

An algorithm for parametric communities detection in networks

Andrea Bettinelli*

DTI, Università degli Studi di Milano, via Bramante 65, Crema, Italy

Pierre Hansen†

GERAD, HEC, Montréal, Canada


Leo Liberti‡

LIX, École Polytechnique, F-91128 Palaiseau, France

(Dated: February 3, 2012)

Abstract

Modularity maximization is extensively used to detect communities in complex networks. It has been shown however that this method suffers from a resolution limit: small communities may be undetectable in the presence of larger ones even if they are very dense. To alleviate this defect, various modifications of the modularity function have been proposed as well as multi-resolution methods. In this paper we systematically study a simple model (first proposed by Pons and Latapy [Theor. Comp. Sci. **412**:892-900 (2010)] and similar to the parametric model of Reichardt and Bornholdt [Phys. Rev. E **74**:016110 (2006)]) with a single parameter α which balances the fraction of within community edges and the expected fraction of edges according to the configuration model.

An exact algorithm is proposed to find optimal solutions for all values of α as well as the corresponding successive intervals of α values for which they are optimal.  This algorithm relies upon a routine for exact modularity maximization and is limited to moderate size instances. An agglomerative hierarchical heuristic is therefore proposed to address parametric modularity detection in large networks. At each iteration the smallest value of α for which it is worthwhile to merge two communities of the current partition is found. Then merging is performed and the data updated accordingly. An implementation is proposed with the same time and space complexity as the well-known Clauset Newman Moore heuristic (CNM) [Phys. Rev. E **70**:066111 (2004)].

Experimental results on artificial and real world problems show that (i) communities are detected by both exact and heuristic methods for all values of the parameter α ; (ii) the dendrogram summarizing the results of the heuristic method provides a useful tool for substantive analysis, as illustrated particularly on *Les Misérables* data set; (iii) the difference between the parametric modularity values given by the exact method and those given by the heuristic is moderate; (iv) the heuristic version of the proposed parametric method, viewed as a modularity maximization tool, gives better results than the CNM heuristic for large instances.

*andrea.bettinelli@unimi.it

†Also at LIX, École Polytechnique, F-91128 Palaiseau, France; pierre.hansen@gerad.ca

‡liberti@lix.polytechnique.fr

I. INTRODUCTION

Networks, or graphs, are a versatile and powerful tool in the study of complex systems arising in telecommunication, transportation, social sciences and biology, see Newman's recent book²⁰ for a detailed introduction. A network $G = (V, E)$ consists of a set V of vertices and a set E of edges which are pairs of vertices. The number $|V|$ of vertices of G is usually denoted by n and called its order. The number $|E|$ of edges of G is usually denoted by m and called its size. The degree k_i of a vertex i is equal to the number of edges to which it is incident or, in other words, to its number of neighbors. The density d of a network G is the ratio of its number of edges to the possible number of edges, i.e., $d = \frac{2m}{n(n-1)}$.

A much studied phenomenon in networks is the presence of communities, i.e., subsets of vertices among which edges joining two vertices of that community are more dense than edges joining a vertex from that community to a vertex of another one. In the last few years, detection of communities in networks has been the subject of intense study, mostly among the physics research community, see Fortunato¹⁰ for an in-depth survey with over 400 references.

A first requirement when studying communities in networks is to have a precise definition of a community. Many proposals have been made and discussed¹⁰. The most used one appears to be the *modularity*, first proposed by Newman and Girvan²³. It expresses the idea that for a network to be modular its communities should contain more inner edges than expected. One must therefore define a null model which corresponds realistically to the network under study. The classical Erdős-Renyi model⁹, in which the probability of having an edge is uniform does not qualify as the distribution of degrees of real life networks tend to follow a power law²⁴. The adopted configuration model^{17,19} is that one where edges are drawn at random with the same expected distribution of degrees as in G . For a given partition of the network into communities the modularity function Q is:

$$Q = \sum_s (e_{ss} - a_s^2) \tag{1}$$

where s is the index of the community, e_{ss} denotes the fraction of the edges with both endpoints within community s , a_s is the sum of the degrees of the vertices of community s divided by the sum of degrees of the whole network, so that a_s^2 denotes the expected fraction of edges within that community, assuming they are drawn at random while keeping the same

distribution of degrees. As shown in Ref. 22 modularity can also be written as

$$Q = \frac{1}{2m} \sum_{i,j \in V} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(C_i, C_j) \quad (2)$$

where, for any entity i , C_i is the community containing that entity, A_{ij} are the components of the adjacency matrix of G (equal to 1 if there is an edge between vertices i and j and 0 otherwise), $\delta(C_i, C_j)$ is the Kronecker symbol, equal to 1 if C_i and C_j denote the same community and 0 otherwise.

While Equation (1) or (2) can be used to compute the modularity of a given partition of G into communities, they can also be used to maximize modularity, giving the optimal modularity value, the optimal number of communities and the corresponding partition. A few exact algorithms^{1,13,14,32} and many heuristics^{6,22,26,30,31} have been proposed for that purpose. Exact algorithms are presently limited to instances with a few hundred entities. Some of the heuristics, e.g. simulated annealing, are also very time consuming for large instances; other very fast heuristics can handle instances with over a million entities. An example is the efficient implementation by Clauset, Newman and Moore⁶ of the greedy agglomerative hierarchical heuristic of Newman²¹.

While the modularity function has been found to be illuminating in many applications, some criticisms have been raised^{5,11,12,18}. The main concern appears to be the existence of a resolution limit: in the presence of large communities, smaller ones may be undetectable even if they are very dense. More precisely, if the total degree of a community is smaller than the resolution limit ($\sqrt{m/2}$), then one cannot be sure whether this cluster is actually a single community or the union of several ones (this phenomenon also occurs in weighted networks³ and for the more general Reichardt and Bornholdt²⁷ function). Observe that in Equation (2) the number of edges of G within communities, i.e., $\sum_{i,j \in V} A_{ij}$, is divided by twice the size m of the network, while the expected number of edges $\sum_{i,j \in V} k_i k_j$ is divided by four times the square of this number. Consider then increasingly large networks. If their density remains constant, or in other words if the sum of the degrees increases quadratically with the order, both terms remain of the same order of magnitude. If the average degree, instead of the density, remains constant the second term becomes negligible and communities grow even if there is no change in their neighbourhood within the network.

Several papers address the resolution problem in different ways. Arenas et al.² propose to add self-loops of strength r to each vertex, obtaining a modified modularity formula. They

make a sweep in the range of r , and determine for each r the maximum modularity with extremal optimization⁸ or tabu search. Meaningful cluster structures correspond to plateaus in the plot of the number of clusters versus r . The method is able to disclose the community structure of several benchmark graphs; as a drawback it is very slow, since the modularity maximum has to be computed for many values of r .

Reichardt and Bornholdt²⁷ show that it is possible to reformulate the problem of community detection as the problem of finding the ground state of a spin glass model. The authors introduce the following Hamiltonian

$$\mathcal{H} = - \sum_{i \neq j} (A_{ij} - \gamma p_{ij}) \delta(C_i, C_j) \quad (3)$$

where p_{ij} denotes the link probability between communities i and j according to the null model and γ is a parameter. The modularity Q is related to (3) as $Q = -\mathcal{H}/m$, provided that $\gamma = 1$ and $p_{ij} = \frac{k_i k_j}{2m}$. When $\gamma \rightarrow 0$ the minimum energy is obtained when all nodes are assigned into the same community; in this case $\mathcal{H} = 2m$. When $\gamma \gg 1$ communities are broken into smaller pieces. Thus, it is possible to explore different resolution levels by varying the value of the parameter γ .

