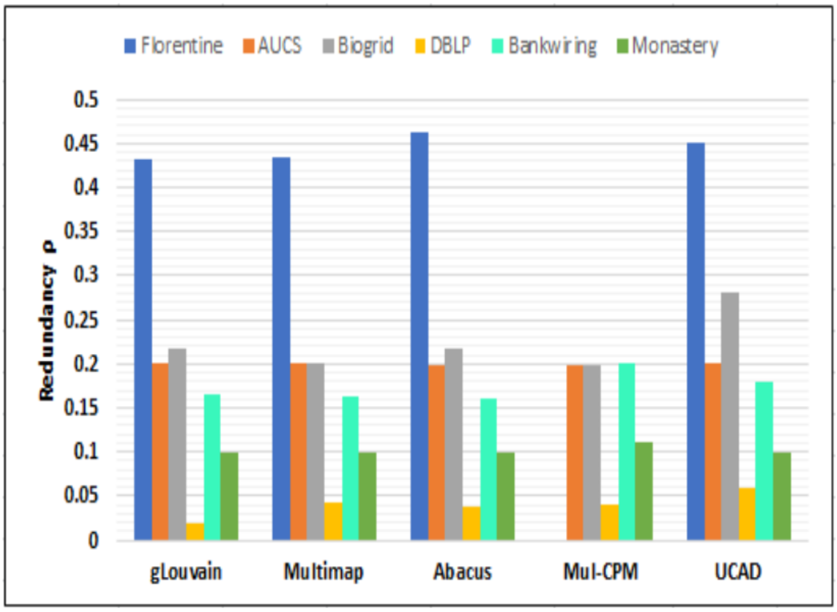


Figure 5.11: Redundancy values distribution

dimensions, all the methods compute larges metric values as they find communities containing both dimensions simultaneously, except Mul-CPM. Indeed, Mul-CPM computes for this measure a null value for Florentine because every community it finds has links in one dimension at a time. On the other hand, each method provides lower metric values for DBLP. It could be due to the fact that DBLP has the weakest $A_{dim} = 1.35$, as reported in Table 5.1. This reflects the fact that very few co-authors participated simultaneously to several conferences. Impact of relevant dimensions. To make the results on the importance of relevant dimensions via RMC_{dens} more meaningful, two datasets were used, in particular a small one and a large-scale one as shown in Tables 5.4 and 5.5 respectively. We notice that there are two dimensional subspaces used: DF_l and D_l.

The UCAD approach is compared to the MDLPA, Multimap and ABACUS approaches which consider the relevance of dimensions. Unlike MDLPA which focuses exclusively on the D_l subspace, ABACUS and Multimap merge both the subspaces D_l and DF_l. The Table 5.4 shows an improvement in the RMC_{dens} values on AUCS, over the subspace D_l, by comparing UCAD(D_l) whose value is 0.45 to UCAD(DF_l) whose value is 0.204. MDLPA(D_l) has a high RMC_{dens} value 0.483 compared to UCAD(D_l), due to its non-overlapping nature. Indeed, its D_l community subspaces are large, unlike UCAD, which is overlapping both on dimensions and on nodes. This leads to a reduction of the size of its D_l community subspace. The same behavior is observed in table 5.5 for DBLP dataset.

ABACUS and Multimap produce high values of RMC_{dens} compared to UCAD, because they take into account several overlaps in the discovered structures. However, the communities obtained by UCAD are more significant. Indeed UCAD considers the similarity of the attributes which are in this context the relevant dimensions. Since co-authors in DBLP could not only commonly classified according to their venues, they could also be classified based on their score in a specified journal to which the authors have submitted.

Table 5.4: Mesoscopic measure evaluation for AUCS

	MCD			RMC_{dens}		
	Low	$\mu \pm \sigma$	High	Low	$\mu \pm \sigma$	High
MDLPA (DF_l)	0.42	0.47 ± 0.04	0.55	0.16	1.69 ± 0.05	0.18
MDLPA (D_l)	0.53	0.6 ± 0.06	0.71	0.35	0.41 ± 0.07	0.483
ABACUS (DF_l)	0.69	0.71 ± 0.04	0.49	0.50	0.55 ± 0.23	0.59
Multimap (DF_l)	0.67	0.69 ± 0.01	0.13	0.207	0.219 ± 0.01	0.23
UCAD (DF_l)		0.38			0.204	
UCAD (D_l)		0.32			0.45	

Table 5.5: Mesoscopic measure evaluation for DBLP

	MCD			RMC_{dens}		
	Low	$\mu \pm \sigma$	High	Low	$\mu \pm \sigma$	High
MDLPA (DF_l)	0.0071	0.0075 ± 0.001	0.008	0.0005	0.00051 ± 0.0001	0.0006
MDLPA (D_l)	0.0067	0.007 ± 0.001	0.0079	0.00042	0.00045 ± 0.17	0.00047
ABACUS (DF_l)	0.08	0.009 ± 0.01	0.11	0.011	0.0115 ± 0.08	0.081
Multimap (DF_l)	0.087	0.088 ± 0.001	0.089	0.015	0.015 ± 0.072	0.016
UCAD (DF_l)		0.86			0.112	
UCAD (D_l)		0.072			0.090	

In fact, if an article is described by other attributes as key-words, affiliation, etc. then authors are grouped with respect to their higher matching of attributes. These methods also prevail in this metric because DF_l considers all dimensions of the community. As ABACUS uses a partition integration approach and itemsets with high support values, it generates communities with better RMCdens for networks which are relatively smaller in terms of nodes (AUCS, Florentine, Bankwiring and Monastery), while low RMC_{dens} for sparser and larger networks (DBLP and Biogrid).

From a structural point of view, we perform another evaluation of the dimension relevance. The Table 5.6 in which lower values are in bold, describes the RMC_{dens} measure on an assessed DS_l subspace. It corresponds to dimensions whose score is upper than a threshold ξ. In other words, D_l is replaced by DS_l in Formula 5.13. This threshold stands for the average of dimensions. According to the results of the Mul-CPM method on Florentine ($RMC_{dens} = 1.0$), the nodes of each community share only one relevant dimension, with respect to the two existing dimensions one. On the other hand, results of gLouvain on the same dataset ($RMC_{dens} = 0.5$) shows that it detects communities in which nodes of the same community simultaneously consider both dimensions to be relevant. It can be concluded via these results that the more the nodes of the same community have relevant dimensions in common, the lower the RMC_{dens} value. The lower values in Table 5.6, of this measure, obtained by UCAD, reflects the fact that the communities generated are *densely relevant.*

Table 5.6: Relevance MultiCommunity Density (RMC_{dens}) based on DS_l

Dataset	gLouvain	ABACUS	Multimap	Mul-CPM	UCAD
Florentine	0.5	0.75	0.5	1	0.25
AUCS	0.2	0.59	0.23	0.58	0.15
Biogrid	0.048	0.28	0.16	0.115	0.03
DBLP	0.0025	0.081	0.0021	0.0075	0.0008
Bankwiring	0.167	0.41	0.167	0.69	0.135
Monastery	0.1	0.57	0.1	0.79	0.1

Table 5.7: Multilayer modularity evaluation

Dataset	gLouvain	ABACUS	Multimap	Mul-CPM	UCAD
Florentine	0.015	0.175	0.215	0.227	0.255
AUCS	0.02	0.071	0.323	0.218	0.375
Biogrid	0.048	0.208	0.156	0.185	0.23
DBLP	0.031	0.065	0.0071	0.301	0.28
Bankwiring	0.17	0.141	0.177	0.086	0.205
Monastery	0.11	0.18	0.042	0.247	0.114

Global measure evaluations

Given the variation in the structural information of each dataset and the processing principle of each approach, the unique use of local metrics (ρ, MCD, TMC_{dens}, RMC_{dens}) is not sufficient to establish, objectively and fairly, the merits of each approach considered in the comparison. In fact, a cover where communities consist of a single pair of nodes will have high scores for some of these metrics. In order to offset this impact, two global metrics that take into account inter-community interactions should be considered to evaluate the results of community detection algorithms. These include multislice modularity of Mucha [113] and the proposed modularity dimension formulated in Equation 5.6.

Multislice modularity evaluation. For experiments, we fix the parameters of the multislice modularity stated in Section 3.2 in Chapter 3 such as $\omega = 1$ and $\gamma_s = 1$. We could see that the values of the multilayer modularity of Mucha [113] are improved by UCAD as shown in Table 5.7. These improved values of the multilayer modularity for most datasets reflect the additional value of our method, i.e. the consideration of stability in the determination of the relevant dimensions. Almost for all datasets UCAD has high modularity values, except on DBLP and Monastery which have extreme values of A_{deg} and ONC, as presented in dataset description in Table 5.1. Indeed, the statistics presented in that table show that Monastery is dense as it has higher values of $A_{deg} = 38.8$ and $OCN = 0.78$, indicating that an algorithm that integrates multiple attribute similarity relations can obtain weak results for datasets with concentrated nodes. On the other hand, DBLP is sparse with low $A_{deg} = 3.8$ and $OCN = 0.018$, expressing the fact that an algorithm density-based similarity can also obtain weak results for datasets with sparse nodes. We believe that these low values of modularity do not reflect poor partitioning, since some authors studied the limits of modularity [60]. According to them, a large value for the modularity maximum does not necessarily mean that a graph has community structure.

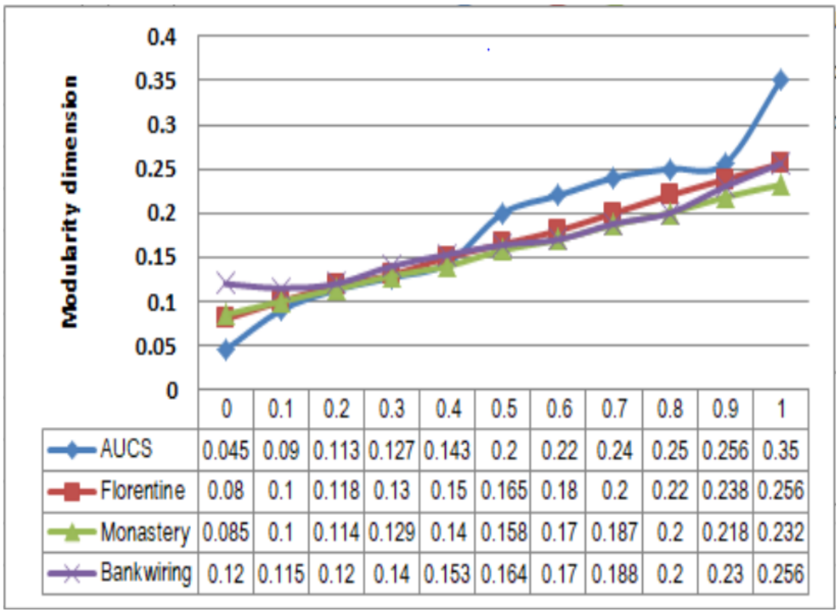
Modularity dimension evaluation. The Modularity dimension Q_{dim} (See Definition 5.3.4) is a quality function optimized by UCAD only. Indeed, UCAD describes each node by a subspace $RD(u)$ (see Section 5.2.2). $RD(u)$ allows to deduce D_l subspace of relevant dimensions of the multi-community C_l through a matching criterion, and it is used in Q_{dim}. On the other hand, other approaches do not describe the nodes by the relevant dimensions or are not preprocessed to extract them. If considering them, they operate through the clustering process. Likewise, this paragraph focuses on the evaluation of the values of Q_{dim}, also with regard to the involved weighting factor α.

Unlike the other methods which have many parameters, UCAD has only one, namely the α weighting factor. The natural question is how to choose α. Note that the results are quite stable with respect to α. Its value is a priori difficult to choose, with no domain knowledge. But we observed a behavior of the metric in two cases of our study :

1. $\alpha \in [0; 0.5[$: Q_{dim} is weak.
2. $\alpha \in [0.5; 1[$: Q_{dim} is higher but under 0.5.

In the first case, the information on the relevance of dimensions is more prevalent on structural information; in other words, there is more attention to the similarity of dimensions than to the density of links. Sim_D is the inverse of the Jaccard index. The smaller this index, the more Sim_D increases, and vice versa. This index is small when the nodes have a slightly similar neighborhood, therefore Sim_D grows when there is dissimilarity between the relevant dimensions of the nodes. Likewise, it decreases for similarity in relevant dimensions. So, small values of Q_{dim} denote an optimal partitioning in this first case. In the second case, the structural information prevails on relevance dimension similarity. This is like considering Newman's property on classical modularity, which assumes that for high modularity values, we tend towards a better partitioning. For increasing α values (case 2), the Q optimum is achieved ($\alpha = 1$ then $Q = 1$) when all nodes are connected two-by-two and belong to the same community. Thus, high values of Q_{dim} denote a better partitioning of the network.

Figure 5.12 shows that UCAD discovers optimal covers since values of modularity dimension are weak in the first case, and high, although below 0.5, in the second case. To summarize, there is a balance between Q and Sim_D for all values of α that leads to better values of Global modularity as shown in Table 5.7.



	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
◆ AUCS	0.045	0.09	0.113	0.127	0.143	0.2	0.22	0.24	0.25	0.256	0.35
■ Florentine	0.08	0.1	0.118	0.13	0.15	0.165	0.18	0.2	0.22	0.238	0.256
▲ Monastery	0.085	0.1	0.114	0.129	0.14	0.158	0.17	0.187	0.2	0.218	0.232
× Bankwiring	0.12	0.115	0.12	0.14	0.153	0.164	0.17	0.188	0.2	0.23	0.256

Figure 5.12: Q_{dim} validation through α study

5.6 Conclusion

In this chapter, we addressed the problem of community discovery in multidimensional networks via the proposition of an approach called UCAD. Specifically we gave a new definition of this problem by integrating attributes on the process of enrichment of properties of the multigraph by relevant dimensions, in order to discover more significant communities of interest. To this end, we provided a measure called *stability*, aimed at computing relevant dimensions for each node, and proposed a solution that takes them into account.

Afterwards, we proceeded by a weighting-based aggregation scheme to deduce a monodimensional attributed and weighted graph. Then we adapted the Louvain's optimization based algorithm to aggregated network, to make it deterministic, stressing on overlapping discovery of multi-communities. We also provided a quality function aimed to characterize the validity of the found multi-communities. This function stands for the linear combination of both structural information and relevant dimensions of nodes in the multigraph. Thus, UCAD seems to be a more appropriate solution to the problem of community discovery in attributed multidimensional networks. The similarity between entities and relational information are of great interest for networks whose nodes are described by attributes and relevant dimensions. Our outcomes are more meaningful, since they produce better results of the metrics when focusing on relevance of dimensions. The proposed solution provides a basis for future research on this direction, in particular the consideration of directed multigraphs.

CHAPTER 6

General conclusion

The observation that interactions between individuals manipulating large amounts of data can be modeled by complex systems. The analysis of these complex networks has produced enormous challenges for researchers. In this thesis we focused on the communities of interest detection in complex networks. Our motivation is that an individual's interest is the leitmotiv of its involvement in the society. Since the methods for detecting communities of interest are based on semantics that require a prior knowledge of the network, then topology could be used to identify these communities of interest.

In this thesis we focused on three main objectives. The first objective was to develop a method of community of interest in directed networks. The second objective was to develop another method dealing with networks whose nodes are described by attributes. The last objective was to investigate the detection of communities of interest in multidimensional networks.

6.1 Summary of the thesis

We summarize the works achieved in this thesis by grouping them into major themes.

Literature review

In Chapter 2, we have presented the generalities on complex networks. More specifically, we stated basic concepts handled by complex networks and their properties and tools for their manipulation.

In Chapter 3, we have presented a review community detection methods on different type of networks of the context, namely directed, attributed and multidimensional, in addition to their applications areas.

Design of heuristics

In Chapter 4, we have proposed two new methods of community of interest detection in directed networks. Unlike the first heuristic, the second one deals with attributes. They perform better than state-of-the-art methods with respect to the density of triads and allow to group nodes according to their interest based both on the structural equivalence through their homogeneous topology and similar attributes.

In Chapter 5, we have proposed a method of community of interest discovery in multidimensional networks. This new heuristic performs better in term of the level of activity involving nodes of the same community. Thus, identified communities are densely relevant since they group nodes of the same interest with regard to their relevant dimension

6.2 Future directions

We propose several ideas and perspectives built on the work presented in this thesis.

General perspectives

In this thesis we use attributes on nodes in directed networks. Edges are sometimes also enriched of attributes. Thus, it would be interesting to handle them in the process of community detection.

In this study, we focused on some properties of complex networks, disregarding the dynamical evolution property. Another perspective concerns the tracking of communities of interest over time, and therefore deal with dynamic aspect, as an individual may change his or her choices or preferences after an unknown period of time.

Also, we could be interested in graph embedding using deep neural networks to model community detection. This remains an open question that deserves to be studied.

Perspectives in heuristics evaluation

This perspective concerns a thorough evaluation of the proposed heuristic for multicommunity of interest detection based on topological features of the network. Instead of using the state-of-the-art complex networks to enhance the implemented communities of interest, we plan to set up a platform for the interconnection of agricultural sector agents. It will aim to build a multidimensional network of users in the agricultural sector. Thereafter, the resulting social network of agriculturists will be applied to the methods in order to discover communities of interest. This construction is achieved by putting in communication individuals or companies who would like to exchange their knowledge and/or technical expertise in their agricultural activities. We hope that this would contribute to the development of the agricultural sector through exchanges of experience and thus increase yields and the economy of our country.

Bibliography

[1] L. A. Adamic and N. Glance. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery*, pages 36–43, 2005.

[2] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *nature*, 466(7307):761–764, 2010.

[3] R. Albert and A.-L. Barabási. Statistical mechanics of complex networks. *Reviews of modern physics*, 74(1):47, 2002.

[4] R. Albert, H. Jeong, and A.-L. Barabási. Diameter of the world-wide web. *nature*, 401(6749):130–131, 1999.

[5] A. Amelio and C. Pizzuti. Community detection in multidimensional networks. In *2014 IEEE 26th International Conference on Tools with Artificial Intelligence*, pages 352–359. IEEE, 2014.

[6] A. Armstrong and J. Hagel. The real value of online communities. *Knowledge and communities*, 74(3):85–95, 2000.

[7] M. Azaouzi, D. Rhouma, and L. B. Romdhane. Community detection in large-scale social networks: state-of-the-art and future directions. *Social Network Analysis and Mining*, 9(1):23, 2019.

[8] A.-L. Barabási. Scale-free networks: a decade and beyond. *science*, 325(5939):412–413, 2009.

[9] A.-L. Barabási and R. Albert. Emergence of scaling in random networks. *science*, 286(5439):509–512, 1999.

[10] M. J. Barber and J. W. Clark. Detecting network communities by propagating labels under constraints. *Physical Review E*, 80(2):026129, 2009.

[11] F. Battiston, V. Nicosia, and V. Latora. Structural measures for multiplex networks. *Physical Review E*, 89(3):032804, 2014.

[12] A. Bavelas. Communication patterns in task-oriented groups. *The journal of the acoustical society of America*, 22(6):725–730, 1950.

[13] M. Berlingerio, M. Coscia, and F. Giannotti. Finding and characterizing communities in multidimensional networks. In *2011 International Conference on advances in social networks analysis and mining*, pages 490–494. IEEE, 2011.

[14] M. Berlingerio, M. Coscia, F. Giannotti, A. Monreale, and D. Pedreschi. Multidimensional networks: foundations of structural analysis. *World Wide Web*, 16(5-6):567–593, 2013.

[15] M. Berlingerio, F. Pinelli, and F. Calabrese. Abacus: frequent pattern mining-based community discovery in multidimensional networks. *Data Mining and Knowledge Discovery*, 27(3):294–320, 2013.

[16] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[17] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang. Complex networks: Structure and dynamics. *Physics reports*, 424(4-5):175–308, 2006.

[18] B. Boden, S. Günnemann, H. Hoffmann, and T. Seidl. Mining coherent subgraphs in multi-layer graphs with edge labels. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1258–1266, 2012.

[19] I. M. Bomze, M. Budinich, P. M. Pardalos, and M. Pelillo. The maximum clique problem. In *Handbook of combinatorial optimization*, pages 1–74. Springer, 1999.

[20] P. Bonacich. Power and centrality: A family of measures. *American journal of sociology*, 92(5):1170–1182, 1987.

[21] J. A. Bondy, U. S. R. Murty, et al. *Graph theory with applications*, volume 290. Macmillan London, 1976.

[22] C. Bothorel, J. D. Cruz, M. Magnani, and B. Micenkova. Clustering attributed graphs: models, measures and methods. *arXiv preprint arXiv:1501.01676*, 2015.

[23] O. Boutemine and M. Bouguessa. Mining community structures in multidimensional networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 11(4):1–36, 2017.

[24] U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner. On modularity clustering. *IEEE transactions on knowledge and data engineering*, 20(2):172–188, 2007.

[25] R. L. Breiger, S. A. Boorman, and P. Arabie. An algorithm for clustering relational data with applications to social network analysis and comparison with multidimensional scaling. *Journal of mathematical psychology*, 12(3):328–383, 1975.

[26] R. L. Breiger and P. E. Pattison. Cumulated social roles: The duality of persons and their algebras. *Social networks*, 8(3):215–256, 1986.

[27] D. Cai, Z. Shao, X. He, X. Yan, and J. Han. Community mining from multi-relational networks. In *European Conference on Principles of Data Mining and Knowledge Discovery*, pages 445–452. Springer, 2005.

[28] R. Campigotto, J.-L. Guillaume, and M. Seifi. The power of consensus: Random graphs have no communities. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 272–276, 2013.

[29] V. Carchiolo, A. Longheu, M. Malgeri, and G. Mangioni. Communities unfolding in multislice networks. In *Complex Networks*, pages 187–195. Springer, 2011.

[30] T. Chakraborty, S. Srinivasan, N. Ganguly, A. Mukherjee, and S. Bhowmick. On the permanence of vertices in network communities. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1396–1405, 2014.

[31] J. Chen, O. Zaïane, and R. Goebel. Local community identification in social networks. In *2009 International Conference on Advances in Social Network Analysis and Mining*, pages 237–242. IEEE, 2009.

[32] N. Chouchani and M. Abed. Online social network analysis: detection of communities of interest. *Journal of Intelligent Information Systems*, 54(1):5–21, 2020.

[33] A. Clauset, M. E. Newman, and C. Moore. Finding community structure in very large networks. *Physical review E*, 70(6):066111, 2004.

[34] G. P. Clemente and R. Grassi. Directed clustering in weighted networks: A new perspective. *Chaos, Solitons & Fractals*, 107:26–38, 2018.

[35] D. Combe, C. Largeron, M. Géry, and E. Egyed-Zsigmond. I-louvain: An attributed graph clustering method. In *International Symposium on Intelligent Data Analysis*, pages 181–192. Springer, 2015.

[36] M. Coscia, F. Giannotti, and D. Pedreschi. A classification for community discovery methods in complex networks. *Statistical Analysis and Data Mining: The ASA Data Science Journal*, 4(5):512–546, 2011.

[37] T. Dang and E. Viennet. Community detection based on structural and attribute similarities. In *International conference on digital society (icds)*, pages 7–12, 2012.

[38] M. De Domenico, A. Lancichinetti, A. Arenas, and M. Rosvall. Identifying modular flows on multilayer networks reveals highly overlapping organization in interconnected systems. *Physical Review X*, 5(1):011027, 2015.

[39] M. De Domenico, M. A. Porter, and A. Arenas. Muxviz: a tool for multilayer analysis and visualization of networks. *Journal of Complex Networks*, 3(2):159–176, 2015.

[40] P. De Meo, E. Ferrara, G. Fiumara, and A. Provetti. Generalized louvain method for community detection in large networks. In *2011 11th international conference on intelligent systems design and applications*, pages 88–93. IEEE, 2011.

[41] I. Derényi, G. Palla, and T. Vicsek. Clique percolation in random networks. *Physical review letters*, 94(16):160202, 2005.

[42] M. E. Dickison, M. Magnani, and L. Rossi. *Multilayer social networks.* Cambridge University Press, 2016.

[43] Y. Ding. Community detection: Topological vs. topical. *Journal of Informetrics*, 5(4):498–514, 2011.

[44] F. G. Domgue and N. Tsopze. Analyse des réseaux sociaux: Communautés et rôles dans les réseaux sociaux. In *Proceedings of the 12th Colloque africain sur la recherche en informatique et mathématiques appliquées, CARI'14*, pages 157–164, 2014.

[45] F. G. Domgue, N. Tsopze, and A. Ahouandjinou. Nouvelle approche de clustering par kernel-pattern via la densité en triades. In *CORIA*, pages 327–342, 2017.

[46] F. G. Domgue, N. Tsopze, and R. Ndoundam. Community structure extraction in directed network using triads. *International Journal of General Systems*, pages 1–24, 2020.

[47] F. G. Domgue, N. Tsopzé, and N. René. Novel method to find directed community structures based on triads cardinality. In *Proceedings of the 13th Colloque africain sur la recherche en informatique et mathématiques appliquées (CARI'16)*, pages 8–15, 2016.

[48] F. G. Domgue, N. Tsopze, and N. René. Finding directed community structures using triads. In *Conference de Recherche en Informatique, CRI'17*, pages 327–342, 2017.

[49] F. G. Domgue, N. Tsopze, and N. René. Towards a hybrid model of semantic communities detection. In *Proceedings of the 14th Colloque africain sur la recherche en informatique et mathematiques appliquées, CARI'18*, pages 257–264, 2018.

[50] F. G. Domgue, N. Tsopze, and N. René. Multidimensional networks: A novel node centrality metric based on common neighborhood. In *Proceedings of the 15th Colloque africain sur la recherche en informatique et mathématiques appliquées, CARI'20*, pages 250–258, 2020.

[51] W. E. Donath and A. J. Hoffman. Lower bounds for the partitioning of graphs. In *Selected Papers Of Alan J Hoffman: With Commentary*, pages 437–442. World Scientific, 2003.

[52] L. Donetti and M. A. Munoz. Detecting network communities: a new systematic and efficient algorithm. *Journal of Statistical Mechanics: Theory and Experiment*, 2004(10):P10012, 2004.

[53] X. Dong, P. Frossard, P. Vandergheynst, and N. Nefedov. Clustering on multi-layer graphs via subspace analysis on grassmann manifolds. *IEEE Transactions on signal processing*, 62(4):905–918, 2013.

[54] N. Durak, A. Pinar, T. G. Kolda, and C. Seshadhri. Degree relations of triangles in real-world networks and graph models. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1712–1716, 2012.

[55] M. G. Everett and S. P. Borgatti. Regular equivalence: General theory. *Journal of mathematical sociology*, 19(1):29–52, 1994.

[56] I. Falih, N. Grozavu, R. Kanawati, and Y. Bennani. Anca: Attributed network clustering algorithm. In *International Conference on Complex Networks and their Applications*, pages 241–252. Springer, 2017.

[57] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. *ACM SIGCOMM computer communication review*, 29(4):251–262, 1999.

[58] E. Ferrara, P. De Meo, S. Catanese, and G. Fiumara. Detecting criminal organizations in mobile phone networks. *Expert Systems with Applications*, 41(13):5733–5750, 2014.

[59] G. Fischer. External and shareable artifacts as opportunities for social creativity in communities of interest. In *University of Sydney*. Citeseer, 2001.

[60] S. Fortunato. Community detection in graphs. *Physics reports*, 486(3-5):75–174, 2010.

[61] S. Fortunato and M. Barthelemy. Resolution limit in community detection. *Proceedings of the national academy of sciences*, 104(1):36–41, 2007.

[62] S. Fortunato and D. Hric. Community detection in networks: A user guide. *Physics reports*, 659:1–44, 2016.

[63] J.-C. Fournier. *Théorie des graphes et applications: Avec exercices et problèmes*. Lavoisier, 2011.

[64] L. C. Freeman. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41, 1977.

[65] L. C. Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1978.

[66] E. Garfield. Citation analysis as a tool in journal evaluation. *Science*, 178(4060):471–479, 1972.

[67] L. Getoor. Link-based classification. In *Advanced methods for knowledge discovery from complex data*, pages 189–207. Springer, 2005.

[68] M. Girvan and M. E. Newman. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12):7821–7826, 2002.

[69] R. Guimera, M. Sales-Pardo, and L. A. N. Amaral. Modularity from fluctuations in random graphs and complex networks. *Physical Review E*, 70(2):025101, 2004.

[70] R. Guimerà, M. Sales-Pardo, and L. A. N. Amaral. Module identification in bipartite and directed networks. *Physical Review E*, 76(3):036102, 2007.

[71] K. Gunce, V. Labatut, and H. Cherifi. Relation entre transitivité et structure de communauté dans les réseaux complexes. 2011.

[72] J. R. Gusfield. *Community: A critical response*. Harper & Row New York, 1975.

[73] M. Hamann, E. Röhrs, and D. Wagner. Local community detection based on small cliques. *Algorithms*, 10(3):90, 2017.

[74] M. Hmimida and R. Kanawati. Community detection in multiplex networks: A seed-centric approach. *Networks & Heterogeneous Media*, 10(1):71, 2015.

[75] C. B. N. Kaledje. *Detection and dynamic of local communities in large social networks*. PhD thesis, 2014.

[76] R. Kanawati. Licod: Leaders identification for community detection in complex networks. In *2011 IEEE Third International Conference on Privacy, Security, Risk and Trust and 2011 IEEE Third International Conference on Social Computing*, pages 577–582. IEEE, 2011.

[77] R. Kanawati. Détection de communautés dans les grands graphes d'interactions (multiplexes): état de l'art. 2013.

[78] R. Kanawati. Seed-centric approaches for community detection in complex networks. In *International Conference on Social Computing and Social Media*, pages 197–208. Springer, 2014.

[79] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[80] I. Keller and E. Viennet. A characterization of the modular structure of complex networks based on consensual communities. In *2012 Eighth International Conference on Signal Image Technology and Internet Based Systems*, pages 717–724. IEEE, 2012.

[81] B. W. Kernighan and S. Lin. An efficient heuristic procedure for partitioning graphs. *The Bell system technical journal*, 49(2):291–307, 1970.

[82] J. Kim and J.-G. Lee. Community detection in multi-layer graphs: A survey. *ACM SIGMOD Record*, 44(3):37–48, 2015.

[83] J. Kim, J.-G. Lee, and S. Lim. Differential flattening: A novel framework for community detection in multi-layer graphs. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 8(2):1–23, 2016.

[84] Y. Kim, S.-W. Son, and H. Jeong. Finding communities in directed networks. *Physical Review E*, 81(1):016103, 2010.

[85] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter. Multilayer networks. *Journal of complex networks*, 2(3):203–271, 2014.

[86] C. Klymko, D. Gleich, and T. G. Kolda. Using triangles to improve community detection in directed networks. *arXiv preprint arXiv:1404.5874*, 2014.

[87] D. Knoke. *Political networks: the structural perspective*, volume 4. Cambridge University Press, 1994.

[88] G. Krings and V. D. Blondel. An upper bound on community size in scalable community detection. *arXiv preprint arXiv:1103.5569*, 2011.

[89] Z. Kuncheva and G. Montana. Community detection in multiplex networks using locally adaptive random walks. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, pages 1308–1315, 2015.

[90] S. Laan, M. Marx, and R. J. Mokken. Close communities in social networks: boroughs and 2-clubs. *Social Network Analysis and Mining*, 6(1):20, 2016.

[91] A. Lancichinetti and S. Fortunato. Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117, 2009.

[92] A. Lancichinetti and S. Fortunato. Limits of modularity maximization in community detection. *Physical review E*, 84(6):066122, 2011.

[93] A. Lancichinetti and S. Fortunato. Consensus clustering in complex networks. *Scientific reports*, 2:336, 2012.

[94] A. Lancichinetti, S. Fortunato, and J. Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New journal of physics*, 11(3):033015, 2009.

[95] A. Lancichinetti, M. Kivelä, J. Saramäki, and S. Fortunato. Characterizing the community structure of complex networks. *PloS one*, 5(8):e11976, 2010.

[96] E. A. Leicht and M. E. Newman. Community structure in directed networks. *Physical review letters*, 100(11):118703, 2008.

[97] J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: densification laws, shrinking diameters and possible explanations. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 177–187, 2005.

[98] H. Li, Z. Nie, W.-C. Lee, L. Giles, and J.-R. Wen. Scalable community discovery on textual data with relations. In *Proceedings of the 17th ACM conference on Information and knowledge management*, pages 1203–1212, 2008.

[99] X. Li. Directed lpa: Propagating labels in directed networks. *Physics Letters A*, 383(8):732–737, 2019.

[100] X. Li, G. Xu, L. Jiao, Y. Zhou, and W. Yu. Multi-layer network community detection model based on attributes and social interaction intensity. *Computers & Electrical Engineering*, 77:300–313, 2019.

[101] Z. Li, J. Liu, and K. Wu. A multiobjective evolutionary algorithm based on structural and attribute similarities for community detection in attributed networks. *IEEE transactions on cybernetics*, 48(7):1963–1976, 2017.

[102] D. Liben-Nowell and J. Kleinberg. The link-prediction problem for social networks. *Journal of the American society for information science and technology*, 58(7):1019–1031, 2007.

[103] S. Lin, Q. Hu, G. Wang, and S. Y. Philip. Understanding community effects on information diffusion. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 82–95. Springer, 2015.

[104] W. Liu, T. Suzumura, H. Ji, and G. Hu. Finding overlapping communities in multilayer networks. *PloS one*, 13(4):e0188747, 2018.

[105] F. Lorrain and H. C. White. Structural equivalence of individuals in social networks. *The Journal of mathematical sociology*, 1(1):49–80, 1971.

[106] Z. Lu, Y. Wen, and G. Cao. Community detection in weighted networks: Algorithms and applications. In *2013 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 179–184. IEEE, 2013.

[107] J. MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

[108] M. Magnani, B. Micenkova, and L. Rossi. Combinatorial analysis of multiple networks. *arXiv preprint arXiv:1303.4986*, 2013.

[109] M. Magnani and L. Rossi. The ml-model for multi-layer social networks. In *2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 5–12. IEEE, 2011.

[110] F. D. Malliaros and M. Vazirgiannis. Clustering and community detection in directed networks: A survey. *Physics reports*, 533(4):95–142, 2013.

[111] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual review of sociology*, 27(1):415–444, 2001.

[112] M. Meilă and W. Pentney. Clustering by weighted cuts in directed graphs. In *Proceedings of the 2007 SIAM international conference on data mining*, pages 135–144. SIAM, 2007.

[113] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela. Community structure in time-dependent, multiscale, and multiplex networks. *science*, 328(5980):876–878, 2010.

[114] M. E. Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[115] M. E. Newman. Coauthorship networks and patterns of scientific collaboration. *Proceedings of the national academy of sciences*, 101(suppl 1):5200–5205, 2004.

[116] M. E. Newman. Detecting community structure in networks. *The European physical journal B*, 38(2):321–330, 2004.

[117] M. E. Newman. A measure of betweenness centrality based on random walks. *Social networks*, 27(1):39–54, 2005.

[118] M. E. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical review E*, 69(2):026113, 2004.

[119] B. Ngonmang, M. Tchuente, and E. Viennet. Local community identification in social networks. *Parallel Processing Letters*, 22(01):1240004, 2012.

[120] V. Nicosia and V. Latora. Measuring and modeling correlations in multiplex networks. *Physical Review E*, 92(3):032805, 2015.

[121] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri. Extending the definition of modularity to directed graphs with overlapping communities. *Journal of Statistical Mechanics: Theory and Experiment*, 2009(03):P03024, 2009.

[122] J. F. Padgett and P. D. McLean. Organizational invention and elite transformation: The birth of partnership systems in renaissance florence. *American journal of Sociology*, 111(5):1463–1568, 2006.

[123] S. Papadopoulos, Y. Kompatsiaris, A. Vakali, and P. Spyridonos. Community detection in social media. *Data Mining and Knowledge Discovery*, 24(3):515–554, 2012.

[124] P. Pons and M. Latapy. Computing communities in large networks using random walks. In *International symposium on computer and information sciences*, pages 284–293. Springer, 2005.

[125] A. Prat-Pérez, D. Dominguez-Sal, J. M. Brunat, and J.-L. Larriba-Pey. Shaping communities out of triangles. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1677–1681, 2012.

[126] F. Radicchi, C. Castellano, F. Cecconi, V. Loreto, and D. Parisi. Defining and identifying communities in networks. *Proceedings of the national academy of sciences*, 101(9):2658–2663, 2004.

[127] U. N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical review E*, 76(3):036106, 2007.

[128] P. K. Reddy, M. Kitsuregawa, P. Sreekanth, and S. S. Rao. A graph based approach to extract a neighborhood customer community for collaborative filtering. In *International Workshop on Databases in Networked Information Systems*, pages 188–200. Springer, 2002.

[129] M. Rosvall and C. T. Bergstrom. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences*, 105(4):1118–1123, 2008.

[130] Y. Ruan, D. Fuhry, and S. Parthasarathy. Efficient community detection in large networks using content and links. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1089–1098, 2013.

[131] G. Salton, A. Wong, and C.-S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[132] V. Satuluri and S. Parthasarathy. Symmetrizations for clustering directed graphs. In *Proceedings of the 14th International Conference on Extending Database Technology*, pages 343–354, 2011.

[133] M. Seifi. *Cœurs stables de communautés dans les graphes de terrain.* PhD thesis, Paris 6, 2012.

[134] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000.

[135] S. Souravlas, A. Sifaleras, and S. Katsavounis. A novel, interdisciplinary, approach for community detection based on remote file requests. *IEEE Access*, 6:68415–68428, 2018.

[136] C. Stark, B.-J. Breitkreutz, T. Reguly, L. Boucher, A. Breitkreutz, and M. Tyers. Biogrid: a general repository for interaction datasets. *Nucleic acids research*, 34(suppl_1):D535–D539, 2006.

[137] K. Steinhaeuser and N. V. Chawla. Identifying and evaluating community structure in complex networks. *Pattern Recognition Letters*, 31(5):413–421, 2010.

[138] Y. Sun and J. Han. Mining heterogeneous information networks: principles and methodologies. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 3(2):1–159, 2012.

[139] Y. Sun, Y. Yu, and J. Han. Ranking-based clustering of heterogeneous information networks with star network schema. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 797–806, 2009.

[140] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to data mining.* Pearson Education India, 2016.

[141] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su. Arnetminer: extraction and mining of academic social networks. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 990–998, 2008.

[142] L. Tang and H. Liu. Community detection and mining in social media. *Synthesis lectures on data mining and knowledge discovery*, 2(1):1–137, 2010.

[143] L. Tang, X. Wang, and H. Liu. Uncoverning groups via heterogeneous interaction analysis. In *2009 Ninth IEEE International Conference on Data Mining*, pages 503–512. IEEE, 2009.

[144] L. Tang, X. Wang, and H. Liu. Community detection via heterogeneous interaction analysis. *Data mining and knowledge discovery*, 25(1):1–33, 2012.

[145] F. W. Taylor. *Scientific management.* Routledge, 2004.

[146] N. A. Tehrani and M. Magnani. Partial and overlapping community detection in multiplex social networks. In *International Conference on Social Informatics*, pages 15–28. Springer, 2018.

[147] J. Travers and S. Milgram. An experimental study of the small world problem. In *Social Networks*, pages 179–197. Elsevier, 1977.

[148] C. E. Tsourakakis. Fast counting of triangles in large real networks without counting: Algorithms and laws. In *2008 Eighth IEEE International Conference on Data Mining*, pages 608–617. IEEE, 2008.

