Assessing the Quality of Multilevel Graph Clustering

François Queyroi, Maylis Delest, Jean-Marc Fédou, Guy Melançon

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

Assessing the Quality of Multilevel Graph Clustering

Maylis Delest · Jean-Marc Fédou · Guy Melançon · François Queyroi

the date of receipt and acceptance should be inserted later

Abstract Hierarchical clustering of graphs is a useful strategy to mine, explore and visualize graphs. Popular approaches define *ad hoc* procedures to decide how subgraphs are subdivided or nested. The popularity of graph hierarchies certainly relates to the relevance of multilevel models appearing in the natural and social sciences. For instance, current models in biology (genomics and/or proteomics) try to capture the multilevel nature of networks formed by various biological entities; cities and worldwide city systems in geography can also be described as multilevel networks. In our opinion, a theory supporting these multilevel clustering approaches is yet to be developed. Indeed, to the best of our knowledge there are no known optimization multilevel criteria guiding the construction of a hierarchy of clusters: the hierarchy basically is an artefact of an iterative procedure. The main results of this paper contribute to such a multilevel clustering theory, by designing and studying a multilevel modularity measure for hierarchically clustered graphs, explicitly taking the nesting structure of clusters into account. The multilevel modularity we propose generalizes a modularity measure introduced by Mancoridis *et al.* in the context of reverse software engineering. The measure we designed recursively traverses the hierarchy of clusters and computes a one-variable polynomial encoding the intra and inter-cluster densities appearing at all levels in a hierarchical clustering. The resulting polynomial reflects how the graph combines with the hierarchy of clusters and can be used to assess the quality of a hierarchical clustering. We discuss archetypal examples as proof-of-concept. We also look at how this multilevel modularity acts on a popular real world example.

Keywords graph clustering · graph hierarchies · hierarchical clustering · multilevel modularity

CR Subject Classification G.2.1 Combinatorics · I.5.3 Clustering

Maylis Delest, Guy Melançon, François Queyroi
Université de Bordeaux, CNRS, LaBRI, INRIA Bordeaux – Sud-Ouest, France
E-mail: *first-name.last-name*@labri.fr

Jean-Marc Fédou
Université de Nice,CNRS UMR 6070 I3S, France
E-mail: fedou@unice.fr

1 Introduction

Identifying community structures and outliers remains a central task when mining graphs [9]. Numerous graph clustering strategies and algorithms have been developed, where a majority of them aim at modularity maximisation (see for instance recent survey papers [8], [7] and [29]). The results in this paper precisely relate to the situation where optimal modularity is assessed using a quality measure. Candidate measures have been introduced by several authors. The Newman's Q modularity [21] measures the difference between the observed proportion of links within clusters and its expected value in a random graph with the same degree sequence [16]. Other clustering quality measures have been studied and used to benchmark algorithms, such as the average Normalized Cut [28].

Related work : This paper focuses on a clustering quality measure inspired by Mancoridis *et al.* [18] (denoted as MQ) defined in terms of intra-cluster density versus inter-cluster connectivity ratios. In a manner similar to Newman and Girvan using modularity together with edge betweenness [21], Auber *et al.* [3] used Mancoridis' MQ quality measure combined with an edge statistics in an effort to identify bridges between communities and obtain multilevel clustering for small world networks. Examples successfully clustered using MQ are (sub)graphs of the Internet movie database (IMDB), the worldwide air passenger traffic [1] or the co-citation network built from the IEEE InfoVis proceedings [11].

Two main strategies are used to produce a hierarchy of clusters (nested subgraphs). Divisive approaches usually first produce a clustering of a graph (a set partition of its vertices), and then iterate over each subgraph until some stopping condition is met. Agglomerative approaches first consider clusters formed of single vertices and merge them into larger groups following some criteria or objective function. After such a hierarchy of clusters is produced, either the hierarchy can be preserved and manipulated as is, or a "cut" must be decided, based on some other criteria to find a best possible clustering out of the hierarchy. Deciding of an optimal cut, or deciding of the optimal depth for the hierarchy is a difficult question. In our view, the main reason why iterative (divisive or agglomerative) strategies cannot reasonably guide the overall nesting process is clear. They fail to evaluate the very hierarchical character of the clustering they produce. This is the question we address here: find a criteria evaluating the relevance of the hierarchy. Applying a modularity measure to obtain clusters $C_1, C_2, \ldots$ and then independently re-apply the measure on each cluster, and so forth, does not explicitly take the nesting structure into account. That is, even if a best possible clustering is sought for at each iteration step, the overall quality of the multilevel clustering needs to be measured or assessed. To the best of our knowledge, although many authors designed *ad hoc* algorithms producing hierarchical clusterings of a graph, none of them provided an accompanying multilevel modularity. There is one exception however [17], where the authors compute a multilevel classification of concepts into categories based on a numerical evaluation of the resulting hierarchies. Their approach however does not transfer in the context of multilevel graph clustering.Another interesting approach is due to Pons and Latapy[24]: they propose an extension of well-known quality measures which includes a scale parameter and they define a post-processing procedure to retrieve the most relevant cuts from a dendrogram (binary clustering tree). However this extension can not be used to compare the qualities of several clustering trees.

Contribution : Our results can be seen as a contribution to theoretical foundations for hierarchical graph clustering. A multilevel presentation of information through quotient graphs provides a useful abstraction of the initial data. More importantly, current studies confirm the absolute presence of hierarchies either in nature itself or in abstract human construction such as language. Current evolutionary models in biology try to capture the multilevel nature of networks formed by various biological entities [31]. The same holds for cities and city systems in geography [25]. Obviously, approaches claiming to unfold such structures in networks should rely on sound principles and methodology for hierarchical graph clustering.

The multilevel criteria we present and discuss in this paper generalizes a one level criteria first introduced by Mancoridis *et al.* [18]. We focused our effort on Mancoridis' MQ modularity measure for several reasons, one being that it possesses interesting statistical properties [10], the other being that it nicely admits a multilevel generalization, making it a good candidate quality measure among others. Our multilevel measure collects values along a traversal of all clusters and sub-clusters ending into a polynomial whose coefficients reflect how the graph combines with the hierarchy of clusters. We borrowed ideas from standard techniques in algebraic combinatorics where such polynomials appear when enumerating recursive discrete objects. The idea is to exploit a variable q to keep track of the intrinsic depth of objects. In most cases, the objects can be described by formal languages generated by algebraic grammars, generally called *attribute grammars* after a counting variable q is introduced [12, 19]. A first attempt at defining this multilevel measure was conjectured by some of the authors of the present paper[10]. This previous work was not theoretically comprehensive and not experimentally successful.

Section 3 motivates the design this one variable multilevel modularity. The whole discussion incrementally builds towards the full generalization by going through a careful examination of MQ and its underlying mechanism. Looking at archetypal case studies, Section 4 provides a rationale for such an adaptation of Mancoridis' original formulation. In Section 5, we look at two real world examples to assess of the relevance of our multilevel modularity. First, we compare several algorithms recursively applied to a college football network that has been the focus of previous work. Secondly, we present a evaluation of a classic hierarchical clustering procedure applied to a French commuting network. While the data of the first example are publicly available, the data of the second are unfortunately not.