Ronhovde and Nussinov²⁹ present a technique based on the Potts model similar to that of Reichardt and Bornholdt²⁷. The main difference is the absence of a null model term. In a subsequent paper²⁸, the authors introduce a stability criterion for the partitions, based on the computation of the similarity of partitions obtained for the same γ and different initial conditions.

Pons and Latapy²⁵ propose a method consisting of the optimization of multiscale quality functions, including the multiscale modularity

$$Q_\alpha = \sum_i \alpha e_{ii} - (1 - \alpha) a_i^2.$$

They suggest that the length of the α -range $[\alpha_{min}(C), \alpha_{max}(C)]$, for which a community C appears in a maximum modularity partition, is a good indicator of the stability of the community. The authors define the relevance function of a community at a scale α and use it to reveal the most meaningful partitions.

Multi-resolution versions of modularity like (3) and (I) are still affected by resolution limit. Indeed, they are subject to two conflictual effects: the tendency to merge small communities, and the tendency to split large communities. Real

networks are characterized by the coexistence of clusters of very different sizes, well described by power law distributions. Thus, a single value of the parameter α able to capture all the structure of the network might not exist. For this reason we propose a method to solve parametric modularity maximization for all values of the parameter, in order to identify relevant communities at different resolution levels.

The paper is organized as follows: in Section II we study the model proposed by Pons and Latapy²⁵ and present an exact algorithm to find optimal solutions for all values of α as well as an agglomerative hierarchical heuristic to address parametric modularity detection in large networks; in Section III computational experiments on artificial and real world networks are presented. Finally in Section IV we draw some conclusions.

II. ALGORITHMS FOR PARAMETRIC MODULARITY

We consider the following parametric modularity function, as done by Pons and Latapy²⁵.

$$Q_\alpha = \sum_i \alpha e_{ii} - (1 - \alpha) a_i^2, \quad (4)$$

where $0 \leq \alpha \leq 1$. The parameter α balances the number of edges within communities and the expected number of edges within those communities in the null model. The value $\alpha = \frac{1}{2}$ corresponds to usual modularity. Brandes et al.⁴ have proved that modularity maximization is *NP*-hard. This immediately implies that the parametric modularity maximization considered in this paper is also *NP*-hard. Exact algorithms for modularity maximization^{1,4,32} are easily extended to the maximization of the parametric modularity Q_α for any fixed value of α . For instance, expressing modularity maximization as a clique partitioning problem, following^{1,4,13,14}, leads to:

$$\begin{aligned} \max \quad & \sum_{i < j \in V} \frac{1}{m} \left(\alpha A_{ij} - (1 - \alpha) \frac{k_i k_j}{2m} \right) x_{ij} - \sum_{i \in V} (1 - \alpha) \frac{k_i k_i}{2m} \\ \text{s.t.} \quad & x_{ij} + x_{jk} - x_{ik} \leq 1 & \forall 1 \leq i < j < k \leq n \\ & x_{ij} - x_{jk} + x_{ik} \leq 1 & \forall 1 \leq i < j < k \leq n \\ & -x_{ij} + x_{jk} + x_{ik} \leq 1 & \forall 1 \leq i < j < k \leq n \\ & x_{ij} \in \{0,1\} & \forall 1 \leq i < j \leq n \end{aligned} \quad (5)$$

where the binary variables x_{ij} are associated with the edges of the network and equal to 1 if and only if the endpoints v_i and v_j of the corresponding edge belong to the same community. The set of feasible solutions of (5) corresponds exactly to all partitions into cliques of the vertex set V . Each such partition corresponds to an equivalence relation on the entities. Indeed, the corresponding relation satisfies reflexivity (we can assume $x_{ii} = 1$ for each entity i as the x_{ii} do not appear in (5)), symmetry (similarly since (5) only mentions indices i, j with $i \leq j$, we may set $x_{ij} = x_{ji}$ for $i > j$, as we consider undirected networks) and transitivity (encoded by the constraints of (5)). This problem is an Integer Linear Program (ILP) with one parameter in the objective function.

All optimal solutions of (5) are integer. **So at least one optimal solution of (5) is the result of the maximization of a parametric linear function on the (unknown) convex hull \mathcal{C} of its integer solutions: in other words, a linear program on \mathcal{C} , which is a polyhedron with integer extreme points. It is well known⁷ that linear programs always attain at least one of their optima at extreme points of the polyhedron defined by their constraints.** If we fix the variables of (5) to an integer extreme point vector \bar{x} of \mathcal{C} , the objective function of (5) becomes a linear function of α : to each extreme point there corresponds therefore a linear function $Q_\alpha(\bar{x})$ in α . For any α , the optimal solution of (5) is on the upper envelope of this set of linear functions; i.e., on a convex piecewise linear function. Moreover, this function is monotonically increasing in α because the derivative of Q_α with respect to α is strictly positive (see Fig. 1).

It follows that there is a sequence of consecutive intervals of α (possibly reduced to a point) such that, for each successive interval, there is a solution of (5) which is optimal in the whole interval. The problem is then to determine all breakpoints of the curve Q_α in function of α , i.e., the highest points of intersection of the lines $Q_\alpha(\bar{x})$ as functions of α for a given partition \bar{x} , as α ranges between 0 and 1.

At a generic iteration t of our algorithm, we have a value α_t ; we solve (5) to find an optimal partition x^t and the corresponding modularity value Q^t optimal for α_t . Next, we determine whether α_t is the next breakpoint after α_{t-1} : we compute the intersection α^* of the two lines at α_{t-1} and α_t defined respectively by $Q_\alpha(x^{t-1})$ and $Q_\alpha(x^t)$ (see Fig. 2, left) and a corresponding optimal partition x^* with modularity Q^* , using (5) for $\alpha = \alpha^*$. Now there are three cases for Q^* : (a) it is at the top end of the interval of possible values; (b) it is at the bottom end; (c) it lies between the two interval endpoints (see Fig. 2, right).

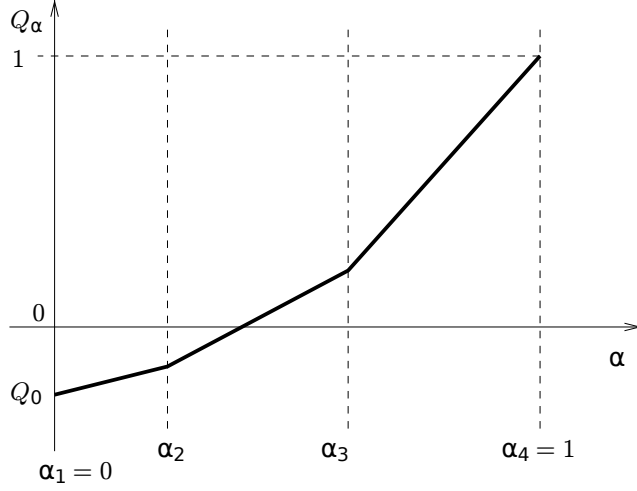


Figure 1. Q_α is a monotonically increasing convex piecewise linear function of α .