[149] T. Vicsek. Complexity: The bigger picture. *Nature*, 418(6894):131–131, 2002.

[150] L. Wang, T. Lou, J. Tang, and J. E. Hopcroft. Detecting community kernels in large social networks. In *2011 IEEE 11th International Conference on Data Mining*, pages 784–793. IEEE, 2011.

[151] S. Wasserman, K. Faust, et al. Social network analysis: Methods and applications. 1994.

[152] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world'networks. *nature*, 393(6684):440–442, 1998.

[153] X. Wu and V. Kumar. *The top ten algorithms in data mining*. CRC press, 2009.

[154] J. Yang and J. Leskovec. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems*, 42(1):181–213, 2015.

[155] J. Yang, J. McAuley, and J. Leskovec. Community detection in networks with node attributes. In *2013 IEEE 13th International Conference on Data Mining*, pages 1151–1156. IEEE, 2013.

[156] T. Yang, Y. Chi, S. Zhu, Y. Gong, and R. Jin. Directed network community detection: A popularity and productivity link model. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 742–753. SIAM, 2010.

[157] W. Zhan, Z. Zhang, J. Guan, and S. Zhou. Evolutionary method for finding communities in bipartite networks. *Physical Review E*, 83(6):066120, 2011.

[158] Y. Zhang, L. Gao, and C. Wang. Multinet: multiple virtual networks for a reliable live streaming service. In *GLOBECOM 2009-2009 IEEE Global Telecommunications Conference*, pages 1–6. IEEE, 2009.

[159] D. Zhou, T. Hofmann, and B. Schölkopf. Semi-supervised learning on directed graphs. *Advances in neural information processing systems*, 17:1633–1640, 2004.

[160] Y. Zhou, H. Cheng, and J. X. Yu. Graph clustering based on structural/attribute similarities. *Proceedings of the VLDB Endowment*, 2(1):718–729, 2009.

[161] G. Zhu and K. Li. A unified model for community detection of multiplex networks. In *International Conference on Web Information Systems Engineering*, pages 31–46. Springer, 2014.

APPENDICES

A Extended Abstract

A.1 Context and Goal

Context. With the proliferation of social media and mobile applications, users are constantly interacting, sharing documents, images/videos and messages, etc. These interactions can be modeled by a complex system. A *complex system* is a system possessing some emergent properties, due to the interactions of its constituting objects [17, 114]. It can be encountered in many different domains such as biology, physics, computer science, sociology, etc. [3, 114]

Network modeling consists in representing such systems through *complex networks* [3, 17, 114] using *nodes* to represent the objects and *links* for their interactions. For this reason, complex systems broaden the understanding of the topological real-world networks' properties, such as small-world effect, Scale-free, Homophily and Community structure. Likewise they help in the analysis of semantics and functioning of the systems of interest. In its most basic form, a complex network contains only nodes and links; it can then be qualified of *plain network*. However, one can introduce a richer information in this model, depending on the considered system, modeling needs and constraints, which makes it a very flexible tool. Thereby, the network can be *directed* [114] if the relations between objects are asymmetric. It can be represented by multiple dimensions, where each dimension represents one type of relationship between nodes, leading to *multidimensional* network [138]; it also can be an *attributed* or *assigned* network [114] when some attributes are added to nodes or links in order to a better description of the model, etc. The analysis of these networks investigates their objects which generally tend to group into communities, according to their similarity or cohesive connectivity.

Goal. The thesis presented in this manuscript focuses its study on the uncovering of *communities of interest* being those whose nodes are interest-based similar, meaning that they share the same idea. Because real networks are increasingly enriched by relevant informations on the interactions between entities, we focus on directed and multidimensional networks. According to these two types of studied graphs, the interest is based both on topological and relational properties of links respectively. Indeed, for directed networks, the community detection approach stresses on the directionality of *in-links*, while for multidimensional networks, the community discovering method deals with the relevant dimensions to built its communities of interest. The following main goals arise:

1. Our first aim is to propose a method for extracting communities in directed graphs, based on the consideration of the incoming links to the nodes of interest, using triads. Triads are structures based on the homophily property of terrain graphs [86]. Therefore, interest-based similarity, such as in social network analysis, exhibits the idea that two entities going inward a third named as their common friend, have a higher probability of belonging to the same community. Indeed, the incoming link reflects the semantics of adhesion to the same idea as the node of interest, hence the notion of triad for directed graphs. The underlying goal behind it lies in revealing the in-centric nodes' importance. Furthermore, when these directed networks are assigned, we propose a method that simultaneously takes into account the directionality of edges and attributes of nodes to extract communities of interest.

2. Our second objective is to propose a method for community discovery in multidimensional graphs that includes the neighborhood quality and consequently the nodes interest based on their involvement level in their interactions. The interest is expressed by the relevance dimension-based similarity of nodes. The dimension relevance is assessed by the neighborhood stability of a node in that dimension, being dimensions in which the node owns more stable neighbors. The implied purpose is to show that a node's membership to a community depends on its level of activity in the dimensions included in that community, i.e. to establish that relevant dimensions are profitable for the community of interest extraction.

Methodology. The methodology used to achieve these outcomes is described below: For the first goal related to community detection in directed networks:

- Define a similarity measure for kernel nodes' extraction
- Extract the kernels by taking into account the interest principle based on the triads.
- Build communities centered around these kernels

As far as the second aim for discovering communities in multidimensional networks is concerned, the implemented protocol is described below.

- Define a new centrality measure based on the stability of a node's neighborhood
- Extract relevant dimensions of nodes based on the stability centrality measure
- Construct an assigned monodimensional network based on relevant dimensions
- Extract communities from the monodimensional network

A.2 Contribution

Our contribution in this thesis has three aspects: – Community detection Heuristic on directed networks, – Heuristic for community detection in directed and attributed networks – and Community discovery Heuristic on multidimensional networks.

The main purpose is to define a new way of looking at community of interest, different from the one discussed in the literature which focuses on semantics through ontologies [32]. This method is mainly limited by the fact that despite all works done on validation, they are still subject to discussion as knowledge do not only evolves but also there is no evidence that ontology always captures all the knowledge in the field. In order to consider real network features, we will be able to reuse and/or adapt existing topology-based solutions to uncover communities of interest in our context (directed and multidimensional graphs). As a result, the three most prominent contributions are listed below, according to the type of graphs of the context.

Heuristic for Community detection on directed networks. The first heuristic is related to community detection in directed networks. It allows to detect communities densely linked by triads, since communities' members are centered around kernels, being structures consisting in dense triads. To detect communities, we first define *Kernel degree*, a similarity measure based on both triads and Jaccard index, to measure the strength of the kernel vertices' similarity. Afterwards, the kernels reflecting the nodes of interest sets are extracted through kernel degree measure. Then we define *NCI* (Node Community Index), a merging measure of non-kernel nodes to kernels, in order to detect communities of interest consisting in triad-based densely nodes. Finally, we merge non-kernel nodes to kernel for which the NCI measure is maximized. This contribution has been the subject of 4 publications, namely one paper in an international journal [46], and 2 papers in international conferences [45, 47] and one paper in national conference [48].

Novel quality function. In order to take into account both relational and topological information, we propose a "modularity hybrid" quality function. It is a combination of 3 types of information: relational information based on link connectivity, topological information based on link directionality, and information based on node attributes. The modularity hybrid includes an hybrid similarity that investigates the topological aspect by applying the Kernel degree similarity measure implemented in the first contribution. This similarity measure contains informations on attributes and directionality and is joined to structural information to transform the directed attributed graph into a weighted one. Then, the resulting graph is applied to an hierarchical agglomerative algorithm to extract the communities qualified as more meaningful. This contribution has been the subject of one publication in a conference [49]

Heuristic for Community discovery multidimensional networks. This contribution focuses on the implementation of UCAD(commUnity disCovery method in Attributedbased multiDimensional networks) for community discovery in multidimensional networks. We use some topological graph properties to define a novel centrality called stability, needed for computing relevant dimensions. Then, we extract relevant dimensions of nodes based on the stability centrality measure. Afterwards, we enrich the attributes of nodes by their relevant dimensions. A dimension aggregation approach is then used to design a monodimensional attributed network. Finally, through a modified version of an hierarchical agglomerative method, we extract communities. This contribution has been the subject of one publication in an international conference [50] and one submitted paper in international revue, currently under revision.

Publications. This thesis leads to six publications with reading committee (5 conferences and 1 journal):

1. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Community structure extraction in directed network using triads.* International Journal of General Systems. 49(8): 819-842 (2020) https://doi.org/10.1080/03081079.2020.1786379

2. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Multidimensional networks: A novel node centrality metric based on common neighborhood.* In: 15^{th} Edition of CARI (Colloque sur la Recherche en Informatique et Mathématiques Appliquées), Sénégal, 2020.

3. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Communities in directed networks: Towards a hybrid model of semantic communities detection.* In: 14^{th} Edition of CARI (Colloque sur la Recherche en Informatique et Mathématiques Appliquées), Stellenbosch, South Africa, 2018.

4. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Finding directed community structures Using triads.* In 3^{rd} Edition of CRI (Conférence de Recherche en Informatique), Yaoundé, Cameroon, November, 2017.

5. Félicité Gamgne Domgue, Norbert Tsopze and Arnaud Ahouandjinou. *Nouvelle approche de clustering par kernel-pattern via la densité en triades : Optimisation de la métrique Kernel Degree Clustering.* In: CORIA'2017, Marseilles, France, March, 2017.

6. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Novel method to find directed community structures based on triads cardinality.* In: 13^{th} Edition of CARI (Colloque sur la Recherche en Informatique et Mathématiques Appliquées), Hammamet, Tunisia, October, 2016.

This thesis is an extension of our master thesis which leads to one publication with reading committee.

7. Félicité Gamgne Domgue, and Norbert Tsopze. *Analyse des réseaux sociaux: Communautés et rôles dans les réseaux sociaux.* Accepted for presentation in CARI'2014, Colloque sur la Recherche en Informatique et Mathématiques Appliquées, SaintLouis, Sénégal, October, 2014.

A.3 Perspectives

General perspectives. In this thesis we use attributes on nodes in directed networks. Edges are sometimes also enriched of attributes. Thus, it would be interesting to handle them in the process of community detection.

In this study, we focused on some properties of complex networks, disregarding the dynamical evolution property. Another perspective concerns the tracking of communities of interest over time, and therefore deal with dynamic aspect, as an individual may change his or her choices or preferences after an unknown period of time. Also, we could be interested in graph embedding using deep neural networks to model community detection. This remains an open question that deserves to be studied.

Perspectives in heuristics evaluation. This perspective concerns a thorough evaluation of the proposed heuristic for community of interest detection based on topological features of the network. Instead of using the state-of-the-art networks to enhance the implemented communities of interest, we plan to build new networks by setting up a platform for the interconnection of agricultural sector agents. Thereafter, the resulting social network (either directed or multidimensional) of agriculturists will be applied to the methods in order to discover communities of interest. This construction is achieved by putting in communication individuals or companies who would like to exchange their knowledge and/or technical expertise in their agricultural activities. We hope that this would contribute to the development of the agricultural sector through exchanges of experience and thus increase yields and the economy of our country.

B Résumé étendu

B.1 Contexte et Objectif

Avec la prolifération des médias sociaux et des applications mobiles, les utilisateurs interagissent en permanence en se partageant des documents, des images/vidéos et des messages, etc. Ces interactions peuvent être modélisées par un système complexe. Un système complexe est un système possédant certaines propriétés émergentes, dues aux interactions de ses objets constitutifs : [17, 114]. Il peut être rencontré dans de nombreux domaines différents tels que la biologie, la physique, l'informatique, la sociologie, etc. [3, 114]

Le *network modeling* permet de représenter ces systèmes par des *réseaux complexes* [3,17,114] constitués de noeuds pour représenter les entités et de liens pour représenter leurs interactions. Pour cette raison, les systèmes complexes favorisent une compréhension plus simplifiée des propriétés topologiques des réseaux réels, telles que l'effet petit-monde, la propriété *sans echelle*, l'homophilie et la structure communautaire. De même, ils aident à l'analyse de la sémantique et du fonctionnement des systèmes d'intérêt. Dans sa forme la plus élémentaire, un réseau complexe ne contient que des nœuds et des liens ; il peut alors être qualifié de réseau *basique*. Cependant, on peut introduire une information plus riche dans ce modèle, en fonction du système considéré, des besoins de modélisation et des contraintes, ce qui en fait un outil très flexible. Ainsi, le réseau peut être *orienté* [114] si les relations entre les objets sont asymétriques. Il peut être structuré en plusieurs dimensions, où chaque dimension représente un type d'interaction entre les nœuds, conduisant à un réseau *multidimensionnel* ; il peut également s'agir d'un réseau *attribué* [114] lorsque certains attributs sont affectés aux nœuds ou aux liens afin de mieux décrire le modèle. L'analyse de ces réseaux porte sur leurs objets qui tendent généralement à se regrouper en communautés, en fonction de leur similarité ou de la cohésion de leur connectivité.

Goal. La thèse présentée dans ce manuscrit concentre son étude sur la détection des communautés d'intérêt. Il s'agit des communautés dont les nœuds sont affiliées au même intérêt, c'est-à-dire qu'elles partagent la même idée. Comme les réseaux réels sont dotés d'amples informations pertinentes sur les interactions (liens) entre les entités, nous nous concentrons à la fois sur les réseaux orientés pour lesquels l'information est relative à l'orientation des liens, et sur les réseaux multidimensionnels pour lesquels l'information porte sur le type de liens. Selon ces deux types de graphes étudiés, l'intérêt est basé à la fois sur les propriétés topologiques et relationnelles des liens. En effet, pour les réseaux orientés, la méthode de détection des communautés met l'accent sur l'orientation des liens entrant, tandis que pour les réseaux multidimensionnels, la méthode de détection des communautés se focalise sur l'extraction des dimensions pertinentes pour construire ses communautés d'intérêt. Les principaux objectifs sont les suivants :

- Notre premier objectif est de proposer une méthode pour extraire les communautés dans des graphes orientés, basée sur la prise en compte des liens entrants vers les nœuds d'intérêt, en utilisant des triades. Les triades sont des structures basées sur la propriété d'homophilie des graphes de terrain [86]. Par conséquent, tel que le stipule l'analyse des réseaux sociaux, l'idée de similarité des noeuds s'exprime par le fait que deux entités orientées vers une troisième nommée comme leur ami commun (noeud d'intérêt) ont une plus grande probabilité d'appartenir à la même communauté. En effet, les liens entrants reflètent la sémantique d'adhésion à la même idée que le nœud d'intérêt, d'où la notion de triade pour les graphes orientés. L'objectif sous-jacent est de révéler l'importance des nœuds *centrés* connectés aux liens entrants. Par ailleurs, lorsque ces reseaux orientés sont attribués, nous proposons une methode de prise en compte simultanée de l'orientation des liens et des attributs pour extraire des communautés d'intérêt.
- Notre deuxième objectif est de proposer une méthode d'extraction de communautés dans les graphes multidimensionnels qui tient compte du type de voisinage des noeuds et non de la cardinalité de ce voisinnage. Ainsi elle se focalise sur l'intérêt des nœuds, dépendamment de leur niveau d'implication dans leurs interactions. En effet, l'intérêt est exprimé par la similarité des dimensions pertinentes des noeuds. La pertinence de la dimension dépend de la stabilité du voisinage d'un nœud dans cette dimension, c'est-à-dire les dimensions dans lesquelles le nœud possède des voisins plus stables constituent son ensemble de dimensions pertinentes. L'objectif implicite est de montrer que l'appartenance d'un nœud à une communauté dépend de son niveau d'activité dans les dimensions incluses dans cette communauté, c'est- à-dire d'établir que les dimensions pertinentes sont bénéfiques pour l'extraction de la communauté d'intérêt

Méthodologie. La méthodologie utilisée pour atteindre ces objectifs est décrite cidessous : Pour le premier objectif lié à la détection des communautés dans les réseaux orientés :

- Définir une mesure de similarité des noeuds noyaux
- Extraire les noyaux (noeuds coeur) constituant les noeuds d'intérêt via les triades.

- Créer des communautés centrées autour de ces noyaux

En ce qui concerne le deuxième objectif de détection de communautés dans des réseaux multidimensionnels, le protocole mis en œuvre est décrit ci-dessous.

- Définir une nouvelle mesure de centralité basée sur la stabilité du voisinnage d'un noeud
- Extraire les dimensions pertinentes des nœuds sur la base de cette centralité
- Construire un réseau attribué monodimensionnel basé sur les dimensions pertinentes
- Extraire les communautés du réseau monodimensionnel

B.2 Contribution

Notre contribution dans cette thèse comporte trois aspects : – une heuristique de dé- tection de communautés sur les graphes orientés, – une heuristique pour la détection de communautés dans les réseaux orientés attribués – et une heuristique de détection de communautés sur les graphes multidimensionnels.

L'objectif principal est de définir une nouvelle façon de considérer la communauté d'intérêt, différente de celle qui est discutée dans la littérature et qui se concentre sur la sémantique à travers les ontologies [33]. Cette méthode est principalement limitée par le fait que, malgré tous les travaux réalisés sur la validation, ils sont toujours sujets à une discussion car non seulement les connaissances évoluent mais aussi il n'y a pas de preuve que l'ontologie capture toujours toutes les connaissances dans le domaine. Afin de pallier cette limite tout en prenant en compte les caractéristiques réelles du réseau, nous proposons de réutiliser et/ou adapter les solutions existantes basées sur la topologie pour découvrir des communautés d'intérêt dans notre contexte (graphes orientés et multidimensionnels). En conséquence, les trois contributions les plus marquantes sont énumérées ci-dessous, selon le type de graphes du contexte.

Heuristique pour la détection de communautés dans les graphes orientés. La première heuristique est liée à la détection de communautés d'intérêt dans les graphes orientés. Elle permet de détecter des communautés basées sur les triades, puisque les membres des communautés sont centrés autour de noeuds noyaux qui sont des structures constituées de triades denses. Pour détecter ces communautés, nous définissons d'abord le *Kernel degree*, mesure de similarité basée à la fois sur les triades et le coefficient de Jaccard, pour mesurer la force de la similarité des noeuds noyaux et les extraire. Ensuite, à partir de ces noeuds noyaux reflétant les ensembles de nœuds d'intérêt, nous définissons *NCI (Node Community Index)*, une mesure de fusion des nœuds non-noyaux aux noeuds noyaux, afin de détecter les communautés d'intérêt consistant en des nœuds denses basés sur des triades. Enfin, nous fusionnons les nœuds non-noyaux au noyau pour lequel la mesure NCI est maximale. Cette contribution a fait l'objet de 4 publications, à savoir un article dans une revue internationale [46], et 2 articles dans des conférences internationales [45,47] et un article dans une conférence nationale [48].

Nouvelle fonction qualité. Afin de prendre en compte les attributs de noeuds dont pourraient être dotés les noeuds du graphe orienté, nous proposons une fonction de qualité "hybride de modularité". Il s'agit d'une combinaison de 3 types d'informations: informations relationnelles basées sur la connectivité des liens, informations topologiques basées sur la directionalité des liens, et informations basées sur les attributs des nœuds. La modularité hybride est constituée d'une mesure de similarité hybride qui modélise l'aspect topologique basée sur la mesure de similarité des noeuds noyaux mise en œuvre dans la première contribution. Cette mesure de similarité contient des informations sur les attributs et la directionnalité des liens. Elle est associée aux informations structurelles pour extraire les noeuds d'intérêt. Ensuite, par application d'un algorithme d'agglomération hiérarchique, l'on procède à l'identification des communautés qualifiées de plus significatives. Cette contribution a fait l'objet d'une publication dans une conférence internationale [49].

Heuristique de détection de communautés d'intérêt dans les graphes multidimensionnels. Cette contribution se concentre sur la mise en œuvre de la méthode UCAD (commUnity disCovery method in Attributed-based multiDimensional networks) pour l'extraction des communautés dans des réseaux multidimensionnels. Nous utilisons certaines propriétés topologiques des graphes pour définir une nouvelle mesure de centralité appelée stabilité, nécessaire à l'identification des dimensions pertinentes. Ensuite, les dimensions pertinentes determinées, nous enrichissons les attributs des nœuds par leurs dimensions pertinentes. Une approche d'aggrégation des dimensions est alors utilisée pour concevoir un réseau monodimensionnel attribué. Enfin, grâce à une version modifiée d'une méthode agglomérative hiérarchique, nous construisons les communautés. Cette contribution a fait l'objet d'une publication dans une conférence internationale [50] et de deux articles soumis dans des revues internationales. Ils sont actuellement en cours de révision.

Publications. Cette thèse a donné lieu à six articles scientifiques avec commité de lecture (5 conférences et 1 journal):

1. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Community structure extraction in directed network using triads.* International Journal of General Systems. 49(8): 819-842 (2020) https://doi.org/10.1080/03081079.2020.1786379

2. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Multidimensional networks: A novel node centrality metric based on common neighborhood.* In: 15^{th} Edition of CARI (Colloque sur la Recherche en Informatique et Mathématiques Appliquées), Sénégal, 2020.

3. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Communities in directed networks: Towards a hybrid model of semantic communities detection.* In: 14^{th} Edition of CARI (Colloque sur la Recherche en Informatique et Mathématiques Appliquées), Stellenbosch, South Africa, 2018.

4. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Finding directed community structures Using triads.* In 3^{rd} Edition of CRI (Conférence de Recherche en Informatique), Yaoundé, Cameroon, November, 2017.

5. Félicité Gamgne Domgue, Norbert Tsopze and Arnaud Ahouandjinou. *Nouvelle approche de clustering par kernel-pattern via la densité en triades : Optimisation de la métrique Kernel Degree Clustering.* In: CORIA'2017, Marseilles, France, March, 2017.

6. Félicité Gamgne Domgue, Norbert Tsopze and René Ndoundam. *Novel method to find directed community structures based on triads cardinality.* In: 13^{th} Edition of CARI (Colloque sur la Recherche en Informatique et Mathématiques Appliquées), Hammamet, Tunisia, October, 2016.

Cette thèse est une extension de notre mémoire de master qui avait donné lieu à une publication avec commité de lecture.

7. Félicité Gamgne Domgue, and Norbert Tsopze. *Analyse des réseaux sociaux: Communautés et rôles dans les réseaux sociaux.* Accepted for presentation in CARI'2014, Colloque sur la Recherche en Informatique et Mathématiques Appliquées, SaintLouis, Sénégal, October, 2014.

B.3 Perspectives

Perspectives générales. Dans cette thèse, nous nous intéressons aux attributs sur les nœuds dans les réseaux orientés. Les liens sont parfois aussi enrichis d'attributs. Ainsi, il serait intéressant de les manipuler dans le processus de détection des communautés.

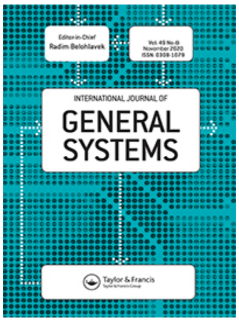
Dans cette étude, nous nous sommes concentrés sur certaines propriétés des réseaux complexes, en négligeant la propriété basée sur l'évolution du réseau. Une autre perspective concerne le suivi des communautés d'intérêt dans le temps, et traite donc de l'aspect dynamique, car un individu peut changer ses choix ou ses préférences après une période de temps inconnue.

Nous pourrions également être intéressés au graph embedding en utilisant des réseaux de neuronnes profonds pour modéliser la détection des communautés. Cette question reste ouverte et mérite d'être étudiée.

Perspectives dans l'évaluation des heuristique. Cette perspective concerne une évaluation approfondie de l'heuristique proposée pour la détection de communautés d'intérêts basée sur les caractéristiques topologiques du réseau. Au lieu d'utiliser les réseaux de l'état de l'art pour améliorer les communautés d'intérêt mises en œuvre, nous entendons construire des nouveaux réseaux tant orientés que multidimensionnels, en mettant en place une plateforme d'interconnexion des opérateurs du secteur agricole. Par la suite, les réseaux sociaux d'agriculteurs qui en résulteront seront appliqués aux méthodes developpées dans cette thèse, afin de découvrir les communautés d'intérêt. Cette construction est réalisée en mettant en communication les personnes morales ou physiques qui souhaiteraient échanger leurs connaissances et/ou leur expertise ou technicité dans leurs activités agricoles. Nous espérons que cela contribuera au développement du secteur agricole par le biais d'échanges d'expériences ainsi que l'amélioration du rendement et l'économie de notre pays.

C List of publications

C.1 Community structure extraction in directed network using triads



International Journal of General Systems

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ggen20

Community structure extraction in directed network using triads

Félicité Gamgne Domgue , Norbert Tsopze & René Ndoundam

INTERNATIONAL JOURNAL OF GENERAL SYSTEMS
2020, VOL. 49, NO. 8, 819–842
https://doi.org/10.1080/03081079.2020.1786379

Taylor & Francis
Taylor & Francis Group

Community structure extraction in directed network using triads

Félicité Gamgne Domgue[a,b], Norbert Tsopze[a,b] and René Ndoundam[a]

[a]Département d'informatique, Université de Yaoundé 1, Yaoundé, Cameroun; [b]Sorbonne Université, IRD, UMMISCO, Bondy, France

ABSTRACT

Community detection in directed networks appears as one of the most relevant topics in the field of network analysis. One of the common themes in its formalizations is information flow clustering in a network. Such clusters can be extracted by using triads, expected to play an important role in the detection of that type of communities since communities could be centered round core nodes called *kernels*. Triads in directed graphs are directed sub-graphs of three nodes involving at least two links between them. To identify communities in directed networks, this paper proposes an in-seed-centric scheme based on directed triads. We also propose a new metric of the communities' quality based on the triad density of communities. To validate our approach, an experiment was conducted on some networks showing it has better performance on triad-based density over some state-of-the-art methods.

ARTICLE HISTORY
Received 11 January 2020
Accepted 3 April 2020

KEYWORDS
Networks analysis; directed networks; kernel; community detection; triad

1. Introduction

One of the recurrent research topics in Network Analysis is community detection. In directed networks, it appears as one of the dominant research works because of links' direction that should be preserved. For instance, clusters in the directed hyperlink structure of the Web refers to sets of web pages that share some common topics. Community detection methods for directed graphs (Maliaros 2013) focus either on link structure or semantic properties to detect communities. Measuring the quality of partitions also resides in the optimization of some metrics, like the directed modularity, as it focuses on the connectivity of nodes. However, this widely popular measure stresses on the density of links within groups without discriminating the direction of edges (Li 2019). Indeed, it does not implement the idea that an edge from a low out-degree but high in-degree node to an opposite case node should be considered of a bigger value.

In order to improve the segmentation quality, the proposed method that extracts core nodes based on triad density has been proposed in this work. It introduces the notion of

CONTACT Tsopze Norbert ✉ tsopze.norbert@gmail.com

kernel degree as a combination of *Neighborhood Overlap (NO)* and *Triad Weight (TW)*. Our specific weighting scheme is based on an extension of the idea that, in "good" communities, information can be centralized by kernels and attainable within a community more easily than between communities. Therefore, our approach expresses the idea of detecting groups of nodes with homogeneous in-link structure (e.g. citation-based clusters) through triads, and gives the possibility to *kernel* nodes to own more common neighbors. The main question to assess the performance of the proposed algorithm may arise such as: How can we profitably use triads to quantitatively discover communities in a directed network ? What advantages does the development of a community discovery method based on kernel has over other methods?

The specific contributions of this paper are:

- We propose a new in-seed-centric based clustering scheme that points up triadic closure of structures.
- We introduce a new concept called *kernel degree* using information about directed triads to improve community detection in directed networks.

The paper is organized as follows. Section 2, structured in two paragraphs, presents in the first some research works related to directed network clustering and in the second one a detailed description of triad concept such as addressed by other methods with limits of their contributions, then, an introduction to related works on kernels community detection is exhibited. In Section 3, we formally introduce and define several concepts used into the proposed clustering method; therefore, a description of the proposed technique is detailed in three steps: Kernel candidates' generation, kernel extraction and community computation. To validate our approach and the choice of any concept pointed up into Section 3, 4 is an experiment study through some metrics (*d-modularity* and *triad density*) that shows the performance of our method, over some state-on-the-art methods. Section 5 concludes this work.

2. Related works

2.1. Community detection in directed networks

Finding clusters in directed networks is a challenging task with several important applications in a wide range of domains. Some methods ignore the direction of links (*Walktrap* Pons and Latapy 2005; *Edge Betweenness* Newman 2004; *Label Propagation* Raghavan and Albert 2007; *Louvain* Blondel et al. 2008), while others like Zhou, Hofman, and Schlkopf (2005), Satuluri and Parthasarathy (2011) and Kim, Son, and Jeong (2010), Clemente and Grassi (2018) propose new ways as to how edge directionality can be utilized in the clustering task. The latter focuses on the extension of tools and measures developed for undirected case, as the ones based on the optimization of the so-called *directed modularity* (Nicosia, Mangioni, and Malgeri 2009) and the *directed clustering coefficient* (Clemente and Grassi 2018). The formers convert a directed graph into bipartite, undirected and weighted one, this enabling to utilize the richness and complexity of existing methods to find communities in undirected graphs. Directed modularity measure has been demonstrated in Nicosia, Mangioni, and Malgeri (2009) to be expressed as:

$$Q_d = \frac{1}{2m}\Sigma_{ij}\left(A_{ij} - \frac{k_i^{out}k_j^{in}}{2m}\right)\delta(c_i, c_j). \tag{1}$$

Yet, Fortunato (2010) describes how this measure has a limit resolution, the difficulty for the measure to extract small scale communities. To make up for this limit, Gautier and Lancichinetti (Lancichinetti and Fortunato 2009; Krings and Blondel 2011) proposed to inform some parameters about either the number of communities or whether the method should extract small communities or not. These parameters can greatly affect the accuracy of an approach if the values provided by the user are incorrect. Moreover results could not be satisfactory in triad-based clustering from real-world networks, if the user parametrizes community size to 2 vertices. As the modularity optimization has been showed as NP-hard problem (Brandes et al. 2008), many authors proposed some heuristics to optimize the known modularity function. Santiago et al. in Santiago and Lamb (2017) propose seven heuristics among which coarsening merger, moving node, and multilevel heuristics. In D$\tilde{Z}$amić, Aloise, and Mladenovic (2019) authors proposed a variant of the Variable Neighborhood Decomposition Search (VNDS) heuristic called Ascent-Descent VNDS for maximizing the modularity and a new neighborhood structure. The approach consists of accepting the better decomposed subproblem solution in ascent way and also worse subproblem solution with some probability in the descent way. However, modularity ignores the impact of in-link degree of nodes. In order to make good use of the directionality, a recent heuristic based on constrained directed label propagation algorithm (CDLPA) is proposed in Li (2019). The authors consider the balance growth of communities through an improvement of LPA for directed networks. CDLPA is effective for datasets with a monotonous degree distribution of nodes, and overcomes the imbalance growth of communities limit. Indeed, it assumes that communities must have a similar capacity of nodes; therefore it constrains the membership of a node towards a community to which it is not strongly connected.

Likewise, by exploring the idea of extending tools, authors in Clemente and Grassi (2018) propose a new local clustering coefficient for directed and unweighted networks. Starting from existing coefficient designed for the weighted and undirected case, they propose to take into account the triangles formed by the neighbors of a node v_i, through a preserving the initial idea of the clustering coefficient. Extending tools and algorithms does not improve the scalability of these methods. To speed up the graph clustering, some authors propose some "parallel" models. This is the case of the works in Souravlas, Sifaleras, and Katsavounis (2019) where the authors propose an incremental approach combining parallel processing techniques with threaded binary trees. The idea consists in both transforming the directed graph into a weighted networks with irregular topologies and using a stepwise path detection strategy, so that each step finds a link that increases the overall strength of the path being detected. The obtained results are overlapped communities and one of the main advantage is the possibility to affect the newly introduced node to a cluster. The same team also has a new parallel approach where social networks information (like user profile or requested data) is combined with distributed systems information to identify users' membership to a community. They also introduce a new metric, based on data requests and use it as the belonging degree of a node in a certain formed community. Moreover, Shaojie et al. (2018) designed an approximate optimization parallel algorithm called *Picasso* by integrating Mountain model, based on graph theory to approximate the selection of nodes needed for merging, and Landslide algorithm based approximate optimization, which is used to update the modularity.

In order to make up for these aforementioned restrictions and enhance one of the fundamental properties of real networks, namely *homophily*, which is the tendency of neighboring nodes to share the same center of interest, some authors have focused on the triads.

2.2. Triad-based methods

Triads, initially studied by authors Wasserman et al. in Wasserman and Katerin (1994) in social network analysis were introduced by Serrour and Arenas (2011) to identify communities of different types. Triads are considered as wedges, i.e paths of length 2 by Klymko, Gleich, and Kolda (2014). Thus, a triad can be integrated into a triangle. In directed graphs, the process of extracting semantic structures should take into account either "in" or "out" directionality of links for meaningful interpretation. Therefore, it becomes interesting to specify those of nodes centered around *kernels* (set of influential nodes inside a group) according to *in-direction*. Then, kernel community detection methods are considered as seed-centric approaches (Kanawati 2014) because of the influence of nodes centralizing information. Using triads enlarges the possibility to consider low-degree nodes instead of high degree nodes called "hub nodes". That way, low-degree nodes will not be isolated at the end of the community detection process.

In the same way, Tsourakakis in Tsourakakis (2008) initiated the study of degree labeled triangle. He argued that low-degree vertices form fewer triangles than higher degree vertices. At this stage, to make up for this limit, the purpose of this new approach is to cluster low-degree nodes so that they should be more linked together around a kernel and could more easily access to central retained information. So, it takes into account in-links to the kernel and vertices with low-degree.

Some methods, like Wang's approach (Wang 2011) explored the problem of detecting community kernels, in order to either exhibit different influence or different behavior of vertices inside a structure for easily interpreting results, then uncovering the hidden community structure in large social networks. Wang in his model sets the size of the partition and proceeds by a random choice of node to initiate the kernel. This constitutes a considerable drawback. In fact, providing accurate values input parameters, including the number of communities, requires a priori knowledge of the network to be analyzed. Whereas, in practice, such knowledge is not always available. To make up for this limit, the proposed approach integrates the idea that in-degree value of nodes helps to better structure influential nodes and easy information flow. This includes topological structure based on triads and conducts to semantic community structures.

Unlike modularity which does not allow to the incoming degree of nodes a significant value, the new *triad density* measure implemented in this work (see Definition 3.4) takes into account the strength of their incoming degree values by exhibiting triad-based structure of the resulting partition. Also, we focus on kernels because they represent community core and lead to meaningful structures. In this paper, a triad is a set of 3 vertices linked through at least 2 edges. It could be represented in two categories, namely opened and

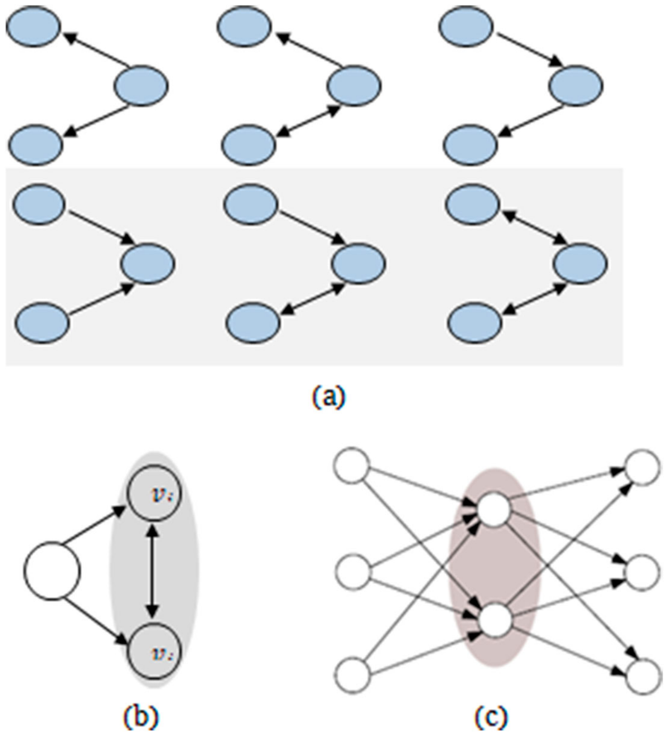


Figure 1. Basic structures of our kernel community model. (a) Examples of opened triads. (b) A toy example of Closed triad and (c) Common neighbors'toy example.

closed triads as shown in Figure 1(b). In undirected networks, there is only two types of triad, a path of length 2 for the opened case and a triangle for the closed one. In directed networks, there are many types of opened cases as shown in Figure 1(a). This work focuses on *in-seed-centric* (Kanawati 2014) approach because of the influence of nodes centralizing information, a good closed relationship pattern (see Definition 3.1) and ability to concentrate information between nodes (see Figure 1(c)). This is not the case for all types of directed triads. As an example, in a blog readership network, there are two types of "bloggers": "writers" who generate influential blogs read by many others, and "readers" who read a lot but seldom write anything for others to read.

3. Community detection method

This method makes use of centric-based approaches through extracting subgraphs induced by co-parent structures, called *Kernel*. Seed-Centric approaches for Community Detection in Complex Networks $G = (V, E)$ generally follows these principal steps (Kanawati 2014):

(1) Seed computation;
(2) Seed local community computation;

(3) Community computation out from the set of local communities from step 2.

The algorithm for finding structures in directed networks we propose here makes use of an introduced centrality measure *kernel degree* based on triads cardinality, as described in the following section.

This method is structured in three steps: (i) *Generation of Kernel candidates*, (ii) *Kernel extraction*, (iii) *Community computing*. Before describing these steps, we define some basic concepts in Section 3.1.

In this method, we consider a Community as a set of vertices centered round kernel, and easily accessing to the central information retained by that kernel. Otherwise, a community seems to be a subgraph yielded by kernel. This way, we consider the number of communities as one of the metrics of quality partition evaluation in experiments.

3.1. Basic terminology and concepts

Considering a given directed graph $G = (V, E)$ with $n = |V|$ the number of vertices and $m = |E|$ the number of edges. An edge e_{ij} connects vertex v_i with vertex v_j. We now give some useful definitions:

Definition 3.1 (Pattern-based cluster)**:** We refer to pattern-based clusters as *triad-based clusters*, since they represent structures in which nodes are similar among them as they, in majority, point to their kernel.

Definition 3.2 (Kernel degree)**:** Intuitively, *kernel degree* measures the strength or weight of the kernel vertex similarity. Its value between a pair of vertices v_i and v_j is evaluated using the formula:

$$K_{ij} = \frac{|\Delta_{ij}|}{|\Delta_j|} \times \frac{|\Gamma_j^{in} \cap \Gamma_i^{in}|}{|\Gamma_j^{in} \cup \Gamma_i^{in}| - \theta}. \tag{2}$$

In Equation (2), Γ_i^{in} stands for the *in-Neighborhood* set for a vertex v_i. We use Δ_{ij} to represent a set of triad involving both v_i and v_j, and Δ_j to represent a set of triad in which v_j is involved.

