Assessing the Quality of Multilevel Graph Clustering

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

Assessing the Quality of Multilevel Graph Clustering

Maylis Delest, Guy Melançon, François Queyroi
Université de Bordeaux, CNRS, LaBRI & INRIA
Bordeaux – Sud-Ouest, France
firstName.lastName@labri.fr

Jean-Marc Fedou
Université de Nice,CNRS UMR 6070 I3S, France
fedou@unice.fr

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.

Categories and Subject Descriptors

I.5.3 [**Pattern Recognition**]: Clustering; G.2.1 [**Discrete mathematics**]: Combinatorics

General Terms

Graph clustering, graph hierarchies, hierarchical clustering, multilevel modularity

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 [25]). 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 [15]. Other clustering quality measures have been studied and used to benchmark algorithms, such as the average Normalized Cut [24].

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

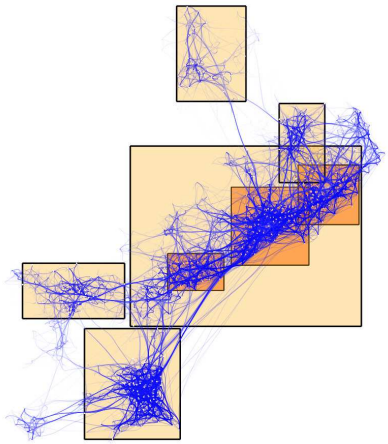
to evaluate the very hierarchical character of the clustering they produce. This is the question we address here: find a criteria evaluating the relevance of the hierarchy. Applying a modularity measure to obtain clusters $C_1, C_2, \ldots$ and then independently re-apply the measure on each cluster, 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 [16], 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.

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 [27]. The same holds for cities and city systems in geography [23]. Obviously, approaches claiming to unfold such structures in networks should rely on sound principles and methodology for hierarchical graph clustering.

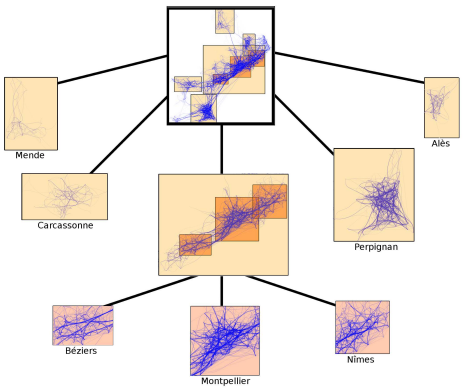
Fig. 1 gives an example of such a "natural" hierarchy. The overall image (a) shows a graph describing flows of daily commuters between towns and cities in the South-West of France. Nodes correspond to towns and cities and are positioned using their geospatial coordinates, while edges are drawn using edge bundling [17]. This type of graphs is of interest to study polycentrism in urban systems [22]. The application of a hierarchical clustering algorithm enables the identification of denser activity regions and leads to a multilevel representation of the overall network as shown in (b). Our experience working with geographers validates this three 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[1].

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

[1]The data used here is unfortunately not publicly available.



(a) Network of daily commuters in the South-West of France. The underlying colored boxes suggest the presence of a hierarchy in the data.



(b) A hierarchy has been identified and allows to recursively decompose the original data into a hierarchy of nested subgraphs.

Figure 1: The network of daily commuters (a) has been hierarchically decomposed using an iterative approach (b).

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 in [10] but did not lead to any substantial results.

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 a real world example that has been the focus of previous work, to assess of the relevance of our multilevel modularity. Some concluding remarks point at potential directions for future work.

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. 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 [26]. 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]:

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

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, and accordingly set $\delta_i = \binom{|C_i|}{2}$ and $\delta_{ij} = |C_i||C_j|$. We then get:

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

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).

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.

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 $V(t) = \bigcup_{t' \in \sigma(t)} V(t')$ 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 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 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 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 $\mathbf{T}$.

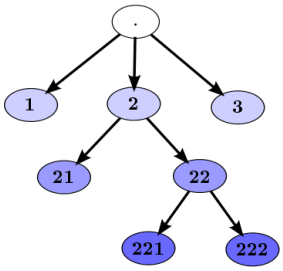


Figure 2: A labeled tree encoding a hierarchical clustering. All paths from the root to a cluster C_w are described using *words*.