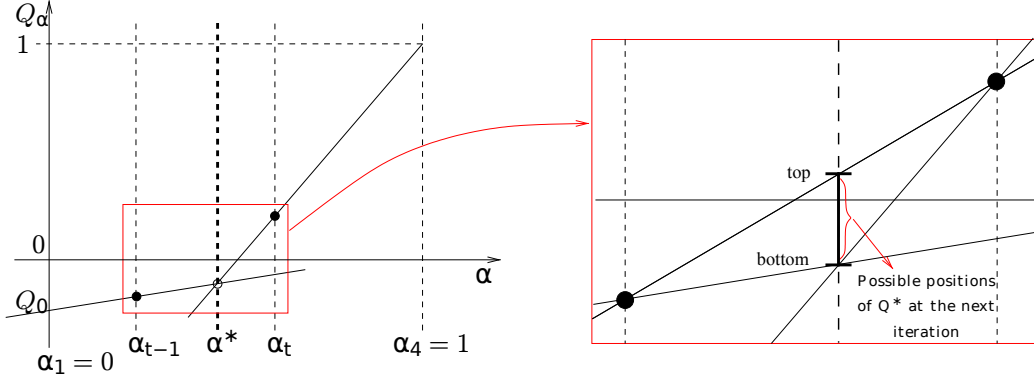


Figure 2. Finding the next breakpoint. The optimal modularity value must be on the emphasized segment in the right hand side frame.

In case (a), α^* is not a breakpoint, so α_t is the next breakpoint after α_{t-1} : for, suppose it is not, then the next breakpoint after α_{t-1} would be smaller than α_t , say $\tilde{\alpha}$ with associated optimal modularity value \tilde{Q} . This breakpoint would define a nonconvex piecewise linear function Q_α , as shown in Fig. 3.

In case (b), α^* is the next breakpoint after α_{t-1} and α_t is the next breakpoint after α^* : for suppose there were a different breakpoint $\tilde{\alpha}$ between α_{t-1} and α^* , then its optimal modularity value \tilde{Q} would be smaller than $Q' = Q_{\tilde{\alpha}}(x^{t-1})$; this would mean that x^{t-1} is a better partition than the one corresponding to \tilde{Q} , contradicting optimality of \tilde{Q} (see Fig. 4).

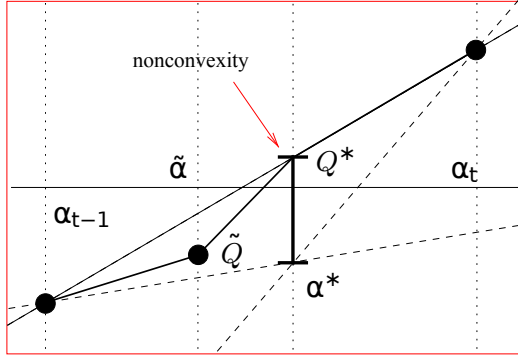


Figure 3. A proof sketch for case (a).

The argument when $\tilde{\alpha}$ lies between α^* and α_t is similar.

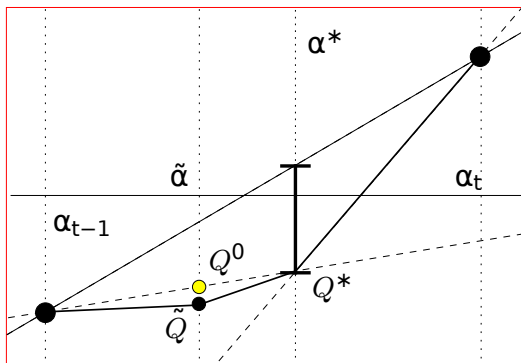


Figure 4. A proof sketch for case (b).

In case (c), α^* may or may not be a breakpoint. In such cases, we update $\alpha_t = \alpha^*$ and repeat. The process terminates in finite time because there can only be finitely many breakpoints.

In general, in order to find a value $\alpha_t > \alpha_{t-1}$, corresponding to a putative breakpoint, at the next iteration, we use an agglomerative approach: we find the smallest value of α for which it is worthwhile to merge two communities with at least an inter-community edge (merging non-adjacent communities cannot improve modularity). This implies finding α such that:

$$\alpha = \min_{rs} \left\{ \frac{2a_r a_s}{e_{rs} + 2a_r a_s} \right\}, \quad (6)$$

where e_{rs} is the fraction of edges joining communities r and s , a_r and a_s are the expected fractions of edges with one endpoint at least in these communities. Indeed, merging com-

munities r and s gives a larger community with a fraction $e_{rr} + e_{rs} + e_{ss}$ of the edges and an expected fraction $a_r^2 + a_s^2 + 2a_r a_s$ of the edges. This is worthwhile for a change in modularity $\Delta Q_\alpha \geq 0$ i.e., $\Delta Q_\alpha = (\alpha e_{rs} - 2(1 - \alpha)a_r a_s) \geq 0$, then the equality case leads to (6).

The steps of the exact algorithm are as follows:

1. **Initialization.** Set $t = 1$ and $\alpha_t = 0$. Consider the initial solution x^t with n communities each containing one entity and its value $Q^t = Q_0 = -\sum_i \frac{k_i^2}{2m^2}$. Let the expected fraction of edges with one entity in community i be equal to $a_i = \frac{k_i}{2m}$ for all communities $i = 1 \dots n$. Set $e_{ij} = \frac{1}{m}$ if an edge joins vertex i to vertex j , with $i < j$, and $e_{ij} = 0$ otherwise.
2. **Tentative optimal solution.** If x^t has a single community, print all values of α_t, Q^t and the corresponding partitions x^t , then stop. Otherwise, increase t by 1. Consider the set of all pairs (C_r, C_s) of adjacent communities in the previous partition x^{t-1} . Compute the new tentative value α_t using (6). Let C_{r^*} and C_{s^*} be the two communities to be merged at level α_t . Obtain x^t by replacing C_{r^*} and C_{s^*} by their union in x^{t-1} and compute the new value $Q_{\alpha_t}(x^t) = \sum_s \alpha_t e_{ss} - (1 - \alpha_t)a_s^2$.
3. **Optimality test.** Find the next breakpoint α_t after α_{t-1} using the arguments above, and update x^t and Q^t . Then return to 2.

The algorithm terminates when either $\alpha_t = 1$ or all entities are in the same community. Termination is guaranteed because $\alpha_t \geq \alpha_{t-1}$, there is only a finite number of breakpoints and in case $\alpha_t = \alpha_{t-1}$ one can apply a perturbation technique to avoid cycling.

The algorithm just described uses frequently a routine for clique partitioning, which is an *NP*-hard problem. This currently limits the size of instances that can be solved in reasonable time to a few hundred entities. As many community detection problems are larger, or even much larger, a heuristic variant is in order. In fact it suffices to remove the optimality test of the previous algorithm and iterate the tentative optimal solution test until the stopping condition is satisfied. The resulting heuristic, which we call α -aggregation, is close to the greedy heuristic proposed by Clauset Newman and Moore⁶. The objective at each step is the minimum increase in the value of α which justifies merging in the proposed heuristic, instead of the maximum increase in modularity in the CNM heuristic. It is important to note that similar data structures may be used:

1. A sparse matrix containing e_{ij} for each pair i, j of communities with at least one edge between them. Each row is memorized as a balanced binary tree.
2. A sparse matrix $\bar{\alpha}$ containing the minimum value of α which justifies merging for each pair i, j of communities with at least one edge between them.
3. A max-heap containing the largest element of each row of the matrix $\bar{\alpha}$ and the labels i, j of the corresponding communities.
4. An ordinary vector array with elements a_i .

Hence, the time complexity of the α -aggregation heuristic is $O(ml \log n)$, where l is the depth of the dendrogram describing the community structure.