The first term is based on triads, and promotes the *Triad Weight* through a kernel; Given two vertices v_i and v_j, a standard way to compute the percentage of triads they form together is to compute the ratio between the total number of triads in which the pair of vertices is included (numerator) and the total number of triads in which vertex v_j is contained (denominator). The second term promotes the *Neighborhood Overlap* of v_i and v_j vertices. It concerns a Jaccard Index variant (Pang-Ning, Steinbach, and Kumar 2005), which consists in measuring neighborhood similarity of two vertices so that they could belong to the same kernel. Unlike the Jaccard Index which does not consider the connectivity between the nodes because it just computes the common neighbors of 2 vertices v_i and v_j, Neighborhood Overlap integrates the fact that there could be or not an edge between v_i and v_j. That is why we use the θ parameter in the denominator to compare 2 similar kinds of neighbor sets. In fact, in the numerator, one vertex can belong to the in-neighborhood of another, and vice versa. θ can take different values 0, 1 and 2, depending on the connectivity of v_i and v_j vertices.

- $\theta = 0$ if $(v_i, v_j) \notin E$ and $(v_j, v_i) \notin E$
- $\theta = 1$ if $(v_i, v_j) \in E$ and $(v_j, v_i) \notin E$
- $\theta = 2$ if $(v_i, v_j) \in E$ and $(v_j, v_i) \in E$

Definition 3.3 (Kernel)**:** Kernel is a set of vertices with the same neighborhood, so that these neighbors expand gradually inward the kernel, according to a threshold.

The Kernel is formally defined as

- $K = \{v_1, \ldots, v_i, \ldots, v_{|K|}\}, v_i \in V$
- $\forall v_i, v_j \in K, \Gamma_i^{in} \simeq \Gamma_j^{in},$
- $\forall i \neq j \backslash v_i, v_j \in K, K_{ij} > \sigma.$

The threshold value σ is expressed in the Section 3.3.2 below. Throughout the article, we will use the network of Figure 3(a) of Wang (2011) to illustrate the steps of the proposed method. It contains 14 nodes and 32 edges. The notations are simplified by abbreviating the names of the nodes as follows: Demi Moore (DM), Oprah Winfrey(OW), Al Gore (AG), Barack Obama (BO), Ashton Kutcher (AK).

Definition 3.4 (Triad Density)**:** The triad density of a partition is a ratio that conceals difference between real number of triads in that partition and maximal possible number of triads in the whole graph.

$$TriadDens = \frac{\Sigma_i^n |\Delta_{C_i}|}{\binom{3}{|V|}}, \tag{3}$$

where the numerator expresses the number of triads from the overall communities, and the denominator denotes that combination value equals to $|V|!/(|V|-3)!3! = \frac{1}{6}(|V|(|V|-1)(|V|-2))$, with $|\Delta_{C_i}|$ being the number of triads in the community C_i and n, the number of communities. $TriadDens = 0$ if vertices are isolated or if $|V| < 3$. Otherwise, $TriadDens = 1$ if the graph is complete, i.e there is bidirectional edge between every pair of vertices.

After defining useful concepts, the next sections (Sections 3.2, 3.3 and 3.2.3) present how does the process of extracting kernels spread out and how does communities are finally generated. We describe the kernel-based community model through the Algorithm 2 below. We group these steps into three main phases:

(1) Kernel candidates' generation (from step 1 to step 3) at the end of which the structure *KDict* contains these candidates;
(2) Kernel extraction (step 4 and step 5) where t kernels formed by triads are extracted from the candidates;
(3) Community computing process (Step 6) where kernels are extended by migration of the non-kernel nodes to kernels.

3.2. Kernel candidates' generation

This step consists in pruning the list of the nodes of the network to preserve those that are eligible to eventually be kernels. It spreads out into three subtasks: extract the list of node

Algorithm 1 The kernel-based community model for community detection

```
Require: Directed graph $G = (V, E)$
Ensure: List of Communities $C = \{C^{(1)}, ..., C^{(t)}\}$
    Step 1: Compute In-degree central and pruned list $CL$ according to the degree average of the graph. The list is in decreasing order of degree
    Step 2: Compute Kernel Dictionary $KDict$ based on each distinct pair of $CL$ such as $KDict = [((v_i, v_j), K_{ij})]$
    Step 3: Compute Interclass inertia vector $I$ according to $K_{ij}$ values of $KDict$
    Step 4: Compute a threshold $\sigma$ being the standard deviation of the vector $I$
    Step 5: Extraction of kernels as described in Algorithm 1 from Line 3 to Line 13.
    Step 6: Community building through non-kernel nodes migration, as described in Algorithm 2
```

degrees through computing a pruned centrality list; then compute the values of the weights between pairs of nodes from the previous list, through computing kernel dictionary; finally grouping these couples according to their neighbors' similarity through the computation of an inter-class inertia vector.

3.2.1. In-degree centrality list computing

This step consists in determining a list of nodes sorted in descending order of their in-degree; that list is called *Centrality List* (CL). So that those with maximal in-degree are more eligible than those with a low in-degree. Then, pruning from the list *in–pendant* and *in–isolated* vertices i.e. those of nodes with an *in –degree* below the *in–degree graph average*, as inspired by Steven and Martin (2016) who defined a *pendant* as vertex with a single neighbor which has degree 1. This filtering step improves performance and allows simplifying assumptions later when deciding whether to include a vertex into a kernel. For instance, in a citation network, an *in–pendant* or *in–isolated* vertex corresponds to an author whose the area search does not interest other researchers, so removing these nodes with an in-degree below 2 improves the processing speed and produces more semantic results later.

For illustration on the network in Figure 2(a), the CL contents is: $CL = ['AG', 'BO', 'AK', 'DM', 'OW']$ because they have an in-degree above 2, being average degree of the network.

3.2.2. Kernel dictionary computing

This step consists in computing kernel degree values for every pair $(v_i, v_j) \in CL$ (See Definition 3.2 above).

Therefore it represents these values by a kernel dictionary called ($KDict$) whose items are structured as (key_dict, $value\_dict$). key_dict is any *unordered* pair of nodes from CL pruned list, and $value\_dict$ is the corresponding kernel degree K_{ij} of these pairs. ($KDict$) will be used in the kernel extraction approach as defined in the next section. The size of $KDict$ equals $n(n-1)/2$ in the worst case (when all the nodes of the network belong to CL). $KDict$ is sorted in decreasing order of K_{ij}. Formally, $KDict = [((v_i, v_j), K_{ij})]$.

For illustration, *KDict = [((“DM”, “OW”),* 1.6*), ((“AG”, “BO”),* 0.595 *), ((“AK”, “OW”),* 0.32*), ((“AK”, “DM”),*0.267*), ((“BO”, “DM”),* 0.0635*), ((“AG”, “DM”),* 0.057*), ((“BO”, “OW”),* 0.0158*), ((“AG”, “OW”),* 0.0143*), ((“BO”, “AK”),* 0.013*), ((“AG”, “AK”),* 0.012*)].*

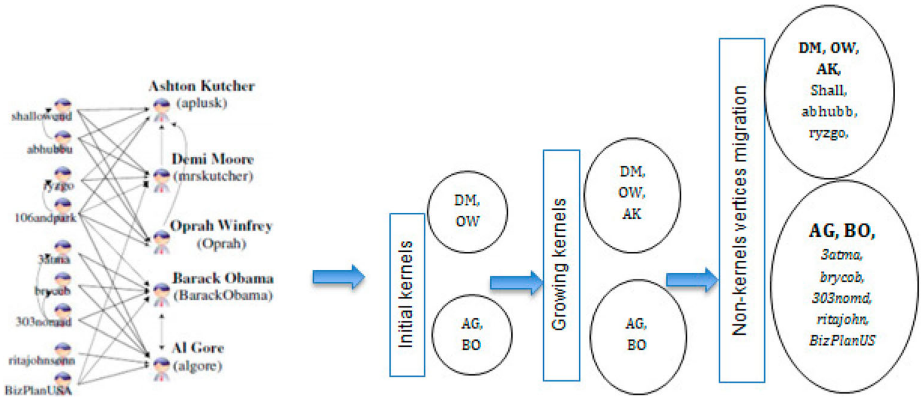


Figure 2. Illustration of the model on network in Figure 3(a).

Let us remember that a *kernel* in this paper is a set of nodes owning a common central in-degree overlapping neighborhood. This task of extracting kernels focuses on determining those of nodes more eligible to belong to kernel via interclass inertia.

3.2.3. Interclass inertia computation

Given that the clustering main goal is to form homogeneous groups, the measure used here is *Inter-class Inertia*, and the list of the inter-class inertia values is named I. This list is based on *KDict* dictionary. In fact, high inter-class inertia values indicate that objects tend to be more dissimilar, and consequently should belong to distinct groups. So, it divides objects into two groups, those eligible to belong to a kernel and those not eligible. The delimitation of two groups is done by a comparison of values from Inter-class Inertia List with a computed *Standard Deviation* σ on I. This way, vertex pairs (i, j) whose Inter-class Inertia value is larger than σ are more eligible to belong to kernels. The Inter-class Inertia between 2 sub-groups G_1 and G_2 is expressed as

$$I(G_1, G_2) = |G_1|(\mu_1 - \mu)^2 + |G_2|(\mu_2 - \mu)^2 \tag{4}$$

$|G_1|$ and $|G_2|$ are respectively the number of edges in groups G_1 and G_2. μ_1, μ_2, and μ are respectively the average *kernel degree* for G_1, G_2 and G.

The Figure 3(a) Network presents distinct groups G_1 and G_2 respectively as the following, and the corresponding Inter-class Inertia of *KDict* as : for $G_1 = \{$*(DM, OW)*$\}$ *and* $G_2 = \{$*(AG, BO), (AK, OW), (AK, DM), (BO, DM), (AG, DM), (BO, OW), (AG, OW), (BO, AK), (AG, AK)*$\}$, the Inter-class Inertia for these groups is 1.987. Then, the following pair of nodes in *KDict* list moves from G_2 to G_1, and their contents become: $G_1 = \{$*(DM, OW), (AG, BO)*$\}$ *and* $G_2 = \{$*(AK, OW), (AK, DM), (BO, DM), (AG, DM), (BO, OW), (AG, OW), (BO, AK), (AG, AK)*$\}$, and the Inter-class Inertia for these groups is 1.705. We change the G_1 and G_2 contents and so on. The interclass inertia vector is progressively computed and its contents are presented as follows: $I = [\,1.987, 1.705, 1.359, 1.162, 0.844, 0.627, 0.439, 0.297, 0.186, 0.131]$. Afterwards, a threshold helpful for kernel extraction process is computed, as detailed in the next section.

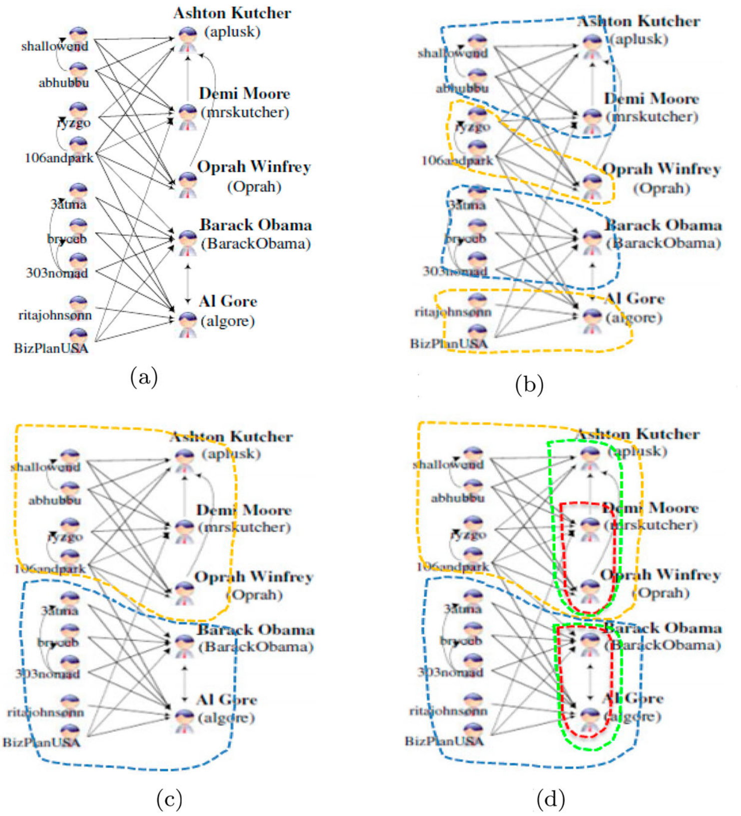


Figure 3. An illustration of outputs from the extract of Twitter network. (a) Extract for Twitter social network used in Wang (2011). (b) Output by Newman's algorithm. (c) Output by Wang's algorithm and (d) Output by our algorithm.

3.3. Kernel extraction approach

This section firstly presents the properties fulfilled by kernels, and secondly it describes the threshold on which the kernels are structured.

3.3.1. Kernel properties

To select kernel nodes, some metrics inherent to vertex centrality have been studied: Common neighbors (Xu, Xu, and Zhang 2015), Distance (Cosinus, Euclidian . . .), Jaccard Index (Steinhaeuser and Chawla 2008), geodesic (short path) (Newman 2004), Clustering Coefficient (Latapy, Magnien, and Del Vecchio 2008). The retained metrics in this paper combined through *kernel degree* (see Equation (2)) to strengthen the similarity of kernel vertices are: the "common neighbors" corresponding to Triad Weight, the left term of the formula and the variant "Jaccard Index" corresponding to Neighborhood overlap, the right term of the formula. Their combination leads to scalable results. Indeed, our empiric tests on metric taken separately show the superiority of *kernel degree* on various networks, as evaluated in Section 4.2.

The phase begins by initiating kernels with distinct pair of vertices possessing the highest corresponding Inter-class Inertia, through the mileage of the *KDict* list. Given that an initiating kernel vertex r of a kernel t between the initiating pair of vertices $\{r, u\}$. If a vertex p in *KDict* is coupled to another one q with whom the *kernel degree* K_{pq} is lower than its *kernel degree* K_{pr} with the initiating kernel vertex r, p immediately migrates to that kernel t. So the kernel t will be made of $\{r, u, p\}$. Then those already belonging to the kernel will not be treated in the future steps. The vertices belonging to the kernel own almost the same neighbors. The approach proposed here makes use of a new concept *Kernel Degree* K_{ij} as defined in the Definition 3.2, that measures the strength of a kernel according to a threshold. This concept is based on the *triadic membership* to emphasize the semantic proximity that links kernel members conducting to efficient centralization of information over the network.

We require that the kernel fulfills the following properties:

(1) Every kernel contains distinct pair of vertices with inter-class inertia upper than a threshold.
(2) The kernel vertices have higher *kernel degree* values, proportionally to the degree distribution of the graph.
(3) Given an initiating pair (i, j) and a border vertex k in a kernel, the neighborhood overlap cardinality of (i, j) must be higher than the neighborhood overlap cardinality of any neighbor t of (i, j, k). Formally, Given $\forall(i, j) \backslash i, j \in CL$, and $k \in K$, $|\ \Gamma_{i,j} \cap \Gamma_k\ | \geq |\ \Gamma_{i,j} \cap \Gamma_t\ |$, where $\Gamma_{i,j} = \Gamma_i \cap \Gamma_j$.

3.3.2. Standard deviation σ

To compute Kernels, we focus on a threshold, which is the standard deviation from inter-class inertia list I. Unlike the well-known meaning of the standard deviation, we observe during the experimental phase that the higher the standard deviation σ computed from a set nodes, the more likely they possess an almost common neighborhood. As a matter of fact, as illustrated in Figure 5, a lower standard deviation indicates that these vertices have a quasi-null common-neighborhood cardinality. Because of the power-law degree distribution in real-life networks, very little nodes get a high in-degree widely above the in-degree average. We make the assumption that according to Leskovec, Kleinberg, and Faloutsos (2005), there tend to be a few “hub” vertices with a very high degree and great number of vertices with a much lower degree. In the case of directed graphs, the concept of hub vertices depend on the in-degree or the out-degree value. This paper stresses on in-degree vertices, meaning that they receive more information from the other vertices than “non-hub” vertices. The standard deviation is expressed as

$$\sigma = \sqrt{\frac{1}{n} \Sigma_{i=1}^{n} (x_i^2) - \mu^2}, \tag{5}$$

where $\mu = (1/n)\Sigma_{i=1}^{n} x_i$ indicates s_i average (or mean), and x_i indicates every element of the interclass inertia array. A kernel is initially made of a pair of vertices, and expands progressively by adding vertices which are in couple with kernel members, whose the corresponding *kernel degree* value is above σ. This leading to an expansion of the starting kernel. As shown in Figure 3(d), initial kernels are surrounded of red dashed lines, and grow progressively (see green dashed lines in Figure 2(d)). The impact of nodes belonging to kernels, to the remaining of the entire graph, is null, because discovered communities are disjointed. We make use of a denoted *Key* variable which could be any pair/couple of vertices of *KDict*. In fact, each eligible *key* is integrated into a new kernel, after confirming its non-existence anywhere in the list of kernel vertices. This merging step improves performance and allows simplifying assumptions later when deciding whether to choose the favorite kernel by a *non-kernel* vertex (the kernel not belonging to a kernel).

The implementation for kernel is presented in Algorithm 2, which extracts a list of kernels named *ListK*, from the overall nodes of the graph. $standard\_deviation(I)$ is the function computing the standard deviation from the inter-class inertia vector I. *inkey* represents a boolean array of distinct nodes from $KDict'$ reflecting whether they are in a kernel or not. $neighbor(e)$ returns the other member of the pair of nodes defined by key in *KDict*, orderless.

Algorithm 2 Kernel extraction

```
Require: Directed graph $G = (V, E)$
Require: $I$ inter-class inertia vector //corresponding to the vector $I$ in the explanation //above.
Ensure: Structured-by-key Kernels set called $ListK$
Initialization : $\sigma \longleftarrow standard\_deviation(I), ListK \longleftarrow \emptyset$;
$\forall$ $distinct\ e \in KDict.key\_dict, inkey[e] = False$
$KDict' \longleftarrow KDict$ such as $KDict.value\_dict > \sigma$
while $\exists e \in KDict'.key\_dict/inkey[e] = False$ do
    if $inkey[neighbor(e)] = False$ then
        $Key \longleftarrow (e, neighbor(e))$
        $ListK \longleftarrow ListK \cup Key$
    else
        $K \longleftarrow K \cup \{e\}/neighbor(e) \in K$ and $K \in ListK$
    end if
    $inkey[e] \longleftarrow True$
    $inkey[neighbor(e)] \longleftarrow True$
end while
Return ListK
```

Let us illustrate this idea through an example considering the network in Figure 3(a). The standard deviation value for that network is $\sigma = 0.62$. It is the threshold on which kernels are to be built. The model computes the first kernel K_1 initialized by nodes "*DM*" and "*OW* " for which the associated inertia in I is $1.987 \geq 0.62$; thereafter, K_1 is extended by the node AK because AK is in the couple with the other nodes already assigned to kernels (See *KDict* in Section 3.2), with corresponding inertia of 1.359, 1.162 (See I list in Section 3.2.3); the second kernel K_2 is initialized by "*AG*" and "*BO*" for which the associated inertia in I is $1.705 \geq 0.62$. The process is repeated on the other i values in I for which $I[i] \geq \sigma$; and if the corresponding $KDict[i]$ pair nodes are already keys or associated values of keys, they are just omitted. Figure 3(d) shows in green dashed lines the kernels.

3.4. Community computing process

After extracting kernels, the other nodes not into the kernels, called *non-kernels* vertices, remain. The process of generating *global communities* (communities containing both kernels and non-kernels vertices) consists in migrating *non-kernels* vertices to the kernel with whom they have a maximal number of links, as defined in the Formula (7). It is an iterative optimization process of the number of connection each non-kernel vertex owns with the kernel.

Node Community Index (NCI). This measure we defined corresponds to an extent membership function. It consists in determining the membership of a vertex, according to its maximum number of "in-connections" (incoming edges) to the kernel or "out-connections" (outgoing edges from x) and the number of nodes in the kernel.

A vertex x migrates to kernel K if $K = argmax_l(NCI(x, K_l))$ where l is a kernel number; then NCI is defined as $NCI : V \times K \longrightarrow \mathbb{R}_+$ so:

$$NCI(x, K) = \frac{m_{out}(x, K)}{\min(m_{out}(x), n_K)}, \tag{6}$$

where $m_{out}(x, K)$ is the number of *outgoing* edges from x pointing to a kernel K, $m_{out}(x)$ is the total number of *outgoing* edges from x or its out-neighborhood cardinality, and n_K is the kernel size, i.e the number of vertices in the kernel. The pseudo-code of this migration approach is described in the following Algorithm 3. It presents in line 1 the initialization of Communities named G_i by their corresponding kernel K_i computed in the preceding kernel extraction step (see Section 3.3). From line 2 to line 6, the method computes for each non-kernel node its Node Community Index (NCI) and puts it in the kernel (or growing community) whose NCI is maximal. In line 7, results which are global communities (communities not growing, but definitely computed) are produced.

Algorithm 3 Algorithm for non-kernels vertices migration

```
Require: Communities Kernels $ListK = \{K_1, K_2, ..., K_t\}$
Require: $NonKernelSet = \{G.nodes \setminus \cup Ki\}$ //nodes $x$ of $G$ not belonging to any $K_i$
Ensure: Global Communities $G_K = \{G_1, G_2, ..., G_t\}$
1: $\forall i \in \{1, ..., t\}, G_i \longleftarrow K_i$
2: for $x \in$ NonKernelSet do
3:     Compute $NCI(x, G_i)$ for each $G_i$
4:     $G^* \longleftarrow argmax(NCI(x, G_i))$
5:     $G^* \longleftarrow G^* \cup \{x\}$
6: end for
7: Return $G_K$
```

Non-kernel vertices for the Figure 2(a) Network are listed below: *shallowend, abhubbu,ryzgo, 106andpark, 3atma, brycob, 303nomad, ritajohnsonn, BizPlanUSA*. The number of communities is visibly 2. We see in Figure 3(c) that Wang extracts the same partition as well as our model. Nevertheless, he sets the number of communities to detect. In other words, if its input on the number of communities was 1, his result would had been different from ours.

Complexity analysis: In view of the size of G with n the number of vertices and m the number of edges, the complexity is assessed according to each phase.

The first phase of constructing candidate kernels is assessed in 3 ways as shown in Section 3.2: Step 1 in Algorithm 1 computes a Centrality list CL. Assume that the length of

CL is $p = n{-}k$. The complexity of this sorted degree-based centrality list CL is $(p)\log(p)$. Step 2 in Algorithm 1 computes the kernel dictionary *KDict*. Its computation is assessed considering the right and left sides of the kernel degree measure K_{ij} : Given n_i and n_j the number of neighbors of nodes v_i and v_j respectively. The left side namely triad weight is assessed as follows: the numerator is the intersection of neighbors of nodes v_i and v_j. So the numerator complexity is $O(n_i + n_j)$. The denominator is $O(n)$, in the worst case. This worst case is reached when v_j get all of the other nodes $(n - 1)$ as neighbors. For p elements of CL, we will have $O(pn)$. The right side namely Jaccard index variant, possesses a complexity of $O(n_i + n_j)$. Thus the complexity of the sorted Kernel dictionary computation is $O(pn + nlogn)$. Step 3 computes the interclass inertia vector. Its complexity is $O(p^2)$. So the first phase of kernel candidate's generation is $pn + (p)\log(p)$ or $n^2 + (n)\log(n)$ in the worst case.

The second phase of kernel extraction namely Step 5 in Algorithm 1, is assessed as follows: given that KDict is pruned considering the threshold, and that its remaining elements are copied in *KDict*', let us assume that the size of *KDict*' is s, the number of distinct element; thus, to obtain kernels, we compare one element of *KDict*' to the other, so the complexity is $O(s^2)$. In the worst case when the number of nodes involved in pairs of *KDict*' is n, the complexity of Kernel extraction is $O(n^2)$.

The third phase based on migration of non-kernel nodes to kernels in order to constitute final communities (Step 6) is assessed as follows: Suppose that t is the number of kernels and L the number of non-kernel nodes. So the complexity will be $O(Lt)$. In the worst case, we have $(n - 2)$ non-kernel nodes with one kernel. Thus, complexity in the worst case is $O(n)$.

The global complexity of the proposed model is $O(n^2 + nlogn)$.

4. Empirical evaluation and experiments

In this section, we show experiment results. We assess a variety of models on two main tasks: Triad density of the partition and modularity evaluation. In order to evaluate kernels, the study of the *kernel degree* measure will be made on the illustration on Figure 3(a) network as shown in Figure 5, and tested through some criteria as described in Section 4.1 below; and the experiments will not focus on *kernel degree* metric, but on three criteria : partition *triad density* referenced by *TriadDens* defined through the formula 3, partition quality through *d-modularity* Q_d defined in Formula (1) and the number of communities each partition of experimented datasets get. Experiments were performed on a *DELL* Computer with *Windows* 8.1 OS 64 bytes, *Intel Pentium Core Duo CPU* of 4.2 Ghz and $7G_o$ of *RAM*. In terms of software, we used *Python* 3.7.1 for the implementation of our solution, *Gephi* 0.9.1[1] (Bastian and Hetmann 2009) and R[2] (Team, R. Core 2013) for graph structure visualization. We start by describing the four datasets from real-life graphs used in the experiments.

4.1. Datasets

In the following experiments, we use a neural network, a blog network and two paper citation networks. Information about each graph can be found in Table 1.

Celegansneural network. This is a weighted, directed network representing the neural network of C.Elegans. The weighted parameter is not taken into account in this work. There are 297 nodes and 2345 links. This dataset possesses 5 communities as obtained by Tianbao, Yun, and Shenghuo (2010).

Political Blog Network. This is a directed and unconnected network of hyperlinks between a set of weblogs about US politics. In this network, there is a total of 1490 nodes and 19, 090 links. Seeing that the new approach is based on connected networks, the largest connected subgraph with the highest number of links and nodes is the one taken into account throughout the execution of the approach.

Paper Citation Networks. We use the Cora paper citation network and the Citeseer paper citation network processed by Getoor et al[3]. There are 2708 nodes connected by 5429 links in Cora network, for 3327 nodes and 4732 links in Citeseer network.

The phenomenon described by these datasets follows a power-law in-degree distribution except the in-degree distribution in Cora network. The scatter plots for in-degree valuation of nodes are presented in Figure 6. In fact, a small number of vertices possess a high in-degree value, implying that a small amount of nodes have high quasi-common neighborhood cardinality, while larger nodes have less common neighbors. Yet, the in-degree in Cora dataset follows a rather uniform distribution with in-degree not larger than 5. We suspect such a distribution is due to the small scale of the Cora dataset which leads to many references, and therefore in-links, inside the dataset.

The goal of experiments is to demonstrate the influence of in-links emphasized by the method, as the numbers of authors quoting an article favors to delimit a topic area among a pioneer area (the center node or set of nodes). In other words, our goal is to evaluate if our new *kernel degree* based metric yields the link semantic of communities in directed networks, in accordance with triad-based community definition. The empirical evaluation of the new approach, to show its performance, is compared to some of the state-of-the-art methods: *Walktrap* (Pons and Latapy 2005), *Edge Betweenness* (Newman 2004), *Label Propagation* (Raghavan and Albert 2007), *Louvain* (Blondel et al. 2008) and *CDLPA* (Li 2019). The latter was assessed on the Cora dataset exclusively, as the authors considered this dataset in their study.

Illustration from Figure 3(a) *network*. To illustrate the results of our approach, based on Figure 3(a) Network, Table 4 shows some results and compares them to kernel degree-based approach. Thus the kernel degree approach and Walktrap method present the same results on the triad density and the modularity with the same number of communities, contrary to Louvain method, although detecting 2 communities, computes a modularity of 0.395. Label and Edge Betweenness methods compute respectively 5 and 7 communities with lowest triad density and modularity values. Visibly, our approach extracts expected structures better on this Figure 3(a) illustration, than the other methods, as pointed up in Figure 4.

Table 1. Characteristics of the test graphs.

Networks	Nodes	edges	Comm
Celegansneural	297	2345	5
Polblogs	1,490	19,090	–
Citeseer	3,327	4,732	–
Cora	2708	5,429	–

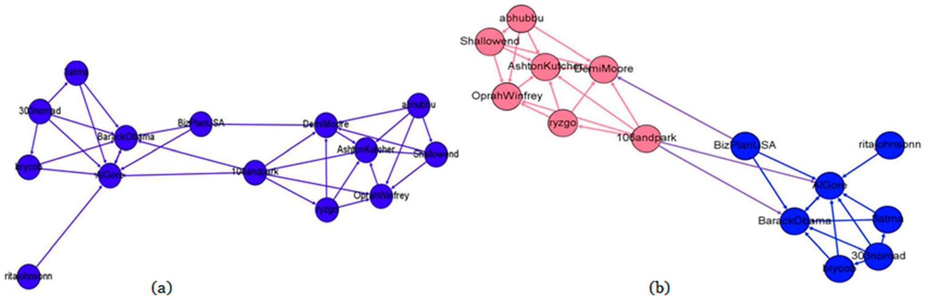


Figure 4. Graphical visualization of structures obtained from the network in Figure 3(a) when the threshold is descent or ascent. (a) Descent threshold ($I[eij] < \sigma$) leads to one community. (b) Ascent threshold ($I[eij] > \sigma$) leads to two communities.

Table 2. Using metric comparison.

	Figure 3(a) Network		Celegansneural	
Metric	#Comm	*TriadDens*	#Comm	*TriadDens*
Kernel-degree	2	0.64	5	0.711
Neighborhood overlap	2	0.64	91	0.20
Triad weight	2	0.64	73	0.254

Table 3. σ choice evaluation.

	Figure 3(a) Network		Celegansneural	
Inertia Criteria	#Comm	*TriadDens*	#Comm	*TriadDens*
$I[e_{ij}] > \sigma$	2	0.6428	5	0.711
$I[e_{ij}] < \sigma$	1	0.417	103	0.065

4.2. Kernel degree metric and threshold σ evaluation

4.2.1. Kernel degree metric evaluation

To appreciate the powerfulness of the *kernel degree* formula, let us consider two networks namely Figure 3(a) Network and Celegansneural network for better results' visualization. *kernel degree* computes the similarity strength between kernel vertices; in other words, it determines the kernel power. Both *Triad Weight* and *Neighborhood Overlap* (Definition 3.2) are associated to reinforce this similarity, because, when taken separately, the expected results are not obtained, as presented in the Table 2. In fact, for the Figure 3(a) Network, results are the same regardless of the criteria (2 communities with the same triad density and same modularity). But for the Celegansneural network, using separately *Neighborhood Overlap* or *Triad Weight* leads to results (91 and 73 communities respectively) far from expected one as demonstrated by Klymko, Gleich, and Kolda (2014), Tianbao, Yun, and Shenghuo (2010) who detect 5 communities. Furthermore, taken separately, they lead to a computation of weak values of triad density, contrary to the new composite *kernel degree* metric which computes a better triad density of 0.711, close to the triad density value of 0.78 obtained by Klymko.

Table 4. Community detection performance where the best performances are in bold.

Datasets	Methods	TriadDens	Modularity	#of Communities
Twitter network	Edge-Betweenness	0.0857	0.187	7
	Walktrap	**0.6428**	**0**.410	2
	Label Propagation	0.34	0.306	5
	Louvain	**0**.6428	0.395	2
	Kernel Approach	**0.6428**	**0.410**	2
Celegans Neural	Edge-Betweenness	0.0004	0.081	194
	Walktrap	0.0458	0.363	21
	Label Propagation	0.0135	0.0027	29
	Louvain	0.608	0.379	6
	Kernel Approach	**0.711**	**0.393**	5
Polblogs	Edge-Betweenness	0.0064	0.1872	55
	Walktrap	**0.67**	**0.4302**	12
	Label Propagation	0.0026	0.386	244
	Louvain	0.0085	0.427	274
	Kernel Approach	**0.5732**	**0.429**	34
Citeseer	Edge-Betweenness	0.0	0.5344	738
	Walktrap	0.0	0.811	593
	Label Propagation	0.0	0.491	842
	Louvain	0.079	0.886	466
	Kernel Approach	**0.407**	**0.8907**	121
Cora	Edge-Betweenness	0.0516	0.3999	1028
	Walktrap	0.2131	0.756	265
	Label Propagation	–	–	–
	Louvain	**0.313**	**0.808**	100
	Kernel Approach	0.0853	0.212	1107
	CDLPA	–	0.6042	–

4.2.2. Threshold σ evaluation

As far as the threshold σ is concerned, the empirical experiments show that when taking descent values of the interclass inertia, meaning those less than σ, expected results are not obtained. For illustration, as seen from the Table 3, our approach performs the best in both datasets. The Figure 3(a) Network, for the first case $(I[e_{ij}] > \sigma)$ contains 2 communities with a triad density of 0.6428, and Celegansneural 5 communities with a high value of triad density equals to 0.711; contrary to the second case $(I[e_{ij}] < \sigma)$ for which Figure 3(a) Network just contains 1 community with a low triad density of 0.417 and Celegansneural 103 communities with 0.065 triad density value. This result means that the Figure 3(a) Network partition is not well structured for this second case. Figure 4 illustrates the comparison of these both σ considerations. Higher inter-class inertia values indicate better kernel-based triad structures and therefore, finding vertices with similar neighbors whose inter-class inertia values are upper than threshold provides a method for extracting the underlying kernel structure. The Figure 5 shows the analysis made on the idea that the more the inter-class inertia is upper than a threshold σ, the more the kernel degree values are large, meaning better triad-based structures.

4.3. Performance on community detection

4.3.1. Quality measure evaluation

The community detection performances for different models on the four datasets are given in Table 4. The Figure 3(a) Network contains visibly 2 communities as shown on

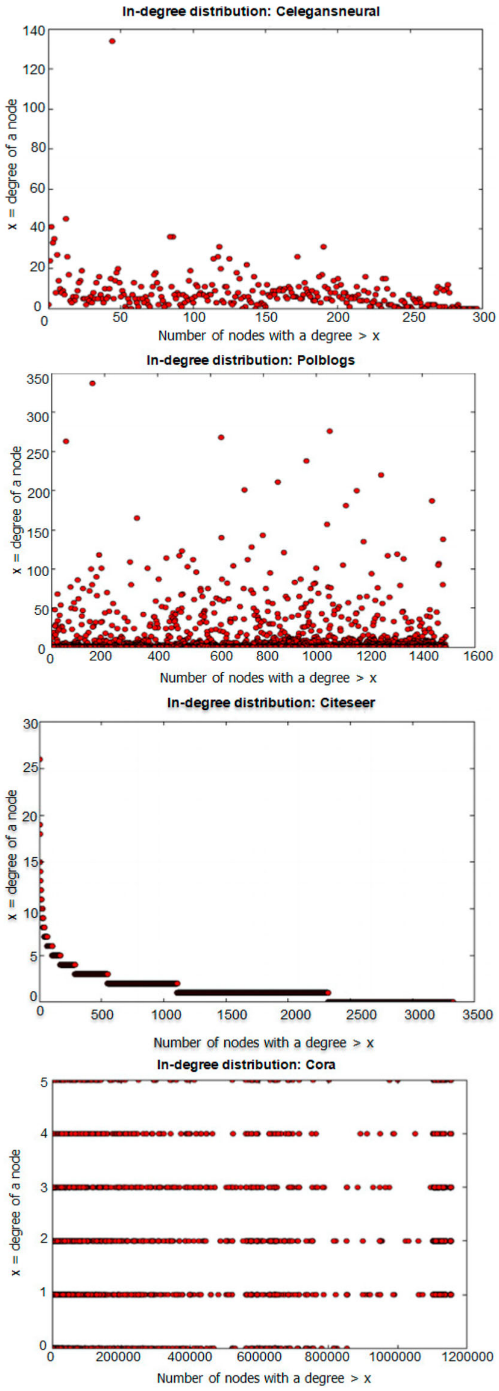


Figure 5. In-degree distribution on dataset nodes.

the Figure 4; Celegansneural network is used to illustrate the new approach methodology and its hidden idea. With this dataset, both the expected number of communities and *TriadDens* metrics are evaluated.

As shown in Table 4, Edge Betweenness approach focus on links between nodes by searching the central edge (geodesic) meaning the bridge linking two communities. It detects 194 communities, with the weakest *TriadDens* (0.0857). Contrary to the other models, the Kernel approach confirms the 5 communities detected by Klymko, Gleich, and Kolda (2014), with a higher *TriadDens* (0.711) close to the triad density (0.78) of Klymko; likewise, the higher modularity (see 0.393) proves its performance on the partition quality.

As far as Label Propagation method is concerned, a node moves from one community to another if, its neighbors share the same label. Hence, for the polblogs network, it computes the high community number (244) with the weakest *TriadDens* (0.0026). Walktrap and Kernel approach methods perform in all of the criteria: they produce small communities (12 and 34 respectively), with high *TriadDens* of (0.67 and 0.5732 respectively) and the best modularity value of 0.4302 and 0.429 respectively as shown in the Table 4. This result indicates that models of "what is a growing-community" are somehow in agreement with the notion of Kernel-degree measure. But Walktrap performs the best since it considers unconnected partitions, indicating that it captures the so-called outliers by Ester, Kriegel, and Sander (1996), which are anomalous nodes (belonging to none of the communities). Meanwhile, Louvain method results are not so interesting with a *TriadDens* of 0.0085. This result could be due to the fact that Louvain's method stresses on the modularity optimization. Indeed, this measure does not implement the higher consideration of nodes with higher incoming edges and weak outcoming edges than the opposite.

In the Table 4, through results presented for Citeseer dataset, Kernel approach improves values of modularity and triad density. As shown in Figure 6, more than 80% of the nodes have a degree between 0 and 3 and the remaining nodes have a degree between 4 and 25. Since the majority of the nodes have such a low degree, it means that the method will produce few kernels, and therefore few communities. In other words, the resulting structure will have more followers than leaders, more citations than articles containing them. This behavior reflects the reality insofar as for 2 articles, one could have about 50 articles in the bibliography. According to Figure 1 which presents the triads considered in our approach, we can deduce that it is quite normal that the proposed method produces this high value of triad density, compared to the low values obtained by the other methods. Moreover, the value of modularity obtained by our approach is not very far from Louvain because of link density that the latter takes into account.

The null modularity values for the other methods in the Table 4 illustrates better type of scorpus that our method performs on. In fact, contrary to the other datasets, Citeseer follows the deepest power law distribution, because it possesses a hub node (node with a higher degree distant from the other nodes degrees), as presented in Citeseer subfigure of Figure 6. Tsourakakis (2008) confirms the plausibility of these results by its argumentation that low-degree nodes form fewer triangles than higher degree nodes; and according to Durak, Pinar, and Kolda (2012), citation networks are dominated by heterogeneous triangles; like this, triads are included into triangle. So results on Citeseer, a citation network type, seem to be valid in regard of both precedent demonstrations.