2 Mancoridis' modularity

Mancoridis *et al.* [18] proposed a modularity measure they called MQ (standing for *Modularity Quality*) evaluating the quality of a clustering (of a graph) as a difference between internal and external connectivity ratios. That is, the ratio between the number of connections observed in a given module or between two given modules and the maximum possible number of such edges. Obviously, MQ applies to any graph and clustering although it was first introduced in the context of reverse software engineering to cluster graphs induced from references between source code files.

Let $G = (V, E)$ be a graph where V and E respectively denote the set of nodes (also called vertices) and edges of G. Let $\mathbf{C} = (C_1, \ldots, C_k)$ be a *clustering*, that is, the subsets $C_i \subset V$ are pairwise disjoint and cover $V = \cup_{i=1}^{k} C_i$. Given two clusters C_i, C_j, we define e_{ij} as the number of edges connecting vertices of C_i to vertices of C_j (or vice versa). In this context, e_{ii} denotes the number of edges *within* C_i.

The modularity measure $\widetilde{MQ}$ we now define slightly extends Mancoridis' original modularity, and involves internal and external connectivity ratios for each cluster C_i, respectively denoted as α_i and β_i. We also need to specify upper bounds δ_i and δ_{ij} on the number of edges lying within C_i or between C_i and C_j (depending on a reference graph model, see forthcoming examples and sections). Moreover, we assign a weight x_i associated with each cluster C_i and we set $X = \sum_{i=1}^{k} x_i$. In a sense, the quantity X can be seen as a weight associated with the whole graph G, or more precisely to the set of vertices V. We furthermore require that these weights to be *additive*, meaning that if C_i is decomposed into (pairwise disjoint) sub-clusters $C_{i1}, \ldots, C_{ik_i}$, we then have $x_i = \sum_{p=1}^{k_i} x_{ip}$.

Definition 1 The *internal connectivity ratio* of the cluster $C_i \in \mathbf{C}$ is defined as the relative amount of internal edges in cluster C_i and equals:

$$\alpha_i = \frac{e_{ii}}{\delta_i} \tag{1}$$

Remark 1 A natural upper bound δ_i for subgraph density is $\binom{|C_i|}{2}$ when dealing with simple graphs (undirected, no loops). This definition implicitly sets the complete graph as a reference model where cluster density is measured against a clique of comparable node size. However, finding a subset of nodes $C_i \subset V$ maximizing α_i in this case is a NP-hard problem. This has motivated the use of alternate definitions for edge density [30]. Finally, we do not consider here the particular case where the δ are null. The situation could however arise when computing the density of a singleton.

Definition 2 The *external connectivity ratio* of the cluster $C_i \in \mathbf{C}$ is defined as a weighted mean of the relative amount of external edges between C_i and the other clusters and equals:

$$\beta_i = \frac{1}{X - x_i} \sum_{j \neq i} \frac{x_j e_{ij}}{\delta_{ij}} \tag{2}$$

Remark 2 A natural upper bound δ_{ij} for external density subgraph density, which furthermore matches the internal density $\delta_i = \binom{|C_i|}{2}$ discussed in the previous remark, is $\delta_{ij} = |C_i| \cdot |C_j|$. This definition implicitly sets the complete bipartite graph as a reference model.

Definition 3 Let G be a graph, and $\mathbf{C} = (C_1, \ldots, C_k)$ be a clustering of G. The generalized modularity (denoted $\widetilde{MQ}$) is defined as:

$$\widetilde{MQ}(G; \mathbf{C}) = \frac{1}{X} \sum_{i=1}^{k} x_i (\alpha_i - \beta_i) \tag{3}$$

The quantity in Eq. (3) should be seen as a weighted average of the ratio difference (between the quantities defined in Eq. (1) and Eq. (2)). That is, larger cluster have a higher ratio $\frac{x_i}{X}$ and correspondingly have more impact on the final value computed in Eq. (3).

Example 1 Let us briefly show how Mancoridis' original definition can be recovered from Eq. (3). First set uniform weights for all clusters, that is $x_i = 1$, for all $i = 1,\ldots,k$. We consider directed graphs and allow loops. Take as reference graphs the (directed) complete graph, and the directed bipartite graph. Accordingly set $\delta_i = |C_i|^2$ and $\delta_{ij} = 2|C_i||C_j|$. Eq. (3) then unfolds as the original MQ measure [18]:

$$MQ(G;\mathbf{C}) = \frac{1}{k}\sum_{i=1}^{k}\left(\frac{e_{ii}}{|C_i|^2} - \frac{1}{k-1}\sum_{j\neq i}\frac{e_{ij}}{2|C_i||C_j|}\right) \tag{4}$$

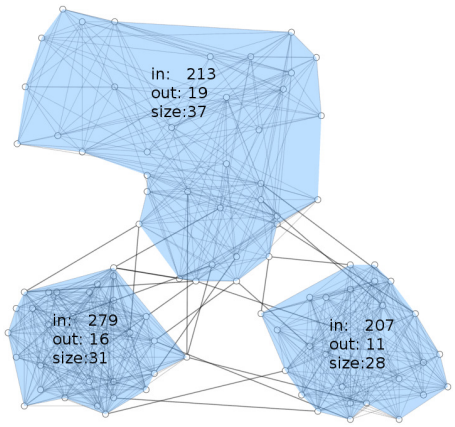


Fig. 1 Flat clustering of a small simple network ($n = 96$). The three cluster C_i are drawn using convex hulls. The *in*, *out* and *size* quantities are e_{ii}, $\sum_{j\neq i} e_{ij}$ and $|C_i|$ respectively.

Example 2 We now consider simple graphs (undirected, no loops) and use the size of a cluster C_i as its weight ($x_i = |C_i|$). Take as reference graphs, the complete graph and bipartite complete graphs, we have $\delta_i = \binom{|C_i|}{2}$ and $\delta_{ij} = |C_i||C_j|$. Then:

$$\widetilde{MQ}(G;\mathbf{C}) = \frac{1}{n}\sum_{i=1}^{k}\left(\frac{2e_{ii}}{|C_i|-1} - \frac{1}{n-|C_i|}\sum_{j\neq i} e_{ij}\right) \tag{5}$$

additionally assuming $|C_i| \geq 2$, $\forall i = 1,\ldots,k$, and where we set $n = |V|$. Mancoridis' original definition (as used in [3]) considers clusters to be of equal importance and simply averages the density of all clusters, while the identity we use here computes a weighted average again giving more impact to larger clusters (see also [6] who pointed at this improvement). Looking at the example given in Fig. 1 we have

$$\begin{aligned}\widetilde{MQ}(G;\mathbf{C}) &= \frac{1}{96}\left(\frac{2\times 213}{37-1} - \frac{19}{96-37}\right)\\ &+ \frac{1}{96}\left(\frac{2\times 279}{31-1} - \frac{16}{96-31}\right)\\ &+ \frac{1}{96}\left(\frac{2\times 207}{28-1} - \frac{11}{96-28}\right)\\ &\simeq 0.47\end{aligned}$$

Roughly speaking, $\widetilde{MQ}$ (as defined in Eq. (5)) seeks at finding dense subgraphs assigning a maximum score to cliques (complete subgraphs). As a result, $\widetilde{MQ}$ tends to prefer small cliques to larger but less dense subgraphs. Using the de Moivre-Laplace theorem, one can show that when G is a random Erdös-Rényi graph [13] with link probability p, and for a fixed clustering $\mathbf{C}$, the quantity defined in (5) can be approximated by a Gaussian distribution of zero mean (we also need to assume $i > 1$). This observation corresponds to the idea that the probability of finding a clustering of random graph where clusters have a much larger inner connectivity ratio than external connectivity ratio is rather small.