Property 1. We have:

$$\widetilde{MQ}(G; \mathbf{T}; q) = \frac{1}{X} \sum_{w \in \mathcal{L}_{\mathbf{T}}} x_w \sum_{v \prec w} q^{|v|-1} \left(\prod_{u \prec v} \alpha_u - \beta_u \right) \quad (7)$$

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. The Prop. 1 thus provides a natural interpretation for the coefficient of q^p, we denote $[\widetilde{MQ}(G, \mathbf{T}, q); q^p]$. Indeed, let C be a cluster in $\mathbf{T}$ with depth $p+1$. We need to multiply differences between inner and outer connectivity ratios for each cluster sitting on the path to C. The coefficient $[\widetilde{MQ}(G, \mathbf{T}, q); q^p]$

is then obtained by summing this quantity over all clusters at depth $p+1$, as given in Prop. 2.

Property 2. 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:

$$[\widetilde{MQ}(G,\mathbf{T},q);q^p] = \frac{1}{X}\sum_{w\in\mathcal{D}_p} x_w \prod_{u\prec w}(\alpha_u - \beta_u) \tag{8}$$

Eq. (8) 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, then as edges in $G(C_i)$ 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 (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. (8):

$$\begin{aligned}\widetilde{MQ}(G;\mathbf{T}) &= \int_0^1 \widetilde{MQ}(G;\mathbf{T};q)dq \\ &= \frac{1}{X}\sum_{p=0}^{h(\mathbf{T})-1}\frac{1}{p+1}\sum_{v\in\mathcal{D}_p} x_v \prod_{u\prec v}(\alpha_u-\beta_u)\end{aligned} \tag{9}$$

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 ($x_i = |C_i|$).

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. 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 subclusters $[C_1,C_2]$.

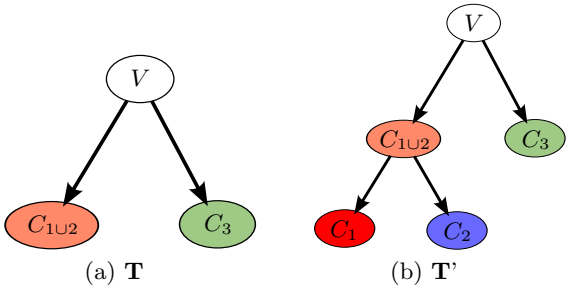


Figure 3: Two different hierarchical clusterings of a graph built from three cliques.

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 $[C_1,C_2]$. The tree $\mathbf{T}$ 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:

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

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 $C_{1\cup 2}$ into two sub-clusters, while they should be kept as a single cluster when their inter-connectivity ratio approaches higher values.

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):

Ratios	**Case 1**	**Case 2**	**Case 3**	**Case 4**
b_{12}	0.4	0.8	0.8	0.8
b_{13}	0.4	0.0	0.0	0.5
b_{23}	0.4	0.0	0.0	0.5
b_{34}	0.4	0.0	0.8	0.0

Fig. 5 shows the curves of the four polynomials we get. Table 1 collects the averaged modularities (see Eq. (9)) for each of these hierarchical clusterings. Based on these elements, we can conclude:

- 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.
- When b_{12} and b_{34} are much greater than all others b_{ij}, then the complete tree is the best available option.
- The linear tree becomes the best candidate tree when $b_{12} \gg b_{13} \simeq b_{23} \gg$ others b_{ij}.
- Note that, for all values $q \in [0,1]$, the $\widetilde{MQ}$ curve of the flat clustering does not indicates it as a good option, even in case 1. However, its averaged modularity (see Table 1) does show it as a reasonable candidate for case 1.

5. APPLICATION ON A COLLEGE FOOTBALL NETWORK

In this section we show how multilevel modularity $\widetilde{MQ}$ can be used to compare hierarchical clusterings of real world networks. We consider an example borrowed from [14] 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 (we do not consider the independent teams here). The graph is of limited size and contains 110 vertices and 568 edges. Although games are more likely to occur within a conference, they also seem to depend on the geographical proximity of the teams' hometowns.

The College Football graph has been clustered using four different algorithms. Three of them actually produce flat clusterings and have been iterated over clusters in order to obtain multilevel clusterings. We used the Strength Clustering algorithm [3], Newman's Fast Agglomerative algorithm [20] and the MLR-MCL algorithm [24]. We also used the Louvain algorithm [4] that actually produces a hierarchical clustering. We directly used the source code provided

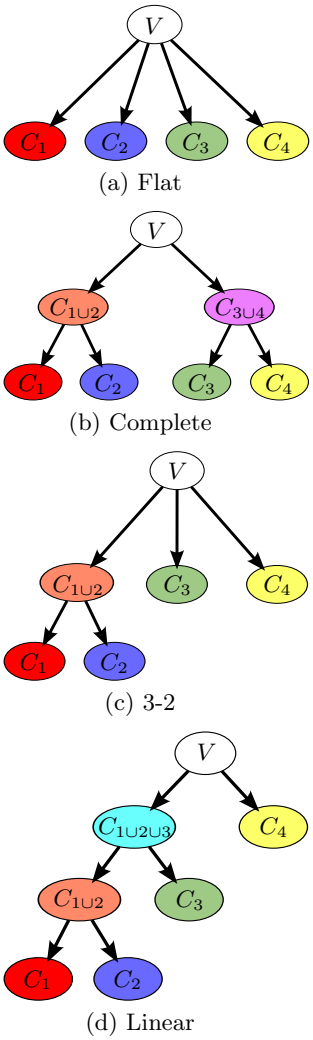


Figure 4: Different clusterings of size 4.

