Finding Overlapping Communities in Networks Using Propositional Satisfiability

Said Jabbour *, Nizar Mhadhbi*
Badran Raddaoui** Lakhdar Sais *

*CRIL - CNRS UMR 8188, University of Artois
F-62307 Lens Cedex, France
{jabbour, mhadhbi, sais}@cril.fr
**SAMOVAR, Télécom SudParis, CNRS, Univ. Paris-Saclay
F-91011 Evry Cedex, France
badran.raddaoui@telecom-sudparis.eu

Résumé. Community detection is a fundamental issue for understanding the structure of large and complex networks such as social, biological and information networks. In this paper, we propose a new approach to detect overlapping communities in large complex networks. We first introduce a parametrized notion of a community, called *k-linked community*, allowing us to characterize node/edge centered k-linked community with bounded diameter. Such community admits a node or an edge with a distance at most $\frac{k}{2}$ from any other node of that community. Next, we show how the problem of detecting node/edge centered k-linked overlapping communities can be expressed as a Partial Max-SAT optimization problem. Then, we propose a post-processing strategy to limit the overlaps between communities. An extensive experimental evaluation on real-world networks shows that our approach outperforms several popular algorithms in detecting relevant communities.

1 Introduction

Many complex interactions can be represented by networks, which are set of nodes connected by edges. Such connections might represent different type of relations between individuals or entities. In social networks an edge represents some kind of social interaction, while in the world of information networks, an edge represent logical connections such as hyper links and citations. Nodes in networks can be organized into *communities*, which often correspond to groups of nodes that share common properties, roles or functionnalities, such as functionally related proteins, social communities, or topically related webpages.

One of the most important task when studying networks is that of identifying communities. Communities correspond to groups of nodes in a graph that share common properties or have similar roles. Indeed, detecting and analyzing communities is of great interest in several application domains, including clustering web clients who have similar interests, identifying clusters of customers in the network of customers-products purchase relationships of online

retailers (e.g. Amazon), etc. Several efficient algorithm for discovering communities in complex networks have been proposed. Let us mention for example, the most popular algorithm based on non-negative matrix factorisation Lee et Seung (2001), the spectral clustering methods Newman (2006a) and the edge betweenness based approach Girvan et Newman (2002). Some of them recuire several parameters such as the number of expected communities Lee et Seung (2001); Newman (2006a), while others involve for example the computation of the shortest paths between pairs of nodes Girvan et Newman (2002).

In this paper, we introduce a parametrized notion of communities, called k-linked community, allowing us to characterize node/edge centered k-linked community admitting a node or an edge with a distance at most $\frac{k}{2}$ from any other node of the community. This can be seen as a way to look for communities of bounded diameter. Our approach is only dependent on this single parameter k, and does not require any other knowledge about the network or about the number of expected communities.

Our proposed overlapping communities detection framework is based on an appropriate encoding of the centered k-linked community detection task as a partial maximum satisfiability (Partial Max-SAT) optimisation problem. It allows us to benefit from the recent advances in propositional satisfiability and its optimisation variants. Finally, we propose a post-processing strategy to limit the overlaps between communities. Our proposed framework follows the recent data mining research trend exploiting two powerful declarative models, namely constraint programming and propositional satisfiability. Indeed, several data mining tasks including pattern mining Guns et al. (2011) and clustering Gilpin et Davidson (2011) have been modeled and solved using these two well-known declarative and flexible models.

The paper is organized as follows. After some preliminary definitions about propositional satisfiability and community detection (Section 2), we describe our SAT-based framework for overlapping community detection for both centroid node based community and centroid edge based community (Section 3). An extensive and comparative experimental evaluation on many real-world datasets is presented in Section 4. Finally, we overview the related works before concluding.

2 Formal Preliminaries

In this section, we provide some preliminaries, key definitions and notational conventions.

2.1 Propositional Logic and SAT Problem

Let $\mathcal{L}$ be a propositional language defined inductively from a finite set $\mathcal{PS}$ of propositional symbols, the boolean constants $\top$ (*true* or 1) and $\bot$ (*false* or 0) and the standard logical connectives $\{\neg, \wedge, \vee, \rightarrow, \leftrightarrow\}$ in the usual way. We use the letters x, y, z, etc. to range over the elements of $\mathcal{PS}$. Formulas of $\mathcal{L}$ are denoted by A, B, C, etc. A *literal* is a propositional variable (x) of $\mathcal{PS}$ or the negation of a variable ($\neg x$). The two literals x and $\neg x$ are called complementary. A *clause* is a (finite) disjunction of literals, i.e., $a_1 \vee \ldots \vee a_n$. For every propositional formula $\mathcal{A}$ from $\mathcal{L}$, $\mathcal{P}(\mathcal{A})$ denotes the symbols of $\mathcal{PS}$ occurring in $\mathcal{A}$. A *Boolean interpretation* $\mathcal{I}$ of a formula $\mathcal{A}$ is a truth assignement of $\mathcal{PS}$, that is, a total function from $\mathcal{P}(\mathcal{A})$ to $\{0, 1\}$. A *model* of a formula $\mathcal{A}$ is a Boolean interpretation $\mathcal{I}$ that satisfies $\mathcal{A}$, i.e.