III. EXPERIMENTS

A. Artificial examples

We have first tested the two version of our algorithm on three well known artificial examples from the literature. The first one¹¹ consists in a ring of cliques each one joined to the next by a single edge. Specifically, we consider a ring of 30 cliques with 5 vertices each. The functions $Q^*(\alpha)$ for the exact algorithm and the α -aggregation heuristic are presented in Figure 5 and are seen to coincide. The dendrogram of the heuristic is given in Figure 6. While the usual modularity (which corresponds to $\alpha = \frac{1}{2}$) merges cliques by pairs of consecutive ones, a partition into 30 cliques is clearly apparent on the dendrogram for $0.031 \leq \alpha \leq 0.423$.

The second example¹¹ is a graph consisting of two cliques with 20 vertices joined by an edge plus two small cliques of 5 vertices also joined by an edge and both joined by an edge to one of the large cliques, see Figure 7. The functions $Q^*(\alpha)$ are presented in Figure 8 and again coincide. The dendrogram, given in Figure 9, clearly displays the 4 clique partition for $0.331 \leq \alpha \leq 0.375$.

The third example² is a graph represented in Figure 10 which consists of a clique on 10 vertices followed by a chain of stars. The functions $Q^*(\alpha)$ are given in Figure 6 and once again coincide. There is no value of α for which the optimal partition into 5 communities

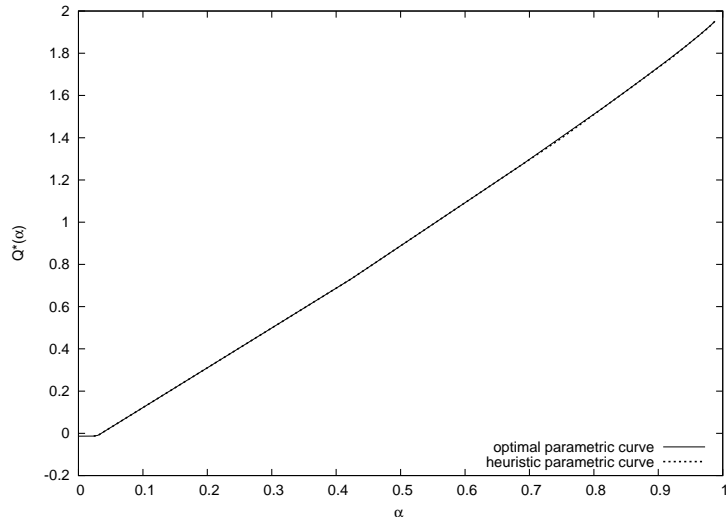


Figure 5. Optimal and heuristic parametric curves of modularity values for the ring of cliques network.

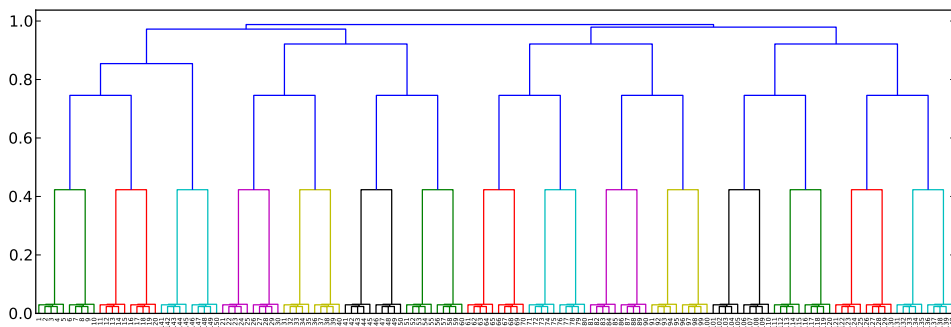


Figure 6. Dendrogram generated by the α -aggregation heuristic for the ring of cliques network (colors online).

is obtained. For $\alpha = 0.5$ there are 4 communities. However, it is clearly seen that for $0.063 \leq \alpha \leq 0.343$ one of these communities splits into two.

B. Real world examples

We first consider the classical karate club example of Zachary³³. The vertices correspond to the 34 members of a karate club and the edges to friendship relations between them as observed over a two year period by Zachary. Moreover, at one stage following a dispute between the club administrator and the karate teacher, the club split into two. Since then

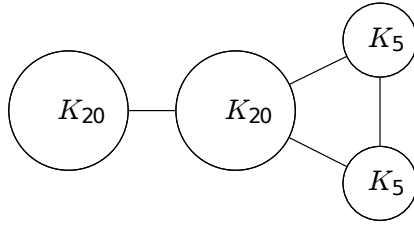


Figure 7. The second artificial network.

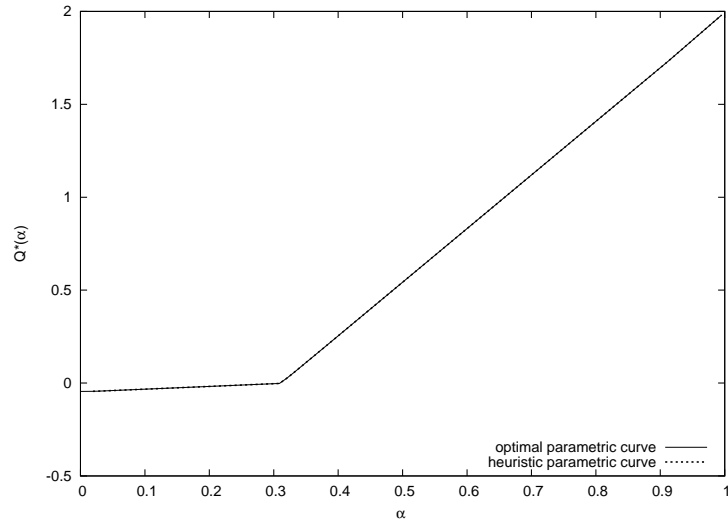


Figure 8. Optimal and heuristic parametric curves of modularity values for the second artificial network.

Zachary's data has been very often used to see how well communities detection methods are able to predict the split. The two $Q(\alpha)$ curves are represented in Figure 13. This time the heuristic is not always accurate: optimal values are obtained for $\alpha \leq 0.26$, i.e., in 25 out of 33 cases. The absolute error never exceeded 0.0341 with an average of 0.0042. The dendrogram summarizing the resolution is given in Figure 14. The partition in two communities is obtained for $0.5478 \leq \alpha \leq 0.7797$. It misclassifies two entities, i.e., 10 and 29. Both communities split into two at high level of α , i.e., 0.5301 and 0.5478. The partition into four communities obtained for $\alpha = \frac{1}{2}$ has a modularity value of 0.3893 while the optimal one is 0.4198. It misclassifies four entities, i.e., 1, 10, 12, 29. Pons and Latapy²⁵ suggest to consider, among the more general formulas, the length of the interval $\alpha_{max} - \alpha_{min}$ between values of α at which a community is absorbed into a larger one and at which it is formed as an indicator of its relevance. Following that criterion one observes that in addition to the

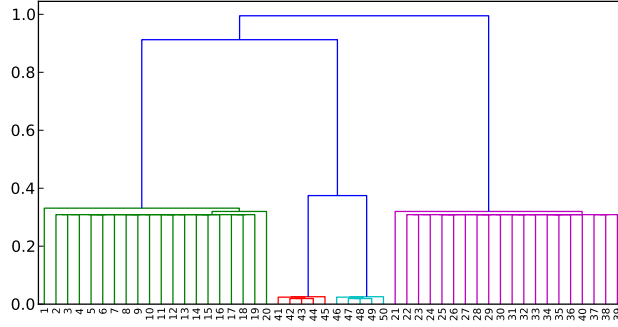


Figure 9. Dendrogram generated by the α -aggregation heuristic for the second artificial network (colors online).

