2014 IEEE 26th International Conference on Tools with Artificial Intelligence

Community Detection in Multidimensional Networks

Alessia Amelio
National Research Council of Italy (CNR)
Inst. for High Perf. Comp. and Net. (ICAR)
Via P. Bucci 41C, 87036 Rende(CS), Italy
Email: amelio@icar.cnr.it

Clara Pizzuti
National Research Council of Italy (CNR)
Inst. for High Perf. Comp. and Net. (ICAR)
Via P. Bucci 41C, 87036 Rende(CS), Italy
Email: pizzuti@icar.cnr.it

***Abstract*—The paper proposes a new approach to detect shared community structure in multidimensional networks based on the combination of multiobjective genetic algorithms, local search, and the concept of temporal smoothness, coming from evolutionary clustering. A multidimensional network is clustered by running on each slice a multiobjective genetic algorithm that maximizes the modularity on such a slice and, at the same time, minimizes the difference between the community structure obtained for the current layer and that found on the already considered dimensions. Experiments on synthetic and real-world datasets show the ability of the approach in discovering latent shared clustering of objects.**

***Keywords*-multidimensional networks; social networks; community detection; evolutionary computation; multiobjective genetic algorithm;**

I. Introduction

In the last decades a lot of attention has been devoted to characterize and study the dynamics of complex systems. Many real-world systems can be modeled as networks whose nodes are the objects constituting the system, and the links represent the connections among them. One of the most important research activities in complex networks is to understand the relationship between network structure and emerging individual and/or collective behavior. Research in this field mainly focused on studying networks having a single type of link. Real-life networks, however, are intrinsically multidimensional since objects are connected by different relationships. Thus, the approach adopted till now of aggregating all links roughly mirrors reality because important information on the system structure and organization could be discarded. Individuals, in fact, often participate in different activities with different strength and by playing diverse roles. A more general formalism that avoids the loss of relevant information, since links related to each aspect are considered important, is constituted by *multidimensional networks* [1], [1], [2], also called *multilayer* [3], *multiplex* [4], [5], [6], *multirelational* [7], [8]. A multidimensional network consists of a set of dimensions, where each dimension network represents an aspect of the individual activity, i.e. the connections among individuals in that dimension.

In the last years there has been an increasing interest in complex networks presenting multiple connections between pairs of individuals. In a multidimensional network grouping actors by considering only one type of interaction may lead to misleading community structure because of the insufficient information used. The exploitation, instead, of the knowledge coming from all the actor activities can help in finding accurate latent group organization among individuals.

The objective in a multidimensional network is to uncover a shared community structure among objects such that a quality function is optimized in all the dimensions.

In this paper a new method, named *MultiMOGA* (*Multi-dimensional MultiOjective Genetic Algorithm*), to deal with the problem of detecting a shared community structure in multidimensional networks is proposed. The approach adapts the idea of temporal smoothness introduced by Chakrabarti et al. [9] underlying evolutionary clustering, and combines it with multiobjective Genetic Algorithms [10], [11] and local search. The new concepts of *facet quality* $\mathcal{FQ}$ and *dimensional sharing* $\mathcal{SC}$ are introduced and used as competing objectives to optimize. Facet quality means that the clustering $\mathcal{CS}_i$ computed for the i-th dimension under consideration should be as accurate as possible. Thus it guarantees that $\mathcal{CS}_i$ maximizes the quality function at the best. Dimensional sharing allows to obtain a clustering $\mathcal{CS}_i$ that does not differ too much from the clustering $\mathcal{CS}_{i-1}$ obtained so far on the i-1 already considered dimensions. The multiobjective genetic algorithm *MultiMOGA* iteratively optimizes both facet quality and sharing cost. The community structure $\mathcal{CS}_i$ obtained for the i-th dimension is considered the best sharing community structure among itself and all the $i-1$ already examined dimensions.

As pointed out, slice networks can be rather different because many interactions could be present in a slice network, but be missing in another one. As a consequence, isolated nodes in a network will not be assigned to any community. In order to dampen this phenomenon, we perform a local label propagation for each isolated node v by considering the neighbors of v in all the dimensions, and then assigning to v the most recurring class label of its neighbors in the current community structure $\mathcal{CS}_i$.

Experiments on synthetic and real-world networks show that the combination of multiobjective optimization and local label propagation, along with the dimensional sharing con-

1082-3409/14 $31.00 © 2014 IEEE
DOI 10.1109/ICTAI.2014.60

cept, allow the detection of accurate community structures in multidimensional networks.

The paper is organized as follows. The next section formalizes the problem of community detection in multidimensional networks. Section III gives a brief overview of the most recent proposals in this context. Section IV formalizes the problem as a multiobjective optimization problem. Section V describes the proposed approach. In section VI the results of the experiments are reported. Finally, section VII concludes the paper and suggests future developments.