$\mathcal{I}(\mathcal{A}) = 1$. A formula $\mathcal{A}$ is satisfiable if there exists a model of $\mathcal{A}$. We denote by $\mathcal{M}(\mathcal{A})$ is the set of all models of $\mathcal{A}$.

As usual, every finite set of formulas is considered as the conjunctive formula whose conjuncts are the elements of the set. A formula in *conjunctive normal form* (CNF) is a (finite) conjunction of clauses. The SAT problem consists in deciding wether a given CNF formula admits a model or not. This well-known NP-Complete problem has seen spectacular progress these recent years.

SAT has seen many successful applications in various fields such as electronic design automation, debugging of hardware designs, artificial intelligence, and data mining. Several SAT extensions have been proposed to deal with optimisation problems. For example, the Max-SAT Problem seeks the maximum number of clauses that can be satisfied. In this paper, we consider one of these optimisation variants referred to as Partial Max-SAT problem. Partial Max-SAT sits between SAT and Max-SAT problems. While SAT requires all clauses to be satisfied, Partial Max-SAT relaxes this requirement by considering two kind of clauses, hard and soft. Partial MaxSAT is the problem of finding an optimal assignment to the variables that satisfies all the hard clauses, while satisfying the maximum number of soft clauses. Given n relaxable clauses, the objective is to find an assignment that satisfies all non-relaxable clauses together with the maximum number of relaxable clauses (i.e. a minimum number k of these clauses get relaxed). Partial Max-SAT can thus be used in various optimization tasks, e.g. multiple property checking, FPGA routing, etc. In these scenarios, simply determining that an instance is unsatisfiable (UNSAT) is not enough. In this paper, we consider the Partial Max-SAT WPM3 solver, the winner of the last Max-SAT SAT evaluation (Ansótegui et al. (2015)).

2.2 Overlapping Community Detection

In this subsection, we discuss the classic problem of detecting overlapping community structure in networks, , and review three traditional quality metrics.

A network is an undirected graph $\mathcal{N} = (V, E)$ where V is a set of nodes and $E \subseteq V \times V$ is a set of edges. We denote by n (respectively m) the number of nodes (respectively edges) in $\mathcal{N}$. The *degree* of a node $u \in V$, denoted d_u, is the number of edges connected to it. The length of the shortest path between two nodes $u, v \in V$ is called the *distance* between the nodes, noted $dist(u, v)$. Given an edge $e = (u, v) \in E$ and a node $w \in V$, the distance between e and w is defined as $dist(e, w) = \min\{dist(u, w), dist(v, w)\}$. In graph theory, a *community* is described as a set of nodes densely connected internally. In real-world networks, nodes are organized into densely linked sets of nodes that are commonly referred to as *network communities*, clusters or modules. Notice that communities in networks often overlap as nodes can belong to multiple communities at once. Network *overlapping community detection* problem consists in dividing a network of interest into (overlapping) communities for intelligent analysis. It has recently attracted significant attention in diverse application domains. Identifying the community structure is crucial for understanding structural properties of the real-world networks. Various methods have been proposed to identify the community structure of complex networks (see Fortunato (2009); Leskovec et al. (2010b) for an overview).

Quality Metrics :

Several measures have been proposed for quantifying the quality of communities in networks (see Leskovec et al. (2010a) for a comparative study of quality measures). In this paper, we adopt three well-known metrics to assess the performance of our method :

Modularity.

The most widely used metric for measuring the quality of network's partition into communities (without a ground-truth) is Newman's *modularity* function Newman et Girvan (2004). The idea of modularity-based community detection is to try to assign each node of the given network to a community such that it maximizes the modularity value of the whole network. Modularity quantifies the community strength by comparing the fraction of edges within the community with such fraction when random connections between the nodes are made. Networks with high modularity have dense connections between the nodes within communities but sparse connections between nodes in different communities. The modularity function has several variants, but these variants share the same principle. Without the loss of generality, we use the following equation of modularity, an extension of Newman's modularity function designed to support overlapping communities proposed in Shen et al. (2009). For the given community partition of a network $\mathcal{N} = (V, E)$ with m edges, an extended modularity EQ is given by :

$$EQ = \frac{1}{2m} \sum_{C \in C_{\mathcal{N}}} \sum_{u,v \in C} \frac{1}{O_u O_v} \left[A_{uv} - \frac{d_u d_v}{2m} \right]$$

with $C_{\mathcal{N}}$ the set of communities in $\mathcal{N}$; O_u the number of communities to which the node u belongs and A_{uv} is the element of the adjacency matrix representing the network.

F1 score.

Let $\mathcal{N} = (V, E)$ be a network, and $\hat{C}$ (respectively C^*) the set of (respectively ground truth) communities associated to $\mathcal{N}$. The average F1 score measure aims to quantify the level of correspondence between C^* and $\hat{C}$. More precisely, we need to determine which $C_i \in C^*$ corresponds to which $\hat{C}_i \in \hat{C}$. The F1 score is defined as the average of F1 score of the best matching ground-truth community to each detected community, and the F1 score of the best matching detected community to each ground-truth community Yang et Leskovec (2013). More formally, this function is defined as follows :

$$\frac{1}{2} \left(\frac{1}{|C^*|} \sum_{C_i \in C^*} F_1(C_i, \hat{C}_{g(i)}) + \frac{1}{|\hat{C}|} \sum_{\hat{C}_i \in \hat{C}} F_1(C_{g'(i)}, \hat{C}_i) \right)$$

where the best matching g and g' is defined as follows : $g(i) = \arg\max_j F_1(C_i, \hat{C}_j)$, $g'(i) = \arg\max_j F_1(C_j, \hat{C}_i)$, and $F_1(C_i, \hat{C}_j)$ is the harmonic mean of Precision and Recall.