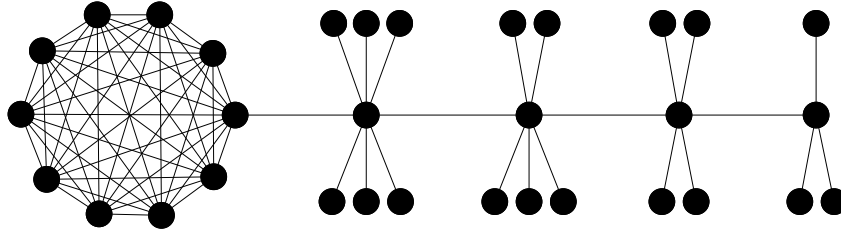


Figure 10. The third artificial network

four communities obtained for $\alpha = \frac{1}{2}$ several other ones are relevant: $\{1, 5, 6, 7, 11, 12, 17\}$ which is clearly visible on the network as 1 is a cut vertex and has $\alpha_{max} - \alpha_{min} = 0.5301 - 0.3036 = 0.2265$, $\{2, 18, 20, 22\}$ with $\alpha_{max} - \alpha_{min} = 0.4507 - 0.2000 = 0.2507$, $\{9, 31\}$ with $\alpha_{max} - \alpha_{min} = 0.3936 - 0.1136 = 0.2800$, $\{14\}$ with $\alpha_{max} - \alpha_{min} = 0.2778 - 0 = 0.2778$, $\{27, 30\}$ with $\alpha_{max} - \alpha_{min} = 0.2676 - 0.0488 = 0.2188$ and $\{24, 25, 26, 28, 32\}$ with $\alpha_{max} - \alpha_{min} = 0.5478 - 0.2571 = 0.2907$.

Our second example is the *Les Misérables* data set as compiled by Knuth¹⁵. This author listed the 77 interacting characters in Victor Hugo's masterpiece. They are represented by vertices. Two vertices are joined by an edge if and only if the two corresponding characters interact in at least one of the many, usually short, chapters of the book. The functions $Q^*(\alpha)$ obtained by both versions of our algorithm are presented in Figure 15. Once again the error of the heuristic is moderate, i.e., null or negligible in 43 cases out of 76, never larger than 0.0251 and equal to 0.0015 on average. The number of communities as a function of α is represented in Figure 17: it appears that most mergings take place for a value of $\alpha \leq 0.3$. There are 9 communities for $\alpha = 0.3$. In order to evaluate if these communities correspond

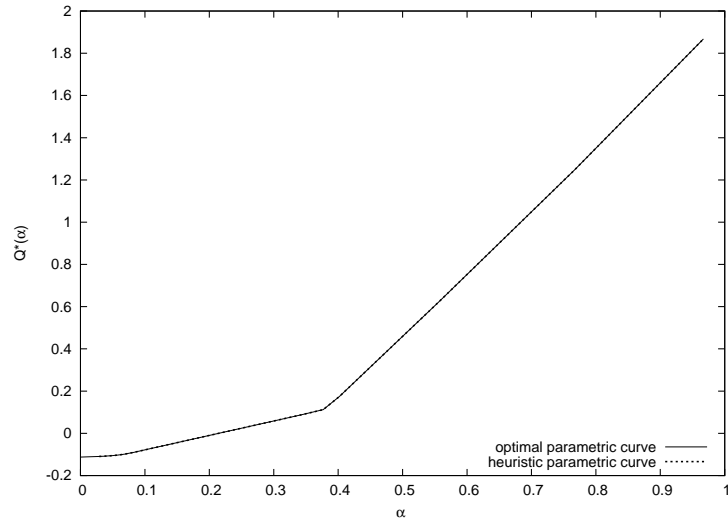


Figure 11. Optimal and heuristic parametric curves of modularity values for the third artificial network.

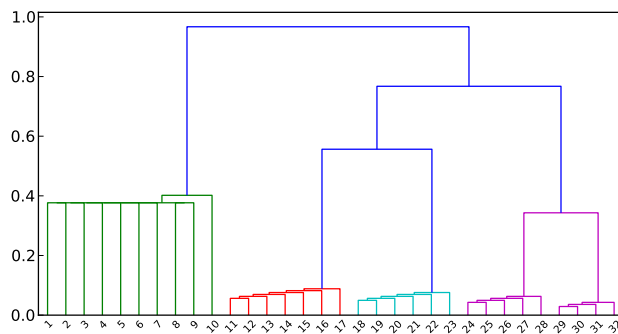


Figure 12. Dendrogram generated by the α -aggregation heuristic for the third artificial network (colors online).

to observable groups of characters in the novel we examine them in turn, seeing also how they have been obtained.

A first community is obtained for $0.0912 \leq \alpha \leq 0.7602$ and includes the first 10 characters. These are bishop Myriel, his servants Mlle Baptistine and Mme Magloire and various persons encountered at some time during his long life: Napoleon, the countess Delo, Geborand, the marquise de Champtercier, Cravatte, the count and an old man. The vertices corresponding to these characters form a star centered around bishop Myriel with the exception of vertices associated with his servants which are also joined together and to one entity from another

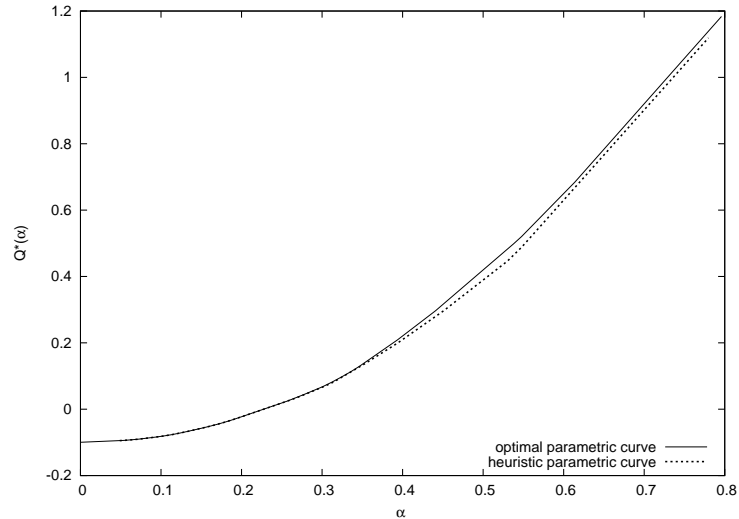


Figure 13. Optimal and heuristic parametric curves of modularity values for the Zachary network.

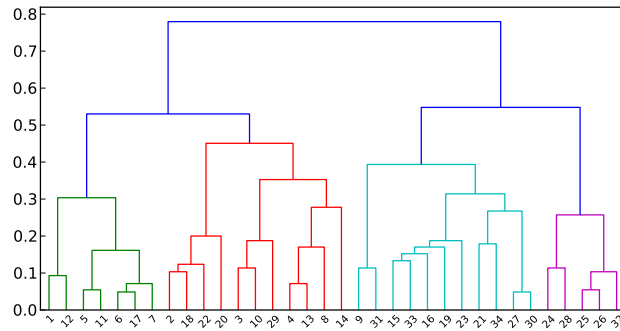


Figure 14. Dendrogram generated by the α -aggregation heuristic for the Zachary network (colors online).

community. Consequently they merge with the other members of the group at a higher level, i.e., $\alpha = 0.0912$.

A second community is obtained for $0.2203 \leq \alpha \leq 0.3618$. It consists of 9 vertices corresponding to the former convict (and hero) Jean Valjean, and to characters he meets just after being released from jail, which often do not treat him well: the innkeeper Labarre, the marquise de R, the baker Isabeau, a small boy Petit Gervais and the horse merchant Scaufflaire. The next three characters are met later: the aged notary Fauchelevent, the gravedigger Gribier and mother Innocent, prioress of a convent.