II. Problem definition

Let $V = \{1, \ldots, n\}$ be a set of individuals or objects, and $\{1, \ldots, d\}$ be a finite set of dimensions. A network $\mathcal{N}_l$ in the l-th dimension can be modeled as a graph $G_l = (V_l, E_l)$ where $V_l \subseteq V$ is a set of nodes or vertices, and E_l is the set of links that connect elements of V_l in the l-th dimension. A multidimensional network is a sequence $\mathcal{N} = \{\mathcal{N}_1, \ldots, \mathcal{N}_d\}$ of *slice networks*. $\mathcal{N}$ can thus be represented as a sequence $\mathcal{G} = \{G_1, \ldots, G_d\}$ of graphs, where each $G_l = (V_l, E_l)$, for $l = 1, \ldots, d$, is the graph modeling network $\mathcal{N}_l$ in the l-th dimension. A dimension thus represents one of the d slices, also called *facets, layers*, of the network.

Figure 1 shows a toy network $\mathcal{N} = \{\mathcal{N}_1, \mathcal{N}_2, \mathcal{N}_3\}$ of six nodes and three different types of interactions. Each dimension is depicted in Figure 1(b),(c),(d) respectively, while the corresponding adjacency matrices are reported in Figure 1(e). Note that in $\mathcal{N}_2$ nodes 4 and 6 are isolated, and in $\mathcal{N}_3$ node 2 has no connections.

A community in a network $\mathcal{N}_l$ is a group of vertices $V_{l_i} \subseteq V_l$. Let C_i^l denote the sub-graph corresponding to a community in $\mathcal{N}_l$.

A clustering, or community structure, $\mathcal{CS}_l = \{C_1^l, \ldots C_k^l\}$ of a network $\mathcal{N}_l$ is a partitioning of G_l in groups of nodes that maximizes a quality function Q. Furthermore, for each couple of communities C_i^l and $C_j^l \in \mathcal{CS}_l$, $V_{l_i} \cap V_{l_j} = \emptyset$.

Our objective is to *uncover a shared community structure $\mathcal{CS}$ among the objects of the multidimensional network $\mathcal{N}$ such that the quality function Q is optimized in all the d dimensions*.

A shared community structure for the toy example is clearly given by $\mathcal{CS} = \{C_1, C_2\}$, where $C_1 = \{1, 2, 3\}$ and $C_2 = \{4, 5, 6\}$. In the next section we briefly review the most recent approaches to multidimensional community detection.

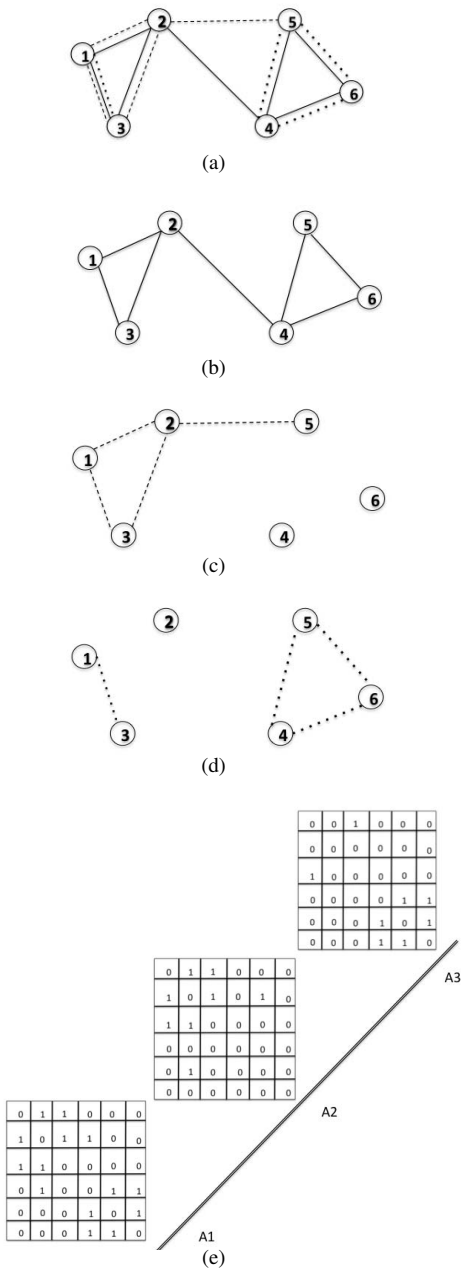


Figure 1. Example of 3-Dimensional network.

III. Related Work

Though real-world networks are often multidimensional, research mainly focused on one dimensional networks. Many approaches to detect communities in single link networks have been proposed, and several overviews have been published [12], [13], [14]. In the last years, however, the interest in complex networks presenting multiple connections between pairs of individuals is increasing, mainly due to the rapid growth of online social networking. In fact, people connect and interact each other by using a variety of social media, and perform different activities that generate multiple relations, and consequently, multidimensional networks. In the following some of the most up-to-date approaches are described.