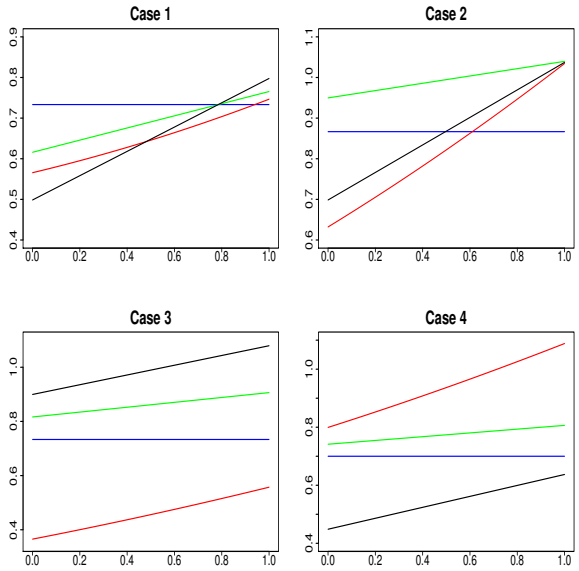


Figure 5: $\widetilde{MQ}$ curves for flat (Blue), 3-2 tree (Green), Complete (Black) and Linear (Red) cluster trees.

Tree	Case 1	Case 2	Case 3	Case 4
Flat	**0.73**	0.86	0.73	0.7
3-2	0.69	**0.99**	0.86	0.77
Complete	0.64	0.87	**0.98**	0.54
Linear	0.65	0.82	0.45	**0.92**

Table 1: Averaged modularities. The value for the best hierarchical clustering is given in bold.

by the respective authors, then ran the algorithm and visualized the results using the Graph Visualization framework Tulip [2].

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. To ease the comparison with the grouping of teams/nodes into conferences, teams/nodes are colored according to the conference they belong to. As one could expect, the five 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.

Newman's Agglomerative (Fig. 6(a)) and Louvain (Fig. 6(b)) algorithms tend to group conferences located in a same region. In both cases, the Louisiana part of the Sun Belt conference is merged with the South Eastern conference. The Strength (Fig. 6(c)) and MLR-MCL (Fig. 6(d)) algorithms produce hierarchies that are close to the division into conferences. Both algorithms refine some of the conferences into denser sub-clusters which makes sense geographically, although they disagree on the Sun Belt conference. They agree on splitting the Mid-American conference into two cliques, one gathering teams closer to Ohio and the other gathering teams closer to Michigan. These five hierarchies can be compared using $\widetilde{MQ}$. Fig. 7 and Table 2 report the resulting polynomials and averaged modularities.

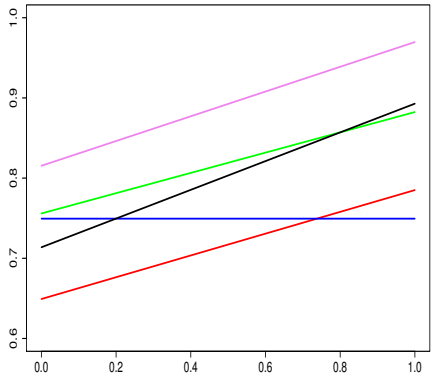
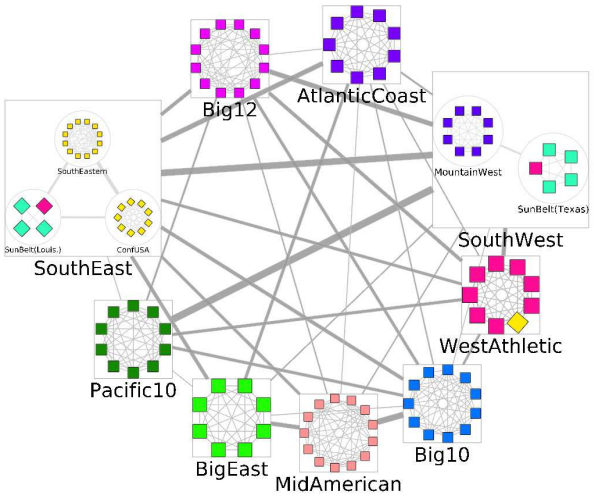


Figure 7: $\widetilde{MQ}$ curves for Conferences partition (Blue), Agglomerative clustering (Red), Strength clustering (Green), MLR-MCL clustering (Violet) and Louvain clustering (Black).