3 Multilevel Modularity

3.1 Basic idea

The extension of $\widetilde{MQ}$ to hierarchical graph clustering relies on a recursive definition involving a variable q.

Observe first that $\widetilde{MQ}$ in Eq. (3) can be computed by going through each individual edge, testing whether it connects nodes belonging to a same cluster or to different ones. The terms in Eqs. (1) or (2) can then be seen as positive or negative weights assigned to edges of the graph. Leaving all averaging constants and edge densities aside these weights end up being ± 1.

When dealing with multilevel clustering, our goal is to take the depth at which an edge acts into account. It may occur that an edge remains internal as we drill down the hierarchy over several levels. The intuition here is that this edge should be assigned a positive weight $1 + q + \cdots + q^r$ depending on the depth r of the deepest cluster it resides in. Conversely, an external edge joining two different clusters should be assigned a negative weight depending on the depth of the two clusters it connects in the hierarchy. Now, the situation becomes intricate since an edge might well be internal starting from the root down to some level of the hierarchy, while it becomes external and connects two distinct lower level clusters. It is this combinatorial complexity we need to capture here.

3.2 Multilevel recursive definition

Let T be a *rooted tree*, that is a directed graph where *leaf nodes* have no successors, and each node has a unique *parent* node, except for the root node. Let $\sigma(t)$ denote the set of all *siblings* having t as common parent node in $\mathbf{T}$. We denote by $h(\mathbf{T})$ the *height* of $\mathbf{T}$, that is the length of a longest path from the root to a leaf node.

A *hierarchically clustered graph* $G = (V, E, \mathbf{T})$ comes equipped with a cluster tree $\mathbf{T}$ where each node $t \in \mathbf{T}$ corresponds to a subset $V(t) \subset V$, subject to the constraints $V(t) = \bigcup_{t' \in \sigma(t)} V(t')$ and $V(t') \cap V(t'') = \emptyset$ for any two siblings $t', t'' \in \sigma(t)$. By definition, all (subsets associated with) siblings C_i ($i = 1,\ldots,k$) having the root node as direct ancestor provide a flat clustering of the graph. Some of these subsets then refine into hierarchically clustered graphs $G(C_i) = (C_i, E(C_i), \mathbf{T}(C_i))$, where $G(C_i) = (C_i, E(C_i))$ denotes the subgraph induced from C_i and $\mathbf{T}(C_i)$ denotes the hierarchy induced from the subtree rooted at C_i. That

is, $G(C_i) = (C_i, E(C_i), \mathbf{T}(C_i))$ itself recursively decomposes into a lower level hierarchical clustering. Note that we do not require that the lowest level clusters be single nodes $v \in V$. For sake of simplicity, we shall write G_i and $\mathbf{T}_i$ to denote $G(C_i)$ and $\mathbf{T}(C_i)$ respectively. We also identify clusters C_i with the subtree $\mathbf{T}_i$ rooted at C_i.

Definition 4 Let $G = (V, E, \mathbf{T})$ be a hierarchically clustered graph with top level clusters $C_1, \ldots, C_k$. For any real number $q \in [0, 1]$, its *multilevel modularity* is defined as:

$$\widetilde{MQ}(G; \mathbf{T}; q) = \begin{cases} \frac{1}{X} \sum_{i=1}^{k} x_i(\alpha_i - \beta_i) \left(1 + q\widetilde{MQ}(G_i; \mathbf{T}_i; q)\right) & \text{if } k > 0 \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

Note that when $\mathbf{T}_i$ is a flat clustering of G, we then have $\widetilde{MQ}(G_i; \mathbf{T}_i; q) = 0$ since $\mathbf{T}_i$ is a leaf node (lowest level cluster) in $\mathbf{T}$. As a consequence, $\widetilde{MQ}$ does coincide with Eq. (3) for flat clustering (a cluster tree of depth one).

The reasons for the bounds on q are obvious. On the one hand, allowing $q < 0$ would bring a negative contribution from internal edges, while external edges would contribute positively. On the other hand, choosing $q > 1$ would lead to an odd situation where bottom clusters of $\mathbf{T}$ may contribute more to $\widetilde{MQ}(G; \mathbf{T}; q)$ than the first level clusters although they represent a refinement of their parent clusters.

3.3 $\widetilde{MQ}$ as weighted paths in a tree

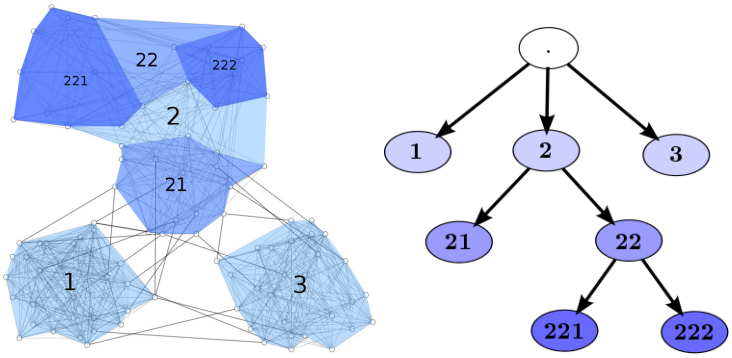
Although Def. 4 introduces a recursive pattern to compute $\widetilde{MQ}(G; \mathbf{T}; q)$ as a polynomial in q, we can provide a combinatorial formula to directly compute the coefficient of q^p.

Now, assume sibling nodes in $\mathbf{T}$ are labeled using distinct integers $1, 2, \ldots$ Any path going from the root node to any other node in the tree can then be described as an integer sequence $w = i_1 \ldots i_r$. We shall call such a sequence a *word* over the alphabet $\{1, 2, \ldots\}$. Fig. 2(b) illustrates this construction: the word encoding the path from the root node is depicted for each node in the tree. Now, given a word $w = i_1 \ldots i_r$, a prefix of w is a word $u = i_1 \ldots i_s$ with $s \leq r$. Note that prefixes incrementally build as we traverse the path from the root and visit all intermediate nodes. We shall write $u \prec w$ when the word u is a prefix of the word w. This happens to be an order relation on words which coincides with the (inverse) set inclusion order on clusters in the hierarchy, so words w uniquely map to a cluster C in the hierarchy. We write $|w|$ to denote the length of the integer sequence w (which also equals the depth of the corresponding cluster in the hierarchy) and $\mathcal{L}_{\mathbf{T}}$ to denote the set of leaf nodes in $\mathbf{T}$.