Normalized Mutual Information (NMI).

This metric adopts the criterion used in information theory to compare the detected communities and the ground-truth communities. Normalized Mutual Information has been proposed as a performance metric for community detection (see Lancichinetti et al. (2009) for details). It provides a real number between zero and one that gives the similarity between two sets of sets of objects. The Normalized Mutual Information is written as :

$$\frac{H(X) + H(Y) - H(X,Y)}{(H(X) + H(Y))/2}$$

where $H(X)(H(Y))$ is the entropy of the random variable $X(Y)$ associated to the partition $C'(C'')$, whereas $H(X,Y)$ is the joint entropy. This variable is equal 1 only when the two partitions C' and C'' are exactly coincident.

3 A SAT-based Framework for Community Detection

Fundamentally, communities allow us to discover groups of interacting objects and the relations between them. A community (also referred to as a cluster) is a set of cohesive nodes that have more connections inside the set than outside. In this section, we propose an appropriate encoding of the community detection task as a SAT optimization problem. Proximity between nodes have been expressed as direct edges expressing formally a direct relation. Individuals can be grouped into the same cluster even if they are not linked directly. Relationships between individuals can be expressed via some proximity conditions. For instance, individuals having much common friends could be considered as very closed to each other. Consequently, the definition of individuals proximity is clearly a fundamental issue, as it have a great impact on the outcome. Next, we establish the main definitions which will be used to formulate our problem. Let us first start by introducing the notion of *k-linked community* as follows :

Definition 1 (k-linked community) *A community is k-linked if the nodes are pairwise k-linked, i.e., the distance between each two nodes is less or equal than k.*

According to Definition 1, a k-linked community has a diameter less or equal than k. Now, to simplify the encoding of the problem of discovering overlapping communities, we focus on the following kinds of k-linked communities called k-linked *centered* communities : those having a centroid node or centroid edge that possesses a distance at most $\frac{k}{2}$ from each other node of the community.

Definition 2 (Node/Edge Centered k-linked Community) *Let $\mathcal{N} = (V, E)$ be a network and $k > 1$ a positive integer. A community $C \subseteq V$ is node (resp. edge) centered k-linked community of $\mathcal{N}$ iff there exists $c \in C$ (resp. $e = (u, v) \in E$ with $u, v \in C$) s.t. $\forall\ w \in C$, $dist(c, w) \leq \frac{k}{2}$ (resp. $dist(e, w) \leq \frac{k}{2}$).*

Obviously, a node centered k-linked community is an edge centered k-linked community, while the converse is not true. Note also that a k-linked community is not necessarily a centered k-linked community. A counter-example consists of the network $\mathcal{N} = (V, E)$ where $V = \{x_1, \ldots, x_8\}$ and $E = \{(x_1, x_2), (x_2, x_3), (x_3, x_4), (x_5, x_6), (x_6, x_7), (x_7, x_8), (x_1, x_5),$

$(x_4, x_8)\}$. Then, $C = V$ is a 4-linked community, while there is neither a node $x_i \in V$ nor edge $e \in E$ with distance at most 2 from all the remaining nodes of C.

Lemma 1 *Let $\mathcal{N} = (V, E)$ be a network, $C \subseteq V$ a community and an integer $k > 1$. If C is a centered k-linked community, then C is also a k-linked community.*

Now, based on the notion of centered k-linked community, community detection is defined as an optimization problem, solving Partial Max-SAT. To do so, our starting point is to find a set of centroids S in the given network. The next step is to formed the communities around the centroids based on a predefined parameter k which represents the diameter of the communities. Clearly, we distinguish the following two cases : k-linked node (resp. edge) centered communities corresponding to an even (resp. odd) value of k.

Next, we propose two appropriate reformulations as an optimization problem for the community detection problem corresponding to node and edge centered k-linked communities, respectively. To achieve this, propositional variables are used for representing the network. Indeed, we associate each node u (resp. edge e) with a propositional variable denoted x_u (resp. y_e) where $x_u, y_e \in \{0, 1\}$. The key idea is that the variables assigned to 1 represent the centroids nodes (resp. edges), i.e., $S_v = \{u \in V \mid \mathcal{I}(x_u) = 1\}$ (resp. $S_e = \{e \in E \mid \mathcal{I}(y_e) = 1\}$). We now describe our SAT-based encodings using such propositional variables.