The most recent proposals to find groups in multidimensional networks can be found in [1], [15]. In particular, in [1] Tang et al. observe that there are two main approaches to

deal with multidimensional networks. The former is a naive strategy that considers a multidimensional network as one dimensional, by using the average interaction network among nodes. This is obtained by computing the average adjacency matrix $\overline{A} = \frac{1}{d}\sum_{i=1}^{d} A_i$, as the average sum of all the d adjacency matrices $A_1, \ldots, A_d$, for each dimension. Then $\overline{A}$ can be used by applying any known community detection algorithm. In particular, the authors apply a spectral method [16], [17], named *Average Modularity Maximization (AMM)* to optimize the concept of modularity of Girvan and Newman [18] (see next section for a formal definition of modularity). The other approach, named *Total Modularity Maximization (TMM)*, consists in optimizing the score function for all the dimensions. Since Tang et al. use modularity, they propose to optimize the total modularity $\overline{Q} = \frac{1}{d}\sum_{i=1}^{d} Q_i$. Besides these two approaches, they propose a new method, named *Principal Modularity Maximization (PMM)* that consists of two main steps. First, for each dimension, the so called *structural features*, corresponding to the top eigenvectors with positive eigenvalues, are extracted, then these features are combined to obtain latent communities. Tang et al. extended their approach in [15] by analyzing four different strategies to integrate structural features. One of the main drawbacks of the proposal is that the number of communities must be given as input parameter.

Li et al. [2], instead of finding a network partitioning, deal with the problem of building a seed-based community for a multidimensional network, i.e. given a seed node, neighboring nodes are added to the seed community, provided that they are similar. Zhang et al. [19] combine friendship networks and user-generated contents to discover user communities that are densely connected and, at the same time, share common content interests. The method is based on matrix factorization and requires the number of communities to be known in advance. Comar et al. [20] proposed a framework to cluster multiple networks by jointly factorizing their adjacency matrices. In particular, the adjacency matrix corresponding to a graph is decomposed into a product of latent factors, and an iterative method is executed to minimize the total distance function between each matrix and the product of its latent factors. The joint clustering approach is presented for two graphs, however, as the authors state, it can be extended to any number of graphs. *MetaFac* is an approach proposed by Lin et al. [21] to extract community structure from multidimensional social data, represented as multiple conjunct data tensors, where each conjunction is realized through a multigraph. The approach decomposes tensors into matrices simultaneously by applying the KL-divergence as approximation cost measure. The number of decompositions, however, generally is not known in advance.

The majority of described methods relies on matrix computations for which input parameters, that determine the number of clusters, must be known a priori. In the next section a method based on multiobjective optimization, which automatically determines the optimal number of clusters, is proposed.

IV. Multiobjective optimization for multidimensional networks

The problem of finding a shared community structure in a multidimensional network can be viewed as the analogous problem in a dynamic network, i.e. a network that evolves by changing its interconnections over time, where each dimension of the multidimensional network corresponds to a time step in the dynamic network. In particular, the evolutionary clustering approach adopted for dynamic networks can be extended for multidimensional networks. The idea of evolutionary clustering proposed by Chakrabarti et al. [9], and refined by Chi et al. [22] for dynamic networks, relies on the concept of *temporal smoothness* that assumes that relevant changes in short time periods are not preferable. This is achieved by optimizing two cost functions: *snapshot quality* $\mathcal{SC}$ is maximized to obtain a clustering, at the current time step, as accurate as possible; *temporal cost* $\mathcal{TC}$ penalizes abrupt shift from one time step to the next one. The cost function:

$$cost = \alpha \cdot \mathcal{SC} + (1 - \alpha) \cdot \mathcal{TC} \tag{1}$$

has thus been defined, where α is an input parameter used to emphasize one of the two objectives. When α = 1 the approach returns the clustering without temporal smoothing. When α = 0, however, the same clustering of the previous time step is produced. A value between 0 and 1 controls the preference degree of each sub-cost. This cost function has been adopted by Lin et al. [23], Kim and Han [24], and Folino and Pizzuti [25]. In particular, in [25] the detection of community structure with temporal smoothness has been formulated as a *multiobjective optimization problem*, where the first objective is the maximization of the snapshot quality, achieved through the optimization of the *modularity* concept [18], and the second objective is the minimization of the temporal cost, fulfilled by maximizing the *Normalized Mutual Information* (NMI) [26] between the community structure obtained at the current time step with that obtained at the previous one.

We propose to modify this framework by substituting the concept of temporal smoothness with that of *dimensional sharing*, snapshot quality with *facet quality* $\mathcal{FQ}$, and temporal cost with *sharing cost* $\mathcal{SC}$. Facet quality thus guarantees that the clustering found for the i-th dimension under consideration maximizes the quality function as much as possible, while the sharing cost means that the clustering of the current facet agrees as much as possible with the clustering obtained for the previously considered i-1 dimensions. In this new framework, given the sequence $A_1, \ldots, A_d$ of adjacency matrices associated with the graphs $\{G_1, \ldots, G_d\}$

modeling a multidimensional network $\mathcal{N} = \{\mathcal{N}_1, \ldots, \mathcal{N}_d\}$, a shared community structure among the networks $\mathcal{N}_i$ can then be obtained by iteratively optimizing both facet quality and sharing cost. The community structure obtained for the last dimension d thus can be considered the best sharing community structure among the d dimensions.

A method to carry out the framework of dimensional sharing is to use a multiobjective genetic algorithm [10], [11] that finds a solution realizing the best trade-off between facet quality of network $\mathcal{N}_i$ and sharing cost with $\mathcal{N}_{i-1}$.