Using these notations we provide a closed formula for the coefficient of $\widetilde{MQ}$. In order to access the contribution of a cluster C in $\mathbf{T}$ with depth $p + 1$, we need to multiply differences between inner and outer connectivity ratios for each cluster located on the path to C. The coefficient $[\widetilde{MQ}(G, \mathbf{T}, q); q^p]$ is then then given by the sum of this quantity over all clusters at depth $p + 1$, as given in Prop. 1.

Property 1 Let $\mathcal{D}_p = \{w \in \mathbf{T}, |w| = p + 1\}$ be the the set of clusters at depth p in $\mathbf{T}$. We have:

$$[\widetilde{MQ}(G, \mathbf{T}, q); q^p] = \frac{1}{X} \sum_{w \in \mathcal{D}_p} x_w \prod_{u \prec w} (\alpha_u - \beta_u) \quad (7)$$



(a) Hierarchical decomposition of the sample graph introduced in Fig. 1.

(b) Tree representation of the hierarchical clustering.

Fig. 2 A labeled tree (right) encoding a hierarchical clustering of a graph (left). All paths from the root to a cluster C_w are described using *words*.

A crucial ingredient to Eq. (7) is the identity $x_i = \sum_{j=1}^{k_i} x_{ij}$, which holds since we assumed the x_i's are additive.

Eq. (7) provides an alternative way to compute $\widetilde{MQ}(G, \mathbf{T}, q)$. Assuming all quantities $(\alpha_u, \beta_u)_{u \in \mathbf{T}}$ are given, the time complexity for computing $\widetilde{MQ}(G, \mathbf{T}, q)$ is however $\mathcal{O}(n \log(n)^2)$ (where $n = |V|$ denotes the number of vertices in G). This is to be compared against a $\mathcal{O}(n \log(n))$ time complexity when using recursion as in Eq. (6).

3.4 Interpreting values of $\widetilde{MQ}$

Observe that $\widetilde{MQ}(G; \mathbf{T}; q)$ achieves our goal since internal edges will be visited several times, once as edges in $G(C_i)$, then as edges in $G(C_{ij})$ and so forth, each time collecting a different power of q as the recursion goes down the hierarchy. The same type of "depth dependent weight" is achieved for external edges. The case where q is close to 1 corresponds to the extreme situation where the weight of an (internal) edge equals its depth in the hierarchy. On the other hand, a value of q close to 0 corresponds to the one-level $\widetilde{MQ}$ value (Eq. 3) applied on the first level of $\mathbf{T}$. As we shall see (in Section 4), the value assigned to q actually plays a role in determining whether one should favor a clustering extending to more or less levels. Roughly speaking, a denser cluster may have a smaller contribution than a cluster sitting at a lower level while being less dense, depending on the value of q (and the depth of the cluster).

Given a hierarchically clustered graph $(G, \mathbf{T})$, and q being considered as a variable, the expression $\widetilde{MQ}(G; \mathbf{T}; q)$ can be seen as a polynomial in q. Obviously, two different clustering trees $\mathbf{T}, \mathbf{T}'$ of a same graph return different polynomials, that may only slightly differ when these two clusterings are "close". Similarly, we expect a larger graph G' equipped with a hierarchical clustering structurally similar to that for G to return a similar polynomial. That is, when plotted as curves over $[0, 1]$, the two polynomials should correspond to similar and close curves. Note that this is more likely to happen when $\mathbf{T}$ and $\mathbf{T}'$ share the same (non labeled) tree

structure, so the polynomials will only vary in their coefficients but will involve the same recursive expansions and powers of q.

Comparing two hierarchical clusterings based on polynomial expressions may be unsatisfactory or insufficient to take decisions. While there is no obvious way to determine the right value for q to run such a comparison, a heuristic is to take the average of $\widetilde{MQ}(G;\mathbf{T};q)$ over $q \in [0,1]$. This can easily be accomplished by computing $\widetilde{MQ}(G;\mathbf{T})$ as an integral using Eq. (7):

$$\begin{aligned}\widetilde{MQ}(G;\mathbf{T}) &= \int_0^1 \widetilde{MQ}(G;\mathbf{T};q)dq \\ &= \frac{1}{X}\sum_{p=0}^{h(\mathbf{T})-1}\frac{1}{p+1}\sum_{v\in\mathcal{D}_p} x_v \prod_{u\prec v}(\alpha_u-\beta_u)\end{aligned} \tag{8}$$

4 Proof of concept: archetypal case studies

We now look at special and simple cases in order to understand how $\widetilde{MQ}(G,\mathbf{T};q)$ actually works. We shall also look at more complex examples later on. We will only consider simple graphs (undirected, no self-loops). We shall use the complete and bipartite complete graphs as reference graphs (cf. Section 2, Ex. 2). Recall that we use the size of a cluster C_i as its weight ($x_i = |C_i|$). These examples are constructed in order to be convinced of the accuracy of our measure. In [10] the measure proposed by the authors did not fit with what one can expect on such examples. This is why the new formula (Eq. (6)) was given.

4.1 A simple case study

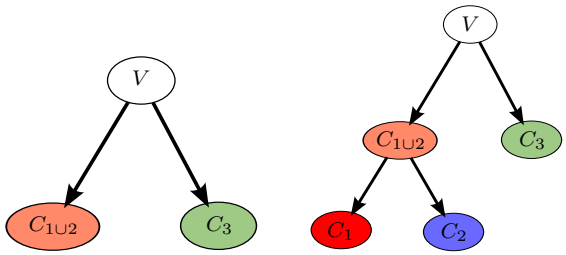
Our multilevel modularity can be used to decide whether to further subdivide a cluster or not. Observe that two trees sharing the same structure on nodes of depth $\leq p$ will have equal coefficients $[\widetilde{MQ};q^r]$ with $r \leq p$. Hence, these cluster trees may only be compared based on local criterion.

A simple example will illustrate this idea. Assume G is a graph formed of three distinct cliques C_1, C_2, C_3 (taken as the archetype of a cluster) of size n. Assume also there are bn^2 edges ($0 \leq b \leq 1$) connecting C_1 to C_2, but that there are no edges between C_3 and either C_1 or C_2. This example allow us to compare analytically the different configurations according to simple $\widetilde{MQ}$ expressions.

Write cluster trees as parenthesized expressions, and consider cluster trees $\mathbf{T} = [C_{1\cup2}, C_3]$ and $\mathbf{T}' = [[C_1, C_2], C_3]$ (see Fig. 3). That is, $\mathbf{T}$ is a flat clustering with a first cluster containing the union of C_1 and C_2, while $\mathbf{T}'$ further divides this cluster into sub-clusters $[C_1, C_2]$.

Since both trees coincide on the first level, comparing their modularity amounts to decide whether there is any benefit to further divide $C_{1\cup2}$ into $[C_1, C_2]$. Now, the internal connectivity ratio for $C_{1\cup2}$ is (see Eq. (1)):

$$\alpha_{1\cup2} = \frac{2n(1+b)-2}{2n-1}$$



(a) **T** (b) **T**'

Fig. 3 Two different hierarchical clusterings of a graph built from three cliques.