3.1 Node Centered k-linked Community

Our encoding consists of a set of constraints. The first propositional formula expresses the fact that if a node u is a centroid ($\mathcal{I}(x_u) = 1$), then the nodes with a distance at most $\frac{k}{2}$ from u are placed to the same community that possesses u as a centroid.

$$\bigwedge_{u \in V} (x_u \rightarrow \bigwedge_{v \in V | dist(u,v) \leq \frac{k}{2}} \neg x_v) \tag{1}$$

Let us remark that constraint (1) can be expressed by a set of binary clauses :

$$\bigwedge_{u \in V} \bigwedge_{v \in V | dist(u,v) \leq \frac{k}{2}} (\neg x_u \vee \neg x_v)$$

After finding the centroids, we still have to determine whether a node u belongs to community C or not depending on the value of k. To achieve this, we use the following formula that affects nodes of the network to communities where they belong to, i.e., nodes that have a distance at most of $\frac{k}{2}$ from the centroid.

$$\bigwedge_{u \in V} \bigvee_{v \in V | dist(u,v) \leq \frac{k}{2}} x_v \tag{2}$$

Proposition 1 *If the constraints (1) $\wedge$ (2) are satisfied, then for all $u \notin S_v$ there exists $v \in S_v$ s.t. $dist(u, v) \leq \frac{k}{2}$.*

Proposition 1 ensures that if $(1) \wedge (2)$ admits a model $\mathcal{I}$, then the nodes corresponding to the variables assigned to 1 ($\{u \in V \mid \mathcal{I}(x_u) = 1\}$) are the centroids and the network can

be partitioned into $|S|$ communities. The communities can then be constructed by finding the nodes with a distance at most $\frac{k}{2}$ from each centroid.

Obviously, the formula $(1) \wedge (2)$ may admits many candidate solutions (i.e. models). However, choosing an arbitrary model do not always guarantee a best partition of the network into communities. To alleviate this problem, we will consider an objective function to optimize over the space of solutions. Then, the node centered k-linked community detection problem can be formulated as the following optimisation problem :

$$\min/\max \sum_{u \in V} x_u \qquad \text{subject to } (1) \wedge (2) \tag{3}$$

3.2 Edge Centered k-linked Community

Now, to derive the formulation of edge centered k-linked community detection problem, we use similar reasoning as for node centered k-linked community, except that we consider centroid edges instead of centroid nodes. To do so, a community is built around an edge $e = (u, v)$ by considering nodes with a distance at most $\frac{k}{2}$ from the edge e. This is equivalent to partition the set of edges into modules and from that modules we can deduce the set of communities of nodes.

In the same way as for centroid nodes, the following formula expresses the fact that if an edge $e = (u, v)$ is a centroid edge ($\mathcal{I}(y_e) = 1$), then the nodes with a distance at most $\frac{k}{2}$ from u or v are assigned to 0.

$$\bigwedge_{e=(u,v)\in E} (y_e \rightarrow \bigwedge_{e' \in E | dist(e',u) \leq \frac{k}{2} || dist(e',v) \leq \frac{k}{2}} \neg y_{e'}) \tag{4}$$

Let us now introduce the following formula that affects nodes of the network to their associated communities, i.e. nodes that have a distance of $\frac{k}{2}$ from the centroid edge e.

$$\bigwedge_{e=(u,v)\in E} \bigvee_{e' \in E | dist(e',u) \leq \frac{k}{2} || dist(e',v) \leq \frac{k}{2}} y_{e'} \tag{5}$$

After fixing the centroids edges, the constraint 5 allows to identify whether a node u belongs to a community C or not from the value of k.

Similarly, to improve the quality of the detected communities, our edge centered k-linked community detection problem is formulated as the following optimisation problem :

$$\min/\max \sum_{e \in E} y_e \qquad \text{subject to } (4) \wedge (5) \tag{6}$$

We will use the notation $\text{CDSAT}^k_{\min/\max}$ to denote the optimization problems (3) and (6).

Example 1 *Let us consider the undirected network $\mathcal{N} = (V, E)$ depicted in Figure 1. Setting $k = 4$ can lead to the following solution of CDSAT$^4_{\max}$: $\mathcal{I} = \{\neg x_1, \neg x_2, \neg x_3, \neg x_4, \neg x_5, x_6, \neg x_7, x_8, \neg x_9, \neg x_{10}, \neg x_{11}\}$. So for that solution, $\mathcal{N}$ can be partitioned into the two communities $C_1 = \{1, \ldots, 6, 7, 11\}$ and $C_2 = \{1, 2, 5, 6, 7, \ldots, 11\}$. In contrast, CDSAT$^4_{\min}$ leads to one community with centroid x_1 and containing all the nodes of $\mathcal{N}$.*

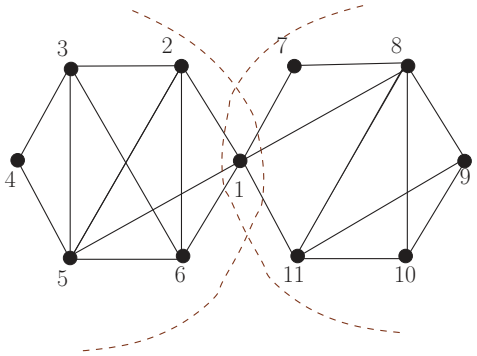


FIG. 1 – *A simple undirected network*

3.3 Overlapping Enhancement

As said before, once the node/edge centroids are found, the communities are formed arround them based on a predefined parameter k. As a result, some nodes can belong to multiple communities as illustrated in Example 1. However, such overlapping can be huge and not significant enough w.r.t. real communities. To overcome this drawback and to allow for an accurate partition of the network, we propose a simple but effective overlaps reduction technique in order to correctly identify dense community overlaps. Starting from a set of communities, each overlapping node will be assigned to its closest communities according to its distance from the centroids of these communities.