Analogously to [25], as *facet quality* $\mathcal{FQ}$ we employ the well known concept of *modularity* introduced by Girvan and Newman [18]. The definition of modularity Q is based on a statistical test where the null model is a uniform random graph model in which any two nodes are connected with uniform probability:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) \delta(C_i, C_j) \qquad (2)$$

A is the adjacency matrix of the graph, m is the number of edges of the graph, k_i and k_j are the degrees of nodes i and j respectively, thus $k_i k_j$ is the expected number of edges between nodes i and j in the null model. δ is the Kronecker function and yields 1 if i and j are in the same community (i.e. $C_i = C_j$), zero otherwise. Values approaching 1 indicate high quality clustering.

Regarding the second objective that must minimize the sharing cost $\mathcal{SC}$, we need to measure how similar the community structure $\mathcal{CS}_i$ of the current facet is to the clustering $\mathcal{CS}_{i-1}$ of the previous facet. To this end we adopt the *Normalized Mutual Information*. Given two partitionings $A = \{A_1, \ldots, A_a\}$ and $B = \{B_1, \ldots, B_b\}$ of a network in communities, let C be the confusion matrix whose element C_{ij} is the number of nodes of the community $A_i \in A$ that are also in the community $B_j \in B$. The normalized mutual information $NMI(A, B)$ is defined as:

$$NMI(A,B) = \frac{-2 \sum_{i=1}^{c_A} \sum_{j=1}^{c_B} C_{ij} log(C_{ij} N / C_{i.} C_{.j})}{\sum_{i=1}^{c_A} C_{i.} log(C_{i.}/N) + \sum_{j=1}^{c_B} C_{.j} log(C_{.j}/N)} \qquad (3)$$

where c_A (c_B) is the number of groups in the partitioning A (B), $C_{i.}$ ($C_{.j}$) is the sum of the elements of C in row i (column j), and N is the number of nodes. If $A = B$, $NMI(A, B) = 1$. If A and B are completely different, $NMI(A, B) = 0$. Thus our second objective at a generic dimension i is to maximize $NMI(\mathcal{CS}_i, \mathcal{CS}_{i-1})$.

It is worth to point out that Mucha et al. [4] proposed a general framework encompassing time evolving, multiscale, and multidimensional networks for determining community structure in multislice networks, i.e. those networks obtained by linking multiple adjacency matrices, where each adjacency matrix can indifferently represent variations across time (dynamic networks), variations across different types of connections (multidimensional networks), or even the same network at different scales. However our proposal is different since we formalize the problem of uncovering community structure in multidimensional networks as the analogous in dynamic networks through the framework of evolutionary clustering. In the next section a detailed description of the *MultiMOGA* method, that relies on Genetic Algorithms and multiobjective GAs, is presented.

MultiMOGA Method:
Input: A multidimensional network $\mathcal{N} = \{\mathcal{N}_1, \ldots, \mathcal{N}_d\}$ of d dimensions
the sequence of graphs $\mathcal{G} = \{G_1, \ldots, G_d\}$ modeling it
Output: A node cluster labeling that partitions $\mathcal{N}$ in the optimal shared community structure

```
1   Perform ClustOneDim on the G_1 graph
2   for each node v_i of G not appearing in G_1
3     Perform LabelAssignment
4   for i = 2 to d
5     Create a population of random individuals whose
      length equals the number n_i =| V_i | of nodes of G_i;
6     Perform a multiobjective GA with objectives
        6.1 FQ = modularity of CS_i
        6.2 SC = NMI(CS_i, CS_{i-1})
7     choose the solution CS_i = {C_1^i, ... C_{k_i}^i} of the
      Pareto front having the maximum modularity value;
8     for each node v_j of G not appearing in G_i
9       Perform LabelAssignment
10  end for
```

(a)

ClustOneDim Method:
Input: The graph G modeling a single dimension network $\mathcal{N}$
Output: The clustering $\mathcal{CS} = \{C_1, \ldots C_k\}$ of G having the best modularity value

```
1   create an initial population of random individuals
    whose length equals the number l of nodes of G
2   while not maxGen
3     decode each individual I = {g_1, ..., g_l} to obtain
      the partitioning C = {C_1, ..., C_k}
      of the graph G in k connected components.
4     evaluate the fitness of the translated individuals
5     create a new population by applying the variation operators
6   end while
```

(b)

LabelAssignment Method:
Input: the sequence of graphs $\mathcal{G} = \{G_1, \ldots, G_d\}$ modeling $\mathcal{N}$ and the current clustering $\mathcal{CS}_i = \{C_1^i, \ldots C_{k_i}^i\}$
Output: A clustering of $\mathcal{N}$ where each node has been assigned a cluster label

```
1   let V̄ = V − V_i
2   for each node v_j ∈ V̄
3     let v_{n_1}, ..., v_{n_t} be the neighbors of v_j in G
      and label(v_{n_1}), ..., label(v_{n_t}) be
      the cluster label of v_{n_i} in CS_i
4     assign to v_j argmax_label {label(v_{n_1}), ..., label(v_{n_t})}
```

(c)

Figure 2. The pseudo-code of the *MultiMOGA* algorithm.