Since the tree **T** is flat, its modularity $\widetilde{MQ}$ is constant (as a polynomial in q). We can furthermore evaluate this situation by letting n increases toward ∞ to obtain expression solely depending on b:

$$\widetilde{MQ}(G;\mathbf{T};q) = \frac{2}{3} + \frac{b}{3}$$
$$\widetilde{MQ}(G;\mathbf{T}';q) = \frac{1}{3}\left(1 + (1+b)\big[1 + q(1-b)\big]\right)$$

Note that we indeed have $\widetilde{MQ}(G;\mathbf{T};0) = \widetilde{MQ}(G;\mathbf{T}';0)$, as expected. The comparison of these two clusterings relies on the value of

$$[\widetilde{MQ}(G;\mathbf{T}';q),q] = \frac{q(1-b^2)}{3}$$

This positive quantity is a decreasing function of b, which confirms an obvious phenomenon: as long as C_1 and C_2 are not too densely interconnected, it makes sense to divide $C_{1\cup2}$ into two sub-clusters, while they should be kept as a single cluster when their inter-connectivity ratio approaches higher values.

We can also compare the two previous trees with the following configuration $\mathbf{T}'' = [C_1, C_2, C_3]$. We have

$$\widetilde{MQ}(G;\mathbf{T}'';q) = 1 - \frac{b}{3}$$

which is obviously a decreasing function of b. If $b = 0.5$, **T** and $\mathbf{T}''$ have equal $\widetilde{MQ}$ values. Note that $\mathbf{T}'$ and $\mathbf{T}''$ quality values overlap in the range $b \in [0, 0.5]$. In this case, a high value of q tends to promote the hierarchical clustering. Actually we have

$$\forall b \in [0,0.5],\ \widetilde{MQ}(G;\mathbf{T}';q) = \widetilde{MQ}(G;\mathbf{T}'';q) \Leftrightarrow q = \frac{2b-1}{b^2-1}$$

which has a nearly linear decreasing behaviour. It means that the more b is high the less we need to promote hierarchy to rank the $\mathbf{T}'$ configuration as best.

As said in Section 3.4, if we have no preference about the value of q to use, a simple solution is to consider as the best the configuration which is above the other for the longest range of q or equivalently compare the average of $\widetilde{MQ}$ as defined in Eq. (8). In this case the clustering trees $\mathbf{T}'$ and $\mathbf{T}''$ have an equal quality for $b \simeq 0.25$. It is reasonable to assume that the tree $\mathbf{T}'$ will be preferred to the flat clustering $\mathbf{T}''$ before **T** because its leaves correspond to the three cliques.

4.2 More complex cluster trees

We now consider cluster trees built from four different clusters and show how $\widetilde{MQ}$ helps predict which is the most relevant hierarchical clustering, depending on the inter-cluster connectivity ratios.

We will here compare four different cluster trees: the *flat* tree, the *3-2* tree, the *complete* tree and the *linear* tree (see Fig. 4). Comparing the modularity of these hierarchical clusterings should help to decide on the appropriate tree structure, since all of these trees have the same leaf clusters. We assume all bottom clusters C_1, C_2, C_3, C_4 to be cliques of equal size n, and we write b_{ij} for the external connectivity ratio between C_i and C_j. We consider four different cases (see Fig. 5) and always assume $b_{14} = 0 = b_{24} = 0$ (cluster C_4 never connects with clusters C_1 or C_2).

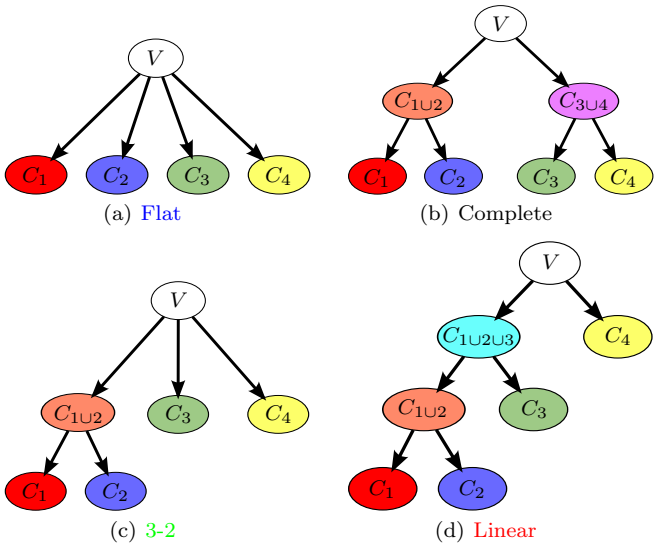
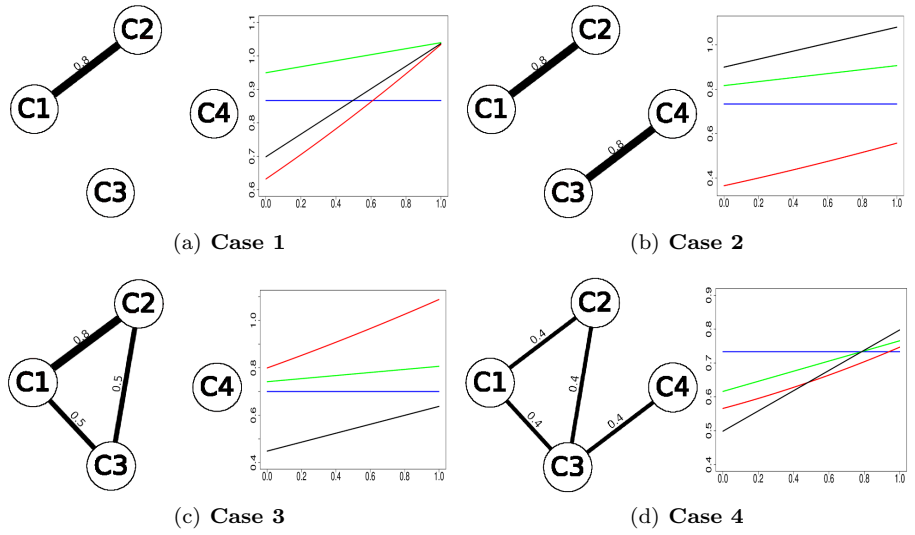


Fig. 4 Different clusterings of size 4.

Fig. 5 shows the curves of the four polynomials we get. Note that the polynomial of the flat tree is constant, while the 3-2 and complete tree have degree one $\widetilde{MQ}$ polynomials. The measure for the linear tree is a quadratic curve. The following conclusions can be made:

- **Case 1 :** When b_{12} is much greater than all others b_{ij}, the modularity $\widetilde{MQ}$ ranks the 3-2 tree as the best option. This obviously is the best possible case between all considered trees.
- **Case 2 :** When b_{12} and b_{34} are much greater than all others b_{ij}, then the complete tree is the best available option.
- **Case 3 :** The linear tree becomes the best candidate when the connectivity ratios verify $b_{12} \gg b_{13} \simeq b_{23} \gg$ others b_{ij}.



(a) **Case 1** (b) **Case 2**

(c) **Case 3** (d) **Case 4**

Fig. 5 For each case a quotient graph representation of the flat clustering is provided (left). The edge label indicates the connectivity ratio between the two cluster it connects (0 when there are no edges). The $\widetilde{MQ}$ curves for flat (Blue), 3-2 tree (Green), Complete (Black) and Linear (Red) cluster trees are also given (right).