A third community is obtained for $0.1749 \leq \alpha \leq 0.3618$ and regroups 5 characters:

the young heroin Cosette, the three servants appointed by Jean Valjean to protect her (Toussaint, woman 1, woman 2) and Valjean's archenemy the policeman Javert, always hot on his tracks.

A fourth community is obtained for $0.0863 \leq \alpha \leq 0.4730$ and corresponds to the 6 protagonists of the Champmathieu affair: the juror Bamatabois, the judge, the thief Champmathieu wrongly believed to be Jean Valjean and the convicts Brevet, Chenildieu and Cochepaille.

A fifth community is obtained for $0.2263 \leq \alpha \leq 0.6015$ and corresponds to the 10 members of the circle of the evil innkeeper Thenardier: himself, his family (Mme Thenardier and their daughters Anzelma and Eponine) and the bandits with which he associates (Boulatruelle, Gueulemer, Montparnasse, Brujon, Babel and Claquesous).

A sixth community is obtained for $0.2480 \leq \alpha \leq 0.7404$ and regroups the heroin Fantine, her friends Favourite, Dahlia and Zephine, the four Parisian students Tholomyes, Listolier, Fameuil and Blacheville who love them and then abandon them, as well as three women who help Fantine after her downfall (Mme Marguerite and the sisters Perpetue and Simplicie).

A seventh community is obtained for $0.1732 \leq \alpha \leq 0.6129$ and regroups members of the noble family of the hero Marius (Pontmercy, Mme Pontmercy, Mme Gillenormand, Mlle Gillenormand, Lt Gillenormand) their friends (Mlle Vaubois and baroness T), and their servant Magnon.

An eighth community is obtained for $0.1331 \leq \alpha \leq 0.3935$ and regroups the street urchin Gavroche, his father Jondrette, two (unnamed) children and the landlady Mme Burgon.

Finally, a ninth community is obtained for $0.2761 \leq \alpha \leq 0.3935$ and corresponds to members of the *friends of the ABC* society (Enjolras, Combeferre, Pouvaire, Feuilly, Courfeyrac, Bahorel, Grantaire and Joly), the innkeeper Mme Hucheloup, and their friends (the church prefect Mabeuf and his maid Mother Plutarch).

These nine communities are shown in Fig. 19.

The second and third communities join to form a tenth community at level $\alpha = 0.3618$ with 14 members due to interactions such as Jean Valjean saving Cosette from the Thenardiens. The eighth and ninth communities merge into an eleventh community at level $\alpha = 0.3935$ due to Gavroche meeting the *friends of the ABC* during the days on the barricades of the 1830 revolution. The tenth community merges with the fourth one at level $\alpha = 0.4730$ to form a twelfth community, notably due to Jean Valjean intervening in the Champmathieu trial under the name of Monsieur Madeleine. The twelfth community merges

with the fifth one at level $\alpha = 0.6015$ due to Jean Valjean fighting with the Thenardier bandits. The eleventh community merges with the seventh one at level $\alpha = 0.6128$ to form a thirteenth community due to Marius joining the revolutionaries. The first community joins the twelfth community at level $\alpha = 0.7602$ to form a fourteenth community due to the charitable attitude of bishop Myriel, which leads Jean Valjean on the path to redemption. The fourteenth community joins the sixth one at level $\alpha = 0.7904$ to form a fifteenth community due to Jean Valjean helping Fantine. Finally the fifteenth community joins the thirteenth one to form a single community with all characters at level $\alpha = 0.8293$.

The dendrogram for the CNM heuristic applied to the same data set is presented in Figure 18. Here the ordinates represent the number of steps of the algorithm rather than the values of α . We compare the nine last communities to be merged in the results of both heuristics. While the 9 communities obtained for $\alpha = 0.3$ with the α -aggregation heuristic appear to be fairly well balanced, with a number of characters between 5 and 11, a similar result does not hold for the 9 last communities obtained with CNM and represented in Fig. 20. Indeed, their cardinalities are: 24, 1, 1, 15, 12, 1, 6, 2 and 15. So on the one hand three isolated individuals and a cluster of only two are obtained and on the other hand some large non homogeneous communities with up to 24 individuals are found. There are just a few similitudes between the communities obtained by both heuristics. For instance the 17th rightmost characters on the dendrogram correspond to the union of the two rightmost communities obtained by the two heuristics. The best modularity partition obtained by CNM has five communities with 26, 15, 13, 6 and 17 entities and a modularity value of 0.5006

C. Classical modularity

As a final test, we compared the values for the classical modularity given by CNM and by the α -aggregation heuristic for $\alpha = \frac{1}{2}$ on 10 test problems for which the optimal value of modularity is known¹ and on 5 larger networks. **Results are reported in Table I.** The new heuristic gives a better solution than CNM in 11 cases out of 15 and the difference in performance appear to increase with the size of the large networks. The average modularity of CNM is 0.6064 and 0.6198 for our heuristic. Note that both heuristics never found the optimal solution which was to be expected as they are simple greedy procedures. As they

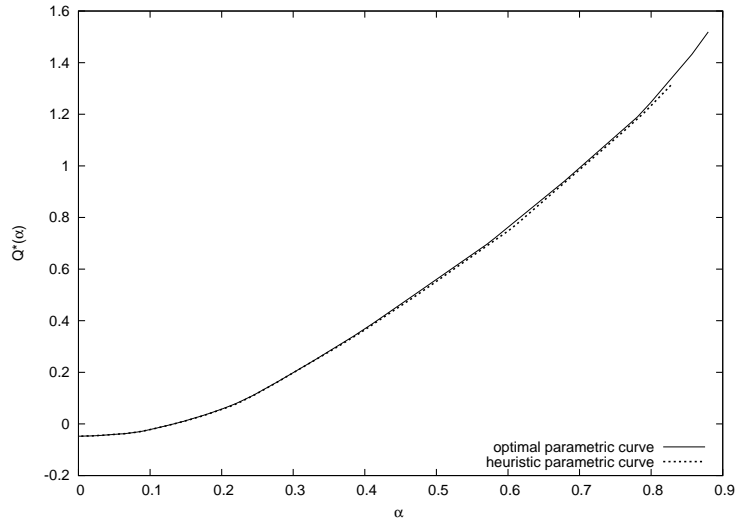


Figure 15. Optimal end heuristic parametric curves of modularity values for *Les Misérables* network.