V. THE *MultiMOGA* METHOD

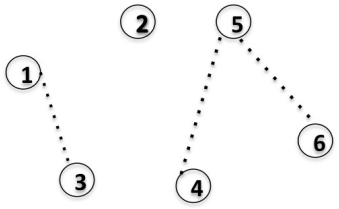
The *MultiMOGA* method is described in Figure 2, and it consists of two main steps. In the former (line 1 in Figure 2(a)) the network $\mathcal{N}_1$ is clustered by employing the algorithm *ClustOneDim* (Figure 2(b)) that uses a genetic approach to optimize the modularity value. In the latter (lines 4-10), a multiobjective genetic algorithm, for each pair of dimensions $\mathcal{N}_i$ and $\mathcal{N}_{i-1}$, tries to optimize the facet quality

$\mathcal{FQ}$ for the graph G_i modeling the current dimension $\mathcal{N}_i$, and the sharing cost $\mathcal{SC}$ computed as the normalized mutual information between the clustering obtained for G_i and that for G_{i-i}.

Both *MultiMOGA* and *ClustOneDim* use the locus-based adjacency representation [27], in which an individual of the population consists of n genes $g_1, \ldots, g_n$, where n is the number of nodes of the network, assuming values in the range $\{1, \ldots, n\}$. A value j assigned to the ith node means that there is a link between nodes i and j, and that in the clustering solution i and j will be in the same cluster. Figure 3(b) shows the locus-based representation of an individual when the network $\mathcal{N}_3$ of the toy example of Figure 1 is considered. In such a case node 1 is connected with node 3, node 3 with node 1, and so on. Note that node 2 is not connected to any other node in $\mathcal{N}_3$. This individual corresponds to the network partitioning $\{\{1,2\},\{4,5,6\}\}$, where node 2 is not assigned to any community. The initialization process of the genetic algorithm (step 1 of the *ClustOneDim* method) assigns to each node i one of its neighbors j at random. The kind of crossover operator adopted is uniform crossover. Given two parents, a random binary vector is created. Uniform crossover then selects the genes where the vector is a 0 from the first parent, and the genes where the vector is a 1 from the second parent, and combines the genes to form the child. The mutation operator, analogously to the initialization process, randomly assigns to each node i one of its neighbors (step 5 of the *ClustOneDim* method).

ClustOneDim returns a clustering $\mathcal{CS}_1 = \{\mathcal{CS}_1^1, \ldots \mathcal{CS}_{k_1}^1\}$ where each node of $G_1 = (V_1, E_1)$ is associated with a class label. However, in general, $V_1 \subseteq V$ and $E_1 \subseteq E$, because any two objects may interact in one dimension, but not in another one, thus nodes may be isolated in some dimension. For each node $v_j \in V - V_1$ we then perform a local label propagation that considers the neighbors $v_{n_1}, \ldots, v_{n_t}$ of v_j in all the dimensions and then assigns to v_j the most recurring class label of its neighbors in $\mathcal{CS}_1$, as explained in the $LabelAssignment$ algorithm (Figure 2(c)). After label assignment the multiobjective genetic algorithm is iteratively executed for the d-1 dimensions by optimizing the two objectives $\mathcal{FQ}$ and $\mathcal{SC}$. For each iteration, the clustering having the best modularity value is chosen from the Pareto front as current solution. Then again the procedure $LabelAssignment$ is run on this solution to assign labels to those nodes having no connections in the current network. The multiobjective optimization is repeated for the next dimension.

Consider the individual in Figure 3(b) relative to network $\mathcal{N}_3$ of the toy example reported in Figure 1. Node 2 has no connections with the other nodes of $\mathcal{N}_3$, thus the method does not assign it to any cluster. However, node 2 has two types of links with nodes 1 and 3, i.e. nodes 1 and 3 are its neighbors in $\mathcal{N}_1$ and $\mathcal{N}_2$. Since nodes 1 and 3 are clustered together in $\mathcal{N}_3$, the $LabelAssignment$ procedure will assign node 2 to the same cluster of its neighbors. The individual after $LabelAssignment$ is thus that reported in Figure 4.



(a)

1	2	3	4	5	6
3	0	1	5	4	5

(b)

Figure 3. Locus-based representation on an individual for network $\mathcal{N}_3$ of the toy example of Figure 1 corresponding to the division $\{\{1,2\},\{4,5,6\}\}$ and node 2 not assigned to any community.

1	2	3	4	5	6
3	1	1	5	4	5

(a)

Figure 4. Individual of Figure 3(b) after label assignment. The network division of $\mathcal{N}_3$ is now $\{\{1,2,3\},\{4,5,6\}\}$.

It is worth to note that the network ordering could influence the performances of *MultiMOGA* because slices are considered sequentially, thus processing first one network instead of another could produce different results. Choosing the best ordering is not an easy task and deserves a deep investigation which is beyond the scope of this paper. Instead of choosing a random order, we employed a heuristic based on the the concept of clustering coefficient of a network. The clustering coefficient has been defined by [28]. Given a node i, let n_i be the number of links connecting the k_i neighbors of i to each other. The clustering coefficient of i is $C_i = 2n_i/k_i(k_i - 1)$. n_i represents the number of triangles passing through i, and $k_i(k_i - 1)/2$ the number of possible triangles that could pass through node i. The clustering coefficient of a graph is the average of the clustering coefficients of the nodes it contains. A high clustering coefficient means that nodes tend to be more connected, thus they are biased to form groups.