For Cora network, results shown in Table 4 guarantee that the new scheme is based on power-law distribution in datasets. Indeed, since Cora follows a uniform in-degree

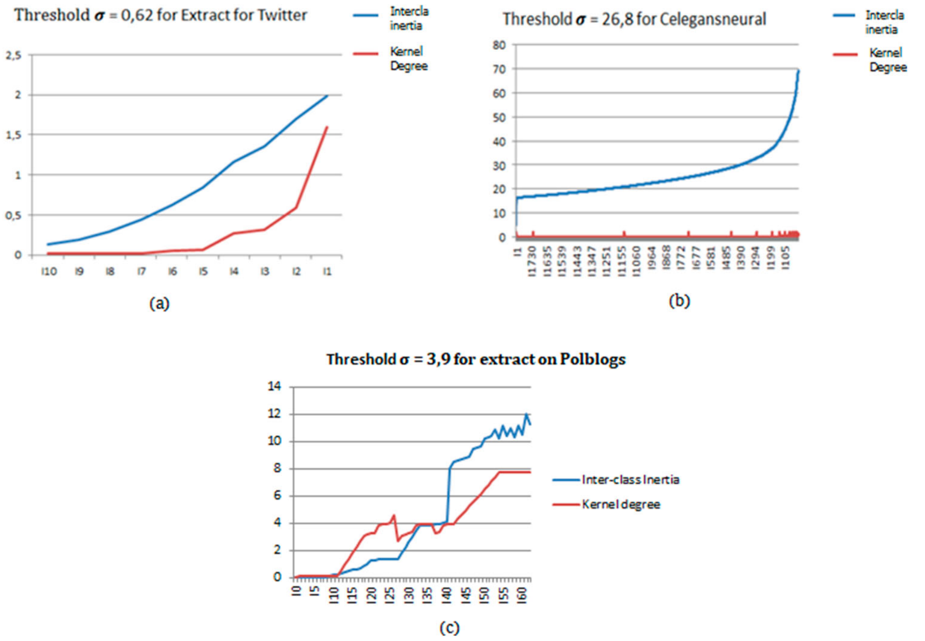


Figure 6. Standard deviation distribution. (a) Threshold based on the network in Figure 3(a). (b) Threshold based on Celegansneural network. (c) Threshold based on Extract from Polblog network.

distribution as shown in Figure 6, the kernel degree-based approach produces weak results; label propagation method fails on this dataset, indicating the fact that nodes already belong to communities containing their whole neighbors, thus most of these vertices do not need to move from one community to another; Louvain's method performs the best. This result is due to the fact that it is based on density of links disregarding the benefit of the node in-degree. Moreover, CDLPA performs better as it consists in balancing structures to reach a computed size. Figure 7 compares the values of modularity on the partitions obtained by each of the methods on the different data sets while Figure 8 shows a comparison of the triad density values obtained by each method.

Summarily, these weak results for Louvain method compared with the proposed approach on the overall of datasets indicate that it focuses solely on link density in the community without no interest of the topology or in-link based semantic of triads into the communities. The Kernel degree-based approach performs the best in all the cases except on the triad density for Cora network. These results also illustrate that most of the time, it is beneficial to use both triad weight and neighborhood overlap measures simultaneously, establishing *kernel degree* formula, to valorize the similarity kernel vertices in a directed network.

4.3.2. Number of communities

An efficient report made from table of results is that the more the number of communities is low the more triad density and modularity values are great. Indeed, the proposed approach shows that the number of communities depends on the depth of the power law distribution.

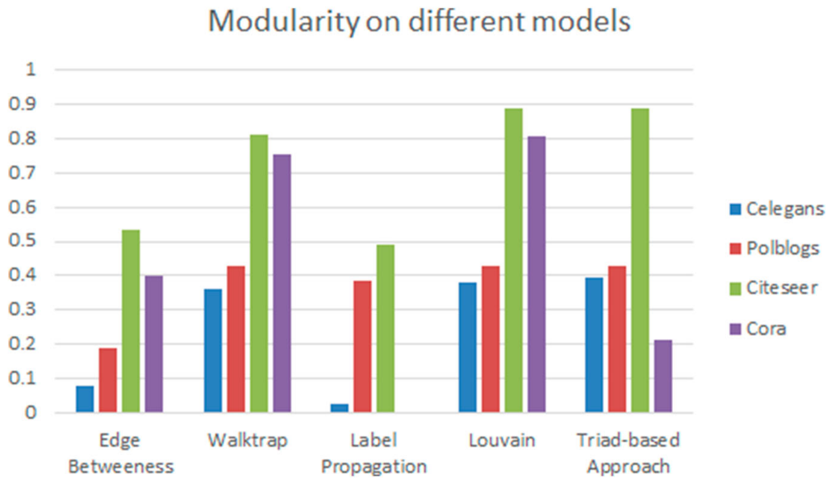


Figure 7. Link density (modularity) as a criteria of network type.

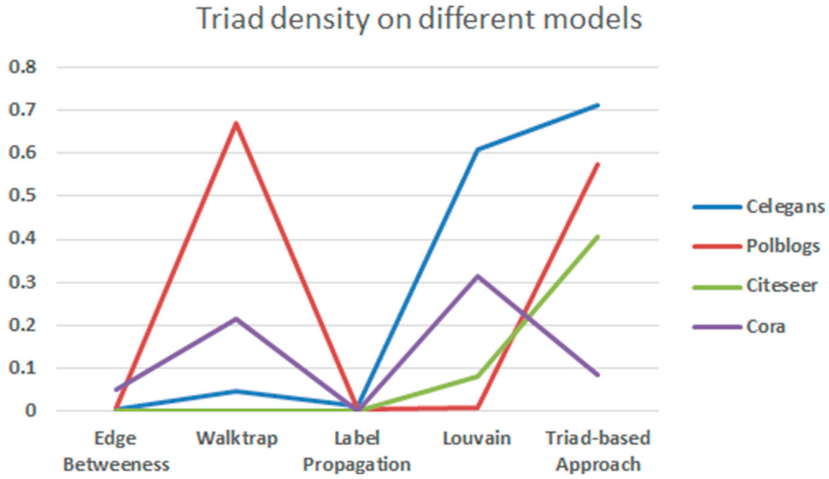


Figure 8. Triad density as a metric of partition evaluation.

The deeper this distribution is (case of Citeseer, Celegans), the fewer communities there are. In our future studies, we will show the relative effects of metrics on the number of communities or vice versa.

5. Conclusion and future work

This paper has described a simple kernel scheme to improve the detection of communities in directed networks, through triad density. It focuses on kernels which are seed nodes centralizing information through their in-degree valuation. Based on the definition of community as a subgraph induced by kernels, the new scheme basis are triads relationships between kernel nodes and their neighbors. Thus, we have defined a metric called *kernel degree*, for computing the similarity between kernel nodes. When the new metric is used, we obtain better triad density and modularity values on some datasets, those following the power-law degree distribution of nodes. Our model captures a significance of communities based on both criteria: density of links and topology of vertices in the graph, meaning communities with higher triad density. We compared the modularity values for each model on the result partition, and we found a substantial improvement in the triad density measure, with appreciable changes in the traditional community detection metrics such as modularity.

The model complexity constitutes a main criteria of effectiveness for any method. With the increasing ways on information access in the era of digital, it becomes important to extend this method to parallel processing, in order to manipulate very large-scale real networks. Also, it is possible to apply weighted graphs to reinforce the strength of the kernel, for community detection results more similar to the real life. We will explore this in our future works.

Notes

1. http://www.aaai.org/ocs/index.php/ICWSM/09/paper/view/154
2. https://www.R-project.org
3. http://www.cs.umd.edu/projects/linqs/projects/lbc/

Disclosure statement

No potential conflict of interest was reported by the authors.

Notes on contributors

Félicité Gamgne Domgue is a PhD student at the Department of Computer Science of the University of Yaounde I. In her research she studies social network analysis and graph mining in order to detect communities.

Norbert Tsopze is a senior lecturer at the Department of Computer Science of the University of Yaounde I and member of the local UMMISCO research team. His research interests include datamining, formal concept analysis, neural network, deep learning, text analysis, social network analysis, classification. He is also director of many Master and PhD students.

Réné Ndoundam is an Associate Professor of Computer Science at Department of Computer Science of University of Yaoundé 1. His interest area of research includes: Automata, Complexity, Steganography and Recommandation systems, graph theory.

References

Bastian, M., S. Hetmann, and M. Jacomy. 2009. “Gephi: An Open Source Software for Exploring and Manipulating Networks” Proceedings of the Third International ICWSM Conference. DOI: 10.13140/2.1.1341.1520

Blondel, V. D., J. L. Guillaume, R. Lambiotte, and E. Lefebvre. 2008. “Fast Unfolding of Communities in Large Networks.” *Journal of Statistical Mechanics: Theory and Experiment* 10: P10008.

Brandes, U., D. Delling, M. Gaertler, R. Gorke, M. Hoefer, Z. Nikoloski, and D. Wagner. 2008. “On Modularity Clustering.” *IEEE Transactions on Knowledge and Data Engineering* 20: 172–188. doi:10.1109/TKDE.2007.190689.

Clemente, G. P., and R. Grassi. 2018. “Directed Clustering in Weighted Networks: A New Perspective.” *Chaos, Solitons and Fractals* 107: 26–38.

Dž̌amić, D., D. Aloise, and N. Mladenovic. 2019. “Ascent–Descent Variable Neighborhood Decomposition Search for Community Detection by Modularity Maximization.” *Annals of Operations Research* 272: 1–15. doi:10.1007/s10479-017-2553-9.

Durak, N., A. Pinar, and T. Kolda. 2012. “Degree Relations of Triangles in Real-world Networks and Graph Models.” *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* 96: 1712–1716.

Ester, M., H. Kriegel, and J. Sander. 1996. “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise.” *Computer Communications* 96: 226–231.

Fortunato, S.. 2010. “Community Detection in Graphs.” *Physics Reports* 486 (3): 75–174.

Kanawati, S. R.. 2014. “Seed-Centric Approaches For Community Detection In Complex Networks.” *In International Conference on Social Computing and Social Media*, 197–208. Springer International.

Kim, Y., S. W. Son, and H. Jeong. 2010. “Finding Communities in Directed Networks.” *Physical Review E* 81 (1).

Klymko, C., D. F. Gleich, and T. G. Kolda. 2014. “Using Triangles to Improve Community Detection in Directed Networks.” *Conference Stanford University*.

Krings, G., and V. D. Blondel. 2011. An Upper Bound on Community Size in Scalable Community Detection.” arXiv preprint:1103.5569.

Lancichinetti, A., and S. Fortunato. 2009. “Community Detection Algorithms: A Comparative Analysis.” *Physical Review E* 80 (5): 056–117.

Latapy, M., C. Magnien, and N. Del Vecchio. 2008. “Basic Notions for the Analysis of Large Two-mode Networks.” *Social Networks* 30 (1): 31–48.

Leskovec, J., J. Kleinberg, and C. Faloutsos. 2005. “Graphs Over Time: Densification Laws, Shrinking Diameters and Possible Explanations.” *In Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*. 177–187.

Li, X. 2019. “Directed LPA: Propagating Labels in Directed Networks.” *Physics Letters A* 383 (8): 732–737.

Maliaros, F. D.. 2013. “Clustering and Community Detection in Directed Networks: A Survey.” arXiv 1308.0971.

Newman, M. E.. 2004. “Detecting Community Structure in Networks.” *The European Physical Journal B-Condensed Matter and Complex Systems* 38 (2): 321–330.

Nicosia, V., G. Mangioni, and M. Malgeri. 2009. “Extending the Definition of Modularity to Directed Graphs With Overlapping Communities.” *Journal of Statistical Mechanics: Theory and Experiment*.

Pang-Ning, T., M. Steinbach, and V. Kumar. 2005. *Introduction to Data Mining*. ISBN 0-321-32136-7.

Pons, P., and M. Latapy. 2005. “Computing Communities in Large Networks Using Random Walks.” *International Symposium on Computer and Information Sciences* 284–293.

Raghavan, U. N., and R. Albert. 2007. “Near Linear Time Algorithm to Detect Community Structures in Large-scale Networks.” *Physical Review E* 96: 36–106.

Santiago, R., and L. C. Lamb. 2017. “Efficient Modularity Density Heuristics for Large Graphs.” *European Journal of Operational Research* 258 (3): 844–865.

Satuluri, V., and S. Parthasarathy. 2011. “Symmetrizations for Clustering Directed Graphs.” *In Proceedings of the 14th International Conference on Extending Database Technology*. 343–354. ACM.

Serrour, B., and S. Arenas. 2011. “Detecting Communities of Triangles in Complex Networks Using Spectral Optimization.” *Computer Communications* 34: 629–634.

Shaojie, Q., H. Nan, G. Yunjun, L. Rong-Hua, H. Jianbin, G. Jun, A. G. Louis, and W. Xindong. 2018. “A Fast Parallel Community Discovery Model on Complex Networks Through Approximate Optimization.” *IEEE Transactions on Knowledge and Data Engineering* 30 (09): 1638–1651.

Souravlas, S., A. Sifaleras, and S. Katsavounis. 2019. “A parallel algorithm for community detection in social networks, based on path analysis and threaded binary trees.” *IEEE Access* 7 (1): 20 499–20 519.

Steinhaeuser, K., and N. Chawla. 2008. “Community Detection in a Large Real-world Social Network.” *In Social Computing, Behavioral Modeling, and Prediction* 42: 168–175.

Steven, L., and M. Martin. 2016. “Close Communities in Social Networks: Boroughs and 2-clubs.” *Social Network Analysis Mining* 6: 20:1–20:16.

Team, R. Core. 2013. “R: A Language and Environment for Statistical Computing.” R Foundation for Statistical Computing.

Tianbao, Y., C. Yun, and Z. Shenghuo. 2010. “Directed Network Community Detection: A Popularity and Productivity Link Model.” *In SIAM Data Mining'10*.

Tsourakakis, C. E.. 2008. “Fast Counting of Triangles in Large Real Networks Without Counting: Algorithms and Laws.” *In 2008 Eighth IEEE International Conference on Data Mining*. 784–793.

Wang, L.. 2011. “Detecting Community Kernels in Large Social Networks.” *IEEE 11th International Conference on Data Mining*. 608–617.

Wasserman, S., and F. Katerin. 1994. “Social Networks Analysis : Methods and Applications.” *Physics Reports* 486: 75–174.

Xu, Y., H. Xu, and D. Zhang. 2015. “A Novel Disjoint Community Detection Algorithm for Social Networks Based on Backbone Degree and Expansion.” *Expert Systems with Applications* 42: 8349–8360.

Zhou, D., T. Hofman, and B. Schlkopf. 2005. “Semi-Supervised Learning on Directed Graphs.” *In Advances in Neural Information Processing Systems (NIPS)*, pp. 1633–1640.

C.2 Multidimensional networks : A novel node centrality metric based on common neighborhood

Multidimensional networks

A novel node centrality metric based on common neighborhood

Gamgne Domgue Félicité, Tsopze Norbert, Ndoundam René

Sorbonne University, IRD, UMMISCO, F-93143, Bondy, France,
University of Yaounde I
Yaounde,Cameroon
felice.gamgne@gmail.com, felicite.gamgne@uy1.uninet.cm
tsopze@uy1.uninet.cm
ndoundam@gmail.com

ABSTRACT. Complex networks have been receiving increasing attention by the scientific community. They can be represented by multidimensional networks in which there is multiple types of connections between nodes. Thanks also to the increasing availability of analytical measures that have been extended in order to describe and analyze properties of entities involved in this kind of multiple relationship representation of networks. These measures focused on quantitative involvement of node through the widely popular and intuitive measure of degree. However, one aspect of such properties have been disregarded so far: entities in such networks are often tied according to the interest they have to their neighbors in the overall dimensions. In this paper, the problem of characterizing multidimensional networks, using a *qualitative* aspect of the node neighborhood, has been studied, through the new defined node centrality measure, *Stability*, to describe the connectivity of nodes that incorporates across-dimension topological features in order to identify the relevant dimensions. We assessed our measure on two real-world multidimensional networks, showing its validity, its meaningfulness and its correlation with a dimension connectivity measure.

RÉSUMÉ. Les réseaux complexes ont reçu beaucoup d'attention de la part de la recherche scientifique. Ils peuvent etre représentés par des réseaux multidimensionnels dans lesquels il existe plusieurs types de relations entre les entités. Plusieurs propriétés decrivant les noeuds et permettant l'extraction de la connaissance sur de tels réseaux, ont été étudiées. La majorité d'entre elles prone l'aspect quantitatif de la connectivité d'un noeud, à l'instar de la centralité de degré. Cependant, un aspect primordial a été omis: celui *qualitatif*, basé sur le type de voisinage d'un noeud. En effet, dans de tels réseaux, les entités sont généralement connectées selon les mêmes centres d'intérêts qu'ils possèdent. Dans ce travail, le problème de caractérisation des réseaux multidimensionnels moyennant l'utilisation de la notion qualitative du voisinage d'un noeud, a été abordé à travers la définition d'une nouvelle mesure de centralité appelée *Stabilité*. Cette dernière permet de décrire la connectivité des noeuds basée sur les caractéristiques topologiques, en vu de déterminer les dimensions pertinentes. L'évaluation de cette mesure s'effectue sur deux réseaux multidimensionnels réels, et montre sa validité et sa correlation à une mesure de connectivité de dimensions.

KEYWORDS : Multidimensionnal networks, Centrality measure, Relevance dimension

MOTS-CLÉS : Reseaux multidimensionnels, mesure de centralité, Pertinence de dimension

1. Introduction

Complex network analysis has received a lot of attention by the scientific researchers, because it helps to better understand the intrinsic behavior of relationships between entities. These relationships could be either of one or several types. Unlike a *monodimensional* network which contains only one type of links between nodes, multidimensional networks contain links which either reflect different kinds of relationship or represent different values of the same kind of relationship among a same set of elementary components. This flexibility allowed to use complex networks to study real-world systems in many fields: sociology, physics, genetics, computer, etc. Such systems can be modeled by multidimensional networks as reported in Figure 1 [1] where on the left we have different types of links, while on the right we have different values (conferences) for one relationship (for example, co-authorship). Multidisciplinary and extensive research works have been devoted to the extraction of non trivial knowledge from such networks [7]. Some of them focused on the characterization of their properties. More precisely, they studied some centrality measurements based on the quantitative neighborhood also called "weight", of each node [1, 6]. These measures, which are certainly relevant, do not take into account a more recent reality. Indeed, with the advent of the Internet and social networking sites, individuals communicate more easily when the majority of their contacts use the same platforms or means of communication as they do. Thus, the qualitative aspect of this neighborhood is important to be considered. From this aspect, a semantics emerges relating to the retention of the same neighbors of a node over all dimensions, namely *Stability*. To the best of our knowledge, however, the literature still misses a systematic qualitative measure for weight-based centrality in the context of correlated multidimensional networks, together with a model of extracting relevant dimensions for each node. The aim of this paper is precisely defining a basic and analytical concept of centrality measure, which takes into account the connectivity redundancy of nodes among dimensions. As questioned in [1], how is it possible to contribute to answering the question *To what extent one or more dimensions are more important than others for the connectivity of a node?*

Contributions: In this work:

– We introduce a novel centrality weighting scheme of nodes called *stability*, in multidimensional network

– We formally define a measure aimed at extracting useful knowledge on relevance dimensions of nodes

– We characterize nodes of the multidimensional network according to their stability

– we empirically test the meaningfulness of our measure, by means of a case study on two realistic networks.

The rest of the paper is organized as follows. Section 2 overviews related works, Section 3 describes the proposed measures to assess the activity level of a node in a dimension. Section 4 presents experimental evaluation, Section 5 concludes the paper.

2. Related works

Multidimensional networks have for a long time been proposed as an alternative to better describe interactions within complex systems [1]. For instance, in social networks, individuals can be connected according to different social ties, such as friendship or family relationship [2]. The extraction of knowledge and analysis of both the local and global properties of such networks remains of interest to scientists. Indeed, multidimensional networks abound with a large amount of information, particularly concerning the various kinds of relationship between entities. Since an individual may have a particular interest for a certain number of dimensions, he could be influential or important in regard to other nodes in these dimensions: they are then qualified as central. So ignoring centrality in multidimensional structures can lead to different ranking results than what one obtains for multidimensional networks [8] .

Centrality, an indicator that quantifies the importance of nodes in a network, comes from the discipline of Social network analysis and has become a fundamental concept in network science with its applications in a range of disciplines. In recent works, many efforts have been devoted to "centrality" measures in order that they are also applicable in multidimensional networks [8]. Examples of these various centrality measures include degree centrality, called overlapping degree in [6]. As they help to extract a knowledge and analyze the network properties related to the questioning of “how important a dimension for a node is”, these measures are based on both relevance dimension and dimension connectivity, since nodes could exist across all dimensions. A multigraph used to model a multidimensional network is denoted by a triple $G = (V, E, L)$ where: V is a set of nodes; L is a set of dimensions; E is a set of edges, i.e the set of triples (u, v, d) where $u, v \in V$ are nodes and $d \in L$ is a dimension. Thus, Berlingerio [1] defined a relevance dimension measure based on the connectivity of dimensions, as described in Equation 1, which computes the fraction of neighbors directly reachable from node v following edges belonging only to the set of dimensions called D with $D \subseteq L$. Likewise, he defined a measure Node Exclusive Dimension Connectivity ($NEDC$) computing the ratio of nodes belonging only to a specific dimension d, as described in Equation 2.

$$DR_{XOR}(v, D) = \frac{|Neighbors_{XOR}(v, D)|}{|Neighbors(v, L)|} \tag{1}$$

$$NEDC(d) = \frac{|u \in V | \exists v \in V : (u, v, d) \in E \wedge \forall j \in L, j \neq d : (u, v, j) \notin E|}{|u \in V | \exists v \in V : (u, v, d) \in E|} \tag{2}$$

where $Neighbors_{XOR}(v, D)$ is the set of neighbors of v belonging only to dimensions D. Despite their popularity and effectiveness in social network analysis, we believe to our knowledge that these measures mainly take into account the quantitative aspect (i.e. degree) of node properties and links, yet the *qualitative* aspect, namely the type of neighboring nodes, would have a significant impact on facilitating communication between an individual and his neighborhood.

Indeed, these measurements focus only on the degree of nodes regardless of the type of neighborhood, to extract the relevant dimensions. According to them, a relevant dimension for a node is quantified by the density of its neighborhood. Thus, if a node has the same number of neighbors on all dimensions, then all these dimensions will be relevant to it. However, in real life situations relating to human relationships, the communication is more obvious, more easy or cheaper among individuals using the same platform of information exchanges. So, the interest a user has for a platform depends on the subscription of his friends to that platform. Therefore, a dimension(platform) would be more relevant for a node(subscriber) if the node has a conservative behavior of its neighborhood over all dimensions. It is this concept that we implement in the next section.

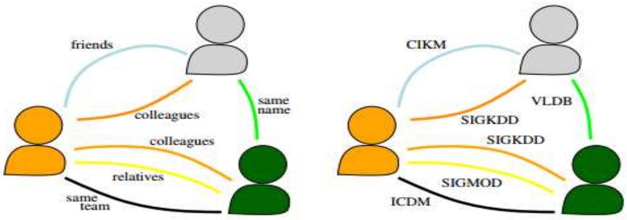


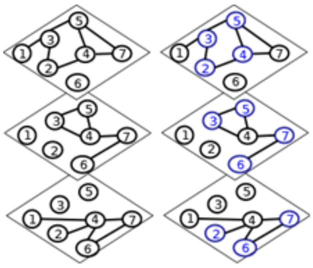
Figure 1: Example of multidimensional networks

3. Multidimensional network analysis

This paragraph presents new defined measures to contribute to knowledge extraction from a multidimensional networks. They concern whether a dimension can be of interest for a node, based on the stability of his neighbors. In the first subsection, the stability centrality is defined, and in the second subsection, we present how to determine the relevant dimensions of a node.

3.1. Stability centrality

The weights of the nodes have been the subject of some studies. According to graph theory, this weight corresponds to the sum of the weights of the edges incident to this node. It is a variant of the degree centrality. This measure shows that the importance of a node depends on the number of communications it establishes with its neighborhood. It corresponds to its activity level in a network. Extending this measure in multidimensional networks, Nicosia et al. [6] studied that the activity of a node in a particular dimension is very often correlated with its activity in another dimension. The authors considered the centrality degree as a measure of the node activity in a dimension. However, the number of neighbors seems to be meaningless when studying behavior of entities in a context of correlated dimensions. Then it becomes necessary to maintain the stable behavior of a node in order to make easy information exchange among its community membership. The stability centrality of a node u is then pointed up and computes the proportion of the common neighborhood of this node between two dimensions p and q, through a *Jaccard index* similarity as defined through Definition 1. It takes into account the structural features across several dimensions.



	1	2	3	4	5	6	7
Dim1	0.0	0.25	0.125	0.45	0.33	0.0	0.33
Dim2	0.0	0.0	0.125	0.33	0.33	0.25	0.66
Dim3	0.0	0.25	0.0	0.28	0.0	0.25	0.66

	1	3	2	6	5	4	7
ε	0.0	0.083	0.166	0.166	0.22	0.35	0.55

Figure 2: An example of connected multidimensional network with 3 dimensions on the left, and on the right, the stability node on the top and threshold for relevance dimension extraction below.

Definition 1 ***(The stability centrality of a node in a dimension)****. Stability centrality of node u in dimension q measures the common neighborhood of a node between q and the other dimensions. The function $Stability : V \times D \rightarrow [0, 1]$ is defined as:*

$$Stability(u, q) = \frac{1}{ndim - 1} \sum_{p=0}^{ndim-1} \frac{\mid \Gamma_u^p \cap \Gamma_u^q \mid}{\mid \Gamma_u^p \cup \Gamma_u^q \mid} \tag{3}$$

whereΓ_u^p denotes the neighborhood of node u in the dimension p and $ndim$ denotes the number of dimensions. We refer to *disassortative stability* when its neighborhood is totally different in all dimensions; Stability tends to be null. Otherwise, it is the *assortative stability*; it tends to its maximal value 1. In this paper, the node with the lowest disassortative stability is unstable and the one with the highest assortative stability is the most stable over the network. As shown in table of Figure 2 above, node 1 possesses a disassortative stability, unlike node 7 which gets an assortative stability.

3.2. Relevance dimension

The concept of dimension relevance of a node studied in [5] stresses on that dimension in which the node has the most important exclusive degree as defined by Berlingerio [1] i.e. it computes the fraction of neighbors directly reachable from node u following edges belonging exclusively to a subset of dimensions D_l as shown in Equation 1. This way does not seem relevant in some real situations, because if a node has the same degree on all dimensions of the network, then all of them will be relevant. Yet, if we consider only those in which the node has a more stable neighborhood, the relevance of the dimensions would be more semantic. The relevant dimensions $RD(u)$ of a node u refers to those dimensions for which the node has a stability centrality greater than or equal to a certain threshold ε. It is described by the function $RD : V \rightarrow D$ as:

$$RD(u) = \{q, |Stability(u, q) \geq \varepsilon\} \tag{4}$$

The threshold ε is defined in the Equation 5. When the node u has a stability centrality whose value is higher than ε, it is said that the node u *is stable for the subset of dimensions* $RD(u)$, or that the *dimensions in the subset* $RD(u)$ *are relevant for the node* u.

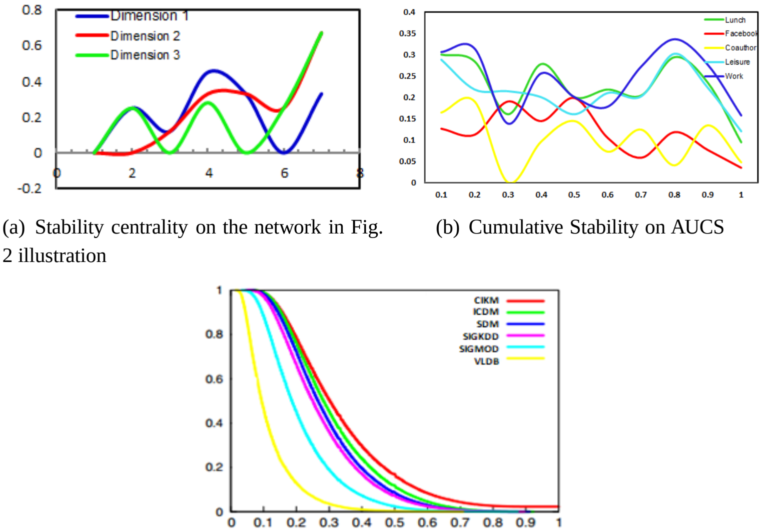
$$\varepsilon = \frac{1}{|D|} \sum_{i=1}^{|D|} Stability(u, i) \tag{5}$$

4. Experimental Evaluation

In this section, we assess the proposed metric on two main sights: its correlation with dimension connectivity and the behavior of nodes according to the values of the metric.

4.1. Correlation with dimension connectivity

This section reports the results obtained by computing the stability measure on two real-world multidimensional network datasets, namely AUCS [4] and DBLP [3]. AUCS, an attributed multidimensional network, models relationships between 61 employees of Aarhus University Computer Science department considering five different aspects: coworking, having lunch together, Facebook friendship, offline friendship, and coauthorship. In



(a) Stability centrality on the network in Fig. 2 illustration

(b) Cumulative Stability on AUCS

(c) Cumulative Stability on DBLP

Figure 3: Stability centrality distribution

DBLP, there are 83901 nodes which correspond to authors, tied by 159302 links, and 50 dimensions represent the top-50 Computer Science conferences. Two authors are connected on a dimension if they co-authored at least two papers together in a particular conference. All the experiments were conducted on an Intel Core $i5 - 8250U$ CPU $@1.60GH_z$, $8GB$ of RAM machine, Windows 10 OS 64 bytes.

Figure 3 reports the cumulative distribution of the stability measure. It denotes the average of nodes' stability on 10 intervals. The latter corresponds to the normalization of the number of nodes. Figure 3(a) is a small dataset. Then there is no need to cumulate the stability centrality of the nodes, unlike the figures 3(b) and 3(c) whose size is important, leading to a normalization of their x-axis. Berlingerio in [1] analyzed the correlation between the DR_{xor} distribution and the Dimension Connectivity values (especially $NEDC$). The authors deduced that DR_{xor} measure is correlated to the $NEDC$ measure. Following them, we analyze the correlation between the stability distribution and the $NEDC$ measure. What can be seen by looking at the Stability distribution and $NEDC$ values, reported in Tables 1-3, is that the Stability distributions seem to be correlated to the $NEDC$ measure. This correlation is not surprising since by definition, the two measures are two different perspectives, one local (Node stability) and one global (Dimension Connectivity), of the same aspect: how much a node is important for the connectivity of a network. We note, in fact, that the stability tends to be higher in conjunction with higher $NEDC$ values.

Table 1: Node connectivity, Node stability, computed on the illustration network in Fig. 2

Dimension	Stability average	NEDC
Dimension 1	0.63	0.85
Dimension 2	0.4	0.7
Dimension 3	0.47	0.7

Table 2: Node connectivity, Node stability, computed on AUCS network

Dimension	Stability average	NEDC
Lunch	0.22	0.15
Facebook	0.10	0.03
Coauthor	0.11	0.05
Leisure	0.20	0.11
Work	0.24	0.25

4.2. Analyzing node behavior

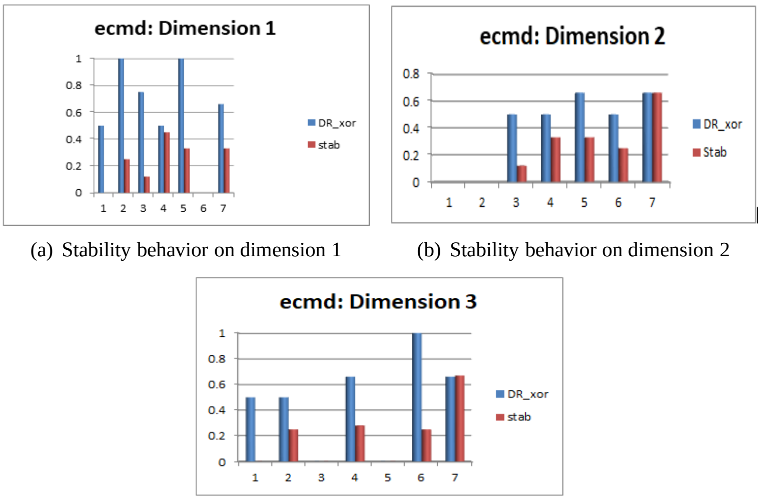
This section describes node behaviors in the overall dimensions, according to its degree and its stability. Assume that the network in Figure 2 represents an exchange of experiences through a multidimensional network between actors of the agriculture area (e.g. farmers), in which a dimension describes a type of crop (banana, onion, rice, etc.). The idea behind the stability metric is that a stable node/actor is the more important because it favors a success in agricultural business, moreover, the other nodes/actors trust in him. Then, the stability of a farmer's neighborhood demonstrates his competence; therefore, he becomes a more reliable source of information. An individual may not be reliable if he loses his regular relationship. In Figure 2, node 7 is an example of this. If the second and third dimensions are removed, it looses its trusted contacts, but still remains present in the network. On the other hand, according to DR_{xor} measure, that node 7 disappears from the network. As shown in Figure 4, this node has the same value of DR_{xor} across all dimensions, but its value of stability is low in the first dimension. Otherwise, the node 1 has a disassortative behavior. Any dimension is relevant for this node, meaning that it is less important across dimensions.

5. Conclusion

We proposed a novel centrality measure based on stability of the neighborhood of nodes. Since an active node on one dimension can remain inactive on the rest of the dimensions, it is possible to study the stability of a node in a multidimensional context, according to its center of interest. Therefore we defined the notion of relevance dimension of a node in order to contribute to the question of how important is a dimension for a node. An assessment on the semantic given to the stability centrality compared to degree centrality was carried out. Likewise, we show how correlated the stability to the dimension connectivity NEDC measure. The next study intends to assess the impact of this measure on the communities obtained from a well-defined model.

Table 3: Node connectivity, Node stability, computed on DBLP network

Dimension	Stability average	NEDC
VLDB	0.00115	0.75
SIGMOD	0.0046	0.97
CIKM	0.15	3.86
SIGKDD	0.0087	1.38
ICDM	0.098	2.45
SDM	0.055	1.44



(a) Stability behavior on dimension 1

(b) Stability behavior on dimension 2

(c) Stability behavior on dimension 3

Figure 4: Stability and DR_{xor} assessments on network in Fig. 2

6. References

[1] M. Berlingerio, M. Coscia , F. Giannotti , A. Monreale , M. Coscia , D. Pedreschi, “Multidimensional networks: foundations of structural analysis”, *World Wide Web*, pp 567–593, (2013).

[2] M. Verbrugge Lois, “Multiplexity in Adult Friendships*”, *Social Forces* vol. 57 (4), pp 1286-1309, (1979).

[3] B. Boden , S. Günnemann , H. Hoffmann , T. Seidl, “Mining coherent subgraphs in multi-layer graphs with edge labels”, *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining* pp 1258–1266, (2012).

[4] M. Magnani , B. Micenkova , R. Rossi, “Combinatorial analysis of multiple networks”, *rXiv preprint arXiv:1303.4986*, (2013).

[5] O. Boutemine , M. Bouguessa, “Mining community structures in multidimensional networks”, *ACM Transactions on Knowledge Discovery from Data (TKDD)* vol. 11,(4), pp 51, (2017).

[6] V. Nicosia , V. Latora, “Measuring and modeling correlations in multiplex networks”, *Physical Review E*, vol. 92, (3), pp 032805, (2015).

[7] P. Mucha , T. Richardson , K. Macon , J. Onnela, “Community structure in time-dependent, multiscale, and multiplex networks”, *American Association for the Advancement of Science*, vol. 328, (5980), pp 876–878, (2010).

[8] M. De Domenico , A. Sole-Ribalta , E. Omodei , S. Gòmez , A. Arenas, “Centrality in interconnected multilayer networks”, *American Association for the Advancement of Science*, vol. , (CoRR abs/1311.2906), (2013).

C.3 Communities in directed networks : Towards a hybrid model of semantic communities detection

Communities in directed networks

Towards a hybrid model of semantic communities detection

Gamgne Domgue Félicité, Tsopze Norbert, Ndoundam René, Arnaud S. R. M. Ahouandjinou

IRD UMI 209 UMMISCO,
University of Yaounde I
Yaounde,Cameroon
felice.gamgne@gmail.com, fgamgne@uy1.uninet.cm
tsopze@uy1.uninet.cm
ndoundam@gmail.com
arnaud.ahouandjinou@univ-littoral.fr, ahou.arn@gmail.com
Laboratoire LISIC, Université du Littoral de la Cote d'Opale(ULCO),62228 Calais, France,

ABSTRACT. Community detection in directed network is of vital importance to find cohesive subgroups. Many existing graph clustering methods mainly focus on the relational structure and vertex properties, but ignore edge directionality during the clustering task, in case of directed graphs. In this paper we propose a hybrid semantic similarity which includes node attribute informations along with the network structure and link semantic. Then by application of a partitioning clustering technique, we evaluate its performance and results on a built textual based dataset with ground truth. We argue that, depending on the kind of data we have and the type of results we want, the choice of the clustering method is important and we present some concrete examples for underlining this.

RÉSUMÉ. La détection des clusters orientés constitue davantage un challenge dans l'analyse des réseaux. Plusieurs approches de clustering s'attardent uniquement sur la structure et les attributs des noeuds, mais ignorent la sémantique portée par les liens dans le cas des graphes orientés. Dans cet article nous proposons une mesure de similarité hybride qui combine les informations structurelles, les attributs et l'orientation des liens. Par application de cette mesure à un algorithme de clustering , nous évaluons les performances de cette nouvelle approche sur un jeu de données que nous avons construit avec vérité de terrain. Selon le type de données exploité et le type de résultats escomptés, nous montrons que le choix de la méthode de classification est important via quelques illustrations.

KEYWORDS : Directed attributed network, Graph clustering, Link semantic, Social network

MOTS-CLÉS : Réseau orienté attribué, Clustering, Sémantique des liens, Réseau social

1. Introduction

Cluster extraction is one of the main tasks of descriptive modelisation in datamining area. Like this, most of graph partitioning methods, useful for strongly connected community detection [7], focus on relational structure, but ignore node properties or attributes. More the recent approaches tended to find cohesive subgroups by combining node attributes with link informations in graph. These informations only concerned the structure data like frequent link-pattern(neighbourhood and leadership). Nevertheless, combining these different data types leads to the problem of semantic classification, because of the "inconsistent" similarity measures omitting the link *semantic* (meaning edge's directionality). A new challenge in community detection consists on meaningful cluster extraction based on three parameters : structure, node attributes and link semantic. In this paper, we propose an hybrid technique dealing with the *semantic based topological* structure of the graph, and we show that with textual attributes joined to vertices, it is possible to extract semantic clusters. We perform our experiments through the construction of an attributed directed network with ground truth, Normalized Mutual Information (NMI) and Density measures are used for evaluations. The work of incorporating structural *semantic* and attribute data has not yet been throughout studied in the context of large social graphs. This is the motivation of our work for which key contributions are summarized next : studying of the relationship between semantic similarity of species in a food web network and showing that the type of data determine the result, thus a textual attribute strengthens the semantic topology and helps to discover more relevant communities.

The document is organized as follows. The Section 2 presents related works based on graphs partitioning methods that take into account both features and structure relationship. The formal description of the idea is presented in Section 3, then some hybrid approaches based on both links and attribute information are suggested in Section 4. An experimental study describing the constructed dataset and the expected results according to the technique are presented in the section 5. After that experiment description, an evaluation on different semi-hybrid and hybrid models are shown in the Section 6, and the Section 7 concludes the study.

2. Related works

The well-known graph clustering techniques use the relationships between vertices to partition the graph into several densely connected components, but do not use the properties of the nodes. The problem is to combine both graph data and attribute data simultaneously in order to detect clusters that are densely connected and similar in the attribute space. Few recent studies have addressed the problem of clustering in attributed networks. Next, we present a classification of the existing methods of clustering in attributed graph based on their methodological principles.

Edge weighting based approaches : In order to integrate the attribute or structure information in the clustering process, these methods define a node attribute similarity that will be used to weight the existing edges. In literature, some relevant approaches have been proposed [1]. The first approach of the following section is based on this idea.