The **Case 4** reveals overlaps between the curves. As a matter of fact this case illustrates a situation where the best clustering option is not that obvious. The variable q is used to favor (when close to 1) or restrain (when close to 0) a deep hierarchical clustering. On one hand take a small q value leads to rank the flat clustering as best. On the other hand a value close to 1 ranks the complete tree as the best choice. Using the averaged modularity criterion (see Eq. (8)), the flat clustering is however the best solution.

5 Application on real world examples

In this section we show how multilevel modularity $\widetilde{MQ}$ can be used to compare hierarchical clusterings of real world networks.

5.1 College football network

We consider an example borrowed from [15] describing the organization of the American College Football season schedule of Division IA Nodes of this graph represent teams and edges connect teams that played together along the season. This graph comes with an obvious clustering criteria since the teams are divided up into 11 conferences[1]. Actually three of them (Big Twelve, South Eastern and

[1] Actually, the groups of teams provided by the authors correspond to the 2001 conferences. Thanks to T.S. Evans, we use here the correct conferences of the 2000 season.

Mid-American) are subdivided into two clusters which leads to a multilevel decomposition of the network.

The graph is of limited size and contains 115 vertices and 613 edges with a mean degree of 10.66 and an average clustering coefficient of 0.4. This last statistics suggests that communities exist in this sparse network. Although games are more likely to occur within a conference, they also seem to depend on the geographical proximity of the teams' hometowns.

We present here an application of our multilevel criteria to evaluate the recursive application of clustering algorithms. This way of producing a hierarchical decomposition is intuitive but not so much studied. It is based on the assumption that similar connectivity pattern can be found at different scale.

The College Football graph has been clustered using three different algorithms. Two of them actually produce flat clusterings and have been iterated over clusters in order to obtain multilevel clusterings. The first is the Hierarchical Clustering[14] with the Jaccard index as similarity metric and the one level $\widetilde{MQ}$ quality measure to select the best threshold value. The second is the MLR-MCL algorithm [28]. We also used the Louvain algorithm [4] that actually produces a hierarchical clustering. We directly used the source code provided by the respective authors, then ran the algorithm and visualized the results using the Graph Visualization framework Tulip [2]. All of these procedure are unsupervised and do not require any parametrization.

The complete and bipartite complete graphs were used as reference graphs for inter and intra connectivity ratios. Our goal was to compare the grouping of teams into conferences with the different hierarchical clusterings output by the different algorithms. As far as the clustering into conferences is concerned, it made sense to set all clusters to have equal weights $x_i = 1$, and be considered equally important whatever their size (number of teams in a conference). As a consequence, weights of leaf clusters in all other hierarchical clusterings were also set to $x_i = 1$. Because we need to insure additivity of these weights, we had to set

$$x_w = \begin{cases} 1 \text{ if } w \in \mathcal{L} \\ |\mathcal{L}_w| \text{ otherwise} \end{cases}$$

where $\mathcal{L}_w$ is the set of T leaves having w as ancestor node.

A visualisation of the results is provided in Fig. 6 using nested graphs. As one could expect, the four hierarchical clusterings agree on a majority of groups, which can be easily explained by the fact that teams of a same conference play together more often.

Louvain (Fig. 6(c)) algorithm tends to group conferences located in a same region. For example Mountain West and Big West conference are merged at the first level. The Hierarchical clustering algorithm (Fig. 6(d)) results in many dense groups which most of the time correspond with conference or the subdivision of some conferences. The MLR-MCL (Fig. 6(b)) algorithm actually produce a hierarchy that is really close to the division into conferences (Fig. 6(a)). Both refine some of the biggest conferences into denser sub-clusters which makes sense geographically, although independent teams are affected to different conference in the case of the MLR-MCL algorithm. These four hierarchies can be compared using $\widetilde{MQ}$.

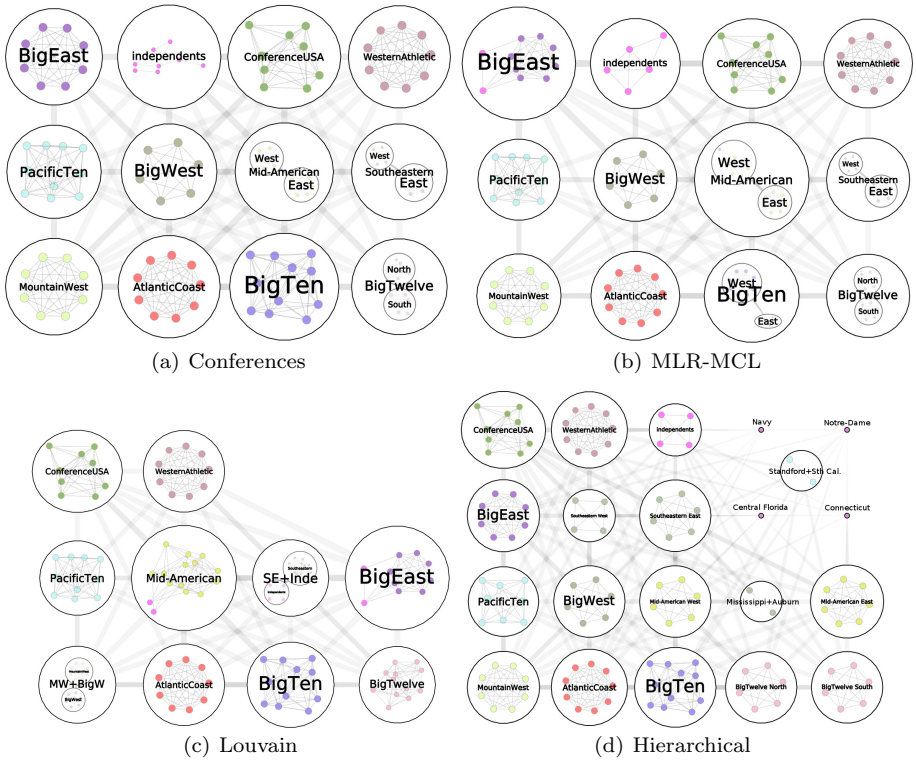


Fig. 6 Nested graph representations of the College Football conferences using several clustering algorithms.

Algorithm	$\widetilde{MQ}(G,\mathbf{T},q)$	$\widetilde{MQ}(G,\mathbf{T})$
MLR-MCL	$0.782064 + 0.143891q$	$0,8540095$
Conferences	$0.774255 + 0.148196q$	$0,848353$
Louvain	$0.688232 + 0.131802q$	$0,754133$
Hierarchical	0.719687	0.719687

Table 1 Polynomials and averaged modularities for the five clusterings of the Football network.

The Fig. 7 reports the resulting polynomials. In Table 1 we rank the algorithms using the averaged modularities as defined in Eq. 8. Several conclusions can be made:

- As said before the MLR-MCL clustering result is very close to the multilevel partition into conference and division. We can see that their respective $\widetilde{MQ}$ curves are very close. Still MLR-MCL produces a better clustering (mostly due to the splitting of the independent teams).
- The lowest level clusters of the Louvain hierarchy 6(c) match the division into conferences. But merging close conferences does not seem to be a good strategy, even if we do not need to promote hierarchy (with a high q) to prefer the Louvain algorithm over the Hierarchical clustering.