In the next section we show that the combination of multiobjective optimization, label propagation, and clustering coefficient based ordering gives good performance results, also compared with the Tang et al. methods [1] on both synthetic networks and a real-world network.

VI. Experimental Results

In this section we present an extensive experimentation to evaluate the algorithm on synthetic networks. Then we consider a real-world network generated from the video sharing web site YouTube. Both types of networks have

Table I
COMPARING THE NMI VALUES RETURNED BY THE MULTIOBJECTIVE APPROACH *MultiMOGA* THAT USES ALL THE 4 DIMENSIONS, AND THE SINGLE-OBJECTIVE APPROACH *ClustOneDim* THAT USES ONLY ONE DIMENSION AT A TIME.

μ	*Strategy*		$\nu = 0.1$	$\nu = 0.3$	$\nu = 0.5$
0.5	One-Dimensional	A1	0.7241 ± 0.2140	0.6474 ± 0.1049	0.4874 ± 0.1997
		A2	0.8530 ± 0.0768	0.7227 ± 0.0879	0.4491 ± 0.1640
		A3	0.7918 ± 0.1427	0.6920 ± 0.0659	0.5525 ± 0.1295
		A4	0.8176 ± 0.0844	0.6121 ± 0.0884	0.5413 ±0.1587
	Multi-Dimensional	*MultiMOGA*	0.9368 ± 0.0118	0.7252 ± 0.0908	0.6309 ± 0.1656
0.8	One-Dimensional	A1	0.8398 ± 0.0920	0.6613 ± 0.1409	0.5959 ± 0.1085
		A2	0.8436 ± 0.0858	0.6448 ± 0.0986	0.5791 ± 0.1539
		A3	0.8421 ± 0.0977	0.6722 ± 0.0924	0.5272 ± 0.1523
		A4	0.8466 ± 0.1184	0.5917 ± 0.1174	0.5210 ± 0.1426
	Multi-Dimensional	*MultiMOGA*	0.9360 ± 0.0510	0.7188 ± 0.0844	0.6683 ± 0.1040

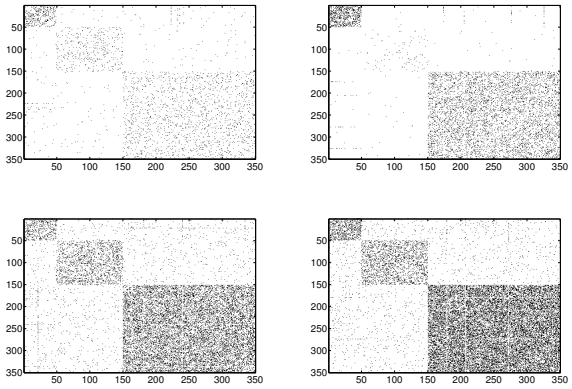


Figure 5. Example of 4-Dimensional synthetic network with $\mu = 0.5$ and $\nu = 0.1$.

been provided by Tang et al., thus a comparison with their methods is showed. As regards the *MultiMOGA* parameters, after properly tuning them, we set population size 300, number of generations 200, crossover fraction 0.9, mutation rate 0.2. For all the experiments, the statistical significance of the results produced has been checked by performing a t-test at the 5% significance level. The significance level was very high being the returned p-values very small. The implementation has been written in MATLAB 7.14 R2012a, using the Genetic Algorithms and Direct Search Toolbox 2.

A. Synthetic networks

We first consider synthetic data sets for which the ground-truth division in communities is known. We executed the method on 50 different generated synthetic networks and computed the normalized mutual information NMI between the obtained clustering and the true community structure. Note that in such a case NMI is used as evaluation measure, and it has no relation with the second objective function we adopted in *MultiMOGA*.

The synthetic data set consists of 350 objects grouped into three clusters of 50, 100, and 200 objects, respectively. The number of dimensions is 4, i.e. the objects can interact in 4 different ways. As reported by Tang et al. [1], members of the same cluster are connected with a random generated within-group probability μ. Between groups interaction probability changes for each dimension. Furthermore, controlled noise is added to the network by connecting any two nodes with probability ν. High values of μ and low values of ν generate well distinguished clusters. Figure 5 shows an example of synthetic network generated with $\mu = 0.5$ and $\nu = 0.1$. From the figure the different patterns of interactions are clearly visible for each dimension.

The first experiment we present compares the NMI values that *MultiMOGA* obtained with those returned by running a single objective genetic algorithm for community detection that uses only one dimension. The GA method we adopted to this end is the *ClustOneDim* algorithm described in the previous section, which actually finds a division of a single-dimensional network in dense groups. From Table I we can observe that *MultiMOGA* always outperforms the one-dimensional approach for two different within-group probability interaction values ($\mu = 0.5$ and $\mu = 0.8$), and for increasing noise values ($\nu = 0.1, 0.3, 0.5$), showing that the multiobjective approach is superior with respect to single dimension based method to uncover shared community structure.