Example 2 *Let us consider again the network* $\mathcal{N} = (V, E)$ *of Figure 1. By enhancing the overlapping, the two communities are reduced to* $C_1 = \{1, \ldots, 6\}$ *and* $C_2 = \{1, 7, \ldots, 11\}$.

Algorithm 1 describes the general feature of our SAT-based node centered k-linked community detection procedure [1]. The algorithm takes as input the network and even integer k and returns a set of overlapping communities. It proceeds as follows : First, we generate the corresponding optimization problem that can be represented as a Partial MaxSAT problem (line 1). Then, a state-of-the-art Weighted Partial MaxSAT solver `WPM3` is used to get an optimal solution (i.e. model) $\mathcal{I}$. Next, the centroids are determined from the obtained model (lines 4-7). Using such centroids, the next step is to build communities by finding the nodes with a distance at most $\frac{k}{2}$ from each centroid. Finally, the cleaning step is called to improve the quality of detected communities (lines 11-13).

4 Empirical Evaluation

4.1 Experiment Settings

In this section, we present an experimental evaluation of our proposed approach. It was conducted on fourteen networks that cover a variety of application areas (e.g. social network,

1. Algorithm 1 can be slightly modified to deal with edge centered k-linked community detection problem.

Algorithme 1 : CDSAT$^{k}_{\min/\max}$

Input : A network $\mathcal{N} = (V, E)$ and an integer $k > 1$
Output : A set of overlapping communities

$\Phi = encodeToOpt(k, G)$;
$\mathcal{I} = solve(\Phi)$;
$S \leftarrow \emptyset$;
for $u_x \in \mathcal{I}$ **do**
  if $\mathcal{I}(u_x) == 1$ **then**
    $C_u \leftarrow \{u\}$;
    $S \leftarrow S \cup C_u$
  end
end
for $v_x \in \mathcal{I}$ **do**
  for $C_u \in S$ **do**
    if $dist(u, v) \leq \frac{k}{2}$ **then** $C_u \leftarrow C_u \cup \{v\}$;
  end
end
for $C_u, C_v \in S \times S$ **do**
  for $w \in V$ **do**
    if $dist(w, u) < dist(w, v)$ **then** $C_v \leftarrow C_v \setminus \{w\}$;
  end
end
return S

collaboration network, political network, game network, purchase network and word adjacencies network (Newman (2006b))) and are briefly described in Table 1 (columns 1 and 2). Some of these networks have ground-truth communities as presented in column 2 of Table 2. We have also chosen three large networks (`Facebook`, `DBLP`, and `Amazon` taken from SNAP (Leskovec et Krevl (2014))) to show the scalability of our model.

We evaluate the performance of our approaches by comparing them with the following most prominent state-of-the-art overlapping community detection algorithms :
(i) *Community-Affiliation Graph Model* (`AGM`) (Yang et Leskovec (2012)),
(ii) *Clique Percolation Method* (`CPM`) (Adamcsek et al. (2006)),
(iii) *Cluster Affiliation Model for Big Networks* (`BIGCLAM`) (Yang et Leskovec (2013)), and
(iv) *Communities from Edge Structure and Node Attributes* (`CESNA`) (Yang et al. (2014)).

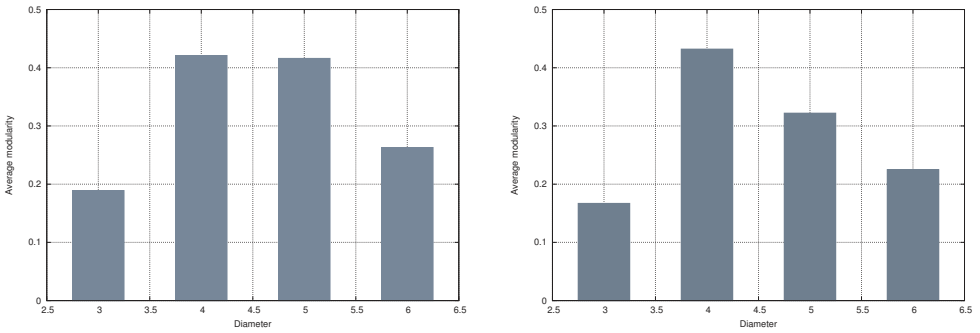
For the `CPM` algorithm, we use the cliques of size equal to 3. For `BIGCLAM` method, user can specify the number of communities to detect, or let the program determine the number of communities from the topology of the network. We opt for the case where the number of communities is not fixed in advance.

The proposed system, referred to as CDSAT$^{k}_{\min/\max}$, was written in Python. Given an input network as a set of edges, our algorithm starts by generating the corresponding optimization problem represented as a Partial MaxSAT problem. To solve this problem, we consider the state-of-the-art Weighted Partial MaxSAT solver `WPM3` (best solver at the last MaxSAT competition [2]) Ansótegui et al. (2015). As finding the optimal solution is NP-hard, in our experiment, we consider the first solution (not necessarily optimal) returned by the solver `WPM3`. For our experimental study, all algorithms have been run on a PC with an Intel Core 2 Duo (2 GHz) processor and 2 GB memory. We imposed 1 hour time limit for all the methods. Last, we use the symbol (-) in Tables 1 and 2 to indicate that the method is not able to scale on the considered network under the time limit.