Pattern-based approaches : These methods focus on the structure or relational property of the graph, based on kernels information Li et al. [2]. In the same way, Gamgne et al. [8] extracted kernels through the neighbourhood overlap. The relationship information is based on either the structural equivalence i.e. two vertices belong to the same cluster if they own the same neighbours or leadership i.e. vertices are connected to the same leader. They defined a *kernel degree* measure which denotes the similarity of nodes in their roles of leader (high in-degree) or follower (low in-degree) as studied by Gamgne et al. [9]. Its limit is that it does not deal with node attributes.

Quality function optimization based approaches : This family of approaches extend the well-know graph based clustering methods to consider both attribute information and topological structure. Authors in [6] proposed an extension of the Louvain algorithm with a modification of modularity by including an attribute similarity metric. [5] propose the **I-Louvain** algorithm which uses the inertia based modularity combined with the Newman's modularity.

Unified distance based approaches : They consist in transforming the topological information of the network into a similarity or a distance function between vertices. Zhou et al. [4] exploit the attributes in order to extend the original graph to an augmented one. A graph partitioning is then carried out on this new augmented graph. A neighborhood random walk model is used to measure the node closeness on the augmented graph. Then, they proposed a **SA-Cluster** algorithm that make use of a random walk distance measure and K-Medoids approach for the measurement of a node's closeness.

All of these methods have the limit that their topological property does not deal with link semantic, meaning edge directionality in directed networks. Yet the majority of real-life networks are represented as directed graphs, and link direction helps in improving partition quality.

We present in the Section 4, methods handling both topological and node attributes and that are easy to use, while the next section shows how formally a generic clustering approach could be implemented.

3. Problem Statement

An attributed graph is denoted as $G = (V, E, W)$, where V is the set of nodes, E is set of edges, and W is the set of attributes associated to the nodes in V for describing their features. Each vertex v_i is described by a real attribute vector $d_i = (w_1(v_i), ..., w_j(v_i), ..., w_m(v_i))$ where $w_j(v_i)$ is the attribute value of vertex v_i on attribute w_j. Into such network, clustering of attributed graph should take into account both structure network and attribute information by achieving a good balance between the following two properties : *(i)* vertices within one cluster are closed to each other in terms of "structure", meaning that vertices are arranged according to a semantic pattern, while vertices between clusters are not patterned; *(ii)* vertices within one cluster are more similar by their attributes than vertices from different clusters that could have quite different attribute values. In this work, we consider that the partitioning process focuses both on *semantic based topology* and node attributes. In others words, the structure concept includes not only link density, but also link semantic. The approach consists in dividing the set of nodes V into a partition of k clusters C_i, such that :

1) $C_i \cap C_j \neq \Phi \; \forall i \neq j$ and $\cup_i C_i = |V|$, where Φ is an empty set,

2) The semantic similarity takes into account three criteria : the link density, the node attribute and the link direction,

3) Vertices within clusters are semantically connected, while the vertices in different clusters are sparsely connected.

Likewise, we assume that an information network like a food web network can be represented by an attributed directed graph. Then, species relationship corresponds to a network in which each vertex represents a species and is described by a vector $d_i = (w_{i1}, w_{i2})$ where w_{i1} is the discrete attribute according to the diet mode (0 for "$carnivorous$" and 1 for "$herbivorous$") and w_{i2} the textual attribute denoting mode of reproduction (either " $oviparous$" or "$viviparous$") ; an edge from node a to node b means that species a is consumed by species b (*"Prey-Predator"* relationship). Thus, partitioning this kind of graph leads to integrate both ($density$ and $semantic$) topological and ($discrete$ or $textual$) attribute knowledge.

4. Clustering Graph models

Approaches for graph clustering described in this section separately handle both relational information and vertex attributes, and differ by their manner of combining relational data and attributes.

4.1. Attribute and Relational based clustering methods

Attribute based clustering method first exploits attributes by graph enrichment through a node attribute similarity (NAS) function [1,4,6]. According to the **SA-Cluster** method [4], the unified random walk distance is applied to an augmented graph. On the other hand, cosine distance between vertices v_i and v_j could be used, as defined as $SimA(v_i, v_j)$ in **SAC1** method [6].

In the relational based clustering model, structural properties are considered first through either a neighbourhood similarity. Li in [2] proposed a hierarchical clustering by filtering process of cores (kernels) based on structural information, then merging them by their attributes similarity. The core filtering is based on a frequent itemsets process through a similarity we labelled here $simS(v_i, v_j)$; it could be based on geodesic distance [7]. Formally, $simS(v_i, v_j) = \frac{1}{1+disS(v_i,v_j)}$. See Sect.4.2 below.

4.2. Semi Hybrid clustering

Semi-hybrid techniques combine simultaneously structural and attribute similarities through a weighted function as in Eq.1. **W-Cluster** and Combe's Model [3] are typical instances of this technique.

$$disG(v_i, v_j) = \alpha disT(v_i, v_j) + \beta disS(v_i, v_j) \qquad (1)$$

$disT$ and $disS$ denote euclidean distance for attribute data and geodesic distance for structure data respectively. A straightforward way to integrate link semantic is to combine relational, attribute and semantic similarities by adding another factor to the Eq.1 as described below.

4.3. Proposed Hybrid Clustering Model

To avoid confusion to that semi-hybrid method (not taking into account link direction), we add semantic property based on edge directionality named $simR(v_i, v_j)$ [8] and we call *semantic clusters* the groups detected from a directed attributed graph partitioning hybrid model. The proposed approach combines simultaneously 3 information data through a Node Attribute and Edge Directionality Similarity ($NAEDS$) as defined in Eq.2. Then, we have applied $NAEDS$ in Louvain's method to find answer of the following question: *Whether semantic communities be detected by dealing with direction of the edges?*

$$simG(v_i, v_j) = \alpha simT(v_i, v_j) + \beta simS(v_i, v_j) + \gamma simR(v_i, v_j) \qquad (2)$$

The equation Eq.2 computes a global Similarity $simG(v_i, v_j)$ between two vertices v_i and v_j by the linear combination of 3 measures respectively corresponding to each type of information. $simT(v_i, v_j)$ is the attribute based similarity. It is an arithmetic average between discrete attribute based similarity $simADiscr(v_i, v_j)$ (determined by counting the number of attribute values nodes have in common) and textual attribute based similarity $simA(v_i, v_j) = \frac{1}{1+\sqrt{\Sigma_d (w_i^d - w_j^d)^2}}$ based on the euclidean distance. $simS(v_i, v_j)$ corresponds to the relational based similarity (see Sect.4.1).
And $simR(v_i, v_j) = \frac{|\Delta_{ij}|}{|\Delta_j|} * \frac{|\Gamma_j^{in} \cap \Gamma_i^{in}|}{|\Gamma_j^{in} \cup \Gamma_i^{in}| - \theta}$ as defined by Gamgne et al. [8], represents edge directionality based similarity which focuses on triad density and neighbourhood of vertices. Then the global similarity measure is used as pairwise similarity measure in the Louvain's method to partition the graph into clusters. The objective is to evaluate the scalability of the method based on this global similarity by extracting semantic clusters. α, β and γ are weighting factors that enable to give more importance to the structural, attribute or semantic similarity. $\gamma = 1 - \alpha - \beta$ and $\alpha, \beta, \gamma \neq 0$.

5. Experimental Study

In this section, we performed extensive experiments to evaluate the performance of the linear combination-based approach on real-world network datasets. All experiments were done on a $2.3GHz$ $Intel\ Pentium\ IV$ PC with $6GB$ main memory, running Windows 8. Python and R package were used for implementations.

5.1. Experimental Datasets and evaluation measures

To our knowledge, there is no referenced benchmark with relational and attributes information handling link semantic (edge directionality). We construct a small ground truth dataset, a food web network, in order to compare each vertex to its real cluster. So, two datasets for experiments are used :

Food web : A typical illustration dataset as shown in Fig.1 is case of food web network where a vertex represents a species and edge the relationship between prey and predator.

Political Blogs Dataset: A directed network of hyperlinks between weblogs on US politics. This dataset contains $1,490$ weblogs with $19,090$ hyperlinks between these weblogs. Each blog in the dataset has an attribute describing its political leaning as either *liberal* of *conservative*.

We use two measures of Density and Normalized mutual information (NMI) to evaluate the quality of clusters generated by different methods.

5.2. Assumptions on food web illustration

Here we enumerate partitioning scenario and present expected results. We consider 5 subsets of vertices A, B, C, D, E describing species diet mode and by their reproduction mode, to be real semantic cluster of the hybrid clustering. The Table 1. shows the described illustration network according to each property :

Table 1: Number of species by nutrition sector and mode of reproduction

Diet Mode		Mode of reproduction	Number
A	Carnivorous	Viviparous	8
B	Carnivorous	Oviparous	3
C	Herbivorous	Viviparous	7
D	Herbivorous	Oviparous	4
E	Vegetables	Asexual or sexual	3
Total			25

– Semi attribute semantic (Textual) : 3 clusters in which species are grouped by their mode of reproduction. The ground truth partition is formally defined as $P_a = \{A \cup C, B \cup D, E\}$.

– Semi Relational-semantic (Neighbourhood) : 3 clusters in which species are grouped by their diet mode. The ground truth partition is formally defined as $P_r = \{A \cup B, C \cup D, E\}$.

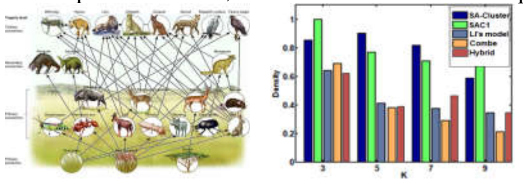
– Semantic : 5 clusters (species categories) : If we want to identify species by their both diet mode and mode of reproduction characteristics, then attributes(textual information), relational and directionality properties should be used. Like this, the resulting partition is $P_s = \{A, B, C, D, E\}$.

6. Model evaluations and results

6.1. Evaluation on illustration dataset

Given that this study focuses on directed attributed graphs which have not yet been investigated in detail, the evaluation consists in checking these assumptions described in Sect.5.2, by evaluating stated models of Sect.4 (M_a, M_r, SH_{ar}). We compare these 3 models (M) and (SH) with the hybrid model (H_s). The synthesis of results is shown in Table.2, according to the Normalized Mutual Information (NMI) measure [1]. Then clusters issued from the ground truth clustering transcripts the following partitions : the group of species by their diet mode (P_r), by their mode of reproduction (P_a), and by the both simultaneously (P_s).

– **Clustering according to textual attributes :** M_a **Model.** In this approach corresponding to the technique in Sect.4.1, the euclidean distance computed on the tex-



(a) Food web illustration network with Diet sectors (b) Density comparison on Poltitical Blogs

Figure 1: Example of datasets and results

Table 2: Results : NMI

Models	P_r	P_a	P_s
M_r	**0.753**	0.350	0.323
M_a	0.741	**0.842**	0.625
SH_{ar}	$[0.028 - 0.291]$	[0.205 – **0.441**]	$[0.085 - 0.397]$
H_s	$[0.098 - 0.217]$	$[0.110 - 0.185]$	[0.558 – **0.895**]

tual attributes helps to weight each edge ; then an unsupervised method is applied to the resulting graph. The method performs well when the ground truth partition is $P_a = \{A \cup C, B \cup D, E\}$ by a higher NMI value (0.842) than considering the partitions P_r or P_s.

– **Clustering according to relations :** M_r **Model.** This method firstly exploits relations and secondly, with attributes handling, it detects communities so that the nodes in the same community are densely connected as well as homogeneous [2]. The NMI value for the ground truth partition $P_r = \{A \cup B, C \cup D, E\}$ is higher (0.753) than its value for the ground truth partition P_a and P_s. This result demonstrates that a technique based on successively relations then attributes, performs well in case of detecting two clusters of species with a densely internal connectivity, corresponding to diet mode.

– **Semi-hybrid attributed based clustering :** SH_{ar} **Model.** As far as this method is concerned, it deals with both types of information simultaneously as studied by Largeron [3] through a weighted distance function. In experiments, the NMI value fluctuates as a function of the weighting factors α and β. It changes its value according to the weighting factor α. NMI is in the interval $[0.028 - 0.291]$ for P_r ground truth and $[0.205 - 0.441]$ for P_a when α values are respectively 0.5 and 0.75. $\beta = 1 - \alpha$. SH_{ar} Model performs the best for the ground truth P_a, meaning that textual attributes describe better the vertices similarity, but produces weak outcomes as proved by [3] for the overall results.

– **Hybrid attributed based clustering :** H_s **Model.** The objective of this hybrid based experiment consists in 2 ways. First it shows that the consideration of the textual attributes improves better the cluster semantics through the highest NMI values as presented in bold in the Table.2. Second it shows that combining simultaneously the three types of information which are link semantic, relational and attribute properties respectively, leads to the highest NMI for that expected partition $P_s = \{A, B, C, D, E\}$. Like this, it detects the five classifying species clusters by their diet and reproduction mode simultaneously with a NMI value of 0.895 when the weighting factors α and β both equal 0.33; NMI value decreases to 0.558 when the weighting factors α and β equal 0.5 and 0.40 respectively, meaning that the negligence of the third factor relating to link semantic property affects the result.

6.2. Evaluation on Polblogs dataset

The Table 3 presents NMI for P_s partition, with $\alpha = \beta = \gamma = 0.33$, while the figure 1b compares Density for each model through the number of cluster. These results strengthen the interpretation according to that high density does not inevitably denote good separation of communities.

Table 3: Results : $Density$

Models	SAC1	SA-Cluster	Li's model	Combe's model	Hybrid model
NMI	0.153	0.350	0.323	0.675	**0.878**

7. Conclusion and future works

This work focused on the presentation of a hybrid clustering approach based on a proposed similarity. This measure takes into account 3 properties : semantic, relational and attributes. As presented below, we obtained different results according to the clustering technique and to the kind of data in the directed attributed food web graph we built.

An illustration on a food web network helped to underline the choice of each method relating to the kind of information (textual or numeric). The experiments show that on the one hand, the consideration of textual documents as attributes in the partitioning process leads to expected results based on the determination of species by their reproduction and nutrition modes simultaneously, and on the other hand, the properties strengthens the cluster semantic as computed through the NMI highest value. Nevertheless it has been difficult to integrate simultaneously two textual attributes relating to both reproduction mode and nutrition mode. For this reason, the second one has been processed as a numeric. Although this method is simple, it is hard to set/tune the parameters as well as interpret the weighted similarity function. Future works intend to apply large real-world networks and study weighting factors distribution.

8. References

[1] K. Steinhaeuser, N. V. Chawla, "Identifying and evaluating community structure in complex networks", *Pattern Recognition Letters*, (2009).

[2] H. Li , Z. Nie , W. C. Lee, "Scalable Community Discovery on Textual Data with Relations", *ACM conference on Information and knowledge management*,pp. 1203-1212, (2008).

[3] D. Combe, C. Largeron , M. Gery, E. Egyed-Zsigmond "Détection de communautés dans des réseaux scientifiques à partir de données relationnelles et textuelles.", *MARAMI*, (2012).

[4] Y. Zhou, H. Cheng , Y. Jeffrey Xu "Graph Clustering Based on Structural/Attribute Similarities", *Adv. Intell. Data Anal.*, pp. 181-192 (2009).

[5] D. Combe, C. Largeron, M. Gery, E. Egyed-Zsigmond "I-louvain: An attributed graph clustering method.", *Adv. Intell. Data Anal. XIV*, pp. 181?192. Springer (2015).

[6] T. Dang, E. Viennet "Community detection based on structural and attribute similarities.", *In: International Conference on Digital Society (ICDS)*, pp. 7-12 (2012).

[7] Newman, M.E., Girvan M. " Detecting community structure in networks." *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 38(2), pp. 321-330, 2004.

[8] F. Gamgne , N. Tsopze, R. Ndoundam, " Novel method to find directed community structures based on triads cardinality." *Proceedings of CARI'16.*, vol. 2016, pp. 8-15, (2016).

[9] F. Gamgne , N. Tsopze, " Communautés et rôles dans les réseaux sociaux." *Actes du CARI'14*, pp. 157-164, (2014).

C.4 Nouvelle approche de clustering par kernel-pattern via la densité en triades : Optimisation de la métrique Kernel Degree Clustering

Nouvelle approche de clustering par kernel-pattern via la densité en triades

Optimisation de la métrique Kernel Degree Clustering

Félicité Gamgne Domgue[*] — Norbert Tsopze[*] — Arnaud S. R. M. Ahouandjinou[**]

[*] *Université de Yaoundé I - Cameroun*
[**] *Université du Littoral de la Cote d'Opale(ULCO),62228 Calais, France*

RÉSUMÉ. *La détection des communautés est devenue un domaine de recherche majeur ces dernières années. Plusieurs algorithmes appliqués aux graphes orientés ont été developpés. Ces derniers se focalisent sur la densité de liens à l'intérieur des communautés et considèrent la relation entre les nœuds comme symmétrique, car ils ignorent l'orientation des liens, ce qui biaise les résultats en produisant des communautés non-significatives. Ce document propose un algorithme basé sur l'extraction des kernels via la distribution des triades, utilisant l'optimisation de la nouvelle métrique Kernel Degree Clustering (KDC), et trouve des communautés plus sémantiques que la modularité, en accord à la notion de centralisation de l'information. Les expérimentations montrent que la nouvelle approche produit les résultats préconisés que ceux produits par certains algorithmes de détection de communautés de l'état de l'art.*

ABSTRACT. *Community detection has become a major active area of research in recent years. A plethora of relevant methods have been implemented for directed graphs. Most of them focus on the density of links, and consider the relationship between nodes as symmetric by ignoring links directionality during their clustering step, this leading to non-semantic results. This paper propose an efficient method based on the extraction of kernels through the distribution of triads in the graph, using Kernel Degree Clustering (KDC) a novel metric to judge the quality of a community partitioning, demonstrated to yield superior results over other commonly used metrics like modularity in conformity with centrality. To validate our approach, we conduct experiments on some networks which show that it has better performance over some of the other state-of-the-art methods and uncovers expected communities.*

MOTS-CLÉS : *Réseaux orientés, Détection des communautés kernel, Clusters basés sur la structure, Triade.*

KEYWORDS: *Directed graphs, Community kernel detection, Pattern-based clusters, Triad.*

1. Introduction

La détection des communautés dans les graphes orientés apparait comme l'un des objectifs majeurs des domaines de la recherche d'informations et de l'analyse des réseaux. Dans son sens premier, la notion de communauté correspond à un ensemble de nœuds densément connectés entre eux et faiblement connectés avec les autres nœuds du réseau (Fortunato, 2010). La détection des communautés peut par exemple aider à faire du marketting viral ; ou dans un réseau de produits fréquemment achetés ensemble, la détection des communautés peut être utilisée pour faire de la recommandation. Au vu de ces diverses perceptions sur la manière dont les objets sont semblables ou similaires, il existe différents algorithmes de clustering formalisant ces pensées respectives. A ce titre, la similarité entre les nœuds d'une même communauté ne tiendra plus seulement compte de la densité de liens, mais aussi des caractéristiques structurelles des nœuds dans les graphes orientés. Ainsi, alors que certains algorithmes de détection de communautés implémentés pour les graphes orientés ignorent l'orientation des liens, d'autres techniques transforment le graphe orienté en graphe non-orienté unipartite et valué (Fortunato, 2010) ou bipartite, et ensuite appliquent les algorithmes de détection de communautés sur les graphes non-orientés pour extraire leurs communautés.

Ces techniques ne sont pas satisfaisantes et ne produisent pas des résultats significatifs parce que la sémantique portée par les liens n'est pas prise en compte. Par exemple, dans un graphe de citation dans lequel les articles sont représentés par les nœuds et les relations telles que "*un article cite un autre article*" sont représentées par les liens orientés. Supposons qu'un article *i* cite un autre *j* (relation *père (j)-fils(i)*)mais pas l'inverse. D'après ces méthodes, la relation de réciprocité ou de symétrie est introduite entre les articles *i* et *j*, ce qui favorise la perte de l'information selon laquelle *j* soit cité par *i*. Dans le but de garder cette sémantique d'orientation des liens, une définition plus générique de la notion de communauté a été introduite par (Malliaros et Vazirgiannis, 2013) comme étant un ensemble de nœuds possédant des caractéristiques homogènes (plus précisément "ensemble de nœuds centrés autour d'autres nœuds, ces derniers possédant les intérêts communs"). Étant entendu que la majorité des graphes réels sont de grande taille et deviennent de plus en plus denses, au vu des multiples et divers outils de manipulation de l'information à l'ère du numérique qui révolutionne la vie quotidienne, il devient plus difficile voire infaisable de les traiter, suite à la taille limitée de la mémoire des machines. Pour ces deux raisons, la complexité et la sémantique portée par les liens, les approches basées sur les kernels semblent être indiquées pour résoudre le problème de détection des communautés dans les grands réseaux. Notre approche se base sur l'extension de l'idée selon laquelle à l'intérieur des "bonnes"communautés se trouvent des nœuds influents, *kernels*, qui centralisent l'information afin qu'elle soit aisément accessible. Les nœuds influents dits nœuds centraux sont traversés par un nombre maximal de triades dans la communauté. Une triade peut se définir comme étant un sous graphe de 3 nœuds impliquant deux liens. Ainsi, les triades constituent les bases de plusieurs structures de communautés (Klymko *et al.*, 2014). Ce travail s'attardant sur l'orientation des liens dans les triades, les contributions spécifiques y afférentes sont entre autres :

– La définition d'un nouveau concept nommé kernel Degree Clustering(KDC) qui mesure la puissance de similarité qui existe entre les nœuds du kernel, et une nouvelle sémantique donnée à la notion de communauté basée sur le voisinage des nœuds du kernel via l'appartenance triadique.

– L'implémentation d'un nouvel algorithme basé sur l'optimisation du KDC pour découvrir les kernels et par la suite les communautés qui en découlent.

– L'amélioration de la qualité des structures obtenues par rapport aux méthodes existantes.

La suite du document est structuré de la manière suivante : La Section 2 est une introduction aux méthodes existantes relatives à cette approche. Dans la Section 3, nous définissons formellement les différents concepts utilisés dans l'approche de clustering proposée. La section 4 présente une forme détaillée de l'implémentation de la nouvelle méthode, suivie de la section 5 qui présente les expérimentations faites pour étudier et évaluer les résultats obtenus. Et enfin la section 6 conclut notre étude.

2. Etude de l'art

Plusieurs approches de détection de communautés se focalisent sur les modèles symétriques qui perdent la sémantique de l'orientation des liens entre les nœuds, un facteur clé distinguant les réseaux orientés de ceux non-orientés. Pour détecter les communautés dans les réseaux orientés, (Malliaros et Vazirgiannis, 2013) présentent des méthodes de transformation du graphe orienté en un graphe non orienté valué, permettant ainsi d'utiliser les concepts éprouvés ainsi que la complexité des modèles existants pour la détection des communautés dans les graphes non-orientés. Ainsi, pour mesurer la qualité de la partition obtenue, ils utilisent une fonction “objectif” parmi plusieurs, dont la plus répandue est la modularité. Cette mesure a pour but de caractériser la qualité d'une partition des sommets d'un graphe au regard de la densité des liens à l'intérieur des groupes et du nombre de liens entre groupes distincts, via la distribution des degrés des sommets. Plusieurs méthodes d'optimisation de la modularité ont été proposées, à l'instar de la méthode d'agglomération gloutonne de (Clauset *et al.*, 2004),et dont la plus répandue et la plus sûre étant celle de Louvain (Blondel *et al.*, 2008). Si elle a eu un succès dans la détection de communautés dans les graphes, il a néanmoins été montré que la modularité possède une limite de résolution (Fortunato et Barthelemy, 2007) qui restreint la possibilité de disposer de petites communautés qui soient bien définies, car plus la taille du graphe croit, plus la qualité de la partition décroit considérablement Il existe également des méthodes basées sur les marches aléatoires (Pons et Latapy, 2005) et (Rosvall et Bergstrom, 2008), consistant en la recherche d'une forme de description des nœuds et les liens, permettant de représenter les marches aléatoires. D'après ces auteurs, la description nécessitant le moins de mémoire via le taux de compression le plus élevé de la marche, est celui sélectionné. En 2010, divers modèles probabilistes de détection de communautés ont été proposés (Malliaros et Vazirgiannis, 2013). Parmi eux, les modèles de bloc stochastiques semblent avoir eu le plus de succès en termes d'extraction de communautés sémantiques, avec des bonnes performances, et offrant des interprétations plausibles. Cependant, leur complexité en pratique parait énorme pour la raison selon laquelle au delà de 20 itérations, l'algorithme s'interrompt et les résultats deviennent invraisemblables. Pour pallier à cette limite de complexité, certains auteurs à l'instar de (Wang *et al.*, 2011) déterminent des kernels afin d'effectuer un traitement local de la détection de communautés avant de l'étendre au graphe tout entier.

Un kernel peut être assimilé à un ensemble de nœuds centraux ou influents à l'intérieur d'un groupe, appelés nœuds graines ou nœuds coeurs par (Kanawati, 2013). Un exemple typique d'algorithmes s'adaptant au type d'approche centrée-noeud sont ceux basés sur la propagation des labels (Raghavan *et al.*, 2007). Dans ce type d'approches, chaque vertex est initialisé par une étiquette ; elles définissent certaines règles simulant la propagation de ces étiquettes tel que l'établit le principe d'infection. La méthode de propagation de label possède l'avantage d'être asymptotiquement efficiente, mais aucune garantie n'est donnée sur la qualité de la partition, précisément dans les réseaux dans lesquels les communautés sont mal structurées. Certaines méthodes explorent le problème de détection de communautés dans les buts suivants : soit réduire le nombre d'itérations de réalisation des actions de l'algorithme, et par conséquent la complexité temporelle des algorithmes définis pour les grands réseaux, soit découvrir la communauté. (Wang *et al.*, 2011) identifie ces membres influents appelés kernel et ensuite propose un algorithme efficient pour déterminer la structure des communautés kernels. Lors de l'exécution de l'algorithme, le noeud initiateur du kernel est choisit aléatoirement parmi tous les nœuds du graphe, et la taille des communautés est fixée, ce qui mène à des résultats arbitraires de communautés. Pour pallier à cette limite, (Klymko *et al.*, 2014) a prouvé que les triangles jouent un rôle important dans la formation des réseaux complexes structurés et convertit un graphe orienté en un autre non-orienté et valué. Cette transformation, bien qu'efficace perd la sémantique portée par les liens au sein d'un réseau orienté. Nous proposons une méthode qui extrait les kernels via les triades et le voisinage des nœuds constituant les propriétés structurelles (orientées “pattern”) dans les grands graphes réels.

3. Formalisation de la méthode

Nous proposons dans cette section le modèle à base de la communauté kernel et une définition des différents concepts y afférents, ainsi que les notations et formulations nécessaires, à base du modèle.

3.1. *Modèle de la communauté Kernel*

(Newman et Girvan, 2004) dans ses travaux initie l'étude des méthodes de détection de communautés basés sur la densité des liens entre les sommets d'un graphe ; ainsi une communauté dans son sens éthymologique correspond à un ensemble de nœuds possédant le plus de relations entre eux qu'avec les autres nœuds du graphe. Cette définition typique de la notion de communautés est la plus répandue des méthodes de clustering dans les graphes non-orientés. Cependant celles-ci ne peuvent pas capturer des structures sémantiques, qui gardent le sens donné à l'orientation d'un lien entre les nœuds d'un graphe, contrairement aux méthodes de clustering pour lesquelles le critère cohésif de la mise en communauté des nœuds serait non pas la densité, mais la topologie accordée à une formulation bien définie de la notion de communauté. Plus précisément, les nœuds dans un graphe orienté pourraient être également groupés selon le critère de voisinage en commun (caractéristique sémantique ou structurelle) qu'ils possèdent et non pas seulement selon la densité de liens (caractéristique relationnelle) reliant les différents nœuds de cette communauté. Par exemple, le réseau de Co-citation signifiant qu'un ensemble de nœuds A, relié à un ensemble de nœuds B, implique une similarité entre les membres de chaque groupe, i.e. les nœuds de A possèdent un comportement similaire vis-à-vis des nœuds membres de B. Dans les graphes orientés, l'orientation des liens donne une impressionnante sémantique au graphe dans son ensemble, et au flux de circulation d'informations en particulier. La Figure 1 exhibe deux situations de structures représentant différents types de pattern orientés “densité en triades” d'une part et “4-cycles” d'autres part (tel que le présentent les zones d'ombre de la figure). Dans un réseau Twitter par exemple, la notion d'autorité est mise en exergue tel qu'illustré par la figure 2(a), à cause de la relation entre un ensemble de nœuds autoritaires appelées blogs Hub (nœuds u et v) et un ensemble de nœuds non populaires appelés “followers”(nœuds x) tel que présenté dans les Figures 2(b) et 2(c).

Ce concept d'autorité (ou de centralité) se traduit par l'optimisation de la notion kernel degree Clustering. La figure 2(a) est une visualisation d'un extrait du réseau de Twitter contenant deux kernels : d'une part les acteurs (Ashton Kutcher, Demi Moore, Opray Winfrey) et d'autre part les politiciens (Barack Obama, Al Gore). Ils constituent en d'autres termes les leaders alors que les nœuds situés à gauche de la figure correspondent à leurs fans ou followers, tel qu'étudié par (Gamgne et Tsopze, 2014). Les communautés kernel décrivent les nœuds possédant le même voisinage entrant (nœuds les plus connectés à un kernel et non pas à un autre). Nous considérons dans ce papier, les liens entrant vers les kernels pour exprimer la puissance de similarité qu'ils possèdent, conformément aux types de graphes qui y sont manipulés (réseau de citation) ; pour mieux illustrer cette formulation, le réseau de Twitter est structuré de pages hub “tweetées”ou aimées par un ensemble de visiteurs, et non l'inverse ; dans un réseau de Citation par exemple, les pionniers d'un domaine de recherche bien précis sont le plus cités par les chercheurs juniors. Initialement, un kernel est constitué d'un ensemble de nœuds centraux via leur degré entrant, obtenus par application de la notion de “triade ”. Ce dernier constitue l'idée de base de cette approche, tout en s'inspirant de la notion d' “appartenance triadique” qui stipule que si deux amis ont un ami en commun, il est fort probable qu'ils soient du même groupe.

3.2. *Terminologie et concepts à base du modèle*

Étant donné un graphe $G(V, E)$ de $n = |V|$ sommets et $m = |E|$ liens. Soit Γ_u l'ensemble des voisins du noeud u. Nous définissons les notions et concepts à base de notre modèle :

Definition 1 *(Puissance de similarité). La puissance de similarité définit le critère ou le degré selon lequel deux ou plusieurs nœuds possèdent le plus grand nombre de voisins communs.*

Definition 2 *(Kernel). Un kernel correspond à un ensemble de nœuds possédant le plus grand nombre de voisins en commun. Ainsi, plus les nœuds d'un kernel possèdent des voisins en commun, plus la puissance de similarité qui les lie est important.*

Definition 3 *(Poids de la Triade). Le Poids de la triade pour chaque paire de nœuds (u, v) dans le graphe G peut être représenté par TW_{uv}. Nous utiliserons l'expression Δ_{uv} pour décrire le nombre de triades(cardinalité de triades) impliquant les nœuds u et v selon le schème présenté par les figures 2(b) et 2(c).*

$$TW_{uv} = \frac{|\Delta_{uv}|}{|\Delta_v|} \qquad [1]$$

Où $|\Delta_v|$ correspond au nombre de triades impliquant le noeud v.

Definition 4 *(Chevauchement de voisinage). Étant donné deux nœuds u et v. Soit Γ_u l'ensemble des nœuds appartenant au voisinage du noeud u, soit Δ_v l'ensemble des nœuds appartenant au voisinage du nœud v. Notons NO_{uv} l'ensemble des nœuds voisins que u et v possèdent en commun.*

$$NO_{uv} = \frac{|\Gamma_v \cap \Gamma_u|}{|\Gamma_v \cup \Gamma_u| - \theta} \qquad [2]$$

où θ peut prendre différentes valeurs fonction de la connectivité existant entre les nœuds u et v (0 lorsque les nœuds ne sont pas liés, 1 lorsqu'il existe un arc entre les nœuds, et 2 lorsqu'il existe une relation de réciprocité entre eux).

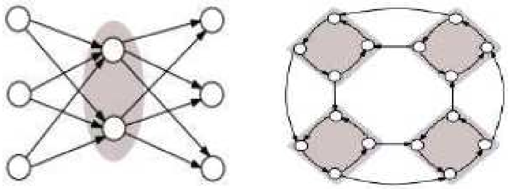
Definition 5 *(kernel Degree Clustering). Le Kernel Degree Clustering d'un couple de sommets u et v est défini par :*

$$KDC_{uv} = TW_{uv} * NO_{uv} \qquad [3]$$

KDC_{uv} peut mesurer de manière particulière le degré de similarité du couple de nœuds (u, v) et de manière générale la puissance ou "force"(notion de similarité) d'un kernel à posséder des voisins en communs.

Definition 6 *(Communauté Kernel). La communauté kernel est un ensemble de nœuds possédant le plus grand voisinage commun, tel que ces voisins centrés autour du kernel par des liens entrant optimisent la mesure kernel Degree Clustering KDC_{uv}.*

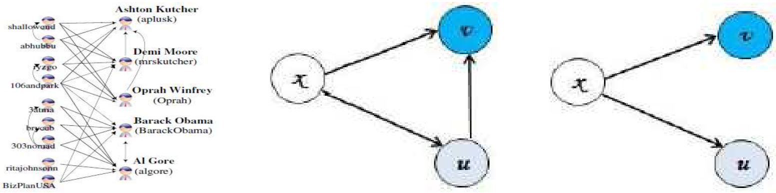
Definition 7 *(Appartenance triadique). Cette notion stipule que si deux individus possèdent un ami en commun, alors il est très probable qu'ils fassent partie du même groupe (ou kernel) sans pour autant devenir absolument des amis.*



((a)) Cluster orienté Citations

((b)) Cluster orienté flot d'informations

Figure 1 – Exemples de clusters basés sur la topologie dans les graphes orientés. Le graphe de gauche (a) represente un cluster de pionniers d'un domaine de recherche donné dans un réseau de citation. Celui de droite (b) expose un graphe de 4 cycles dans un réseau de flots d'informations.



((a)) Illustration du réseau de Twitter

((b)) Triade fermée

((c)) Triade ouverte

Figure 2 – Structures à base du modèle de la communauté kernel.

4. Méthode d'extraction des communautés

La nouvelle approche de détection de communautés est structurée en deux étapes : l'identification des kernels qui sont les nœuds centraux de la communauté, et la migration des autres nœuds vers les kernels avec lesquels ils sont le plus liés. L'algorithme d'extraction des kernels, TRICA (Triads Cardinality Algorithm) que nous proposons ici fait usage du nouveau concept Kernel Degree Clustering (KDC), via l'optimisation de ce dernier, et pour lequel la valeur optimale détermine le degré de similarité des nœuds de la partition kernel. Cette mesure se base sur l'appartenance triadique permettant d'exprimer la sémantique portée par les liens entre les différents membres d'une communauté, favorisant ainsi un accès certain à l'information à travers le réseau. Au lieu de mesurer la qualité de la partition entière de communautés comme le font (Clauset *et al.*, 2004) et (Blondel *et al.*, 2008) dans leur méthode, cette métrique s'applique aux kernels tout en effectuant une optimisation en local du graphe, tel que étudié par (Van Laarhoven et Marchiori, 2016). Nous nous attardons sur la cardinalité des triades communs aux vertex du kernel, correspondant au nombre de voisins en communs que ces derniers possèdent.

4.1. *Algorithme TRICA*

Ce paragraphe propose un algorithme évolutif, basé sur l'optimisation de la métrique KDC défini à la fois pour les graphes orientés et non-orientés. En s'inspirant des propriétés des réseaux réels, l'idée de base sous-jacente à cette métrique est la suivante : les nœuds d'un même kernel doivent favoriser une densité en triades plus importante dans les communautés dont ils sont le centre, en s'appuyant sur leur voisinage entrant. En effet la communauté engendrée par les nœuds centrés autour du kernel devrait contenir un nombre important de triades entre ses membres, et un nombre assez faible de triades entre les nœuds de ces communautés avec les autres nœuds extérieurs à la communauté. Ainsi, la qualité d'un kernel est défini comme la cohésion moyenne de chacun de ses membres avec les autres nœuds du kernel. La cohésion entre un vertex v et un ensemble de nœuds $u \in S$ dont la valeur du NO_{uv} est supérieure à un certain seuil ($\varepsilon = 0.5$ expérimentalement choisit et dont la valeur se verra discutée dans un prochain article) se définit par :

$$KDC_{uv}(u, v \in S) = TW_{uv} * NO_{uv}. \qquad [4]$$

Le terme de gauche TW_{uv} calcule la proportion en triades à l'intérieur des kernels entre les nœuds pris deux à deux ; et celui de droite NO_{uv} détermine la proportion de voisinage en commun que deux nœuds d'un même kernel possèdent. Intuitivement, la métrique Kernel Degree Clustering mesure la force de similarité des membres d'un kernel. La qualité de la partition des kernels correspond à la qualité moyenne de chaque vertex dans son kernel. Ainsi, pour un ensemble S correspondant à un kernel, $KDC(S)$ se définit comme étant la moyenne $\forall x \in S$ de $KDC(x, S)$, et la valeur finale de KDC correspondant à la partition kernel $P = K_1, ..., K_n$ de K (l'ensemble des nœuds membres des kernels) formulé par :

$$KDC(P) = \frac{1}{|V|} \Sigma_{S \in P} \Sigma_{x \in S} KDC(x, S). \qquad [5]$$

Supposons que le réseau à analyser est représenté par un graphe connexe, non-valué G de $n = |N|$ nœuds et $m = |E|$ liens . Cette étape d'identification des kernels se décompose en 4 sous-étapes comme suit :

1) Extraire une liste triée de nœuds centraux selon le critère de leur degré entrant, dans le graphe.

2) Déterminer le voisinage commun de chaque couple (u,v) par le biais d'une variante du coefficient de Jaccard(Fortunato, 2010) tel que décrit par NOuv dans la formule 2

3) Déterminer le poids des triades TW_{uv} (tel que décrit par la formule 1) pour chaque u, v dont le $NO_{uv} \geq \varepsilon$.

4) Calculer la moyenne des KDC_{uv} pour chaque couple (u, v) de la liste des nœuds éligibles à appartenir à un kernel.

5) Stocker v dans le kernel pour lequel KDC_{uv} est optimal.

Ces différentes étapes se répètent jusqu'à l'obtention d'une valeur de KDC_{uv} optimale.

La première étape consiste en la détermination d'une liste de nœuds triée par leur degré entrant. L'opération de tri permet de simplifier la réalisation de la deuxième étape consistant à supprimer aisément de cette liste les nœuds possédant moins de deux voisins, car ce type de nœuds serait probablement disqualifié dans l'idée de faire partie des nœuds centraux dans les kernels, à cause de leur degré entrant variant entre 1 et 0.

Après extraction de cette liste triée et épurée de nœuds classés par ordre décroissant de leur valeur de degré entrant, suit le calcul des NO_{uv} pour chaque couple de nœuds (u, v) pour déterminer ceux des nœuds éligibles à faire partie des kernels, correspondant ainsi aux nœuds dont le couple a pour valuation de NO_{uv} une valeur supérieure à un seuil ε.