Table II
COMPARING THE NMI VALUES BETWEEN THE EVOLUTIONARY COMPUTATION APPROACHES AND SPECTRAL APPROACHES OF TANG ET AL. [1].

	Strategy	*Evolutionary*	*Spectral*
1-D	A1	0.7241 ± 0.2140	0.7237 ± 0.1924
	A2	0.8530 ± 0.0768	0.6798± 0.1888
	A3	0.7918 ± 0.1427	0.6672 ± 0.1848
	A4	0.8176 ± 0.0844	0.6906± 0.1976
4-D	*MultiMOGA*	0.9368 ± 0.0118	-
	PMM	-	0.9351 ± 0.1059
	AMM	-	0.7946 ± 0.1623
	TMM	-	0.9157 ± 0.1137

Table II compares the evolutionary computation methods with the spectral-based approaches proposed by Tang et

al. [1]. The table shows the NMI values of our approach with $\mu = 0.5$ and $\nu = 0.1$, and those of Tang et al., as reported in [1]. We can observe that the results obtained by *MultiMOGA* are superior with respect to the AMM and TMM methods. Regarding PMM, the NMI values of *MultiMOGA* are slightly higher, and our approach is more robust than PMM, having a much lower standard deviation. On the single dimensional methods, the genetic approach always outperforms the spectral approach. It is worth to point out that the spectral approaches need as input parameter the number of communities to find, while the GAs methods automatically determine this value because of the genetic representation that encodes the optimal division of objects with respect to the objective function to maximize.

B. YouTube data

We now consider the real-world YouTube[1] data collection. Tang et al. crawled the data and extracted a user profile network constituted by 15,088 active users interacting in 5 different ways:

A_1: contact: contact network among users;
A_2: co-contact: two users are connected if they add the same user as contact;
A_3: co-subscription: two users subscribe to the same user;
A_4: co-subscribed: two users are subscribed by the same user;
A_5: favorite: two users share the same favorite video.

For a more detailed description of the network refer to [1], [15]). The contact dimension is the sparsest one, while the other dimensions are rather dense and contain noise. For this reason, the contact dimension has a clear network division and a higher modularity value with respect to the other dimensions. The number of communities of the YouTube network is not known, thus it cannot be evaluated with respect to a ground-truth division. To this end Tang et al. suggest a new evaluation method named *cross-dimension network validation* consisting in using d-1 dimensions as training data to obtain a community structure $\mathcal{CS}$, and the excluded i-th dimension as test data to compute the validation score. This means that $\mathcal{CS}$ is considered as the ground-truth division, and the modularity value is measured on network $\mathcal{N}_i$.

Table III
MODULARITY VALUES OBTAINED ON THE YOUTUBE NETWORK BY APPLYING THE CROSS-DIMENSION NETWORK EVALUATION.

test dim	*MultiMOGA*		PMM		
	$Modularity$		$Modularity$		
		nc	nc=20	nc=40	nc=60
A_2	**0.2969**	250	0.2063	0.1521	0.1329
A_3	**0.2012**	277	0.1307	0.0808	0.0656
A_4	**0.2776**	279	0.1844	0.1309	0.1101
A_5	0.0718	260	**0.0947**	0.0574	0.0417

[1] www.youtube.com

We adopted this evaluation method to show the performance of *MultiMOGA* on the YouTube data collection, and to compare our results with those obtained by Tang et al. in [1]. Table III shows the modularity values computed by our method and by PMM when $A_2, \ldots, A_5$ are used as test data and the other 4 dimensions as training data. For example, if A_2 is the test dimension, a clustering to evaluate modularity on A_2 is obtained by using A_1, A_3, A_4, A_5 as training dimensions. Note that A_1 is always included in the training set since it is the only network with a clear group organization, thus allowing to obtain better results on the other networks. As already observed, the PMM method needs the number of clusters to find, thus in the table we report the three different numbers of clusters fixed a priori for the PMM method, namely $nc = 20, 40, 60$. The number of clusters obtained by *MultiMOGA*, instead, is automatically determined and reported in the table. The table points out that *MultiMOGA* obtains rather higher modularity values with respect to PMM, except when the test network is A_5 with number of communities fixed to 20. Furthermore, it is worth to observe that the number of clusters returned by *MultiMOGA* is also much higher than the values fixed for PMM. Note that for PMM, the modularity values decrease as the number of clusters increases. In our case, even if the number of clusters is much higher, modularity is much better too. This experiment highlights the capability of the proposed approach in successfully dealing with the problem of community detection in multidimensional networks.

VII. CONCLUSIONS

The paper presented a multiobjective method to uncover shared community structure in multidimensional networks relying on the new concepts of facet quality and dimensional sharing. The method applies also a local label propagation strategy to compute an accurate clustering shared among all the dimensions. Experiments showed the good results obtained by the proposed approach. A heuristic that exploits the clustering coefficient of a network is also introduced to select the ordering under which networks should be examined. Choosing a best ordering, however, is not an easy task, thus future work aims at investigating the problem to automatically determine the optimal evaluation ordering of networks.

ACKNOWLEDGMENT

The authors are grateful to Lei Tang for having provided the synthetic network generator and the YouTube network.