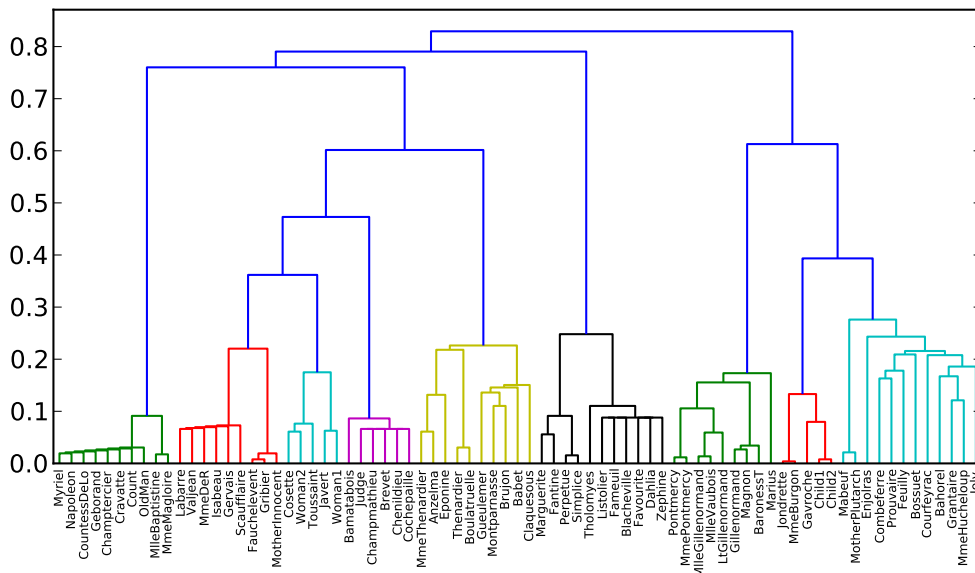


Figure 16. Dendrogram generated by the α -aggregation heuristic for *Les Misérables* network (colors online).

are very quick one might run them both and take the best solution: this raises the average modularity to 0.6240. We also tested the stability of the α -aggregation heuristic against small modifications of the graphs. For each test problem, 10 new graphs have been generated by a single rewiring. Average results are displayed in the last

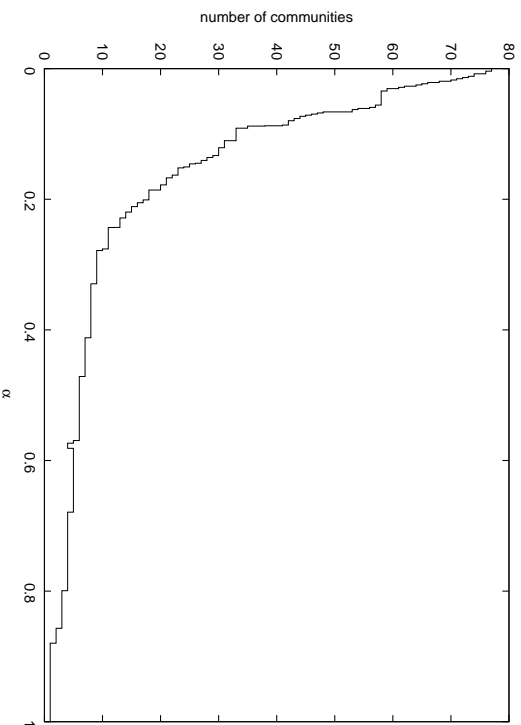


Figure 17. Number of communities as a function of α , *Les Misérables* network.

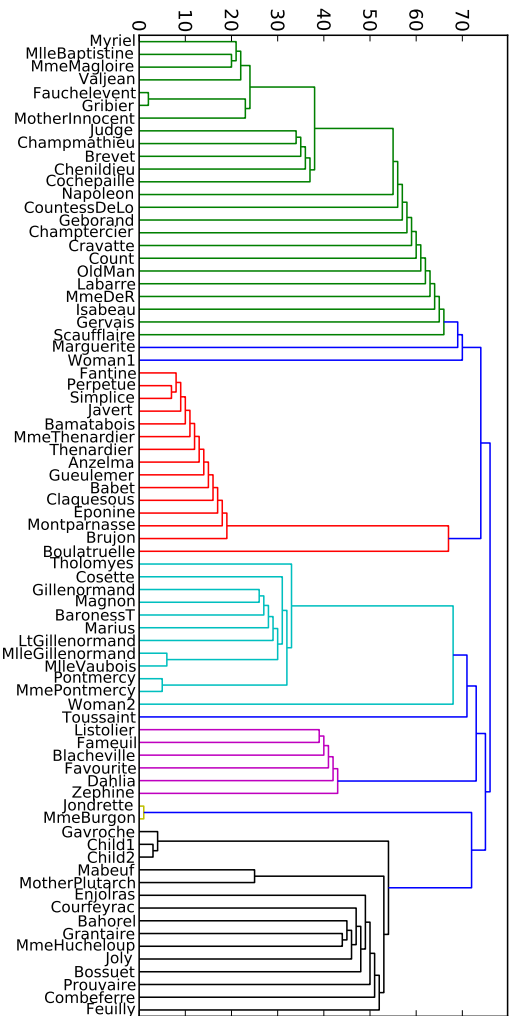


Figure 18. Dendrogram generated by the CNM heuristic for *Les Misérables* network (colors online).

two columns of Table I. The algorithm shows good stability both in solution value and in computational time.

IV. CONCLUSION

We consider a parametric modularity model²⁵ and propose an exact algorithm to find the optimal modularity partitions for all values of the parameter α as well as the intervals of

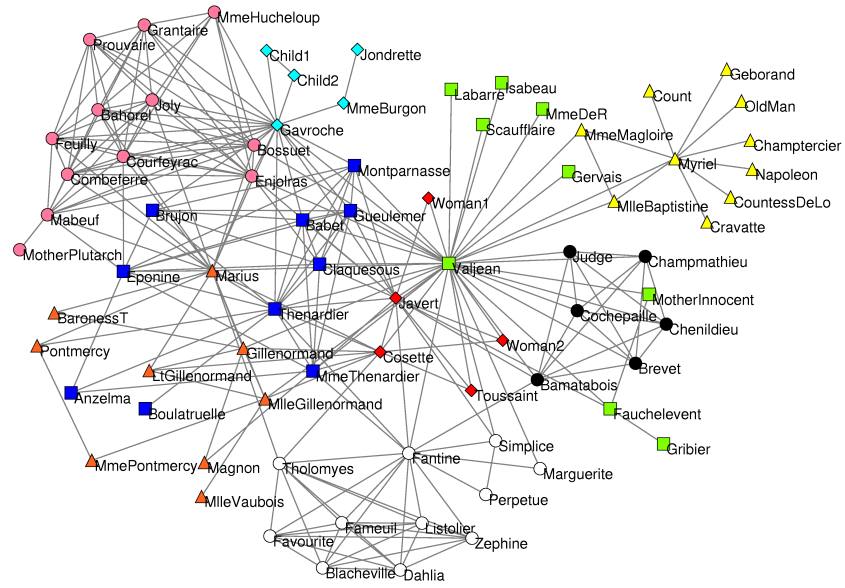


Figure 19. The nine communities identified by the α -aggregation heuristic for the *Les misérables* network (colors online).

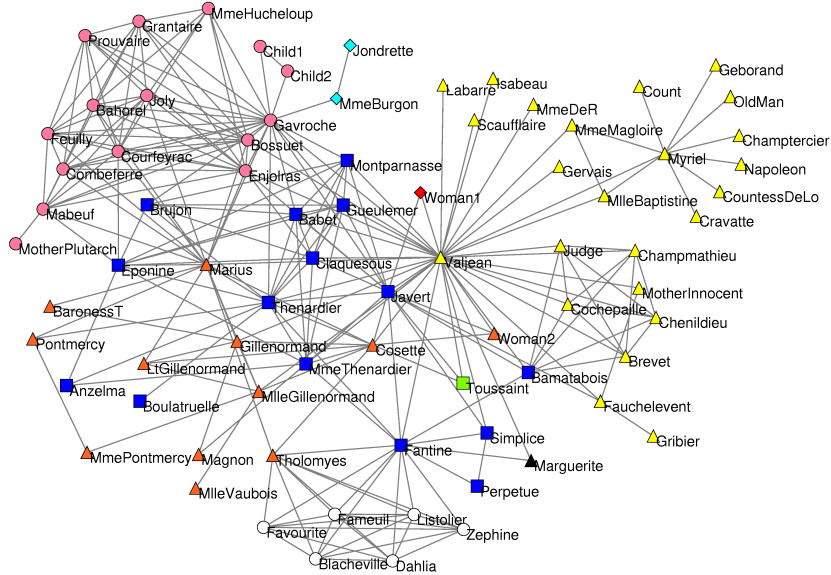


Figure 20. The nine communities identified by the CNM heuristic for *Les Misérables* network (colors online).