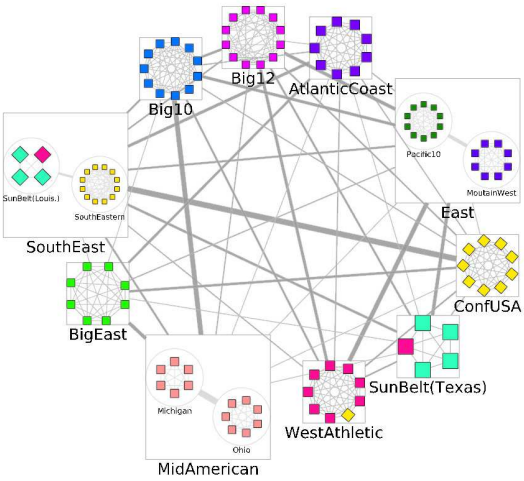
Algorithm	$\widetilde{MQ}(G, \mathbf{T}, q)$	$\widetilde{MQ}(G, \mathbf{T})$
MLR-MCL	$0.815374 + 0.154301q$	$0,8925245$
Strength	$0.755885 + 0.126473q$	$0,8191215$
Louvain	$0.713722 + 0.179126q$	$0,803285$
Divisions	0.749428	0.749428
Agglomerative	$0.649202 + 0.13588q$	$0,717142$

Table 2: Polynomials and averaged modularities for the five clusterings of the Football network.

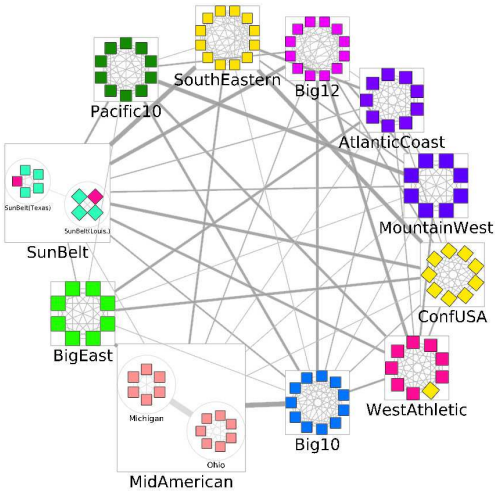
The modularity $\widetilde{MQ}$ ranks the hierarchical clustering produced by MLR-MCL as best. The algorithm is indeed able to split teams within a conference into very dense subgroups. The first level of the hierarchies obtained from the Strength algorithm recovers the organization into conferences. As a matter of fact, this algorithm behaves similarly to MLR-MCL, which have been both ranked as the best available options. It confirms the relevancy of a two level subdivision of conferences into smaller geographical regions. The lowest level clusters of the Louvain hierarchies match the division into conferences. The higher value of the slope for its polynomial compensates the loss in quality in the first level. This is however not the case with the Agglomerative algorithm due to a lower quality grouping of teams on the first level.



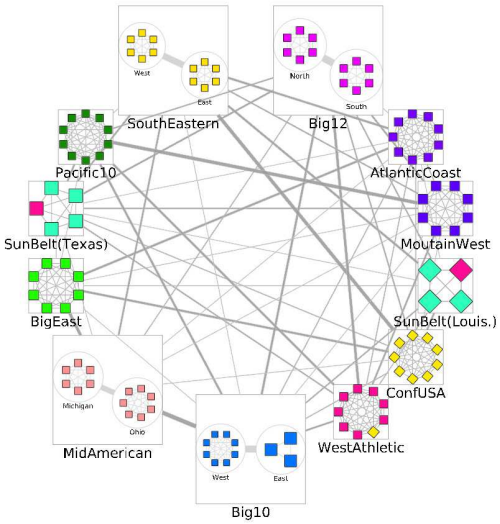
(a) Agglomerative



(b) Louvain



(c) Strength



(d) MLR-MCL

Figure 6: Nested graph representations of the College Football conferences during the season fall 2000 using several clustering algorithms. Meta-edges width represents the number of games between two 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 [24] or edge density criterion (see [5], for instance) are potential candidates we plan to look at.