2. http ://maxsat.ia.udl.cat/introduction/

4.2 Choosing the Best Value of the Diameter

Our CDSAT$^k_{\min/\max}$ algorithms take as input a network and a positive integer k and return a set of overlapping communities. In order to determine the best diameter k, we run CDSAT$^k_{\min/\max}$ on the fourteen considered networks, while varying k from 3 to 6. The Figure 2 summarises the relationship between the average modularity and k. As Figure 2 reveals, the best average modularity is obtained by CDSAT$^4_{\min}$ and CDSAT$^4_{\max}$ with a value of 0.421 and 0.432, respectively. We also observe that the average modularity obtained by both algorithms decreases beyond $k = 4$. Overall, for both algorithms the best average modularity is obtained for $k = 4$. These performances are relatively close. This can be explained by the fact that real-world social networks possess small (average or effective) diameters (e.g. Comellas et al. (2000)). This can be related to the property of the small-world phenomenon observed by several authors on real networks (e.g. Watts et al. (1998)). Also, setting the parameter k is particularly useful for community detection, as it allows for controlling the size of the resulting communities.



CDSAT$^k_{\min}$ CDSAT$^k_{\max}$

FIG. 2 – *Average modularity for CDSAT$^k_{\min/\max}$*

4.3 Comparison with Baseline Algorithms

Results on modularity metric. Table 1 reports the performance comparison between our CDSAT$^4_{\min/\max}$ approaches and the considered methods. Experiments show that our methods outperform every baseline, in most cases, by an interesting margin as shown by the average modularity reported in the last line of Table 1. We observe that across all datasets and modularity metric, CDSAT$^4_{\min}$ yields the best performance in 8 out of 14 networks. We also note that CDSAT$^4_{\min}$ shows a high margin in performance gain against the baselines in two large networks `DBLP` and `Amazon`, and in a collaborations network such as `Coauthorship` (Newman (2006b)). In terms of average performance, CDSAT$^4_{\min}$ outperforms CPM by 111.55%, BIGCLAM by 26.42%, and CESNA by 40.80%. Similarly, we note that CDSAT$^4_{\max}$ outperforms all the other methods in 7 out of 14 datasets. In terms of average performance, CDSAT$^4_{\max}$ outperforms CPM by 117.08%, BIGCLAM by 29.72%, and CESNA by 44.48%. We also observe that CDSAT$^4_{\max}$ gives an important improvement against the baselines in two

large networks Facebook, DBLP, and also in a collaborations network like Coauthorship. On the Lemis (Knuth (1993)), Power grid (Watts et Strogatz (1998)), Pilgrim (Brian et al. (2013)), and Jazz (Gleiser et Danon (2003)) datasets, our methods remain relatively competitive with the best baseline. A possible explanation for this phenomenon is that the WPM3 solver don't return the optimal solution for these datasets. Overall, our methods outperform BIGCLAM, which is the most competing algorithm, on all large real datasets.

Networks	nodes/edges	AGM	CPM	BIGCLAM	CESNA	$\text{CDSAT}^4_{\min}$	$\text{CDSAT}^4_{\max}$
Dolphin	62/159	−0.040	0.304	0.053	0.095	**0.438**	0.297
Karate	34/78	0.200	0.230	0.195	0.180	**0.310**	**0.311**
Risk map	42/83	0.415	0.488	0.194	0.504	**0.571**	**0.528**
Lemis	77/254	0.162	0.205	**0.444**	0.311	0.064	0.419
Word-adj	112/425	0.139	0.031	0.154	0.111	**0.175**	0.098
Football	115/615	0.222	0.199	0.343	0.390	0.286	**0.404**
Facebook	4039/88234	–	–	0.391	0.539	0.449	**0.701**
DBLP	317080/1049866	–	0.293	0.216	0.202	**0.520**	**0.436**
Amazon	334863/925872	–	0.195	0.341	0.430	**0.616**	**0.502**
Books	105/441	0.366	0.265	0.308	0.255	**0.439**	0.345
Power grid	4941/6594	–	0.007	**0.840**	0.586	0.679	0.547
Coauthership	1462/2742	0.619	0.456	0.679	0.031	**0.923**	**0.852**
Pilgrim	34/128	0.368	0.096	**0.415**	0.321	0.312	0.407
Jazz	196/2742	**0.310**	0.022	0.099	0.231	0.112	0.208
Average	N/A	N/A	0.199	0.333	0.299	**0.421**	**0.432**

TAB. 1 – *Modularity based performance of methods on fourteen datasets.*

Results on ground-truth communities. After finding communities in a given network, we can gauge the performance of each community that an algorithm has discovered and whether a ground-truth community has been successfully identified. Table 2 summarizes the evaluation results, with F1 scores of all algorithms on each network. Interestingly, it can be seen that $\text{CDSAT}^4_{\min}$ and $\text{CDSAT}^4_{\max}$ produce more accurate average w.r.t. the ground-truth setting than all the other baseline algorithms. In terms of average performance, $\text{CDSAT}^4_{\min}$ outperforms CPM by 16%, BIGCLAM by 22%, and CESNA by 35.05%. Moreover, notice that $\text{CDSAT}^4_{\max}$ outperforms CPM by 6.49%, BIGCLAM by 12%, and CESNA by 23.98%. In the cases of Karate (W.W. (1977)), Risk map (Cheng et al. (2014)), and DBLP data instances, $\text{CDSAT}^4_{\min}$ and $\text{CDSAT}^4_{\max}$ achieve a closely gain in the F1 score compared to the best baseline (CPM in this case).