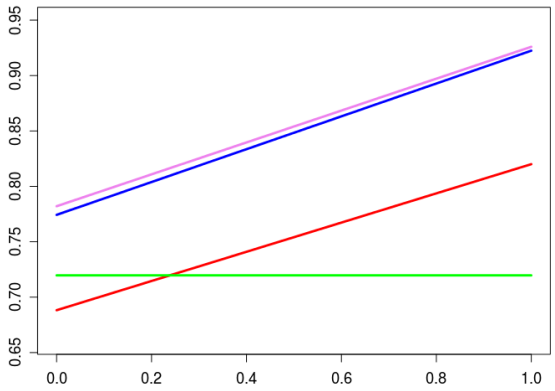


Fig. 7 $\widetilde{MQ}$ curves for Conferences partition (Blue), Hierarchical clustering (Green), MLR-MCL clustering (Violet) and Louvain clustering (Red).

5.2 French commuting network

We now look at a larger example which is the commuters' flows occurring in the French administrative region *Pays-de-la-Loire* using the results of the 1999 French national census (*source : INSEE*). The data are unfortunately not publicly available.

Commuting is defined as the regular travel between the place of living and the place of work[27], these flows are interesting in the study of polycentrism in urban system. Graphs based methods have been used in this context[22, 23]. The graph we use here (Fig. 8) is simple and contains 1502 nodes (cities) which are geolocalized and about 24K edges weighted by the amount of commuters traveling between the cities they connect. There is 162K commuters in this region which represent about 12% of the total labour force. The French national institute of statistics and economical studies (INSEE) provides a two level clustering of French cities using commuters' flows: cities are grouped into *metropolitan areas* which are grouped into *metropolitan regions* (see [26] for more details about this network).

We use for this example a well-known approach which is the Hierarchical clustering[14]. In order to extract dense activity regions, we compute a similarity metric on each edges taking the amount of commuters into account [26]. Then the edges valuated below a given threshold value are filtered out. Finally we consider two nodes as being part of the same cluster if they are still in the same connected component. This procedure can thus provide a hierarchical clustering because choosing multiple threshold values may lead to a multi-scale decomposition of the network.

A clustering quality measure is most of the time use to determine the best threshold value to use. However we can enforce our multilevel modularity quality to find the best hierarchical clustering by evaluating combination of different threshold values. More specifically we are looking for the best two-level clustering to match with the INSEE decomposition.

We compute the hierarchical clusterings for each ordered pair of threshold values $\{(t_1, t_2) \in [0, 1]^2, t_1 < t_2\}$ and we evaluate their quality using $\widetilde{MQ}$. We take the complete and bipartite complete graphs as reference graphs and the size of clusters as the x weight.

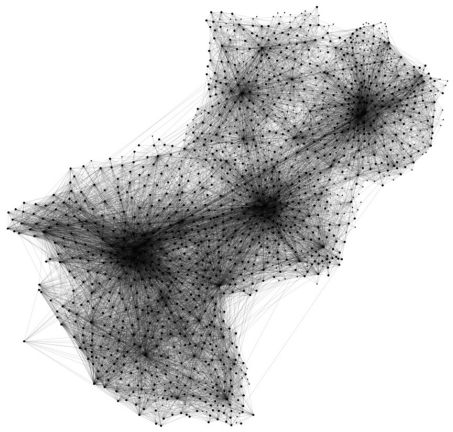
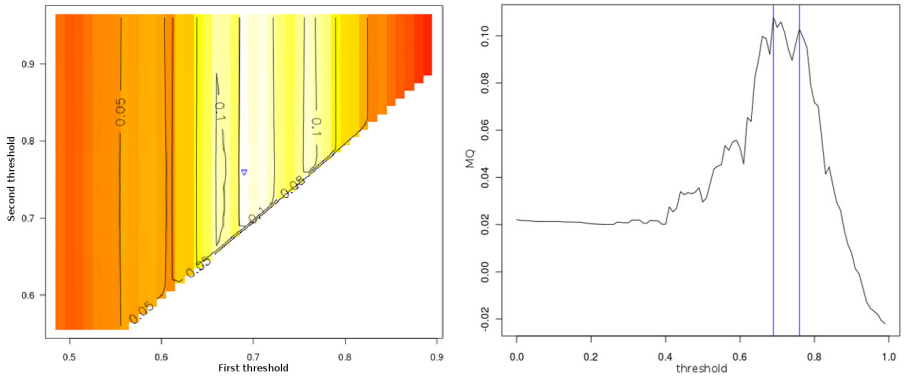


Fig. 8 Commuting network of the French region *Pays-de-la-Loire* (West of France).



(a) Matrix representation of the averaged $\widetilde{MQ}$ values. The color indicates the quality from red (lowest) to white (highest). Labeled contours are also drawn for several levels.

(b) Evolution of $\widetilde{MQ}$ according to the increase of a single threshold value.

Fig. 9 Illustration of the evolution of the quality on the commuting network. The best couple of threshold values is located using a blue triangle (left). These values are reported using blue vertical lines on the right subfigure.

The results are shown in Fig. 9. We can see that choosing two levels has not a strong impact on the quality value: in the matrix representation vertical lines have most of the time the same color. This can explained by the fact that qualities are relatively low: the maximum for a one level clustering is 0.107. The best two-level clustering have an $\widetilde{MQ}$ of 0.11 which is still better. Observe however that the matrix in Fig. 9(a) contains multiple areas having relatively strong $\widetilde{MQ}$ values (above 0.1). They may correspond to potential candidates for alternative hierarchical clusterings. It is also interesting to note that the chosen pair of threshold values also corresponds to local maxima in the evolution of $\widetilde{MQ}$ according to a single threshold (see Fig. 9(b)).

The Fig. 10 shows the two-level clustering we get for this network. We can see that bottom clusters correspond to the commuter belt of big cities. The top

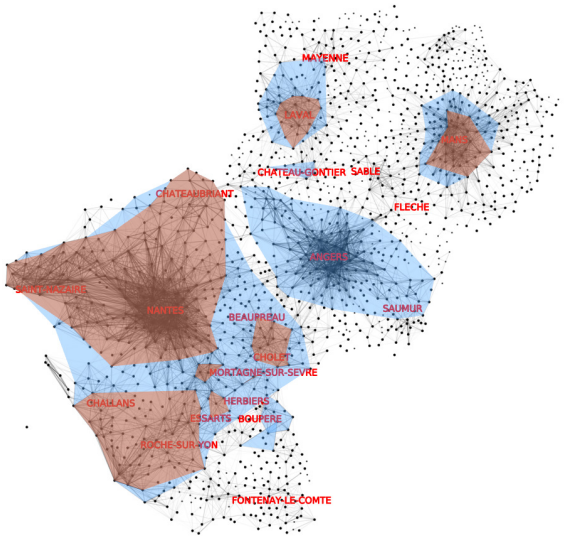


Fig. 10 Hierarchical clustering of the commuting network using the threshold values chosen in Fig. 9(a). Only the groups of cities which gather more than five thousand worker are kept. The clusters are drawn using nested concave hulls. The blue hulls correspond to the first level clustering while the brown one correspond to the second level clustering. Labels indicates the name of several biggest cities in this region.

level clustering merges some close urban cores into larger groups. Our experience working with geographers validates this two level clustering as being relevant, and in a sense as being better than the flat clustering one could consider either by taking only lowest level and smaller clusters, or larger top level clusters. Indeed road and rail infrastructure is very developed in this region: this situation can explained the presence of these large groups.