File				CNM α -aggregation			single rewire	
	n	m	Best known	Q	Q	time (s)	Avg. ΔQ %	Avg. time
Zachary	34	78	0.4198	0.3807	0.3893	0.00	2.28	0.00
Dolphins	62	159	0.5285	0.4954	0.5178	0.00	1.71	0.00
Les Misérables	77	254	0.5600	0.5006	0.5524	0.00	0.94	0.00
protein p53	104	226	0.5351	0.5227	0.5065	0.00	0.80	0.00
books about US politics	105	441	0.5272	0.5019	0.5036	0.00	0.51	0.00
American College Football	115	613	0.6046	0.5497	0.5690	0.00	1.32	0.00
A01_main	249	635	0.6329	0.6071	0.5695	0.01	0.57	0.00
USAir97	332	2126	0.3682	0.3190	0.3435	0.02	0.11	0.01
netscience_main	379	914	0.8486	0.8386	0.8364	0.01	0.14	0.01
Electronic circuit (s838)	512	819	0.8194	0.7976	0.7905	0.00	0.33	0.01
Erdos02	6927	11850	0.7162	0.6703	0.6850	2.18	0.03	2.27
PGPgiantcompo	10680	24316	0.8841	0.8521	0.8628	18.04	0.01	17.94
as-22july06	22936	48436	0.6750	0.6351	0.6418	80.48	0.00	80.20
DIC28_main	24831	71014	0.8468	0.7887	0.8069	147.61	0.03	146.93
cond-mat-2003_main	27519	116181	0.8146	0.6362	0.7220	205.59	0.01	207.36

Table I. Classical modularity. Optimal or best known solution, CNM solution, α -aggregation heuristic solution and computational time for 15 real-world networks; average difference in modularity and average computational time for small modifications of the network.

α values for which they are optimal. We also propose an agglomerative hierarchical heuristic, with the same time and space complexity as the Clauset Newman Moore heuristic⁶, to address large networks. It iteratively merges communities for increasing values of α . Experimental results show that (i) both the exact algorithm and the α -aggregation heuristic effectively detect communities for all values of the parameter α ; (ii) the dendrogram produced by the heuristic method provides a useful tool for substantive analysis; (iii) the difference between the parametric modularity values given by the exact algorithm and by the α -aggregation heuristic is moderate; (iv) for large instances the heuristic version of the proposed parametric method gives better results than the CNM heuristic for the maximization of the classical modularity.

-
- ¹ D. Aloise, S. Cafieri, G. Caporossi, P. Hansen, S. Perron, and L. Liberti. Column generation algorithms for exact modularity maximization in networks. *Phys. Rev. E*, 82:046112, Oct 2010.
- ² A. Arenas, A. Fernández, and S. Gómez. Analysis of the structure of complex networks at different resolution levels. *New Journal of Physics*, 10:053039, 2008.
- ³ J. W. Berry, B. Hendrickson, R. A. LaViolette, and C. A. Phillips. Tolerating the community detection resolution limit with edge weighting. *Physical Review E*, 83:056119, 2011.
- ⁴ U. Brandes, D. Delling, M. Gaertler, R. Gorke, M. Hofer, Z. Nikoloski, and D. Wagner. On modularity clustering. *IEEE Transactions of Knowledge and Data Engineering*, 20:172–188, 2008.
- ⁵ S. Cafieri, P. Hansen, and L. Liberti. Loops and multiple edges in modularity maximization of networks. *Physical Review E*, 81(4):046102, 2010.
- ⁶ A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Phys. Rev. E*, 70:066111, Dec 2004.
- ⁷ G.B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ, 1963.
- ⁸ J. Duch and A. Arenas. Community identification using extremal optimization. *Phys. Rev. E*, 72:027104, 2005.
- ⁹ P. Erdős and A. Rényi. On random graphs I. *Publicationes Mathematicæ*, 6:290–297, 1959.
- ¹⁰ S. Fortunato. Community detection in graphs. *Physics Reports*, 486:75, 2010.

- ¹¹ S. Fortunato and M. Barthelemy. Resolution limit in community detection. *PNAS USA*, 104:36, 2007.
- ¹² B.H. Good, Y.-A. de Montjoye, and A. Clauset. Performance of modularity maximization in practical contexts. *Physical Review E*, 81(4):046106, 2010.
- ¹³ M. Grötschel and Y. Wakabayashi. A cutting plane algorithm for a clustering problem. *Mathematical Programming B*, 45:59–96, 1989.
- ¹⁴ M. Grötschel and Y. Wakabayashi. Facets of the clique partitioning polytope. *Mathematical Programming*, 47:367–387, 1990.
- ¹⁵ D. Knuth. *The Stanford GraphBase: A Platform for Combinatorial Computing*. Addison-Wesley Reading, 1993.
- ¹⁶ A. Lancichinetti and S. Fortunato. *arXiv:1107.1155v1*.
- ¹⁷ T. Luczak. Sparse random graphs with a given degree sequence. In *Random Graphs, vol. 2*, pages 165–182, New York, 1992. Wiley.
- ¹⁸ C.P. Massen and J.P.K. Doye. Identifying communities within energy landscapes. *Physical Review E*, 71:046101, 2005.
- ¹⁹ M. Molloy and B. Reed. A critical point for random graphs with a given degree sequence. *Random Structures Algorithms*, 6:161–179, 1995.
- ²⁰ M. Newman. *Networks: An Introduction*. Oxford University Press, 2010.
- ²¹ M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Phys. Rev. E*, 69:066133, Jun 2004.
- ²² M. E. J. Newman. Modularity and community structure in networks. *PNAS*, 103:8577–8582, 2006.
- ²³ M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69:026113, 2004.
- ²⁴ J. Park and M.E.J. Newman. Origin of degree correlations in the internet and other networks. *Physical Review E*, 68:026112, 2003.
- ²⁵ P. Pons and M. Latapy. Post-processing hierarchical community structures: Quality improvements and multi-scale view. *Theoretical Computer Science*, 412(8-10):892–900, 2011.
- ²⁶ U.N. Raghavan, R. Albert, and S. Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Phys. Rev. E*, 76(3):036106, 2007.

- ²⁷ J. Reichardt and S. Bornholdt. Statistical mechanics of community detection. *Physical Review E*, 74:016110, 2006.
- ²⁸ P. Ronhovde and Z. Nussinov. Multiresolution community detection for megascale networks by information-based replica correlations. *Physical Review E*, 80(1):016109, 2009.
- ²⁹ P. Ronhovde and Z. Nussinov. Local resolution-limit-free potts model for community detection. *Phys. Rev. E*, 81:046114, 2010.
- ³⁰ J. Ruan and W. Zhang. Identifying network communities with a high resolution. *Phys Rev E*, 77:016104, 2008.
- ³¹ S. White and P. Smyth. A spectral clustering approach to finding communities in graph. *5th SIAM International Conference on Data Mining*, 2005.
- ³² G. Xu, S. Tsoka, and L.G. Papageorgiou. Finding community structures in complex networks using mixed integer optimisation. *The European Physical Journal B*, 60:231–239, 2007.
- ³³ W. Zachary. *Journal of Anthropological Research*, 33:452, 1977.