To enlarge the criteria of comparison and offer some intuition about why our methods work well, we propose also to compare the set of considered approaches according to NMI metric. The resulats are reported in Table 3. As we can see from our experimental results, $\text{CDSAT}^4_{\min}$ and $\text{CDSAT}^4_{\max}$ algorithms achieve the best average performance. Compared with the other baselines, ours consistently produces more accurate average with respect to the ground-truth setting among all the other algorithms at detecting overlapping communities. Overall, $\text{CDSAT}^4_{\min/\max}$ algorithms outperform the baselines in nearly all cases. On average, $\text{CDSAT}^4_{\min}$ outperforms CPM by 72.64%, BIGCLAM by 95.72%, and CESNA by 57.08%. Likewise, $\text{CDSAT}^4_{\max}$ outperforms CPM by 54.71%, BIGCLAM by 75.40%, and CESNA by 40.77%. Broadly speaking, our $\text{CDSAT}^4_{\min}$ and $\text{CDSAT}^4_{\max}$ methods give superior overall performance to the four community detection algorithms.

As a summary, experimental results confirm that $\text{CDSAT}^4_{\min/\max}$ methods achieve the overall best performance in terms of the accuracy of the detected overlapping communities.

Networks	Communities	AGM	CPM	BIGCLAM	CESNA	$\text{CDSAT}^4_{\min}$	$\text{CDSAT}^4_{\max}$
Dolphin	2	0.120	0.579	0.628	0.100	**0.749**	**0.659**
Karate	2	0.864	**0.857**	0.629	0.663	0.847	0.851
Risk map	6	0.641	**0.884**	0.694	0.842	0.779	0.769
DBLP	13477	–	**0.596**	0.370	0.310	0.470	0.483
Amazon	75149	–	0.519	0.498	0.642	**0.695**	0.399
Books	3	0.684	0.557	0.549	0.591	**0.804**	0.652
Pilgrim	4	0.773	0.427	0.835	0.652	0.785	**0.892**
Average	N/A	N/A	0.631	0.600	0.542	**0.732**	**0.672**

TAB. 2 – *F1 Score using ground truth.*

Networks	Communities	AGM	CPM	BIGCLAM	CESNA	$\text{CDSAT}^4_{\min}$	$\text{CDSAT}^4_{\max}$
Dolphin	2	**0.434**	0.306	0.195	0.153	0.296	0.173
Karate	2	0.413	0.197	0.204	0.217	**0.621**	0.337
Risk map	6	0.196	0.247	0.405	0.649	0.492	**0.657**
DBLP	13477	–	**0.233**	0.031	0.012	0.175	0.112
Amazon	75149	–	0.178	0.015	0.019	0.173	**0.190**
Books	3	0.320	0.162	0.102	0.152	**0.421**	0.164
Pilgrim	4	0.339	0.166	0.360	0.433	0.389	**0.669**
Average	N/A	N/A	0, 212	0, 187	0, 233	**0.366**	**0.328**

TAB. 3 – *NMI using ground truth.*

Evaluating scalability. Finally, we evaluate the scalability of the different community detection methods by measuring the CPU time (see Table 4). From the results, it can be seen that our algorithms make few seconds to generate all communities for small networks. However, the CPM, BIGCLAM and CESNA baselines are faster than our methods for small networks (up to 200 nodes). We can observe that $\text{CDSAT}^4_{\min}$ and $\text{CDSAT}^4_{\max}$ are third-fastest method overall, when the network becomes larger. Interestingly, we also notice that our algorithms are the second-fastest methods, next BIGCLAM, for `DBLP` and `Amazon`.

Networks	AGM	CPM	BiGCLAM	CESNA	$\text{CDSAT}^4_{\max}$	$\text{CDSAT}^4_{\min}$
Dolphin	6.77	0.09	0.24	0.07	14	8
Karate	35	0.07	0.29	0.07	11	7.15
Risk map	62	0.09	2.84	0.59	38	17
Lemis	200	0.10	0.55	0.09	16	12
Word-adj	60.35	0.09	0.97	0.13	60.60	11
Football	47.71	0.08	1.78	0.13	120.20	420
Facebook	$> 1h$	$> 1h$	240.38	4.81	360.30	480.7
DBLP	$> 1h$	3240	60.56	900.34	720.50	780.40
Amazon	$> 1h$	$> 1h$	60.09	1200.49	780.20	900
Books	19.83	0.12	2.71	0.10	14.35	60.20
Power grid	$> 1h$	0.66	0.81	4.48	420.15	480.25
Co-authership science	360.17	0.07	14.08	0.05	360.58	360.20
Pilgrim	0.61	0.09	0.35	0.07	8.2	7
Jazz	60.02	0.09	2.84	0.59	360.12	120

TAB. 4 – *Comparison in terms of running Time (s).*

5 Conclusion

In this paper, we developed a new framework for detecting overlapping community structure of real-world networks. Our method is based on a partition of the network into modules with bounded diameters. We have shown that the problem of centered k-linked community detection can be expressed as a Partial Max-SAT optimization problem. Extensive experiments based on 14 networks from different sources showed that our approach outperforms the state-of-the-art methods in accurately discovering network communities. These performances are obtained while looking for the first non necessarily optimal solution of the underlying optimisation problem.