REFERENCES

[1] L. Tang, X. Wang, and H. Liu, "Uncoverning groups via heterogeneous interaction analysis," in *The Ninth IEEE International Conference on Data Mining (ICDM'09)*, 2009, pp. 503–512.

[2] X. Li, M. Ng, and Y. Ye, "Multicomm: Finding community structure in multi-dimensional networks," *IEEE Transactions on Knowledge and Data Engineering*, vol. in press, 2013.

[3] M. Kivelä, A. Arenas, M. Barthelemy, J. P. Gleeson, Y. Moreno, and M. A. Porter, "Multilayer networks," *arXiv:1309.7233v3*, 2014.

[4] P. J. Mucha, T. Richardson, K. Macon, M. A. Porter, and J.-P. Onnela, "Community structure in time-dependent, multiscale, and multiplex networks," *Science*, vol. 328, no. 5980, pp. 876–878, 2010.

[5] M. D. Domenico, A. Sole, S. Gómez, and A. Arenas, "Random walks on multiplex networks," *arXiv:1306.0519*, 2013.

[6] F. Battiston, V. Nicosia, and V. Latora, "Metrics for the analysis of multiplex networks," *arXiv:1308.3182v2*, 2013.

[7] D. Cai, Z. Shao, X. He, X. Yan, and J. Han, "Community mining from multi-relational networks," in *9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2005)*, 2005, pp. 445–452.

[8] A. Harrer and A. Schmidt, "Blockmodelling and role analysis in multi-relational networks," *Social Netw. Analys. Mining*, vol. 3, no. 3, pp. 701–719, 2013.

[9] D. Chakrabarti, R. Kumar, and A. Tomkins, "Evolutionary clustering," in *Proc. of the 12th ACM International Conference on Knowledge Discovery and Data Mining (KDD'06)*, 2006, pp. 554–560.

[10] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Ltd, Chichester, England, 2001.

[11] C. A. C. Coello, G. B. Lamont, and D. A. V. Veldhuizen, *Evolutionary Alg. for Solving Multi-Objective Problems*. Springer, 2007.

[12] L. Danon, J. Duch, A. Arenas, and A. Díaz-Guilera, "Community structure identification," *Large Scale Structure and Dynamics of Complex Networks: From Information Technology to Finance and Natural Science,World Scientific*, pp. 93–113, 2007.

[13] S. Fortunato and C. Castellano, "Community structure in graphs," *Encyclopedia of Complexity and Systems Science-Robert A. Meyers (Ed.)Springer*, pp. 1141–1163, 2009.

[14] S. Fortunato, "Community detection in graphs," *Physics Reports*, vol. 486, pp. 75–174, 2010.

[15] L. Tang, X. Wang, and H. Liu, "Community detection via heterogeneous interaction analysis," *Data Mining and Knowledge Discovery*, vol. 25, no. 1, pp. 1–33, 2012.

[16] F. R. K. Chung, "Spectral graph theory," *CBMS Regional Conference Series in Mathematics*, vol. 92, 1997.

[17] M. E. J. Newman, "Finding community structure in networks using the eigenvectors of matrices," *Physical Review E*, vol. 74, no. 3, 2006.

[18] M. E. J. Newman and M. Girvan, "Finding and evaluating community structure in networks," *Physical Review*, vol. E69, p. 026113, 2004.

[19] Z. Zhang, Q. Li, D. Zeng, and H. Gao, "User community discovery from multi-relational networks," *Decision Support Systems*, vol. 54, no. 2, pp. 870–879, 2013.

[20] P. M. Comar, P.-N. Tan, and A. K. Jain, "A framework for joint community detection across multiple related networks," *Neurocomputing*, vol. 76, no. 1, pp. 93–104, 2012.

[21] Y.-R. Lin, J. Sun, H. Sundaram, A. Kelliher, P. Castro, and R. B. Konuru, "Community discovery via metagraph factorization," *TKDD*, vol. 5, no. 3, p. 17, 2011.

[22] Y. Chi, X. Song, D.Zhou, K.Hino, and B. Tseng, "On evolutionary spectral clustering," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 4, Article 17, 2009.

[23] Y.-R. Lin, S. Zhu, H. Sundaram, and B. L. Tseng, "Analyzing communities and their evolutions in dynamic social networks," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 2, Article 18, 2009.

[24] M. Kim and J. Han, "A particle-and-density based evolutionary clustering method for dynamic networks," in *Proc. of the International Conference on Very Large Data Bases (VLDB'09)*, 2009.

[25] F. Folino and C. Pizzuti, "An evolutionary multiobjective approach for community discovery in dynamic networks," *IEEE Transactions on Knowledge and Data Engineering, to appear*, vol. 26, no. 8, pp. 1838–1852, 2014.

[26] L. Danon, A. Díaz-Guilera, J. Duch, and A. Arenas, "Comparing community structure identification," *Journal of Statistical Mechanics*, vol. P09008, 2005.

[27] Y. Park and M. Song, "A genetic algorithm for clustering problems," in *Proc. of 3rd Annual Conf. on Genetic Algorithms*, 1989, pp. 2–9.

[28] D. J. Watt, *Small worlds*. Princeton University Press, 1999.