L'étape de calcul des poids des triades quant à elle se décrit de la manière suivante : d'une part, il est question primo de compter pour chaque couple de nœuds (u, v) le nombre total de triades dans lesquels ils sont impliqués dans le graphe, secundo de supprimer tous les nœuds n'appartenant à aucune triade ; d'autre part l'on compte pour chaque vertex v le nombre total de triades dans lesquels il est impliqué (Δ_v). Cette phase de filtrage aide à l'amélioration des performances et permet de simplifier les hypothèses dans les choix futurs d'un noeud quelconque, dans sa décision à passer d'une communauté à une autre. Notons que ces deux valeurs sont des constantes pendant le processus de détermination des kernels et peuvent être calculées simultanément. Étant donné deux nœuds u et v, une manière classique de dénombrer les triades dans lesquels ils sont impliqués consiste en l'intersection de leur liste d'adjacence en vue de compter leur voisins en communs. Si les nœuds u et v ne possèdent aucun voisin en commun, ils sont mentionnés comme devant appartenir à des communautés distinctes dans la partition résultante du graphe, car ce type de nœuds n'affecte pas la détermination de KDC. Pour dénombrer tous les triades impliquant v, le précédent processus effectué pour les couples, est appliqué pour chaque voisin u de v, v étant le noeud central de degré entrant maximal.

La 5^e étape et la plus importante de ce processus de clustering des kernels consiste en l'optimisation de la mesure KDC_{uv}. L'idée de base sous-jacente au calcul de KDC_{uv} est celle permettant à chaque vertex de mettre à jour de manière répétée les kernels, via une heuristique d'amélioration, tout en évaluant l'ensemble des KDC_{uv}

entre chacune des mises à jour ; et après un certain nombre pré-spécifié d'étapes pour lesquelles KDC_{uv} ne croit plus jusqu'à un certain seuil, le processus s'interrompt. Cette heuristique basée sur le calcul de la moyenne des KDC_{uv} sur l'ensemble des nœuds u, v pour lesquels $NO_{uv} \geq \varepsilon$ permet de fixer une marge dans laquelle l'optimisation de cette métrique se verra varier. En effet, pour une valeur donnée de KDC_{uv} inférieure à la borne inférieure de cette marge (moyenne des KDC_{uv}), certes en deçà de la valeur optimale (borne supérieure), u se verra supprimé du kernel courant pour etre un noeud non-kernel. Sinon u restera dans son kernel courant. Et il migrera du kernel courant vers un autre kernel pour lequel la valeur optimale est atteinte. En fait, après avoir initialisé le kernel par un noeud central v, la combinaison d'autres nœuds u du graphe avec v via le calcul de KDC_{uv} peut conduire aux deux états ci-dessous :

– Migrer : Le vertex migre d'un kernel vers le kernel d'un des nœuds parmi ceux centraux, situé dans son voisinage le plus proche.

– Rester : Le vertex demeure dans son kernel.

Dans le but d'améliorer les performances de l'approche, le vertex doit choisir parmi les actions ci-dessus, celle qui conduirait à l'obtention d'une meilleure valeur optimale de KDC_{uv}. Le pseudo-code associé à TRICA est présenté dans l'algorithme 1.

Algorithm 1 Implementation de la méthode d'Extraction des kernels TRICA

```
Entrées: Graphe orienté $G = (V, E)$
Sorties: $K$ Kernels
Initialisation : $K \leftarrow \emptyset$
$L = Sort(v/d^{in}(v) = max\{d^{in}(t), \forall t \in V\})$ ;
Calculer $NO_{uv}$ et $TW_{uv}$ pour chaque $(u, v) \in V$ tel que $NO_{uv} > \varepsilon$ ;
Calculer la moyenne($KDC_{uv}$)
pour Chaque $u \in L$ faire
    $v = argmax\{d^{in}(t), \forall t \in L\}$ ;
    $KDC^{*} \leftarrow Moyenne(KDC_{uv})$ ;
    répéter
        Calculer $KDC_{uv}$
        Si $KDC_{uv} > KDC^{*}$ alors
            $S \longleftarrow S \cup u$ ;
        Fin si
        $KDC^{*} \longleftarrow KDC_{uv}$
    jusqu'à $KDC_{uv} < KDC^{*}$
    $K \longleftarrow K \cup S$
fin pour
Retourner $K$ ;
```

Algorithm 2 Pseudo code de l'algorithme de migration des nœuds non-kernels

```
Entrées: Communautés kernels $K = K_1, K_2, ..., K_t$
Sorties: Communautés globales $G_k = G_{k1}, G_{k2}, ..., G_{kt}$
$\forall i \in 1, ..., t, G_{ki} \leftarrow \emptyset$
répéter
    $\forall i \in 1, ..., t, R_i \leftarrow K_i \cup G_{ki}$
    pour $i \leftarrow 1$ to $t$ faire
        $S \leftarrow v \notin \cup R_i | \forall j \in 1, ..., t,$
        $|Connexions(v, R_i)| \geq |Connexions(v, R_j)| > 0$
        $G_{ki} \leftarrow G_{ki} \cup S$
    fin pour
jusqu'à Plus de nœuds non-kernels
Retourner $G_k$ ;
```

4.2. *Déduction des communautés globales*

Après l'extraction des communautés kernel, il est question de faire migrer les nœuds non-kernels, ceux n'appartenant à aucun kernel, vers les kernels avec lesquels ils sont le plus liés, via l'orientation “entrante”des liens de ce noeud. Ces nœuds non-kernel migreront vers les kernels et formeront ainsi des “Communautés globales”. Le processus de génération des communautés globales (communautés contenant à la fois les nœuds kernels et non-kernels) consiste en l'exécution des étapes suivantes : initiallement, on étiquette chaque noeud non-kernel comme étant non-associé. Pour chaque noeud non-associé, le ranger dans le kernel avec lequel il possède le plus grand nombre de connexions; le kernel change ainsi d'état pour devenir une communauté globale grandissante. Ce processus est repété jusqu'à ce qu'il n'y ait plus de nœuds non-kernel, tel que décrit par l'algorithme 2.

5. Expérimentation et évaluation de la nouvelle méthode

5.1. *Description des méthodes et des jeux de données*

Afin de montrer la performance de cette approche, l'évaluation empirique s'est focalisé sur une comparaison entre les résultats produits par plusieurs autres méthodes de l'état de l'art parmi lesquels : Walktrap (Pons et Latapy, 2005), Edge Betweenness (Newman et Girvan, 2004), Label Propagation (Raghavan *et al.*, 2007) et Louvain (Krings et Blondel, 2011). Notre méthode peut s'appliquer autant aux graphes orientés que non-orientés. L'expérimentation s'est appuyée sur trois niveaux d'évaluation : le premier concerne la densité en triades ou triad cardinality rate (TCR), dans les communautés résultantes, le second se base sur les valeurs de la modularité dans les partitions obtenues par chacune des méthodes, et la dernière sur le nombre final de communautés, en s'appuyant sur l'intuition selon laquelle plus une partition possède

Tableau 1 – Caractéristiques des jeux de données

Jeu de données	Nombre de nœuds	Nombre de liens
Extrait de Twitter	14	32
Celegansneural	297	2,345
Polblogs	1,490	19,090
Citeseer	3,327	4,732

de communautés, moins elle est dense (en triades). TCR correspond au pourcentage de triades dans la partition toute entière, tel que définit dans la formule ci-dessous.

$$TCR = \frac{\Sigma_i |\Delta_i|}{|\Delta|} \qquad [6]$$

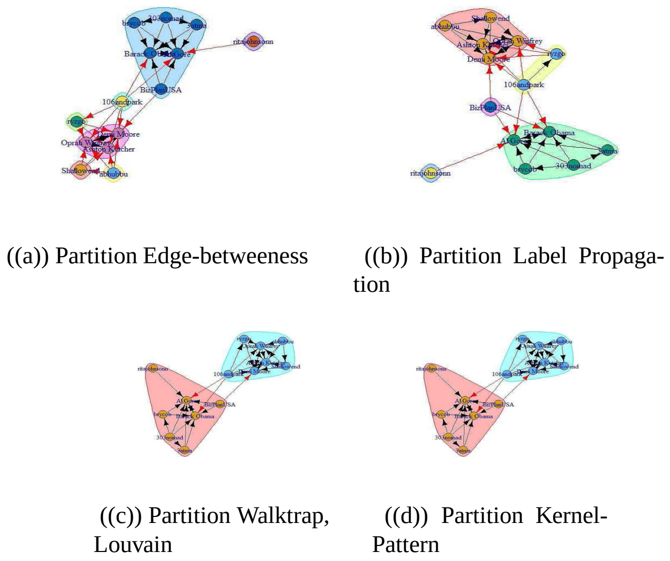
où i correspond à une communauté quelconque et $|\delta|$ le nombre de triades dans le graphe tout entier.

5.2. *Evaluation de la performance des méthodes de détection*

Les performances des différentes méthodes de détection de communautés sur les quatre jeux de données sont respectivement présentées dans les tableaux 2, 3, 4 et 5.

Walktrap détermine la distance (l'homogénéité) entre les communautés et fusionne celles qui sont moins distantes pour produire une nouvelle communauté résultante. L'idée de Walktrap et celle de la nouvelle approche sont semblables dans la mesure où elles ont en commun la notion de fusion des groupes de nœuds voisins (l'un sur la base de la distance, et l'autre sur la base du nombre de voisins en commun) ; ainsi les résultats des deux méthodes sont dans l'ensemble convergentes, tel que présenté ci-dessous : Dans le réseau extrait de Twitter présenté dans le Tableau 2, TRICA et Walktrap obtiennent la même valeur de TCR, soit 0.6428, mais la valeur de la modularité 0.401 obtenue par Walktrap est plus petite que celle obtenue par TRICA, soit 0.410. Les deux méthodes découvrent le même nombre 2 de communautés. La méthode Louvain détecte également 2 communautés, avec une veleur plus petite de la modularité égale à 0.395. Cependant, les méthodes Label Propagation et Edge Betweenness découvrent respectivement 5 et 7 communautés avec des faibles taux de triades ainsi que de valeurs de modularité, ce qui traduit l'insuffisance de ces approches sur l'idée de clusteriser les nœuds appartenant au meme voisinage d'un ensemble de nœuds kernels.

En ce qui concerne le jeu de données Celegansneural network : la méthode Edge Betweenness détermine le lien de centralité d'intermédiarité maximale, c'est à dire celui traversé par le plus grand nombre de géodésiques (plus courts chemins) et se charge de supprimer ce lien, et de façon récursive obtient des communautés. Elle détecte le plus grand nombre de communautés (194), avec un faible taux de triades dans



((a)) Partition Edge-betweeness ((b)) Partition Label Propagation

((c)) Partition Walktrap, Louvain ((d)) Partition Kernel-Pattern

Figure 3 – Visualisation des partitions obtenues par les différentes approches.

conservativeprotest.blogspot.com
digital-democrat.blogspot.com

Figure 4 – Partition Walktrap sur le réseau Polblogs, avec des nœuds bruits appelés outliers.

la partition toute entière (0.0857). Contrairement aux autres méthodes, la nouvelle approche détermine le nombre de communautés attendues (5), car en tant que benchmark, (Klymko *et al.*, 2014) obtiennent cette meme valeur. Par ailleurs, avec un taux élevé de TCR égal à 0.3211), ceci démontre la performance de la nouvelle méthode sur la qualité de la partition résultante, tel que présenté par le Tableau 3.

La méthode Propagation de label consiste à faire déplacer un noeud d'une communauté vers une autre si ses voisins appartiennent à cette communauté de destination. De cette manière, pour le jeu de données polblogs, elle détecte le plus grand nombre de communautés (244) avec le plus faible taux de triades (0.0026). Cependant les mé-

Tableau 2 – Performance des méthodes de détection de communautés sur le réseau Extrait de Twitter, où les meilleures performances sont en gras.

Méthode	TCR	Modularité	Communautés
Edge-Betweenness	0.0857	0.187	7
Walktrap	**0.6428**	0.401	**2**
Label Propagation	0.34	0.306	5
Louvain	0.531	0.395	**2**
Kernel-pattern	**0.6428**	**0.410**	**2**

thodes Walktrap et Kernel-pattern produisent de meilleurs résultats sur tous les critères d'évaluation soit 12 et 34 communautés respectivement, avec les valeurs maximales de TCR (0.67 et 0.5732 respectivement) et les meilleures valeurs de modularité, soit 0.4302 et 0.429 respectivement, tel que présentés dans le Tableau 4. Ces résultats indiquent que les modèles basés sur l'idée de "communauté croissante centrée autour de kernels" est d'une manière ou d'une autre en accord avec la notion d'optimisation de la mesure Kernel Degree Clustering ; en effet, le nombre minimum 12 de communautés avec une valeur de TCR maximale de 0.67 est preuve que la topologie en triades via la méthode Walktrap est la mieux structurée. Cependant, cette dernière ne saurait être meilleure, à cause des données bruitées représentées par les deux nœuds **singletons**, tel que présenté dans la Figure 4, indiquant que la méthode capture des nœuds que (Ester et al., 1996) appelle "outliers", qui constituent des nœuds anormaux ou **bruits** de la partition.

Bien que la méthode Louvain produise la plus grande valeur de modularité (0.886) pour le corpus Citeseer tel que présenté dans le Tableau 5, son nombre de communautés est plus important que celui produit par la nouvelle approche. Ainsi, la méthode Louvain détermine une valeur de TCR (0.213) moins importante que celle obtenue par l'algorithme Kernel-pattern (0.407), ce qui montre la validité de la nouvelle approche sur la sémantique des liens. Citeseer, contrairement aux autres corpus, suit une distribution de la loi de puissance exponentielle, dû au fait qu'il s'agisse d'un réseau de citation dans lequel l'on pourrait être en possession de nœuds de centralité de degré plus importante que les autres nœuds (l'on parle de nœuds "hub"). C'est ce qui expliquerait la valeur nulle des TCR pour les trois méthodes du tableau 5.

Figure 3 permet de visualiser la plausibilité de la méthode d'extraction des communautés sur la base des kernels, TRICA sur le jeu de données extrait de Twitter.

6. Conclusion

Dans ce document, nous nous sommes focalisés sur le problème d'extraction de communautés basé sur les kernel-pattern, une communauté se ramenant à un ensemble de nœuds centrés autour d'un sous groupe de nœuds graines, initiateurs de la commu-

Tableau 3 – Performance des méthodes de détection de communautés sur le réseau Celegansneural, où les meilleures performances sont en gras.

Méthode	TCR	Modularité	Communautés
Edge-Betweenness	0.0004	0.081	194
Walktrap	0.0458	0.363	21
Label Propagation	0.0135	0.0027	29
Louvain	0.2951	**0.398**	6
Kernel-pattern	**0.3211**	0.359	**5**

Tableau 4 – Performance des méthodes de détection de communautés sur le réseau Polblogs, où les meilleures performances sont en gras.

Méthode	TCR	Modularité	Communautés
Edge-Betweenness	0.0064	0.1872	55
Walktrap	**0.67**	**0.4302**	**12**
Label Propagation	0.0026	0.386	244
Louvain	0.1289	0.427	276
Kernel-pattern	**0.5732**	**0.429**	**34**

nauté, possédant quasiment le même voisinage commun. Un kernel correspond ainsi à un outil favorisant la compréhension du rôle et de la structure d'un réseau. Nous avons principalement orienté ce travail dans l'extraction des kernels qui sont considérés comme étant des nœuds influents du réseau. La nouvelle approche proposée se base sur l'optimisation de la mesure Kernel Degree Clustering (KDC) qui définit la puissance de similarité existant entre nœuds d'un même kernel, via la notion de triade représentant les caractéristiques structurelles des grands réseaux réels. Les expérimentations sur la nouvelle approche prouvent que la méthode Kernel-pattern permet de détecter les communautés efficaces attendues, et réalise de meilleurs valeurs de

Tableau 5 – Performance des méthodes de détection de communautés sur le réseau Citeseer, où les meilleures performances sont en gras.

Méthode	TCR	Modularité	Communautés
Edge-Betweenness	0.0	0.5344	738
Walktrap	0.0	0.811	593
Label Propagation	0.0	0.491	842
Louvain	0.213	**0.886**	466
Kernel-pattern	**0.407**	0.707	**121**

qualité des communautés, par rapport à certaines méthodes de l'état de l'art. Cependant, elle ne s'applique pas aux graphes valués. Nos travaux futurs consisteront ainsi à prendre en compte cette propriété de valuation des graphes orientés et s'attardera sur la programmation en parallèle, afin d'améliorer la complexité du modèle.

7. Bibliographie

Blondel V. D., Guillaume J.-L., Lambiotte R., Lefebvre E., « Fast unfolding of communities in large networks », *Journal of statistical mechanics : theory and experiment*, vol. 2008, n^{o} 10, p. P10008, 2008.

Clauset A., Newman M. E., Moore C., « Finding community structure in very large networks », *Physical review E*, vol. 70, n^{o} 6, p. 066111, 2004.

Fortunato S., « Community detection in graphs », *Physics reports*, vol. 486, n^{o} 3, p. 75-174, 2010.

Fortunato S., Barthelemy M., « Resolution limit in community detection », *Proceedings of the National Academy of Sciences*, vol. 104, n^{o} 1, p. 36-41, 2007.

Gamgne D. F., Tsopze N., « Communautes et roles dans les reseaux sociaux », *Proceedings of the 12th Conference Africaine sur la Recherche en Informatique et Mathematiques appliquees*, vol. 12, n^{o} 1, p. 122-188, 2014.

Kanawati R., « Détection de communautés dans les grands graphes d'interactions (multiplexes) : état de l'art », 2013.

Klymko C., Gleich D., Kolda T. G., « Using triangles to improve community detection in directed networks », *arXiv preprint arXiv :1404.5874*, 2014.

Krings G., Blondel V. D., « An upper bound on community size in scalable community detection », *arXiv preprint arXiv :1103.5569*, 2011.

Malliaros F. D., Vazirgiannis M., « Clustering and community detection in directed networks : A survey », *Physics Reports*, vol. 533, n^{o} 4, p. 95-142, 2013.

Newman M. E., Girvan M., « Finding and evaluating community structure in networks », *Physical review E*, vol. 69, n^{o} 2, p. 026113, 2004.

Pons P., Latapy M., « Computing communities in large networks using random walks », *International Symposium on Computer and Information Sciences*, Springer, p. 284-293, 2005.

Raghavan U. N., Albert R., Kumara S., « Near linear time algorithm to detect community structures in large-scale networks », *Physical review E*, vol. 76, n^{o} 3, p. 036106, 2007.

Rosvall M., Bergstrom C. T., « Maps of random walks on complex networks reveal community structure », *Proceedings of the National Academy of Sciences*, vol. 105, n^{o} 4, p. 1118-1123, 2008.

Van Laarhoven T., Marchiori E., « Local network community detection with continuous optimization of conductance and weighted kernel k-means », *Journal of Machine Learning Research*, vol. 17, n^{o} 147, p. 1-28, 2016.

Wang L., Lou T., Tang J., Hopcroft J. E., « Detecting community kernels in large social networks », *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, IEEE, p. 784-793, 2011.

C.5 Finding directed community structures using triads

FINDING DIRECTED COMMUNITY STRUCTURES USING TRIADS

Gamgne Domgue Félicité[1] *Tsopze Norbert*[1]

[1] *Université de Yaoundé I, Département d'Informatique*
BP: 812 Yaoundé, Cameroun
felice.gamgne@gmail.com, fgamgne@uy1.uninet.cm, tsopze@uy1.uninet.cm

ABSTRACT

Community detection in directed networks appears as one of the most relevant topics in the field of network analysis. One common theme in some formalizations is that flows should tend to stay within communities and could be centred round core nodes called kernels. Hence, we expect triads to play an important role. Triads for directed graph are directed sub-graph of 3 *nodes involving at least* 2 *links between them. To identify communities in directed networks, we propose an undirected edge-weighting scheme based on directed triads. We also propose a new metric on quality of the communities that is based on the number of triads that are split across communities. To validate our approach, we conduct experiments on some networks which show that it has better performance on triad-based density over some methods.*

Keywords : *Networks Analysis, Directed graphs, Kernel community detection, Triad*

RESUME

La détection des communautés dans les graphes orientés constitue un domaine crucial de l'analyse des réseaux. Dans une communauté, les noeuds sont plus densément connectés entre eux, et peuvent etre centrés autour d'autres noeuds "coeurs"appelés Kernels. Ainsi nous estimons que les triades sont d'un role capital dans la detection des communautés. Une triade dans un graphe orienté est un sous-graphe de 3 *noeuds impliqués dans au moins* 2 *liens. Afin d'extraire les communautés, nous proposons une methode de transformation du graphe orienté en non orienté et valué, en se basant sur les triades orientées. Et nous proposons une nouvelle mesure de qualité basée sur la densité en triades dans les communautés. La validation de cette approche passe par des experiementations sur des reseaux qui montrent qu'elle découvre des communautés plus significatives, de densité en triades plus importantes que d'autres méthodes de l'état de l'art.*

Mots clés : *Analyse des réseaux, Graphes orientés, Détection des communautés kernel, Triades*

1. INTRODUCTION

A recurrent research theme in Network analysis is community detection. In directed networks, it appears as one of dominant research works, because of the link semantic that should be conserved. Fortunato [2] extends communities definition to be considered as separate entities with their own autonomy. Community assignment methods for directed graphs(see [8]) focus on either density or semantic properties to detect communities; the connectivity between nodes is often used alone to define metrics measuring the "quality" of the assigned groups. Common quality metrics measure (i) the density of links within a group (modularity), (ii) the density of triangles (Weighted Community Clustering- WCC-). These techniques implemented for directed networks are more useful to detect groups with same autonomy but not with the same anatomy, meaning that they ignore graph topology and link directionality during the clustering step [8] In order to improve the quality partition, we propose a generic weighting model that converts a directed graph into an undirected weighted graph and takes into account both density and semantic features. Our specific weighting scheme is based on extending the idea that, within "good" communities, information can be centralized by kernels and accessible in a community more easily than between communities. Therefore, our approach, considering semantic sight, is able to express the idea of detecting groups of nodes with homogeneous in-link structure (e.g., citation-based clusters) through triads, and give the possibility to kernel nodes to own more common neighbours. A triad in this paper is a set of 3 nodes whose at least 2 are the *in-neighbour* nodes (source nodes) of the 3^{rd} vertex (target node).

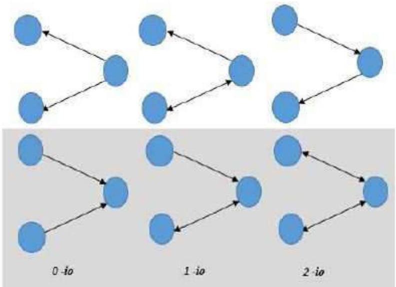


Figure 1: Basic structures of our kernel community model, opened triads.

The approach is structured in three steps: first it transforms the directed graph into undirected weighted one via *Kernel Degree* that measures the similarity strength or common neighbourhood of pair of nodes, then it extracts kernels; like this, we focus on an interclass inertia vector computed from a kernel degree list, on which the extraction of kernels is based, with a threshold fixed by the standard deviation on the interclass inertia vector. The third step consists on enlarge kernels by nodes possessing a maximal connexion number with their kernel, leading to communities, through a proposed concept called *Node Community Index*. The specific contributions of our paper are:

- We proposed a structural and density based clustering scheme that points up features and community semantic.
- we introduce a new concept called *Kernel Degree* using information about directed triads to improve community detection in directed networks.
- In addition, we focus on *Triad Density* measure that constitutes the essence of this approach, for partition evaluation, different from other common measures known in the state-of-the-art, like the modularity, conducting to better triad-based quality improvement over some community detection algorithms.

2. MOTIVATIONS AND BACKGROUND

Triads, initially studied by authors Wasserman and Faust in [17] in social network analysis was introduced by [14] to identify communities of different types. Triads are considered as wedges, i.e paths of length 2 by [5]. Like this, a triad can be integrated into a triangle. In [11], the authors define a "good" community to be a group of nodes that is dense in terms of triangles through Weighted Community Clustering (WCC) measure [12]. Given that semantic properties of the graph should take into account either "in" or "out" directionality of links, it becomes interesting to specify those of nodes centred round kernel (set of influential nodes inside a group) according to one direction. Then, kernel community detection methods are considered as seed-centric approaches [4] because of the influence of nodes centralizing information. Then, using triads enlarges the possibility to consider low degree nodes instead of high degree nodes called "hub nodes" which are solely structured in the majority of real-life networks. In this paper, a **triad** is a set of 3 vertices involved to at least 2 edges. In undirected networks, there is only one type of triad, a path of length 2. In directed networks, we present six triads types as shown in Fig.1. Of these six triads, only a few are relevant for *in-seed-centric* community detection approaches. We focus on *in-seed-centric* [4] approach because of the influence of nodes centralizing information, a good pattern close relationship and ability to concentrate information between nodes. This is not the case for all types of directed triads. As an example, in a blog readership network, there are two types of bloggers: "writers"who generate influential blogs read by many others, and "readers"who read a lot but seldom write anything for others to read. Some methods explored the problem of detecting community kernels, in order to either exhibit different influence and different behaviour of nodes inside a structure for easily analyse and interpret results or uncover the hidden community structure in large social networks. Like this, the new *Triad Density* measure implemented in this work, contrary to modularity, exhibits "in-link"direction to kernel. And we focus on kernels because they represent community core and help to concentrate nodes round them in order to better interpretation of the phenomena.

For detecting communities in directed networks, [8] presents some studies which propose a simple scheme that converts a directed graph into bipartite, undirected and weighted one [13, 19], this enabling to utilize the richness and complexity of existing methods to find communities in undirected graphs. Some hierarchical (agglomerative by *Louvain* [1] or separative called *Edge-Betweenness* algorithm [9]) methods based on the optimization of the so-called *modularity* measure [10] focus on the idea that networks with inherent community structure usually deviate from random graphs. Measuring the partition quality consists of demonstrating whether the expected fraction of edges is null. Yet, Fortunato in [2] describes how this measure has a limit resolution, the difficulty for the measure to extract small scale communities. This limit is remediable [6, 7] but results are not satisfactory in pattern-based clustering from real-world networks. Moreover, Malliaros and al. in their survey [8] present some methods that either focus on transforming a directed graph to an undirected one or describe some of quality measures in detail and also present their extensions to directed networks (e.g *d-modularity*). They confirm that these techniques ignore important information about the direction of the links and miss their semantic. Tsourakakis in [15] initiated the study of degree labelled triangle. He observed that the average number of triangles per degree follows a power-law distribution and the slope of the degree-triangle plot has the negative slope of the degree distribution plot of the corresponding graph. He also argues that low degree nodes form fewer triangles than higher degree nodes. Like this, to make up this limit, the objective of this new approach is to cluster low degree nodes so that they should be more linked together around a kernel so that they could access more easily to central retained information. So it takes into account in-links to the kernel and vertices with low degree.

3. COMMUNITY EXTRACTING METHOD

3.1. Basic terminology and concepts

Given a directed graph $G(V, E)$ with $n = |V|$ vertices and $m = |E|$ edges. An edge e_{ij} connects vertex v_i with vertex v_j. We now give some following useful definitions:

Definition 1 *(Neighbourhood Overlap). Given two vertices v_i and v_j , let Γ_i^{in} be the in-neighbourhood of vertex v_i, let Γ_j^{in} be the in-neighbourhood of vertex v_j . Let NO_{ij} be the in-neighbourhood overlap of v_i and v_j vertices. NO_{ij} is an Index Jaccard variant as:*

$$NO_{ij} = \frac{|\Gamma_j^{in} \cap \Gamma_i^{in}|}{|\Gamma_j^{in} \cup \Gamma_i^{in}| - \theta}$$

θ can take different values 0, 1 and 2, depending on the connectivity of v_i and v_j vertices.

Definition 2 *(Triad Weight). A triad is a subgraph (not necessarily induced) with 2 edges and 3 vertices, one of which is v_i and such that v_j is incident to both edges. The Triad Weight TW_{ij} of any edge e_{ij} in graph G can be defined as:*

$$TW_{ij} = \frac{|\Delta_{ij}|}{|\Delta_j|}$$

We use Δ_{ij} to represent a triad crossing both v_i and v_j according to the scheme presented in the Fig.1, and Δ_j to represent triad in which v_j is involved and is the target node of links.

Definition 3 *(The Kernel Degree). Let the kernel K be a set of central vertices(those owning a maximal central in-degree with a maximal overlapping neighbourhood). The Kernel degree of a pair of vertex v_i and v_j is:*

$$K_{ij} = TW_{ij} * NO_{ij}$$

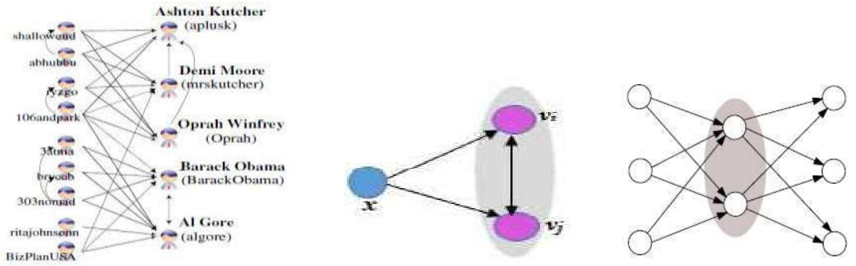
The first term is based on triads, and promotes the triad proportion through a kernel; and the second term promotes the neighbourhood proportion of the kernel nodes. Intuitively, *Kernel degree* can measure the strength of the kernel node similarity.

Definition 4 *(Kernel Community). Kernel community is a set of vertices with the same neighbourhood, such as these neighbours expand inward the kernel, according to the Kernel Degree K_{ij} gradually until a threshold.*

The threshold value is expressed in the section 3.3 below.

Given a graph $< G = (V, E) >$ and C_{ij} a subgraph of G containing v_i and v_j . C_{ij} is called Kernel Community initiated by v_i and v_j if: $\forall u, v \in C_{ij}, \forall w \notin C_{ij}$. We have the following properties:
Property 1 : $K_{uv} \geq K_{uw}$
Property 2 : $\exists t \in C_{ij} / \; t \in \alpha\Gamma_{uv}^{in}$
Property 3 : $|\Delta_{uv}| \geq |\Delta_{uw}|$

$\alpha\Gamma_{uv}^{in}$ is the *α-hop* common in-neighbourhood of vertices u and v (to reach u and v, in-neighbours cross α links) and $\Delta(X, Y) = \cup\Delta_{xy}, x \in X, y \in Y$ and $X, Y \subseteq V$. Then, $\Gamma_A^{in} = \cup_{a \in A}\Gamma_a^{in}$ with $A \subset V$. These properties indicate that: firstly, a *Kernel Degree* computed for a pair of nodes of the same kernel is higher than the *Kernel Degree* computed for nodes of which one belong to kernel and the other not. The standard deviation (discussed below) is the upper bound for kernel computation. Secondly, each node in a community initiated by u and v is in their common in-neighbourhood. To evaluate partitions, *Triad Density* of community initiated by u and v should be higher than the *Triad Density* of community not centred by kernel nodes. This method make use of centric-based approaches [4]. It is structured in three steps: ***(i)****Weighting scheme transformation*, ***(ii)****Kernel extraction*, ***(iii)****Node migration and community computation*.



(a) Extract for Twitter social network [16].

(b) Example of triads in which there is an edge between kernel vertices v_i and v_j

(c) Example of a citation pattern on the right.

Figure 2: Examples of structures expressing triads

3.2. Weighting scheme transformation

The weighting scheme consists on transforming the directed graph into undirected and weighted graph. It spreads out in two subtasks: Computing a pruned central list of node degree, and computing kernel degree corresponding matrix of the directed graph. *In-degree central list definition.* This step consists of determining a list called *LCentral* of in-degree centrality for each node, and put it in decreasing order. So that those with maximal in-degree are more eligible than those with a fair in-degree. Then, pruning from the list those of nodes with an in-degree below 2. This filtering step improves performance and allows simplifying assumptions later when deciding whether to include a vertex into a kernel. For instance, in a citation network, a node with an in-degree equal to 1 or 0 corresponds to an author whose the area search does not interest researchers, so removing these nodes with an in-degree below 2 improves the speed processing. Illustration from Extract for Twitter Network(2a) shows the central list *LCentral* contents as: $LCentral = ['AlGore', 'BarackObama', 'AshtonKutcher', 'DemiMoore', 'OprahWinfrey']$ because they have an in-degree above 2. *Kernel Degree matrix computing.* This step consists on computing kernel degree values K_{ij} (see Definition 3) for every pair $(v_i, v_j) \in LCentral$. As defined in section (section 3.1), The *Kernel Degree* computed from the both Neighbourhood Overlap NO_{ij} (see Definition 1) and Triad Weight TW_{ij} (see Definition 2) concepts, measures the strength of a kernel.

Then, constructing a Kernel Degree square matrix K of K_{ij} with n lines and n columns, where $n = |LCentral|$ is the *LCentral* pruned list size, i.e deprived of nodes with in-degree below 2 , and $K = (K_{ij})$. And finally, representing it through a list called *DicoK* of size $\frac{(n*n)-n}{2}$. n is removed because of the diagonal of matrix K whose K_{ii} values are null and the valuation is divided by 2 because of the symmetric matrix ($K_{ij} = K_{ji}$). Illustration on Extract for Twitter network shows values of K as:

$$K = \begin{pmatrix} 0 & 1.6 & 0.057 & 0.0635 & 0.267 \\ 1.6 & 0 & 0.0143 & 0.0158 & 0.32 \\ 0.057 & 0.0143 & 0 & 0.595 & 0.012 \\ 0.0635 & 0.0158 & 0.595 & 0 & 0.013 \\ 0.267 & 0.32 & 0.012 & 0.013 & 0 \end{pmatrix}$$

The corresponding vector from matrix is represented as $DicoK =$ [((‘DemiMoore’, ‘OprahWinfrey ’), 1.6), ((‘AlGore ’, ‘BarackObama ’), 0.595), ((‘AshtonKutcher ’, ‘OprahWinfrey ’), 0.32), ((‘AshtonKutcher ’, ‘DemiMoore ’),0.267), ((‘BarackObama ’, ‘DemiMoore ’), 0.0635), ((‘AlGore ’, ‘DemiMoore ’), 0.057), ((‘BarackObama ’, ‘OprahWinfrey ’), 0.0158), ((‘AlGore ’, ‘OprahWinfrey ’), 0.0143), ((‘BarackObama ’, ‘AshtonKutcher ’), 0.013), ((‘AlGore ’, ‘AshtonKutcher ’), 0.012)] and the list is in decreasing order of K_{ij}.

3.3. Kernel extracting approach

This task of extracting kernels focus on determining those of nodes more eligible to belong to kernel via interclass inertia, and thereafter, on constructing kernels via a threshold computation.

3.3.1. Interclass inertia computation

Given that the clustering main goal is to group homogeneous groups together, the criteria used here is *Inter-class Inertia I* based on *DicoK* vector. In fact, high inter-class inertia values indicate that objects tend to be more dissimilar, and consequently should belong to distinct groups. So, it divides objects into two groups(initiated by keys), those eligible to belong to a kernel and those not eligible. The delimitation of two groups of key (pair nodes) is done by a comparison of values from Inter-class

Inertia vector I with a computed *Standard Deviation* σ on I. Like this, node pairs (i, j) whose Inter-class Inertia value is upper than σ are more eligible to belong to kernels. The Inter-class Inertia between 2 groups G_1 and G_2 is expressed as

$$I(G_1, G_2) = |G_1|(\mu_1 - \mu)^2 + |G_2|(\mu_2 - \mu)^2 \tag{1}$$

$|G_1|$ and $|G_2|$ are respectively the number of edges in groups G_1 and G_2. μ_1, μ_2, and μ are respectively the *Kernel Degree* average for G_1, G_2 and G. Illustration on Extract for Twitter network presents distinct groups G_1 and G_2 respectively as the following, and the corresponding Inter-class Inertia of *DicoK* as : for $G_1 = \{$(DemiMoore, OprahWinfrey)$\}$ and $G_2 = \{$(AlGore, BarackObama), (AshtonKutcher, OprahWinfrey), (AshtonKutcher, DemiMoore), (BarackObama, DemiMoore), (AlGore, DemiMoore), (BarackObama, Oprah), (AlGore, OprahWinfrey), (BarackObama, AshtonKutcher), (AlGore, AshtonKutcher)$\}$, the Inter-class Inertia for these groups is 1.987. Then, the following pair of nodes in DicoK list moves from G_2 to G_1 and their contents become: $G_1 = \{$(DemiMoore, OprahWinfrey), (AlGore, BarackObama)$\}$ and $G_2 = \{$(AshtonKutcher, OprahWinfrey), (AshtonKutcher, DemiMoore), (BarackObama, DemiMoore), (AlGore, DemiMoore), (BarackObama, Oprah), (AlGore, OprahWinfrey), (BarackObama, AshtonKutcher), (AlGore, AshtonKutcher)$\}$, the Inter-class Inertia for these groups is 1.705. We change the G_1 and G_2 contents and so on; this leading to the Inter-class Inertia vector $I =$ [1.987, 1.705, 1.359, 1.162, 0.844, 0.627, 0.439, 0.297, 0.186, 0.131]. The end of this preprocessing step consists on computing a threshold value beyond of which vertices on *Dicok* are the more eligible to form kernels, as detailed in the next section.

3.3.2. Kernel extraction

To compute Kernels, we focus on a threshold, which is the standard deviation from interclass inertia. So, each vertex must decide if it could belong to the *key membership*(the *kernel initiator vertex* belonging to the pair with the highest corresponding Inter-class Inertia). Inspired by properties of real-life networks [3] based on a power-law degree distribution meaning, the basic idea behind the *kernel degree* metric is that vertices should close more triads with other vertices in the community than with vertices outside of the community. Using this idea, the current phase consists on extracting kernels which are *seeds* or nodes centralizing information through the "in-link"direction by a comparison of the *Inter-class inertia* to the threshold σ, that is the Standard Deviation from that Inter-class Inertia set of node groups.

Standard deviation σ. A low standard deviation indicates that the data points tend to be closed to the mean of the set, while a high standard deviation indicates that the data points are spread out over a wider range of values. Because of the power-law degree distribution in real-life networks, very little nodes get a high in-degree widely above the in-degree average. Like this, we could make the assumption that the higher the standard deviation σ of a node set, the more likely they possess the almost common neighbourhood. As matter of fact, as shown in experiments, a lower standard deviation indicates that these vertices have a quasi-null common-neighbourhood cardinality. This assumption is also applied in [11], but the computation method presented there is not adapted to the vertex centric processing model. The standard deviation formula is :

$$\sigma = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(x_i^2) - \mu^2} \tag{2}$$

where $\mu = \frac{1}{n}\Sigma_{i=1}^{n} x_i$ indicates s_i average (or mean), and x_i indicates every element of the interclass inertia array. The implementation for kernel communities is presented in Algorithm 1.

Let us demonstrate this idea through an example. Consider the network in Figure 2a. The model computes the first kernel initialized by nodes '*Demi Moore*'and '*Oprah Winfrey*'for which the associated inertia in I is $1.987 \geq 0.62$ (where 0.62 is the value of the standard variation σ); thereafter, the vertex '*Ashton Kutcher*'integrates the other, inducing like this a kernel of 3 vertices; the second kernel is initialized by '*Al Gore* 'and '*Barack Obama* 'for which the associated inertia in I is $1.705 \geq 0.62$. The process is repeated on the other i values on I for which $I[i] \geq \sigma$; and if the corresponding $DicoK[i]$ pair nodes are already keys or associated values of keys, they are just omitted.