As a future work, we intend to develop a parallel version to even improve the performance of our optimisation based approach. We also plan to extend our proposed framework to deal with dynamic community detection in networks.

Références

Adamcsek, B., G. Palla, I. J. Farkas, I. Derényi, et T. Vicsek (2006). Cfinder : locating cliques and overlapping modules in biological networks. *Bioinformatics 22*(8), 1021–1023.

Ansótegui, C., F. Didier, et J. Gabàs (2015). Exploiting the structure of unsatisfiable cores in MaxSAT. In *IJCAI*, pp. 283–289.

Brian, B. Dickinson, W. Valyou, et Hu (2013). A genetic algorithm for identifying overlapping communities in social networks using an optimized search space. *Social Networking 02No.04*, 1–9.

Cheng, J., M. Leng, L. Li, H. Zhou, et X. Chen (2014). Active semi-supervised community detection based on must-link and cannot-link constraints. *PLoS 9*(10), 1–18.

Comellas, F., J. Ozón, et J. G. Peters (2000). Deterministic small-world communication networks. *Information Processing Letters 76*(1), 83 – 90.

Fortunato, S. (2009). Community detection in graphs. *CoRR abs/0906.0612*.

Gilpin, S. et I. N. Davidson (2011). Incorporating SAT solvers into hierarchical clustering algorithms : an efficient and flexible approach. In *KDD*, pp. 1136–1144.

Girvan, M. et M. E. J. Newman (2002). Community structure in social and biological networks. *Proc.Natl.Acad.Sci 99*, 7821.

Gleiser, P. et L. Danon (2003). Community structure in jazz. *Advances in Complex Systems 6*, 565.

Guns, T., S. Nijssen, et L. D. Raedt (2011). Itemset mining : A constraint programming perspective. *Artif. Intell. 175*(12-13), 1951–1983.

Knuth, D. E. (1993). *The Stanford GraphBase - a platform for combinatorial computing*.

Lancichinetti, A., S. Fortunato, et J. Kertesz (2009). Community detection algorithms : A comparative analysis. *New Journal of Physics 11*.

Lee, D. D. et H. S. Seung (2001). Algorithms for non-negative matrix factorization. In T. K. Leen, T. G. Dietterich, et V. Tresp (Eds.), *Advances in Neural Information Processing Systems 13*, pp. 556–562. MIT Press.

Leskovec, J., D. P. Huttenlocher, et J. M. Kleinberg (2010a). Predicting positive and negative links in online social networks. In *WWW*, pp. 641–650.

Leskovec, J. et A. Krevl (2014). SNAP Datasets : Stanford large network dataset collection. `http://snap.stanford.edu/data`.

Leskovec, J., K. J. Lang, et M. W. Mahoney (2010b). Empirical comparison of algorithms for network community detection. In *WWW*, pp. 631–640.

Newman, M. E. J. (2006a). Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E 74*.

Newman, M. E. J. (2006b). Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E 74*, 036104.

Newman, M. E. J. et M. Girvan (2004). Finding and evaluating community structure in networks. *Phys. Rev. E 69*(2), 026113.

Shen, H., X. Cheng, K. Cai, et M. Hu (2009). Detect overlapping and hierarchical community structure in networks. *Physica A 388*(8), 1706–1712.

Watts, D. J., P. S. Dodds, et M. E. J. Newman (1998). Collective dynamics of 'small-world' networks. *Nature* (393), 440–442.

Watts, D. J. et S. H. Strogatz (1998). Collective dynamics of small-world networks. *nature 393*(6684), 440–442.

W.W., Z. (1977). An information flow model for conflict and fission in small groups. *Journal of Anthropological Research 33*, 452–473.

Yang, J. et J. Leskovec (2012). Community-affiliation graph model for overlapping network community detection. In *ICDM*, pp. 1170–1175.

Yang, J. et J. Leskovec (2013). Overlapping community detection at scale : a nonnegative matrix factorization approach. In *WSDM*, pp. 587–596.

Yang, J., J. J. McAuley, et J. Leskovec (2014). Community detection in networks with node attributes. *CoRR abs/1401.7267*.

Summary

La détection de communautés est devenue un problème fondamental permettant la compréhension de la structure des réseaux complexes tels que les réseaux sociaux, biologiques ou encore les réseaux d'informations. Dans cet article, nous proposons une approche pour détecter les communautés chevauchantes dans les grands réseaux complexes. Nous introduisons d'abord une nouvelle notion de communauté paramétrée dite *communauté k-liée*. Cela nous permettrait de caractériser les communautés k-liées centrées nœud/arête de diamètre borné. Une telle communauté admet un noeud ou une arête avec une distance au plus $\frac{k}{2}$ des autres noeuds de la même communauté. Ensuite, nous montrons comment le problème de détection de communautés chevauchantes k-liées centrées nœud/arête peut être exprimé sous forme d'un problème d'optimisation Max-SAT partiel. Puis, nous proposons une stratégie de post-traitement pour réduire le chevauchement entre les communautés. Finalement, une évaluation expérimentale extensive sur des réseaux réels montrent que notre approche améliore significativement plusieurs algorithmes de l'état de l'art de détection de communautés.