For now, the multilevel modularity can be used to decide whether to iterate a clustering algorithm and further divide already computed clusters; or it can be used to guide an agglomerative process. 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 favor or to restrain deeper hierarchical clustering. These are obvious issues we need to address.

7. REFERENCES

[1] M. Amiel, G. Melançon, and C. Rozenblat. Réseaux multi-niveaux : l'exemple des échanges aériens mondiaux. *M@ppemonde*, 79(3-2005), 2005.

[2] D. Auber. Tulip - a huge graph visualization framework. In P. Mutzel and M. Jnger, editors, *Graph Drawing Software*, Mathematics and Visualization Series. Springer Verlag, 2003.

[3] D. Auber, Y. Chiricota, F. Jourdan, and G. Melançon. Multiscale navigation of small world networks. In *IEEE Symposium on Information Visualisation*, pages 75–81. IEEE Computer Society, 2003.

[4] V. Blondel, J. Guillaume, R. Lambiotte, and E. Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008:P10008, 2008.

[5] M. Boullé. Data grid models for preparation and modeling in supervised learning. In I. Guyon, G. Cawley, G. Dror, and A. Saffari, editors, *Hand on pattern recognition*. Microtome, 2010.

[6] F. Boutin and M. Hascoët. Cluster Validity Indices for Graph Partitioning. In *IV'04: 8th IEEE International Conference on Information Visualization*, pages 376–381, London (UK), 2004. IEEE.

[7] U. Brandes, M. Gaertler, and D. Wagner. Engineering graph clustering: Models and experimental evaluation. *Journal of Experimental Algorithmics*, 12:(article no. 1.1), 2007.

[8] D. Chakrabarti and C. Faloutsos. Graph mining: Laws, generators, and algorithms. *ACM Computing Surveys*, 38(1):2, 2006.

[9] D. J. Cook and L. B. Holder, editors. *Mining Graph Data*. Wiley, 2006.

[10] M. Delest, J. Fédou, and G. Melançon. A quality measure for multi-level community structure. In *Symbolic and Numeric Algorithms for Scientific Computing, 2006. SYNASC'06. Eighth International Symposium on*, pages 63–68. IEEE, 2007.

[11] M. Delest, T. Munzner, D. Auber, and J.-P. Domenger. Exploring InfoVis Publication History with Tulip (2nd place - InfoVis Contest). In *IEEE Symposium on Information Visualization*, page 110. IEEE Computer Society, 2004.

[12] M. P. Delest and J. M. Fédou. Attribute grammars are useful for combinatorics. *Theoretical Computer Science*, 98(1):65–76, 1992.

[13] P. Erdös and A. Rényi. On random graphs I. *Publ. Math. Debrecen*, 6:290–297, 1959.

[14] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy Science USA*, 99:7821–7826, 2002.

[15] B. Good, Y. De Montjoye, and A. Clauset. Performance of modularity maximization in practical contexts. *Physical Review E*, 81(4):46106, 2010.

[16] I. Jonyer, D. Cook, and L. Holder. . *Journal of Machine Learning Research*, 2:19–43, 2001.

[17] A. Lambert, R. Bourqui, and D. Auber. Winding Roads: Routing edges into bundles. *Computer Graphics Forum*, 29(3):853–862, 06 2010.

[18] S. Mancoridis, B. S. Mitchell, C. Rorres, Y. Chen, and E. Gansner. Using automatic clustering to produce high-level system organizations of source code. In *IEEE International Workshop on Program Understanding (IWPC'98)*, 1998.

[19] M. Mishna. Attribute grammars and automatic complexity analysis. *Advances in Applied Mathematics*, 30(1-2):189–207, 2003.

[20] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Physics Reviews E*, 69:066133, 2004.

[21] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physics Reviews E*, 69(026113), 2004.

[22] G. Pflieger and C. Rozenblat. Discovery and evaluation of graph-based hierarchical conceptual clusters. *Urban Studies (Special Issue: Urban Networks and Network Theory)*, 47(13):2723–2735, 2010.

[23] D. Pumain, editor. *Hierarchy in Natural and Social Sciences*, volume 3 of *Methodos Series*. Springer, 2006.

[24] V. Satuluri and S. Parthasarathy. Scalable graph clustering using stochastic flows: applications to community discovery. In *KDD*, pages 737–746, 2009.

[25] S. E. Schaeffer. Graph clustering. *Computer Science Review*, 1:27–64, 2007.

[26] M. Sozio and A. Gionis. The community-search problem and how to plan a successful cocktail party. In *KDD*, pages 939–948, 2010.

[27] A. Vespignani. Evolution thinks modular. *Nature*, 35(2):118–119, 2003.