3.4. Community computing process

After extracting kernels, it remains the other nodes not into the kernels, called *non-kernels* vertices. The process of generating *global communities* (communities containing both kernels and non-kernels vertices) consists on migrating *non-kernels* vertices to the kernel with whom they have a maximal connexion. It is an iterative optimization of the number of connexion each nonkernel vertex own with the kernel.

4. ILLUSTRATION AND EXPERIMENTS

In this section, we illustrate some graphs analysis based on some criteria and we show experiment results. We evaluate a variety of models on two main tasks: Triad density partition and quality measures evaluation. In the following experiments,

Algorithm 1 Implementation for kernels extraction

```
Require: Directed graph $G = (V, E)$
Require: $DicoK$ vector of node pairs associated to their kernel degree
Require: $I$ inter-class inertia vector //corresponding to the vector $I$ in the explanation above.
Ensure: Structured-by-key Kernels set called $ListK$
Initialisation : $S_k = \sigma$ , $ListK = \emptyset$;
for each item $u \in I$ do
    $(p, q) \longleftarrow$ pair of classes from the item $u$
    if $u >= S_k$ then
        for each distinct item $p_j \in p$ do
            if $p_j \notin Key$ then
                $Key \longleftarrow \{p_j\}$
                Label each $Key$ vertex
            else
                for each unlabelled item $i \in p$ class do
                    if $i \in p$ class Such as $I_{pq} >= S_k$ then
                        $Key \longleftarrow Key \cup \{i\}$
                    end if
                end for
            end if
            $ListK \longleftarrow ListK \cup \{Key\}$
        end for
    end if
end for
$Return\ ListK$
```

we use twiter illustration network, neural network (Celegansneural), blog network (Polblogs) and two paper citation (Cora and Citeseer) networks. Information about each graph can be found in Table 1.

Table 1: Characteristics of the test graphs.

Networks	Nodes	Edges	Comm
Extract for Twitter	14	32	2
Celegansneural	297	$2,345$	5
Polblogs	$1,490$	$19,090$	–
Citeseer	$3,327$	$4,732$	–
Cora	$2,708$	$5,429$	–

4.1. Kernel degree metric and threshold σ evaluation

To demonstrate the idea of the *Kernel Degree* formula, let us consider for example two networks : Twitter illustration network (see Figure 2a) and Celegansneural network, for better visualization of results. *Kernel Degree* computes the similarity strength between kernel vertices. Both the Neighbourhood overlap (Definition 3) and Triad Weight (Definition 4) are associated to reinforce this similarity. Because, when taken separately, the expected results are not purchased, as presented in the Table 2. In fact, for the twitter illustration network, results are the same regardless of the criteria (2 communities with the same triad density and same modularity). But for the Celegansneural network, using separately Neighbourhood overlap or Triad weight leads to results (91 and 73 communities respectively) far from expected one as demonstrated by Klymko and Tianbao [5, 18] who detect 5 communities, with a better triad density for *Kernel Degree* of 0.32.

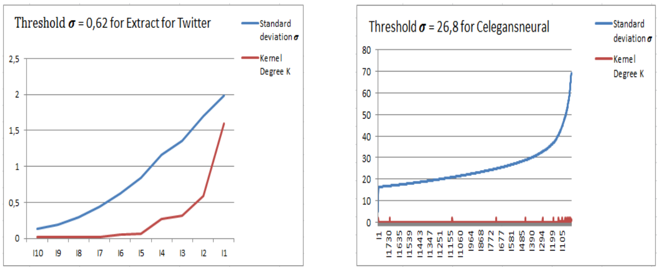
As far as the threshold σ is concerned, the empirical experiments show that when taking values of the interclass inertia less than σ, expected results are not produced. Extract for Twitter social network possesses 1 community and Celegansneural 103 communities. As can be seen from the Table 3, the new approach performs the best in both datasets. The Twitter illustration network, for the first case ($I[e_{ij}] > \sigma$) contains 2 communities with a triad density of 0.6428, contrary to the second case ($I[e_{ij}] < \sigma$) for which Twitter network just contains 1 community with a low triad density of 0.417. This result means that the Twitter partition is not well structured for this second case. Higher standard deviation values indicate better kernel based-triad structures (see Figure 3) and therefore, finding vertices with similar neighbours whose Kernel degree is upper than threshold provides a method for extracting the underlying kernel structure.

Table 2: Using metric Comparison.

		Twitter		**Celegansneural**
	#Comm	Triad Density	#Comm	Triad Density
Kernel-degree	2	0.64	5	0.32
Neighbourhood Overlap	2	0.64	91	0.20
Triad Weight	2	0.64	73	0.254

Table 3: σ choice evaluation.

		Twitter		**Celegansneural**
	#Comm	Triad Density	#Comm	Triad Density
$I[e_{ij}] > \sigma$	2	0.6428	5	0.711
$I[e_{ij}] < \sigma$	1	0.417	103	0.065



(a) threshold for Extract for Twitter network

(b) threshold for Celegansneural network

Figure 3: Standard deviation distribution based on threshold

5. CONCLUSION AND FUTURE WORK

This paper has described a simple kernel scheme to improve the detection of communities in directed networks, through triad cardinality. It focus on kernels which are seed nodes centralizing information through their in-degree valuation. Based on the definition of community as a subgraph induced by kernel vertices, the new scheme basis are triads, meaning the semantic relationship between kernel nodes with their neighbours. Thus, we have defined a new metric called *Kernel Degree*, for computing the similarity between kernel nodes. When the new metric is used, we obtained better triad density values than modularity on datasets. Our model captures semantic communities based on both criteria: density and topology of graphs (graphs with power-law degree distribution and communities with higher triad density). We compared the modularity values for each model on the result partition, and the new approach presents a better modularity on its partition than the other approaches.

The model complexity constitutes a main criteria of effectiveness for any method. With the increasing ways on information access by the era of digital, it becomes important to extend this method to parallel processing, in order to manipulate very large-scale real networks. Likewise, it is possible to apply weighted graphs reinforcing the strength of the kernel, for more community detection results near the real life, as we will study in our future works.

6. REFERENCES

[1] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[2] Santo Fortunato. Community detection in graphs. *Physics reports*, 486(3):75–174, 2010.

[3] Rushed Kanawati. Détection de communautés dans les grands graphes d'interactions (multiplexes): état de l'art. 2013.

[4] Rushed Kanawati. Seed-centric approaches for community detection in complex networks. In *International Conference on Social Computing and Social Media*, pages 197–208. Springer, 2014.

[5] Christine Klymko, David Gleich, and Tamara G Kolda. Using triangles to improve community detection in directed networks. *arXiv preprint arXiv:1404.5874*, 2014.

[6] Gautier Krings and Vincent D Blondel. An upper bound on community size in scalable community detection. *arXiv preprint arXiv:1103.5569*, 2011.

[7] Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117, 2009.

[8] Fragkiskos D Malliaros and Michalis Vazirgiannis. Clustering and community detection in directed networks: A survey. *Physics Reports*, 533(4):95–142, 2013.

[9] Mark EJ Newman. Detecting community structure in networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):321–330, 2004.

[10] Vincenzo Nicosia, Giuseppe Mangioni, Vincenza Carchiolo, and Michele Malgeri. Extending the definition of modularity to directed graphs with overlapping communities. *Journal of Statistical Mechanics: Theory and Experiment*, 2009(03):P03024, 2009.

[11] Arnau Prat-Pérez, David Dominguez-Sal, Josep M Brunat, and Josep-Lluis Larriba-Pey. Shaping communities out of triangles. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, pages 1677–1681. ACM, 2012.

[12] A. Prat-Pérez and D. Dominguez-Sal. High quality, scalable and parallel community detection for large real graphs. In *Proceedings of the 23rd International Conference on World Wide Web*, pages 225–236. WWW '14. New York, NY, USA: ACM, 2014.

[13] Venu Satuluri and Srinivasan Parthasarathy. Symmetrizations for clustering directed graphs. In *Proceedings of the 14th International Conference on Extending Database Technology*, pages 343–354. ACM, 2011.

[14] B. Serrour and S. Arenas. Detecting communities of triangles in complex networks using spectral optimization. *Computer Communications*, 34:629–634, 2011.

[15] Charalampos E Tsourakakis. Fast counting of triangles in large real networks without counting: Algorithms and laws. In *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*, pages 608–617. IEEE, 2008.

[16] Liaoruo Wang, Tiancheng Lou, Jie Tang, and John E Hopcroft. Detecting community kernels in large social networks. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 784–793. IEEE, 2011.

[17] S Wasserman and Katerin Faust. Social networks analysis : Methods and applications. *Physics reports*, 486(3):75–174, 1994.

[18] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. Directed network community detection: A popularity and productivity link model. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pages 742–753. SIAM, 2010.

[19] Denny Zhou, Thomas Hofmann, and Bernhard Schölkopf. Semi-supervised learning on directed graphs. In *Advances in neural information processing systems*, pages 1633–1640, 2005.

C.6 Social Network Analysis : Novel method to find directed community structures based on triads cardinality

Rubrique

Social Network Analysis

Novel method to find directed community structures based on triads cardinality

Gamgne Domgue Félicité[*] — Tsopze Norbert[*] — René Ndoundam[*]

[*] Computer Science Department - University of Yaounde I
BP 812 Yaounde - Cameroon
felice.gamgne@gmail.com, tsopze@uy1.uninet.cm, ndoundam@gmail.com

RÉSUMÉ. La détection des communautés est davantage un challenge dans les l'analyse des réseaux orientés. Plusieurs algorithmes de détection de communautés ont été developpés et considèrent la relation entre les nœuds comme symmétrique, car ils ignorent l'orientation des liens, ce qui biaise les résultats en produisant des communnautés aléatoires. Ce document propose un algorithme plus efficace, TRICA, basé sur l'extraction des kernels qui sont des ensembles de nœuds influents dans le réseau. Cette approche découvre des communautés plus significatives avec une complexité temporelle meilleure que celles produites par certains algorithmes de détection de communautés de l'état de l'art.

ABSTRACT. Community structure extraction is once more a major issue in Social network analysis. A plethora of relevant community detection methods have been implemented for directed graphs. Most of them consider the relationship between nodes as symmetric by ignoring links directionality during their clustering step, this leading to random results. This paper propose TRICA, an efficient clustering method based on kernels which are influencial nodes, that takes into account the cardinality of triads containing those influencial nodes. To validate our approach, we conduct experiments on some networks which show that TRICA has better performance over some of the other state-of-the-art methods and uncovers expected communities.

MOTS-CLÉS : Réseaux orientés, détection des communautés kernel, Triade

KEYWORDS : Directed graphs, Community kernel detection, Triad.

1. Introduction

Community detection in directed networks appears as one of dominant research works in network analysis. The top meaning of community is a set of nodes that are densely connected with each other while sparsely connected with other nodes in the network [1]. This definition is interesting for undirected graphs ; like this many community detection algorithms implemented for directed networks simply ignore the directionality during the clustering step while other technics transform the directed graph into an undirected weighted one, either unipartite or bipartite, and then algorithms for undirected graph clustering problem can be applied to them.

These simplistic technics are not satisfactory because the underlying semantic is not retained. For example, in a food web network, according to them, the community structure will be corporated of predator species with their prays. This reflexion is not quite right. To make up for that idea, a generic definition of community detection consists of clustering nodes with homogeneous semantic characteristics(nodes centred around a set of objects owning the same interest). Our approach is based on extending the idea that within "good"communities, there are influencial nodes [6], ***kernels***, that centralize information, so that it will easily be attainable. Influancial nodes are crossed by a maximal number of triads in a community. A triad is a set of 3 nodes whose at least 2 are the *in-neighbor nodes* (target vertices) of the 3^{rd} vertex, or according to the triadic closure. Consequently, triads are the basis of many community structures [3]. Here we focus on the link orientation in triads. The specific contributions of our paper are :

– we mainly define a new concept named *kernel degree* to measure the strength of the pair of nodes and the similarity of vertices and give a new sense definition to kernel community based on the triadic closure.

– we develop a novel algorithm based on kernel degree to discover kernels and then communities from real social networks.

– We conduct to better quality improvement over the commmunity kernel detection algorithms.

The rest of paper is organized as follows. Section 2 is an introduction to related works. In Section 3, we formally define several concepts used into the proposed clustering method. In Section 4, we develop the algorithm. Section 5 is experiment study and Section 6 concludes this study.

2. Related works

Most approaches focused on symmetric models which lose the semantics of link directions, a key factor that distinguishes directed networks from undirected networks. For detecting communities in directed networks [2], some studies propose a simple scheme that converts a directed graph into undirected one, this enabling to utilize the richness and complexity of existing methods to find communities in undirected graphs, thus, to mesure cluster strength, they use an objective function, *the modularity*. Yet, this mesure has a limit resolution [1]. More recently, various probabilistic models have been proposed for community detection [7]. Among them, stochastic block models are probably the most successful ones in terms of capturing meaningful communities, producing good performance, and offering probabilistic interpretations. However, its complexity is enough because in pratice, if the number of iterations goes beyond 20, the method discontinue and results become insignificant. To make up for this complexity, some authors define “kernels”like described below.

A *kernel* is considered as a set of influencial nodes inside a group. It seems to be information centralizing nodes. Some methods explored the problem of detecting community kernels, in order to either reduce the number of iterations, and consequently the time-complexity of algorithms defined for complex social networks or uncover the hidden community structure in large social networks. [4] identifies those influential members, *kernel* and detects the structure of community kernels and proposed efficient algorithms for finding community kernels. Through these algorithms, there is a random choice of the initial vertex, and the size of communities is fixed, leading to an arbitrary result estimation. To keep going, [3] proved that triangles(short cycles) play an important role in the formation of complex networks, especially those with an underlying community structure [5] and converts directed graph into an undirected and weighted one. This transformation misses the semantic of links. We propose a method which extracts triads based on Social properties to characterize the structure of real-world large-scale networks.

3. Method formalization

We propose in this section the kernel community model and introduce several related concepts and necessary notations.

3.1. Kernel community model

In directed networks, the link direction gives a considerable semantic to the graph and to the information flow. On twitter network for example, the notion of authority is pointed up as illustrated in Fig 1.(a), because of the relationship between a set of authoritative or hub blogs (nodes u and v) and a set of non-popular one called followers (nodes x) as presented in Fig 1.(b) and Fig 1.(c).

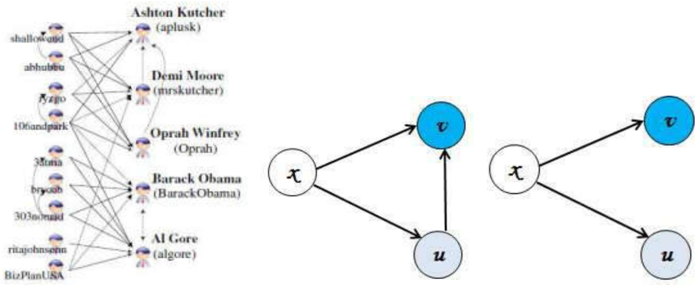
We integrated this concept of authority as one concept named **kernel degree**. Fig 1.(a) is a visualization of an extract from a twitter network. Kernel communities consist of nodes owning the same “in-neighbourhood ”which corresponds to nodes that have more connections to the kernel (and not from the kernel) than a vertex outside the kernel. We consider only ingoing edges to the kernel vertices to express the strength these nodes get in some kind of network treated in this paper; in a twitter network for example, hub blogs are viewed by many others followers and not the opposite; in a citation network for example, authoritative authors like pioneers in a research area are more quoted by the others junior researchers. On the beginning, the kernel consists of two vertices sharing the same properties, leading to the notion of “triad ”which consists of the idea that two vertices of the kernel share the same friend, like defined in the following sub-section.

3.2. Basic terminology and concepts

Given a directed graph $G(V, E)$ with $n = |V|$ vertexes and $m = |E|$ edges. Let Γ_u be the neighborhood vertices set of vertex u. We now give some following useful definitions :

Definition 1 (*Triad weight*). Let the identifier of vertex x in G be j. The triad weight of any edge (u, v) in graph G can be represented as Δ. We can use TW_{uv} to represent the number of triads (triad cardinality) crossing u and v according to the scheme presented in the Fig 1.(b) and Fig 1.(c).

$TW_{uv} = \frac{|\Delta_{uv}|}{|\Delta_j|}$.



(a) Illustration of Twitter Network (b) Closed triad (c) Opened triad

Figure 1. *Basic structures of our kernel community model.*

Definition 2 (*Neighborhood overlap*). Given two vertices u and v, let Γ_u be the set of vertices that are the neighborhood of vertex u, let Γv be the set of vertices that are the neighborhood of vertex v. Let NO_{uv} be the neighborhood overlap of u and v. $NO_{uv} = \frac{|\Gamma_v \cap \Gamma_u|}{|\Gamma_v \cup \Gamma_u|-2}$ if there is an edge between u and v and 0 otherwise.

Definition 3 (*The kernel degree*). The Kernel degree of a pair of vertex u and v is : $K_{uv} = TW_{uv} * NO_{uv}$. K_{uv} can measure the strength of the pair (u, v) and the similarity of nodes.

Definition 4 (*New sense Kernel Community*). A new definition of the kernel community in the sense of this paper is a set of vertices with the same neighborhood such as these neighbors expand inward to the kernel, according the kernel degree K_{uv} gradually until its minimum.

Definition 5 (*Triadic Closure*). If two people in a social network have a friend in common, then there is an increased likelihood that they will become friends themselves at some point in the future.

The algorithm is structured into two steps : detecting kernel communities and then migrating the others vertexes to the kernel to whom they are more connected to.

4. Our Method for extracting communities

The new algorithm is structured in two steps : identifying kernels, then migrating the other vertices to the kernel as described in the following subsections. The algorithm for extracting Kernel communities, TRICA (Triads Cardinality Algorithm) we propose here makes use of a new concept *Kernel degree*, that measures the strength of a kernel gradually until it decreases. This concept is based on the triadic closure for emphasis the semantic proximity that links community members conducting to efficient propagation of information over the network. We focus on triads cardinality that is the number of neighboors two nodes own.

Data set	Vertices	Edges	Types
Extract from Twitter Network	14	31	Directed
American Football Network	115	613	Undirected
Celegansneural	297	2359	Directed

Tableau 1. *Data sets description*

4.1. TRICA algorithm

We assume that the network we want to analyze can be represented as a connected, directed, nonvalued graph G of $n = |N|$ nodes and $m = |E|$ edges. This step for identifying kernels is described in four sub-steps as follow :

1) Detect the *in-central* vertex v, which is the vertex with the maximal in-degree in the graph.

2) Determine the neighborhood overlap of each edge *(u,v)* through a variant of *Jaccard Index*[1] represented by NO_{uv} as defined in **Definition 2**

3) Store neighboorhood vertices u of v like $NO_{uv} > \varepsilon$

4) Compute K_{uv} through the *triad weight* TW_{uv} as described in **Definition 1**. This action is repeated to measure the strength of a kernel gradually until K_{uv} decreases.

These 4 substeps are repeated n/k times, k being the *in-degree* of vertex v.
The space complexity of TRICA is $O(n+m)$, and it runs in time more quickly than some of the state-of-the-art algorithms like shown in experiments.

The TRICA implementation for kernel communities is presented in Algorithm 1.

4.2. Deduction of global communities

After extracting kernels, it remains the other nodes which don't belong to the kernels ; they are called *non-kernels vertices*. The process of generating *global communities* (communities containing both kernels and non-kernels vertices) consists of migrating the other members (belonging to a set called "auxiliary communities") to the kernel whith whom they have a maximum number of connections, as described in Algorithm 2.

Algorithm 1 TRICA implementation for kernels extraction

```
Data: Directed graph $G = (N, E)$
Result: $K$ Kernels
Initialisation : $K = \emptyset$ ;
repeat
    $k = d^{in}(v)/d^{in}(v) = max\{d^{in}(t), \forall t \in V\}$ ;
    Calculate $NO_{uv}$ for each $(u, v) \in E$ ;
    $\Gamma_v[] \leftarrow \{t \in V/\exists t \in V, NO_{tv} > 0,8\}$ ;$\Gamma_v[].sort$ ;$i \leftarrow 1$ ;
    $S \leftarrow \emptyset$ ;
    $j \leftarrow i$ ; $u \leftarrow \Gamma_v[j]$ ;$K^*_{uv} \leftarrow 0$ ;
    repeat
        Compute $K_{uv}$ ;
        if $(K_{uv} > K^*_{uv})$ then
            $S \leftarrow S \cup u$ ;
        end if
        $u \leftarrow \Gamma_v[i++]$ ;
    until $(K_{uv} < K^*_{uv})$ ;
    $K \leftarrow K \cup S$ ;
until $(|V|/k)$
Return $K$ ;
```

5. Experiments

To study the effectiveness and accuracy of TRICA, we compare it with following comparative methods :

– NEWMAN : Method for finding community structure in directed networks using the betweenness based on modularity [6].

Algorithms	Extract from Twitter		American Football		Celegansneural	
	%Δ	Comm Numb	%Δ	Comm Numb	%Δ	Comm Numb
Newmann	98%	2	39%	10	28%	194
Louvain	98%	2	63%	9	35%	5
Weba	98%	2	-	8	-	-
Triad Cardinality	**98%**	2	**70%**	12	**64%**	21

Tableau 2. *Community detection performance on the triad cardinality rate where the best rate are in bold.*

– LOUVAIN : Community detection algorithm based on modularity ; (we use Gephi tool for visualizing LOUVAIN results).

– WEBA [4] :Algorithm for community kernel detection in large social networks.

Algorithm 2 Algorithm implementation for non-kernels vertices migration

```
Data: Communities Kernels $K = \{K_1, K_2, ..., K_t\}$
Result: Global Communities $G_K = \{G_{K_1}, G_{K_2}, ..., G_{K_t}\}$
  Let $N$ be set of auxiliary communities ; $N = \{N_{K_1}, N_{K_2}, ..., NG_{K_t}\}$ ;
  $\forall i \in \{1, ..., t\}, G_{K_i} = \emptyset$;
  repeat
    $\forall i \in \{1, ..., t\}, G_{K_i} = K_i \cup N_{K_i}$ ;
    For $i \leftarrow 1$ to $t$ do
      $S \leftarrow \{v \notin \cup G_{K_i} / \forall j \in \{1, ..., t\}\},$
      $|E(v, G_{K_i})| \geq |E(v, G_{K_j})| > 0$ ;
      $N_{K_i} \leftarrow N_{K_i} \cup S$ ;
      $G_{K_i} \leftarrow K_i \cup N_{K_i}$ ;
    End For
  until (No more vertices can be added)
  Return $G_K$ ;
```

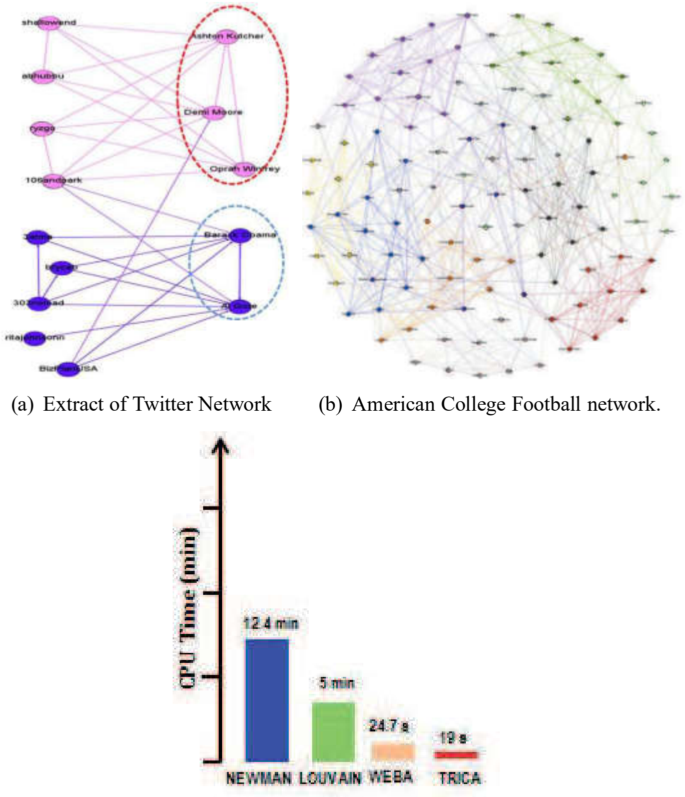
Our method is evaluated on directed and undirected networks. We use two levels of evaluation : The first is based on the time complexity, and the second on the **triad cardinality rate in communities**, that is the percentage of communities in the partition with highest triad cardinality rate. We use the function TCR defined as following, to evaluate our method :

$TCR = \frac{\Sigma_i |\Delta_i|}{|\Delta|}$

where i is one community and $|\Delta|$ the number of triads.

When we apply TRICA on the data sets described in the table 1, results in Fig 2 are following : The Fig 2.(a) illustrates the 2 expected communities of the Extract from Twitter Network, for all of the methods compared, with a triad cardinality rate in communities of 98% with kernels and followers [6]. But TRICA CPU time is better than other methods CPU time, as shown if 2.(c) The table 2 summarizes the comparison with some state-of-the-art methods. It shows that Triad Cardinality algorithm provides the highest triad cardinality rate in communities. As far as the Football network is concerned, Triads cardinality algorithm can divide the network into 12 communities exactly as shown in Fig 2.(b). In this result, 8 communities are completely consistent, this revealed by the triad cardinality rate of 70%. Meanwhile Newmann algorithm can divide it into 10 communities and LOUVAIN into 9. This number of communities does not reflect the real structure of the American College Football network. On the other hand, the result for applying

TRICA to Celegansneural network shown in Table2 presents that TRICA detects 21 communities, while LOUVAIN detects 5 and NEWMAN 194. But the triad cardinality rate is the best, 64%, certifying that our method uncovers a better structure of social networks.



(a) Extract of Twitter Network (b) American College Football network.

(c) Efficiency comparison of TRICA and others algorithms on Twitter Network.

Figure 2. *Results of applying TRICA to data sets.*

6. Conclusion

In this paper, we focus on the problem of kernel community detection in directed graphs, kernels being the key tool for understanding the role of networks and its structure. We mainly interested on extracting kernels which are influential nodes on the network. Our kernel community model define triads according to some social properties to characterize the structure of real-world large-scale network, and we develop a novel method based on the proposed new concept, the *kernel degree* which defines the strength of kernel community. Experiments proved that TRICA detects efficiently expected communities and achieves 20 % performance improvement over some other state-of-the-art algorithms, but it only works for unweighted graphs. Our next work is to optimize Triad cardinality-based property, and adjust it to suit for detecting kernel communities from large-scale directed and weighted networks.

7. Bibliographie

[1] S. FORTUNATO, « Community detection in graphs », *Physics Reports* 486(3) 75-174, 2010.

[2] F. D. MALLIAROS and M.VAZIRGIANNIS, « Clustering and community detection in directed networks : A survey. », *arXiv* 1308.0971, 2013.

[3] C.KLYMKO , D.F GLEICH and T.G KOLDA, « Using Triangles to Improve Community Detection in Directed Networks », *Conference Stanford University*.

[4] LIAORUO WANG , TIANCHENG LOU , JIE TANG and JOHN E. HOPCROFT, « Detecting Community Kernels in Large Social Networks ».

[5] A. PRAT-PÉREZ , D. DOMINGUEZ-SAL , J. M. BRUNAT and J. L. LARRIBA-PEY, « Shaping communities out of triangles. », *In CIKM 12* n° 1677-1681, 2012.

[6] FÉLICITÉ GAMGNE and NORBERT TSOPZE, « Communautés et rôles dans les réseaux sociaux », *in : CARI '14 : Proceedings of the 12th African Conference on Research in Computer science and Applied Mathematics* n° 157 - 164, 2014.

[7] TIANBAO YANG , YUN CHI , SHENGHUO ZHU and YIHONG GONG and RONG JIN,« Directed network community detection : A popularity and productivity link model. », *In SIAM Data Mining'10* n° 2010.

C.7 Analyse des réseaux sociaux : Communautés et rôles dans les réseaux sociaux

Rubrique

Analyse des réseaux sociaux

Communautés et rôles dans les réseaux sociaux

Gamgne Domgue Félicité* — Tsopze Norbert*

* Département d'Informatique - Université de Yaoundé I
BP 812 Yaoundé - Cameroun
felicitedomgue@yahoo.fr, tsopze@uy1.uninet.cm

RÉSUMÉ. Cet article s'intéresse à la détection des communautés et des rôles dans un réseau social modélisé à l'aide d'un graphe orienté. Premièrement, il aborde la problématique de la détection des communautés dans les réseaux sociaux en adaptant l'algorithme de Girvan et Newman aux réseaux modélisés par des graphes orientés et pondérés. Il propose un algorithme séparatif qui s'inspire de celui de M.Girvan et M.Newman et détecte les composantes connexes via la centralité d'intermédiarité en utilisant une fonction de qualité, la modularité. Deuxièment, il définit les rôles des nœuds des communautés découvertes à l'issue de la première phase. Pour cela, il détecte le nœud central ou « leader »en utilisant les mesures de centralité, ainsi que des méthodes pour détecter d'autres rôles tels que l'« externer »ou externeur et le « follower »ou suiveur.

ABSTRACT. We are interested in the detection of communities and roles in complex social network oriented and weighted networks. Firstly, the detection of communities in oriented and weighted graph modeling a social network is presented, using Girvan and Newman algorithm. An adapted Girvan and Newman algorithm is proposed for this issue. Secondly, we focus roles, using the degree-betweeness to detect the central node and we propose methods for other roles like externer and follower.

MOTS-CLÉS : Réseaux sociaux, détection des communautés, détection des rôles, modularité.

KEYWORDS : Social networks, detection of communities, roles of nodes, modularity.

1. Introduction

Les réseaux sociaux sont des structures modélisant les relations sociales (par exemple l'amitié, la collaboration, la parenté, etc.) qui existent entre un ensemble d'individus appelés aussi acteurs. Ces réseaux sont généralement modélisés par un graphe dans lequel les sommets correspondent aux entités sociales (individus ou acteurs) et les liens correspondent aux relations sociales. La détection des communautés et l'identification des nœuds principaux (leaders ou ceux des individus qui mènent le groupe, externers qui sont ceux qui diffusent les informations dans les communautés ainsi constituées, et followers, qui sont les adeptes des communautés) constituent le but de ce travail.

Plusieurs travaux de recherche se sont intéressés à la détection des communautés, une communauté étant un ensemble de noeuds densément connectés entre eux et faiblement connectés avec les autres noeuds du graphe. La modularité est le critère de qualité le plus employé. Une étude comparative des algorithmes de communautés a été présentée dans [2] . Selon cette étude, l'algorithme de Girvan et Newman [1] est l'un des meilleurs en termes de modularité [4] et complexité ; mais reste limité aux graphes non orientés. Nous proposons dans ce travail primo d'adapter l'algorithme de Girvan et Newman au graphes orientés ; secundo, nous proposons des heuristiques pour le choix de la partition à segmenter. Nous identifions en nous basant sur d'autres heuristiques les rôles joués par chaque nœud dans les communautés ainsi constituées. Les motivations de ce sujet sont multiples. Les graphes d'appels téléphoniques (GAT) sont un cas pratique à modéliser par un graphe orienté, puisque les appels et les SMS vont toujours d'un émetteur vers un récepteur. Dans un GAT, nous pouvons déterminer ceux des individus qui collaborent ensemble, vu la fréquence des appels passés, et nous identifions les rôles de chaque noeud dans chacune des communautés détectées. La détection de l'externer dans une communauté peut permettre de limiter la propagation d'une épidémie en l'empêchant de continuer à avoir des contacts avec l'extérieur. Une comparaison expérimentale de cette approche à l'algorithme de détection de communautés de Louvain implémenté dans l'outil Gephi [5] montre que pour les données utilisées, notre approche trouve des communautés avec une modularité plus élevée que celle obtenue de l'algorithme de Louvain [3].

La suite sera organisée de la manière suivante : dans la deuxième section, nous décrirons sommairement quelques méthodes de détection de communautés. D'autres méthodes de détection des communautés dans les graphes orientés et non orientés sont décrits dans [4]. Dans la troisième section, nous présenterons d'abord l'algorithme de Girvan et Newman, puis notre adaptation à la détection des communautés dans le cas des graphes orientés. La quatrième section sera consacrée à la détection des rôles. Nous continuerons par des expérimentations de notre proposition sur un graphe d'appels téléphoniques et sur les données d'un club de karaté. Enfin nous terminerons par une conclusion et des perspectives pour ce travail.

2. Détection de communautés

L'identification de structure de communautés (ISC), appelée aussi détection ou extraction de communautés, a pour but d'identifier toutes les communautés présentes dans un graphe donné. Une structure de communautés dans un graphe $G = (V, E)$ est un ensemble $C_1, C_2, ..., C_k$ tel que :$C_1 \cup C_2 \cup ... \cup C_k = V$ et chaque C_i vérifie la définition de communauté considérée. Cette définition passe par les approches classiques de partitionnement de graphe pour parvenir à des méthodes de détection hiérarchiques ascendantes et descendantes.

2.1. Approches classiques

Il s'agit des méthodes dont l'objectif se rapproche de celui de la détection des communautés. Deux techniques y sont abordées : le bi-partitionnement de graphes et la segmentation des graphes. La première technique cherche à répartir en deux groupes les tâches représentées par les sommets d'un graphe, tout en minimisant les échanges, représentés par les arêtes. Dans cette technique, on peut classer les deux méthodes : la méthode de bissection spectrale et la méthode de bissection en coupe minimale [4]. En ce qui concerne la deuxième technique, il s'agit de voir le problème de détection de communautés comme un problème de classement, et d'analyse générale des données, dans lequel on cherche à regrouper les objets possédant des caractères communs, i.e. respectant les mêmes critères de similarité. La segmentation est une méthode basée sur les nœuds puisqu'elle satisfait certaines propriétés, telles que la mutualité complète (clique : graphe dans lequel tous les sommets sont interconnectés) et l'accessibilité de k membres (k-clique, k-club, k-moyennes, k-hop)[8] qui sont les plus usuelles.

En effet, Girvan et Newman [1] orientent leur idée dans le découpage d'un graphe en communautés avec pour souci la réduction des liens intercommunautaires. Bien que la détection des communautés ait le même but que ces approches dites classiques, elle a ceci de plus que le nombre de communautés et leurs tailles sont inconnus, et le plus important est que la détection des communautés permet de reconnaitre les réseaux ne possédant pas une structure modulaire. Cependant, ils ne s'intéressent pas aux graphes orientés et pondérés. Le paragrphe suivant présente les algorithmes dits hiérarchiques.

2.2. Algorithmes hiérarchiques

Ces méthodes cherchent à diviser le graphe en des structures selon leurs connexions : c'est une approche purement topologique. Encore appelées approches de clustering hiérarchique, elles construisent plutôt une hiérarchie de partitions représentée sous la forme d'un dendrogramme. Les algorithmes de classification hiérarchiques sont de deux types : les méthodes déscendantes [4] dites séparatives et les méthodes ascendantes dites agglomératives [4].

L'idée des méthodes séparatives est de considérer au départ le graphe comme une seule communauté et de diviser progressivement, jusqu'à l'obtention d'un graphe vide c'est-à-dire sans arêtes. Les méthodes existantes diffèrent par la façon de choisir les arêtes à retirer. Parmi ces méthodes, l'une des plus répandues est l'algorithme de Girvan et Newman [4] que nous allons adapter à d'autres types de graphes.

3. Algorithme de Girvan et Newman et adaptation

3.1. Algorithme de Girvan et Newman

L'algorithme de Girvan et Newman [1] constitue l'un des plus usuels algorithmes basés sur le clustering hiérarchique déscendant. Il comporte les étapes suivantes :

1) Calcul de la centralité d'intermédiarité pour chaque arête du graphe connexe de départ

2) Retirer du graphe l'arête de plus grande centralité d'intermédiarité

3) Calculer la modularité de chacune des composantes connexes C_i identifiées

4) Réitérer 2) et 3) jusqu'à l'obtention d'un graphe vide

5) Retourner la partition possédant la plus grande modularité

3.2. Adaptation de l'algorithme au graphe orienté

Pour détecter les communautés dans les graphes orientés et pondérés avec cet algorithme, nous faisons des suppositions suivantes :

1) Nous ne traitons que des graphes connexes ;

2) La segmentation du graphe de départ consiste en la suppression du lien apparaissant le plus grand nombre de fois dans l'ensemble des plus courts chemins déterminés entre tous les noeuds du graphe ;

3) Le plus court chemin est celui dont la somme des poids des arcs qui le composent est minimale.

Afin d'éviter de construire un dendrogramme comme l'algorithme de Girvan et Newman, nous allons proposer une nouvelle heuristique appelée Méthode de la sélection maximale, pour le choix de la composante connexe à segmenter : la composante maximale, celle possédant le plus grand nombre d'arcs entre les noeuds.

3.3. Amélioration : méthode de la sélection maximale

L'algorithme qui améliore celui de Girvan et Newman, à savoir la méthode de la sélection maximale que nous proposons, crée des composantes connexes tout en segmentant le cluster maximal (celui possédant le plus grand nombre d'arcs) de la partition ; ensuite on calcule progressivement la modularité de cette dernière jusqu'à obtention de l'optimum, contrairement au précédent qui choisit à la fin des traitements la partition renvoyant le gain maximal de modularité. Ces heuristiques que nous proposons dérivent du fait qu'intuitivement, la division de la plus grande partition produirait une meilleure structure de communauté. L'algorithme de sélection maximale s'applique sur des graphes orientés et pondérés, et par conséquent se sert de la fonction qualité suivante, pour qualifier chaque partition.

$$Q = \sum_{ij}\left(\frac{p_{ij}}{2p} - \frac{d_i^{in} * d_j^{out}}{(2p)^2}\right)\partial\left(C_i, C_j\right)$$

où d_i^{in} est le degré entrant du noeud i, p le poids total du graphe, p_{ij} le poids de l'arc (i, j) et $\partial()$ est la fonction de Kronecker qui vaut qui vaut 1 si ses paramètres sont égaux et 0 sinon.

A chaque étape, elle détermine la qualité de la partition, et lorsque cette qualité ne croit plus, elle retourne la structure qui en découle. De plus, la recherche de l'arc de centralité d'intermédiarité maximale dépend d'une métrique basée sur la distance (pondérée) des arcs entre les noeuds du graphe. Le pseudocode associé à cette méthode est proposé dans l'algorithme1.

3.4. Analyse et complexité

Il est évident de constater que cet algorithme s'arrête car au pire des cas, nous obtiendrons le dendrogramme comme dans l'algorithme de Girvan et Newman c'est-à-dire N communautés (pour un graphe de N nœuds). Cependant, cet algorithme a une complexité supérieure à celle de l'algorithme de Girvan et Newman : soit $O(mn^2ln(n))$. En effet, contrairement à Girvan et Newman, la recherche de l'arc de centralité d'intermédia-

Algorithm 1 Algorithme : Pseudocode de l'algorithme de Sélection maximale

```
Entrées: Graphe orienté $G = (N, E)$
Sorties: Partition $G$ en communautés
Initialisation : $D = V_1, V_2, ..., V_N$ ; $G' = G$ ; $Qps \leftarrow 0$ ; // $V_i \in N$
Calculer la centralité d'intermédiarité pour chaque arête $e_i$ de $G'$ ;
répéter
    $Q* \leftarrow Q_s$ ;
    $e_m \leftarrow (i, j)$ ;//arc de centralité maximale
    Tant que (Il existe une chaine entre les sommets $i$ et $j$) faire
        Retirer l'arc central maximal $e_m$ du graphe $G'$ ;
        $e_m \leftarrow$ Recherche de l'arc $(i, j)$ de centralité maximale ;
    Fin tant que
    Identifier l'ensemble $C = C_1, ..., C_l$ de toutes les composantes connexes de $G'$
    Mettre à jour $D$ avec les nouvelles composantes de $C$ ;
    $Q_s \leftarrow$ Modularité de la partition ps obtenu ;
    Choisir la composante maximale ;
jusqu'à $(Q* > Q_s)$
Retourner $D$ ;
```

rité maximale est basée sur la distance minimale entre deux nœuds, et cette distance est déterminée avec l'algorithme de Dijkstra qui s'exécute en $O(nln(n))$.