6 Conclusion and future work

We introduced a multilevel modularity in order to assess the relevancy of a hierarchical clustering of a graph. The measure we defined explicitly takes the hierarchical structure into account and computes a polynomial expression whose degree reflects the depth of the hierarchy. This multilevel modularity naturally extends a clustering quality measure that was previously defined and used to cluster graphs [18]. Coefficients of the polynomial associated with a hierarchy can alternatively be described and computed in terms of weighted paths in a tree representing this hierarchy.

Archetypal case studies provide arguments to validate the concept of a multilevel modularity. Simple case studies can be used to reveal how the measure is influenced by connectivity ratios acting at different levels in the hierarchy. Limited cases reveal the relative sensibility of the measure and compares it to traditional plain clustering modularity.

Other modularity measures could allow multilevel extensions by using a depth-based variable q to keep track of how edges interact with the hierarchy. Because of

their combinatorial properties, Newman's modularity [20], the average Normalized Cut [28] or edge density criterion (see [5], for instance) are potential candidates we plan to look at.

We also present several procedures to extract a hierarchical clustering in real world networks. There might however be more complex computing patterns to follow in order to optimize the $\widetilde{MQ}$ value. In this context, the variable q can be tuned to promote or to restrain deeper hierarchical clustering. These are obvious issues we need to address.

References

1. Amiel, M., Melançon, G., Rozenblat, C.: Réseaux multi-niveaux : l'exemple des échanges aériens mondiaux. M@ppemonde **79**(3-2005) (2005)
2. Auber, D.: Tulip - a huge graph visualization framework. In: P. Mutzel, M. Jnger (eds.) Graph Drawing Software, Mathematics and Visualization Series. Springer Verlag (2003)
3. Auber, D., Chiricota, Y., Jourdan, F., Melançon, G.: Multiscale navigation of small world networks. In: IEEE Symposium on Information Visualisation, pp. 75–81. IEEE Computer Society (2003)
4. Blondel, V., Guillaume, J., Lambiotte, R., Lefebvre, E.: Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment **2008**, P10,008 (2008)
5. Boullé, M.: Data grid models for preparation and modeling in supervised learning. In: I. Guyon, G. Cawley, G. Dror, A. Saffari (eds.) Hand on pattern recognition. Microtome (2010)
6. Boutin, F., Hascoët, M.: Cluster Validity Indices for Graph Partitioning. In: IV'04: 8th IEEE International Conference on Information Visualization, pp. 376–381. IEEE, London (UK) (2004). URL http://hal-lirmm.ccsd.cnrs.fr/lirmm-00108948/en/
7. Brandes, U., Gaertler, M., Wagner, D.: Engineering graph clustering: Models and experimental evaluation. Journal of Experimental Algorithmics **12**, (article no. 1.1) (2007)
8. Chakrabarti, D., Faloutsos, C.: Graph mining: Laws, generators, and algorithms. ACM Computing Surveys **38**(1), 2 (2006)
9. Cook, D.J., Holder, L.B. (eds.): Mining Graph Data. Wiley (2006)
10. Delest, M., Fédou, J., Melançon, G.: A quality measure for multi-level community structure. In: Symbolic and Numeric Algorithms for Scientific Computing, 2006. SYNASC'06. Eighth International Symposium on, pp. 63–68. IEEE (2007)
11. Delest, M., Munzner, T., Auber, D., Domenger, J.P.: Exploring InfoVis Publication History with Tulip (2nd place - InfoVis Contest). In: IEEE Symposium on Information Visualization, p. 110. IEEE Computer Society (2004)
12. Delest, M.P., Fédou, J.M.: Attribute grammars are useful for combinatorics. Theoretical Computer Science **98**(1), 65–76 (1992)
13. Erdös, P., Rényi, A.: On random graphs I. Publ. Math. Debrecen **6**, 290–297 (1959)
14. Fortunato, S.: Community detection in graphs. Physics Reports **486**(3-5), 75–174 (2010)
15. Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. Proceedings of the National Academy Science USA **99**, 7821–7826 (2002)
16. Good, B., De Montjoye, Y., Clauset, A.: Performance of modularity maximization in practical contexts. Physical Review E **81**(4), 46,106 (2010)
17. Jonyer, I., Cook, D., Holder, L.: Graph-based hierarchical conceptual clustering. Journal of Machine Learning Research **2**, 19–43 (2002)
18. Mancoridis, S., Mitchell, B.S., Rorres, C., Chen, Y., Gansner, E.: Using automatic clustering to produce high-level system organizations of source code. In: IEEE International Workshop on Program Understanding (IWPC'98) (1998)
19. Mishna, M.: Attribute grammars and automatic complexity analysis. Advances in Applied Mathematics **30**(1-2), 189–207 (2003)
20. Newman, M.E.J.: Fast algorithm for detecting community structure in networks. Physics Reviews E **69**, 066,133 (2004)
21. Newman, M.E.J., Girvan, M.: Finding and evaluating community structure in networks. Physics Reviews E **69**(026113) (2004)

22. Patuelli, R., Reggiani, A., Gorman, S., Nijkamp, P., Bade, F.: Network analysis of commuting flows: A comparative static approach to German data. Networks and Spatial Economics **7**(4), 315–331 (2007)
23. Pflieger, G., Rozenblat, C.: Discovery and evaluation of graph-based hierarchical conceptual clusters. Urban Studies (Special Issue: Urban Networks and Network Theory) **47**(13), 2723–2735 (2010)
24. Pons, P., Latapy, M.: Post-processing hierarchical community structures: Quality improvements and multi-scale view. Theoretical Computer Science **412**(8-10), 892 – 900 (2011)
25. Pumain, D. (ed.): Hierarchy in Natural and Social Sciences, *Methodos Series*, vol. 3. Springer (2006)
26. Queyroi, F., Chiricota, Y.: Visualization-based communities discovering in commuting networks : a case study. Tech. rep. (2011). URL http://hal.archives-ouvertes.fr/hal-00593734/PDF/queyroi_cga.pdf
27. Rouwendal, J., Nijkamp, P.: Living in Two Worlds: A Review of Home-to-Work Decisions. Growth and Change **35**(3), 287–303 (2004)
28. Satuluri, V., Parthasarathy, S.: Scalable graph clustering using stochastic flows: applications to community discovery. In: KDD, pp. 737–746 (2009)
29. Schaeffer, S.E.: Graph clustering. Computer Science Review **1**, 27–64 (2007)
30. Sozio, M., Gionis, A.: The community-search problem and how to plan a successful cocktail party. In: KDD, pp. 939–948 (2010)
31. Vespignani, A.: Evolution thinks modular. Nature **35**(2), 118–119 (2003)