4. Détection des rôles dans la communauté

Le rôle joué par un nœud dans une communauté est aussi important que la détection de cette communauté. Il ne servirait pas à grand-chose d'obtenir des clusters dans les réseaux sociaux sans pouvoir les interpréter. Ainsi pour faciliter cette interprétation, nous proposons de définir les rôles des nœuds, en vous inspirant de la méthode de l'algorithme *Leader-follower*[6]. Nous distinguons trois rôles : le leader, l'externer et le follower.

Pour y parvenir, nous nous appuyons sur la mesure de centralité de degré associée à la centralité basée sur le flux réseau [7] qui s'applique aux graphes pondérés. Les formules employées pour déterminer ces rôles des nœuds de la communauté sont décrites dans cette section. Pour le calcul du leader, nous appliquons les formules ci-dessous :

$C_{in}^{deg}(v_i) = \frac{1}{N-1}\sum_j a_{ji}$ et $C_{out}^{deg}(v_i) = \frac{1}{N-1}\sum_j a_{ij}$

Où a_{ij} est le coefficient de la matrice d'adjacence modélisant le graphe et N l'ordre de cette matrice. Ainsi, le nœud leader dans une communauté est le nœud tel que :

$C^{deg}(v_i) = max\left(C_{in}^{deg}(v_k), C_{out}^{deg}(v_k)\right)$, $v_k \in V$ où max est la fonction retournant la centralité dont la valeur est la plus grande entre celle de ses paramètres.

Quant à la formule du calcul de l'externer dans une communauté, nous proposons de calculer la proportion du degré d'externers pour un nœud i de la communauté C_k, que nous définissons comme étant le poids total d'arcs du nœud i externes à la communauté C_k, divisé par le poids total d'arcs externes à la communauté C_k. Les formules suivantes permettent de définir la connectivité entre un noeud et les extérieurs à sa communauté.

$E_{in}^{deg}(v_i, C_k) = \frac{1}{p_k - m + 1}\sum_j p_{ij}$ et $E_{out}^{deg}(v_i, C_k) = \frac{1}{p_k - m + 1}\sum_j p_{ji}$

où P_k désigne le poids total des arcs externes de la communauté C_k et P_{ij} le poids de l'arc (i, j) et m la taille du graphe. Ainsi, l'externer sera celui possédant la fraction maxi-

Propriétés	GAT	Karaté
Nombre de nœuds	111	34
Nombre total de liens	1263	77
Poids global du graphe	667898	241
Nombre moyen de liens par nœud	6017	7
Nombre de nœuds possédant des arcs sortants	111	25
Nombre de nœuds possédant des arcs entrants	23	25

Tableau 1. *Statistiques descriptives des données*

male, tel que le décrit la formule suivante : $E^{deg}\left(C_k\right) = max\left(E_{in}^{deg}\left(v_i, C_k\right), E_{out}^{deg}\left(v_i, C_k\right)\right)$, $v_i \in V$.

S'agissant des nœuds followers, ce sont ceux qui ne sont ni leaders, ni externers.

5. Expérimentations

Nous avons expérimenté notre approche avec deux heuristiques de sélection de la partition à segmenter : la sélection aléatoire comme dans le cas de Girvan et Newman et la sélection maximale comme nous l'avons définie. Nous allons d'abord présenter les données utilisées, ensuite les résultats obtenus en comparant avec ceux obtenus avec l'algorithme de Louvain implémenté dans l'outil Gephi. Enfin, nous présenterons les rôles que nous avons détectés dans les communautés obtenues sur les graphes utilisés.

5.1. Données

Notre approche a été appliquée sur deux jeux de données : GAT et Karaté. Le jeu de données GAT se rapporte à un graphe d'appels téléphoniques de l'opérateur de téléphonie Orange Cote d'Ivoire et est structuré en deux ensembles. Le premier ensemble trace la mobilité des individus : à une certaine date t, l'on a la possibilité d'avoir la position z d'un abonné x. Et la position concerne les coordonnées en latitude et longitude du pylône auquel se connecte l'abonné. Tandis que le second ensemble présente un graphe de communication entre deux abonnés x et y. Les données sont contenues dans des fichiers d'extension .gml. A partir de ces informations, l'écriture d'un script a permis d'extraire dans un fichier texte .txt, le GAT sous le format suivant : trois colonnes, dont la première désigne la source (ou émetteur), la deuxième désigne la destination (ou récepteur) et la troisième désigne le poids (ou nombre d'appels ou SMS). Pour simplifier le graphe, nous faisons abstraction des dates et heure d'appels. Et en cas de doublons des arcs, l'on cumule les poids correspondants. Par exemple, si l'abonné 1 appelle l'abonné 2 à une date t_1 3 fois et à une autre date t_2 4 fois, nous aurons pour l'arc $(1 \longrightarrow 2)$ la valeur pondérée de 7.

Le jeu de données Karaté présenté sous forme d'un fichier .gml est un réseau d'amis d'un club de karaté dans une université des USA dans les années 1970. Ce fichier, pour être exploitable dans le code source a été transformé par un script pour prendre le format du fichier texte tel que décrit ci-dessus. Le tableau 1 résume les propriétés des différents graphes utilisés pour cette phase d'expérimentation.

5.2. Résultats

Pour aboutir aux résultats, nous avons appliqué aux deux jeux de données les algorithmes suivants : algorithme de Girvan et Newman modifié, Algorithme de Louvain et

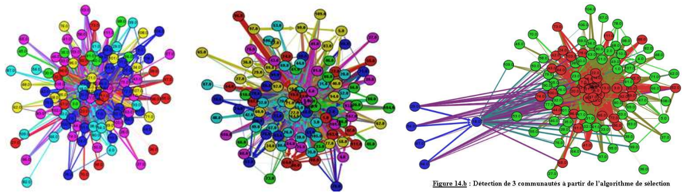


Figure 14.b : Détection de 3 communautés à partir de l'algorithme de sélection

(a) Algorithme de Girvan et Newman, 8 communautés, Q =0,116

(b) algorithme deLouvain, 6 communautés, Q = 0,133

(c) Algorithme de la sélection maximale, 3 communautés, Q = 0,35

Figure 1. *Détection des communautés des données GAT*

algorithme de la sélection maximale. Les figures 1 et 2 les présentent graphiquement. Chaque couleur dans un graphe représente une communauté détectée.

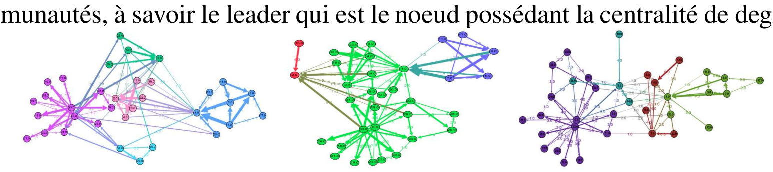
Sur les données de karaté, avec la sélection aléatoire présentée par Girvan et Newman, nous obtenons cinq communautés avec une modularité de $Q = 0,32$, la sélection maximale trois communautés avec une modularité $Q = 0,34$ et l'algorithme de Louvain quatre communautés avec une modularité $Q = 0,46$. Suivant le critère de qualité modularité, l'algorithme de Girvan et Newman est meilleur que les deux autres, mais possède l'inconvénient d'avoir plus de communautés, ce qui implique que la partition pourrait être mal structurée.

Ces trois approches se sont exécutées avec sensiblement le même temps. Cependant l'algorithme de Louvain dans l'outil Gephi fournit les résultats avec un temps légèrement plus faible que les deux autres.

S'agissant des résultats des rôles des nœuds, la figure 3 permet de visualiser la présence des nœuds *leader*, *externer* et *followers* de chaque communauté.

6. Conclusion

La détection de communautés et l'identification des rôles des noeuds dans l'analyse des réseaux sociaux (ARS) contribue à faciliter l'interprétation des phénomènes de la société. L'algorithme de la sélection maximale proposé dans cet article traite des graphes orientés et pondérés, et produit des communautés mieux structurées que celles produites par les algorithmes séparatif de Girvan et Newman et agglomératif de Louvain. Les nœuds jouent un rôle important dans l'ARS, ainsi, nous définissons trois principaux rôles dans les communautés, à savoir le leader qui est le noeud possédant la centralité de degré maxi-

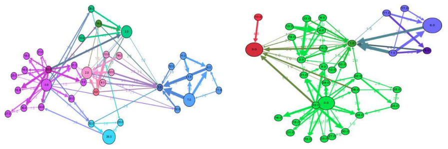


(a) Algorithme de Girvan et Newman, 5 communautés, Q = 0,32.

(b) Algorithme de la Sélection maximale, 3 communautés, Q = 0,34.

(c) algorithme de Louvain, 4 communautés, Q = 0,46

Figure 2. *Détection des communautés des données karaté*



(a) Rôles des noeuds du graphe de Girvan et Newman.

(b) Rôles des noeuds du graphe de la Sélection maximale

Figure 3. *Détection des rôles des données Karaté*

male, l'externer qui est le nœud commiquant avec le plus grand nombre de communautés, et les followers qui sont les autres nœuds de la communauté.

7. Bibliographie

[1] M. GIRVAN and NEWMAN, « Community structure in social and biological networks », *Proceedings of the National Academy of Sciences*, vol. 992, n^{o} 12, 2002.

[2] L. DANON and, A. DIAZ-GUILERA and, J. DUCH and , A.ARENAS, « Comparing community structure identification », *Journal of Statistical Mechanics : Theory and Experiment*, n^{o} 999, 2005.

[3] P. DE MEO , E. FERRARA , G. FIUMARA , A. PROVETTI « Generalized Louvain method for community detection in large networks » 2011.

[4] S. FORTUNATO, « Community detection in graphs. », *Physics Reports*, n^{o} 486, 75-174, 2009.

[5] V. BLONDEL , J. Guillaume , R. Lambiotte and E. LEFEBVRE « Fastunfolding of communities in large networks », *Journal of Statistical Mechanics : Theory and Experiment*, 2008.

[6] S. DEVAVRAT and , T. Zaman « Community detection in networks : The leader follower algorithm, » Workshop on Networks Across Disciplines, NIPS 2010.

[7] S.WASSERMAN and , K.Faust« Social networks analysis : Methods and Applications. », n^{o} 1994.

[8] R. MOKEN « Cliques, clubs, and clans »,n^{o} 161-173, 1979.

D Liste protocolaire

UNIVERSITÉ DE YAOUNDÉ I		THE UNIVERSITY OF YAOUNDE I
Faculté des Sciences		**Faculty of Science**
Division de la Programmation et du Suivi des Activités Académiques		Division of Programming and Follow-up of Academic Affaires
LISTE DES ENSEIGNANTS PERMANENTS		**LIST OF PERMANENT TEACHING STAFF**

ANNÉE ACADEMIQUE 2019/2020

(Par Département et par Grade)

DATE D'ACTUALISATION 03 Mars 2020

ADMINISTRATION

DOYEN : TCHOUANKEU Jean- Claude, *Maitre de Conférences*
VICE-DOYEN / DPSAA : ATCHADE Alex de Théodore, *Maitre de Conférences*
VICE-DOYEN / DSSE : AJEAGAH Gideon AGHAINDUM, *Professeur*
VICE-DOYEN / DRC : ABOSSOLO Monique, *Maitre de Conférences*
Chef Division Administrative et Financière : NDOYE FOE Marie C. F., *Maitre de Conférences*
Chef Division des Affaires Académiques, de la Scolarité et de la Recherche DAASR : MBAZE MEVA'A Luc Léonard, *Professeur*

1- DÉPARTEMENT DE BIOCHIMIE (BC) (38)

N°	NOMS ET PRÉNOMS	GRADE	OBSERVATIONS
1	BIGOGA DIAGA Jude	Professeur	En poste
2	FEKAM BOYOM Fabrice	Professeur	En poste
3	FOKOU Elie	Professeur	En poste
4	KANSCI Germain	Professeur	En poste
5	MBACHAM FON Wilfried	Professeur	En poste
6	MOUNDIPA FEWOU Paul	Professeur	Chef de Département
7	NINTCHOM PENLAP V. épse BENG	Professeur	En poste
8	OBEN Julius ENYONG	Professeur	En poste

9	ACHU Merci BIH	Maître de Conférences	En poste
10	ATOGHO Barbara Mma	Maître de Conférences	En poste
11	AZANTSA KINGUE GABIN BORIS	Maître de Conférences	En poste
12	BELINGA née NDOYE FOE M. C. F.	Maître de Conférences	Chef DAF / FS
13	BOUDJEKO Thaddée	Maître de Conférences	En poste
14	DJUIDJE NGOUNOUE Marcelline	Maître de Conférences	En poste
15	EFFA NNOMO Pierre	Maître de Conférences	En poste

16	NANA Louise épouse WAKAM	Maître de Conférences	En poste
17	NGONDI Judith Laure	Maître de Conférences	En poste
18	NGUEFACK Julienne	Maître de Conférences	En poste
19	NJAYOU Frédéric Nico	Maître de Conférences	En poste
20	MOFOR née TEUGWA Clotilde	Maître de Conférences	Inspecteur de Service MINESUP
21	TCHANA KOUATCHOUA Angèle	Maître de Conférences	En poste

22	AKINDEH MBUH NJI	Chargé de Cours	En poste
23	BEBOY EDZENGUELE Sara Nathalie	Chargée de Cours	En poste
24	DAKOLE DABOY Charles	Chargé de Cours	En poste
25	DJUIKWO NKONGA Ruth Viviane	Chargée de Cours	En poste
26	DONGMO LEKAGNE Joseph Blaise	Chargé de Cours	En poste
27	EWANE Cécile Anne	Chargée de Cours	En poste
28	FONKOUA Martin	Chargé de Cours	En poste
29	BEBEE Fadimatou	Chargée de Cours	En poste
30	KOTUE KAPTUE Charles	Chargé de Cours	En poste
31	LUNGA Paul KEILAH	Chargé de Cours	En poste
32	MANANGA Marlyse Joséphine	Chargée de Cours	En poste
33	MBONG ANGIE M. Mary Anne	Chargée de Cours	En poste
34	PECHANGOU NSANGOU Sylvain	Chargé de Cours	En poste
35	Palmer MASUMBE NETONGO	Chargé de Cours	En poste

36	MBOUCHE FANMOE Marceline Joëlle	Assistante	En poste
37	OWONA AYISSI Vincent Brice	Assistant	En poste
38	WILFRIED ANGIE Abia	Assistante	En poste

2- DÉPARTEMENT DE BIOLOGIE ET PHYSIOLOGIE ANIMALES (BPA) (48)

1	AJEAGAH Gideon AGHAINDUM	Professeur	*VICE-DOYEN / DSSE*
2	BILONG BILONG Charles-Félix	Professeur	Chef de Département
3	DIMO Théophile	Professeur	En Poste
4	DJIETO LORDON Champlain	Professeur	En Poste
5	ESSOMBA née NTSAMA MBALA	Professeur	*Vice Doyen/FMSB/UYI*
6	FOMENA Abraham	Professeur	En Poste
7	KAMTCHOUING Pierre	Professeur	En poste
8	NJAMEN Dieudonné	Professeur	En poste

9	NJIOKOU Flobert	Professeur	En Poste
10	NOLA Moïse	Professeur	En poste
11	TAN Paul VERNYUY	Professeur	En poste
12	TCHUEM TCHUENTE Louis Albert	Professeur	*Inspecteur de service* *Coord.Progr./MINSANTE*
13	ZEBAZE TOGOUET Serge Hubert	Professeur	*En poste*

14	BILANDA Danielle Claude	Maître de Conférences	En poste
15	DJIOGUE Séfirin	Maître de Conférences	En poste
16	DZEUFIET DJOMENI Paul Désiré	Maître de Conférences	En poste
17	JATSA BOUKENG Hermine épse MEGAPTCHE	Maître de Conférences	En Poste
18	KEKEUNOU Sévilor	Maître de Conférences	En poste
19	MEGNEKOU Rosette	Maître de Conférences	En poste
20	MONY Ruth épse NTONE	Maître de Conférences	En Poste
21	NGUEGUIM TSOFACK Florence	Maître de Conférences	En poste
22	TOMBI Jeannette	Maître de Conférences	En poste

23	ALENE Désirée Chantal	Chargée de Cours	En poste
26	ATSAMO Albert Donatien	Chargé de Cours	En poste
27	BELLET EDIMO Oscar Roger	Chargé de Cours	En poste
28	DONFACK Mireille	Chargée de Cours	En poste
29	ETEME ENAMA Serge	Chargé de Cours	En poste
30	GOUNOUE KAMKUMO Raceline	Chargée de Cours	En poste
31	KANDEDA KAVAYE Antoine	Chargé de Cours	En poste
32	LEKEUFACK FOLEFACK Guy B.	Chargé de Cours	En poste
33	MAHOB Raymond Joseph	Chargé de Cours	En poste
34	MBENOUN MASSE Paul Serge	Chargé de Cours	En poste
35	MOUNGANG LucianeMarlyse	Chargée de Cours	En poste
36	MVEYO NDANKEU Yves Patrick	Chargé de Cours	En poste
37	NGOUATEU KENFACK Omer Bébé	Chargé de Cours	En poste
38	NGUEMBOK	Chargé de Cours	En poste
39	NJUA Clarisse Yafi	Chargée de Cours	Chef Div. UBA
40	NOAH EWOTI Olive Vivien	Chargée de Cours	En poste
41	TADU Zephyrin	Chargé de Cours	En poste
42	TAMSA ARFAO Antoine	Chargé de Cours	En poste
43	YEDE	Chargé de Cours	En poste

44	BASSOCK BAYIHA Etienne Didier	Assistant	En poste
45	ESSAMA MBIDA Désirée Sandrine	Assistante	En poste
46	KOGA MANG DOBARA	Assistant	En poste
47	LEME BANOCK Lucie	Assistante	En poste
48	YOUNOUSSA LAME	Assistant	En poste

3- DÉPARTEMENT DE BIOLOGIE ET PHYSIOLOGIE VÉGÉTALES (BPV) (33)

1	AMBANG Zachée	Professeur	Chef Division/UYII
2	BELL Joseph Martin	Professeur	En poste
3	DJOCGOUE Pierre François	Professeur	En poste
4	MOSSEBO Dominique Claude	Professeur	En poste
5	YOUMBI Emmanuel	Professeur	Chef de Département
6	ZAPFACK Louis	Professeur	En poste

7	ANGONI Hyacinthe	Maître de Conférences	En poste
8	BIYE Elvire Hortense	Maître de Conférences	En poste
9	KENGNE NOUMSI Ives Magloire	Maître de Conférences	En poste
10	MALA Armand William	Maître de Conférences	En poste
11	MBARGA BINDZI Marie Alain	Maître de Conférences	CT/ MINESUP
12	MBOLO Marie	Maître de Conférences	En poste
13	NDONGO BEKOLO	Maître de Conférences	*CE / MINRESI*
14	NGODO MELINGUI Jean Baptiste	Maître de Conférences	En poste
15	NGONKEU MAGAPTCHE Eddy L.	Maître de Conférences	En poste
16	TSOATA Esaïe	Maître de Conférences	En poste
17	TONFACK Libert Brice	Maître de Conférences	En poste

18	DJEUANI Astride Carole	Chargé de Cours	En poste
19	GOMANDJE Christelle	Chargée de Cours	En poste
20	MAFFO MAFFO Nicole Liliane	Chargé de Cours	En poste
21	MAHBOU SOMO TOUKAM. Gabriel	Chargé de Cours	En poste
22	NGALLE Hermine BILLE	Chargée de Cours	En poste
23	NGOUO Lucas Vincent	Chargé de Cours	En poste
24	NNANGA MEBENGA Ruth Laure	Chargé de Cours	En poste
25	NOUKEU KOUAKAM Armelle	Chargé de Cours	En poste

26	ONANA JEAN MICHEL	Chargé de Cours	En poste
27	GODSWILL NTSOMBAH NTSEFONG	Assistant	En poste
28	KABELONG BANAHO Louis-Paul-Roger	Assistant	En poste
29	KONO Léon Dieudonné	Assistant	En poste
30	LIBALAH Moses BAKONCK	Assistant	En poste
31	LIKENG-LI-NGUE Benoit C	Assistant	En poste
32	TAEDOUNG Evariste Hermann	Assistant	En poste
33	TEMEGNE NONO Carine	Assistant	En poste

4- DÉPARTEMENT DE CHIMIE INORGANIQUE (CI) (34)

1	AGWARA ONDOH Moïse	Professeur	*Chef de Département*
2	ELIMBI Antoine	Professeur	En poste
3	Florence UFI CHINJE épouse MELO	Professeur	*Recteur Univ.Ngaoundere*
4	GHOGOMU Paul MINGO	Professeur	*Ministre Chargé deMiss.PR*
5	NANSEU Njiki Charles Péguy	Professeur	En poste
6	NDIFON Peter TEKE	Professeur	*CT MINRESI*
7	NGOMO Horace MANGA	Professeur	*Vice Chancelor/UB*
8	NDIKONTAR Maurice KOR	Professeur	*Vice-Doyen Univ. Bamenda*
9	NENWA Justin	Professeur	En poste
10	NGAMENI Emmanuel	Professeur	*DOYEN FS UDs*

11	BABALE née DJAM DOUDOU	Maître de Conférences	*Chargée Mission P.R.*
12	DJOUFAC WOUMFO Emmanuel	Maître de Conférences	En poste
13	EMADACK Alphonse	Maître de Conférences	En poste
14	KAMGANG YOUBI Georges	Maître de Conférences	En poste
15	KEMMEGNE MBOUGUEM Jean C.	Maître de Conférences	En poste
16	KONG SAKEO	Maître de Conférences	En poste
17	NDI NSAMI Julius	Maître de Conférences	En poste
18	NJIOMOU C. épse DJANGANG	Maître de Conférences	En poste
19	NJOYA Dayirou	Maître de	En poste

		Conférences	

20	ACAYANKA Elie	Chargé de Cours	En poste
21	BELIBI BELIBI Placide Désiré	Chargé de Cours	CS/ ENS Bertoua
22	CHEUMANI YONA Arnaud M.	Chargé de Cours	En poste
23	KENNE DEDZO GUSTAVE	Chargé de Cours	En poste
24	KOUOTOU DAOUDA	Chargé de Cours	En poste
25	MAKON Thomas Beauregard	Chargé de Cours	En poste
26	MBEY Jean Aime	Chargé de Cours	En poste
27	NCHIMI NONO KATIA	Chargé de Cours	En poste
28	NEBA nee NDOSIRI Bridget NDOYE	Chargée de Cours	CT/ MINFEM
29	NYAMEN Linda Dyorisse	Chargée de Cours	En poste
30	PABOUDAM GBAMBIE A.	Chargée de Cours	En poste
31	TCHAKOUTE KOUAMO Hervé	Chargé de Cours	En poste
32	NJANKWA NJABONG N. Eric	Assistant	En poste
33	PATOUOSSA ISSOFA	Assistant	En poste
34	SIEWE Jean Mermoz	Assistant	En Poste

5- DÉPARTEMENT DE CHIMIE ORGANIQUE (CO) (35)			
1	DONGO Etienne	Professeur	Vice-Doyen
2	GHOGOMU TIH Robert Ralph	Professeur	Dir. IBAF/UDA
3	NGOUELA Silvère Augustin	Professeur	Chef de Departement UDS
4	NKENGFACK Augustin Ephrem	Professeur	Chef de Département
5	NYASSE Barthélemy	Professeur	En poste
6	PEGNYEMB Dieudonné Emmanuel	Professeur	*Directeur/ MINESUP*
7	WANDJI Jean	Professeur	En poste

8	Alex de Théodore ATCHADE	Maître de Conférences	Vice-Doyen / DPSAA
9	EYONG Kenneth OBEN	Maître de Conférences	En poste
10	FOLEFOC Gabriel NGOSONG	Maître de Conférences	En poste
11	FOTSO WABO Ghislain	Maître de Conférences	En poste
12	KEUMEDJIO Félix	Maître de Conférences	En poste
13	KEUMOGNE Marguerite	Maître de Conférences	En poste
14	KOUAM Jacques	Maître de Conférences	En poste
15	MBAZOA née DJAMA Céline	Maître de	En poste

		Conférences	
16	MKOUNGA Pierre	Maître de Conférences	En poste
17	NOTE LOUGBOT Olivier Placide	Maître de Conférences	Chef Service/MINESUP
18	NGO MBING Joséphine	Maître de Conférences	Sous/Direct. MINERESI
19	NGONO BIKOBO Dominique Serge	Maître de Conférences	En poste
20	NOUNGOUE TCHAMO Diderot	Maître de Conférences	En poste
21	TABOPDA KUATE Turibio	Maître de Conférences	En poste
22	TCHOUANKEU Jean-Claude	Maître de Conférences	*Doyen /FS/ UYI*
23	TIH née NGO BILONG E. Anastasie	Maître de Conférences	En poste
24	YANKEP Emmanuel	Maître de Conférences	En poste

25	AMBASSA Pantaléon	Chargé de Cours	En poste
26	KAMTO Eutrophe Le Doux	Chargé de Cours	En poste
27	MVOT AKAK CARINE	Chargé de Cours	En poste
28	NGNINTEDO Dominique	Chargé de Cours	En poste
29	NGOMO Orléans	Chargée de Cours	En poste
30	OUAHOUO WACHE Blandine M.	Chargée de Cours	En poste
31	SIELINOU TEDJON Valérie	Chargé de Cours	En poste
32	TAGATSING FOTSING Maurice	Chargé de Cours	En poste
33	ZONDENDEGOUMBA Ernestine	Chargée de Cours	En poste

34	MESSI Angélique Nicolas	Assistant	En poste
35	TSEMEUGNE Joseph	Assistant	En poste

6- DÉPARTEMENT D'INFORMATIQUE (IN) (27)

1	ATSA ETOUNDI Roger	Professeur	*Chef Div.MINESUP*
2	FOUDA NDJODO Marcel Laurent	Professeur	*Chef Dpt ENS/Chef IGA.MINESUP*

3	NDOUNDAM Réné	Maître de Conférences	En poste

4	AMINOU Halidou	Chargé de Cours	*Chef de Département*
5	DJAM Xaviera YOUH - KIMBI	Chargé de Cours	En Poste
6	EBELE Serge Alain	Chargé de Cours	En poste
7	KOUOKAM KOUOKAM E. A.	Chargé de Cours	En poste

8	MELATAGIA YONTA Paulin	Chargé de Cours	En poste
9	MOTO MPONG Serge Alain	Chargé de Cours	En poste
10	TAPAMO Hyppolite	Chargé de Cours	En poste
11	ABESSOLO ALO'O Gislain	Chargé de Cours	En poste
12	MONTHE DJIADEU Valery M.	Chargé de Cours	En poste
13	OLLE OLLE Daniel Claude Delort	Chargé de Cours	C/D Enset. Ebolowa
14	TINDO Gilbert	Chargé de Cours	En poste
15	TSOPZE Norbert	Chargé de Cours	En poste
16	WAKU KOUAMOU Jules	Chargé de Cours	En poste

17	BAYEM Jacques Narcisse	Assistant	En poste
18	DOMGA KOMGUEM Rodrigue	Assistant	En poste
19	EKODECK Stéphane Gaël Raymond	Assistant	En poste
20	HAMZA Adamou	Assistant	En poste
21	JIOMEKONG AZANZI Fidel	Assistant	En poste
22	MAKEMBE. S . Oswald	Assistant	En poste
23	MESSI NGUELE Thomas	Assistant	En poste
24	MEYEMDOU Nadège Sylvianne	Assistante	En poste
25	NKONDOCK. MI. BAHANACK.N.	Assistant	En poste

7- DÉPARTEMENT DE MATHÉMATIQUES (MA) (31)

1	EMVUDU WONO Yves S.	Professeur	*Inspecteur MINESUP*

2	AYISSI Raoult Domingo	Maître de Conférences	Chef de Département
3	NKUIMI JUGNIA Célestin	Maître de Conférences	En poste
4	NOUNDJEU Pierre	Maître de Conférences	*Chef service des programmes & Diplômes*
5	MBEHOU Mohamed	Maître de Conférences	En poste
6	TCHAPNDA NJABO Sophonie B.	Maître de	Directeur/AIMS

		Conférences	Rwanda

7	AGHOUKENG JIOFACK Jean Gérard	Chargé de Cours	Chef Cellule MINPLAMAT
8	CHENDJOU Gilbert	Chargé de Cours	En poste
9	DJIADEU NGAHA Michel	Chargé de Cours	En poste
10	DOUANLA YONTA Herman	Chargé de Cours	En poste
11	FOMEKONG Christophe	Chargé de Cours	En poste
12	KIANPI Maurice	Chargé de Cours	En poste
13	KIKI Maxime Armand	Chargé de Cours	En poste
14	MBAKOP Guy Merlin	Chargé de Cours	En poste
15	MBANG Joseph	Chargé de Cours	En poste
16	MBELE BIDIMA Martin Ledoux	Chargé de Cours	En poste
17	MENGUE MENGUE David Joe	Chargé de Cours	En poste
18	NGUEFACK Bernard	Chargé de Cours	En poste
19	NIMPA PEFOUKEU Romain	Chargée de Cours	En poste
20	POLA DOUNDOU Emmanuel	Chargé de Cours	En poste
21	TAKAM SOH Patrice	Chargé de Cours	En poste
22	TCHANGANG Roger Duclos	Chargé de Cours	En poste
23	TCHOUNDJA Edgar Landry	Chargé de Cours	En poste
24	TETSADJIO TCHILEPECK M. E.	Chargée de Cours	En poste
25	TIAYA TSAGUE N. Anne-Marie	Chargée de Cours	En poste
26	MBIAKOP Hilaire George	Assistant	En poste
27	BITYE MVONDO Esther Claudine	Assistante	En poste
28	MBATAKOU Salomon Joseph	Assistant	En poste
29	MEFENZA NOUNTU Thiery	Assistant	En poste
30	TCHEUTIA Daniel Duviol	Assistant	En poste

8- DÉPARTEMENT DE MICROBIOLOGIE (MIB) (18)

1	ESSIA NGANG Jean Justin	Professeur	*Chef de Département*

2	BOYOMO ONANA	Maître de Conférences	En poste
3	NWAGA Dieudonné M.	Maître de Conférences	En poste
4	NYEGUE Maximilienne Ascension	Maître de	En poste

		Conférences	
5	RIWOM Sara Honorine	Maître de Conférences	En poste
6	SADO KAMDEM Sylvain Leroy	Maître de Conférences	En poste

7	ASSAM ASSAM Jean Paul	Chargé de Cours	En poste
8	BODA Maurice	Chargé de Cours	En poste
9	BOUGNOM Blaise Pascal	Chargé de Cours	En poste
10	ESSONO OBOUGOU Germain G.	Chargé de Cours	En poste
11	NJIKI BIKOÏ Jacky	Chargée de Cours	En poste
12	TCHIKOUA Roger	Chargé de Cours	En poste
13	ESSONO Damien Marie	Assistant	En poste
14	LAMYE Glory MOH	Assistant	En poste
15	MEYIN A EBONG Solange	Assistante	En poste
16	NKOUDOU ZE Nardis	Assistant	En poste
17	SAKE NGANE Carole Stéphanie	Assistante	En poste
18	TOBOLBAÏ Richard	Assistant	En poste

9. DEPARTEMENT DE PYSIQUE(PHY) (40)

1	BEN- BOLIE Germain Hubert	Professeur	En poste
2	EKOBENA FOUDA Henri Paul	Professeur	*Chef Division. UN*
3	ESSIMBI ZOBO Bernard	Professeur	En poste
4	KOFANE Timoléon Crépin	Professeur	En poste
5	NANA ENGO Serge Guy	Professeur	En poste
6	NDJAKA Jean Marie Bienvenu	Professeur	Chef de Département
7	NOUAYOU Robert	Professeur	En poste
8	NJANDJOCK NOUCK Philippe	Professeur	*Sous Directeur/ MINRESI*
9	PEMHA Elkana	Professeur	En poste
10	TABOD Charles TABOD	Professeur	Doyen Univ/Bda
11	TCHAWOUA Clément	Professeur	En poste
12	WOAFO Paul	Professeur	En poste

13	BIYA MOTTO Frédéric	Maître de Conférences	DG/HYDRO Mekin
14	BODO Bertrand	Maître de Conférences	En poste
15	DJUIDJE KENMOE épouse ALOYEM	Maître de Conférences	En poste
16	EYEBE FOUDA Jean sire	Maître de	En poste

		Conférences	
17	FEWO Serge Ibraïd	Maître de Conférences	En poste
18	HONA Jacques	Maître de Conférences	En poste
19	MBANE BIOUELE César	Maître de Conférences	En poste
20	NANA NBENDJO Blaise	Maître de Conférences	En poste
21	NDOP Joseph	Maître de Conférences	En poste
22	SAIDOU	Maître de Conférences	MINERESI
23	SIEWE SIEWE Martin	Maître de Conférences	En poste
24	SIMO Elie	Maître de Conférences	En poste
25	VONDOU Derbetini Appolinaire	Maître de Conférences	En poste
26	WAKATA née BEYA Annie	Maître de Conférences	*Sous Directeur/ MINESUP*
27	ZEKENG Serge Sylvain	Maître de Conférences	En poste

28	ABDOURAHIMI	Chargé de Cours	En poste
29	EDONGUE HERVAIS	Chargé de Cours	En poste
30	ENYEGUE A NYAM épse BELINGA	Chargée de Cours	En poste
31	FOUEDJIO David	Chargé de Cours	Chef Cell. MINADER
32	MBINACK Clément	Chargé de Cours	En poste
33	MBONO SAMBA Yves Christian U.	Chargé de Cours	En poste
34	MELI’I Joelle Larissa	Chargée de Cours	En poste
35	MVOGO ALAIN	Chargé de Cours	En poste
36	OBOUNOU Marcel	Chargé de Cours	DA/Univ Inter Etat/Sangmalima
37	WOULACHE Rosalie Laure	Chargée de Cours	En poste

38	AYISSI EYEBE Guy François Valérie	Assistant	En poste
39	CHAMANI Roméo	Assistant	En poste
40	TEYOU NGOUPOU Ariel	Assistant	En poste

10- DÉPARTEMENT DE SCIENCES DE LA TERRE (ST) (43)

1	BITOM Dieudonné	Professeur	*Doyen / FASA / UDs*
2	FOUATEU Rose épse YONGUE	Professeur	En poste
3	KAMGANG Pierre	Professeur	En poste
4	NDJIGUI Paul Désiré	Professeur	Chef de Département
5	NDAM NGOUPAYOU Jules-Remy	Professeur	En poste

6	NGOS III Simon	Professeur	DAAC/Uma
7	NKOUMBOU Charles	Professeur	En poste
8	NZENTI Jean-Paul	Professeur	En poste

9	ABOSSOLO née ANGUE Monique	Maître de Conférences	*Vice-Doyen / DRC*
10	GHOGOMU Richard TANWI	Maître de Conférences	CD/Uma
11	MOUNDI Amidou	Maître de Conférences	*CT/ MINIMDT*
12	NGUEUTCHOUA Gabriel	Maître de Conférences	CEA/MINRESI
13	NJILAH Isaac KONFOR	Maître de Conférences	En poste
14	ONANA Vincent Laurent	Maître de Conférences	*Chef service Maintenance & du Matériel*
15	BISSO Dieudonné	Maître de Conférences	*Directeur/Projet Barrage Memve'ele*
16	EKOMANE Emile	Maître de Conférences	En poste
17	GANNO Sylvestre	Maître de Conférences	En poste
18	NYECK Bruno	Maître de Conférences	En poste
19	TCHOUANKOUE Jean-Pierre	Maître de Conférences	En poste
20	TEMDJIM Robert	Maître de Conférences	En poste
21	YENE ATANGANA Joseph Q.	Maître de Conférences	*Chef Div. /MINTP*
22	ZO'O ZAME Philémon	Maître de Conférences	*DG/ART*

23	ANABA ONANA Achille Basile	Chargé de Cours	En poste
24	BEKOA Etienne	Chargé de Cours	En poste
25	ELISE SABABA	Chargé de Cours	En poste
26	ESSONO Jean	Chargé de Cours	En poste
27	EYONG JOHN TAKEM	Chargé de Cours	En poste
28	FUH Calistus Gentry	Chargé de Cours	*Sec. D'Etat/MINMIDT*
29	LAMILEN BILLA Daniel	Chargé de Cours	En poste
30	MBESSE CECILE OLIVE	Chargée de Cours	En poste
31	MBIDA YEM	Chargé de Cours	En poste
32	METANG Victor	Chargé de Cours	En poste
33	MINYEM Dieudonné-Lucien	Chargé de Cours	*CD/Uma*
34	NGO BELNOUN Rose Noël	Chargée de Cours	En poste
35	NGO BIDJECK Louise Marie	Chargée de Cours	En poste
36	NOMO NEGUE Emmanuel	Chargé de Cours	En poste
37	NTSAMA ATANGANA	Chargé de Cours	En poste

	Jacqueline		
38	TCHAKOUNTE J. épse NOUMBEM	Chargée de Cours	*Chef.cell / MINRESI*
39	TCHAPTCHET TCHATO De P.	Chargé de Cours	En poste
40	TEHNA Nathanaël	Chargé de Cours	En poste
41	TEMGA Jean Pierre	Chargé de Cours	En poste
42	FEUMBA Roger	Assistant	En poste
43	MBANGA NYOBE Jules	Assistant	En poste

Répartition chiffrée des Enseignants de la Faculté des Sciences de l'Université de Yaoundé I

	NOMBRE D'ENSEIGNANTS				
DÉPARTEMENT	**Professeurs**	**Maîtres de Conférences**	**Chargés de Cours**	**Assistants**	**Total**
BCH	9 (1)	13 (09)	14 (06)	3 (2)	**39 (18)**
BPA	13 (1)	09 (06)	19 (05)	05 (2)	**46 (14)**
BPV	06 (0)	11 (02)	9 (06)	07 (01)	**33 (9)**
CI	10 (1)	9 (02)	12 (02)	03 (0)	**34 (5)**
CO	7 (0)	17 (04)	09 (03)	02 (0)	**35(7)**
IN	2 (0)	1 (0)	13 (01)	09 (01)	**25 (2)**
MAT	1 (0)	5 (0)	19 (01)	06 (02)	**31 (3)**
MIB	1 (0)	5 (02)	06 (01)	06 (02)	**18 (5)**
PHY	12 (0)	15 (02)	10 (03)	03 (0)	**40 (5)**
ST	8 (1)	14 (01)	19 (05)	02 (0)	**43(7)**
Total	**69 (4)**	**99 (28)**	**130 (33)**	**46 (10)**	**344 (75)**

Soit un total de	**344 (**75**)** dont :
- Professeurs	**68 (4)**
- Maîtres de Conférences	**99 (**28**)**
- Chargés de Cours	**130 (33)**
- Assistants	**46** (10**)**
( ) = Nombre de Femmes	**75**