Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 119 (2015) 196 – 205

Procedia Engineering

www.elsevier.com/locate/procedia

13th International Conference on Computing and Control for the Water Industry, CCWI2015

Water distribution system clustering and partitioning based on social network algorithms

Armando Di Nardo[a]*, Michele Di Natale[a], Carlo Giudicianni[a], Dino Musmarra[a]
Giovanni Francesco Santonastaso[a], Antonietta Simone[a]

[a]*Department of Civil Engineering, Design, Building and Environment, Second University of Naples, via Roma 29, Aversa, 81031, Italy*

Abstract

The partitioning of water distribution system is a complex process achieved defining network clusters arranged in sectors, with the complete isolation of clusters through gate valves, or arranged in districts, inserting both gate valves and flow meters. The process is generally subdivided in two phases: clustering, aimed to define the shape and dimension of network clusters, and partitioning, aimed to select pipes in which to insert flow meters or gate valves. In this paper, different clustering procedures based on social network community detection and on graph partitioning algorithms, were compared using a real water system and a large battery of performance indices.

Peer-review under responsibility of the Scientific Committee of CCWI 2015

Keywords: Water network partitioning, clustering, social network, graph theory

1. Introduction

Water Network Partitioning (WNP) or Sectorization (WNS) are effective techniques to improve water supply system management and protection [1, 2]. The paradigm of “divide and conquer” applied to smart water network [3] allows to simplify the system maintenance, water losses detection and pressure management [3, 4] and to reduce the negative effects of accidental and intentional contamination in compliance with the *dual use* value [5].

* Corresponding author. Tel.: +39-081-5010202; fax: +39-081-5037370.
E-mail address: armando.dinardo@unina2.it

Peer-review under responsibility of the Scientific Committee of CCWI 2015
doi:10.1016/j.proeng.2015.08.876

The techniques imply the definition of clusters that can be partially isolated from the rest of the network through boundary (or gate) valves and flow meters monitoring permanently inflows and outflows. The clusters equipped with control devices can be called District Meter Area (DMA) [6].

Despite the benefits associated to WNP and WNS, some drawbacks can occur because the gate valves can reduce, also significantly, topologic and energetic redundancy, with a consequent decreasing of node pressures [2]. Further, the number of possible partitioning layouts may be considerably large, depending on the water network extension and looping configuration. In small branched network, it is possible to define districts or sectors based only on empirical suggestions or simple visual criteria [2, 4]; however, in large network the use of mathematical and computational tools becomes extremely necessary [2, 4].

Recently different procedures have been proposed in the literature (a review is given in [2] and [4]) for finding an optimal network partitioning. The proposed procedures are generally subdivided in two phases [4]: a) the clustering, aimed to define the shape and dimension of network subsets based on graph theory; b) the physical partitioning through the device positioning, that is the selection of pipes in which to insert flow meters or gate valves, aimed to define optimal districts or sectors minimizing the number of flow meters to simplify the water balance and to reduce the investment and operational costs, based on iterative procedures or genetic algorithms.

Therefore two approaches proved most effective and were implemented in informatics tools [4, 7] that allow obtaining automatically water network clustering based on graph partitioning and community structure algorithms.

The first one, based on graph partitioning [8], is a technique of Computer Science, developed in order to solve problems that need huge computational power like, for example, simulations based on finite element methods that require distribution of the finite element mesh among different processors. This distribution, in order to improve performance, must be made according to two main rules: 1) a equal number of finite elements has to be allocated to each processor for balancing the workload; 2) a minimum number of adjacent elements between processors has to be found for reducing communication overhead. This problem can be assimilated to partitioning of a computational mesh in *k*-way or in *k*-processors that will perform each computational process. The mesh is commonly schematized by a graph with vertices correspondent to individual computational processes (e.g., finite elements) and with links correspondent to their connections. Thus graph partitioning techniques [8] were developed in Computer Science for the optimal allocation of a computational mesh in parallel or distributed computing architectures. Graph partitioning is used to successfully satisfy these constraints that, in some ways, can be mapped on those of a water network partitioning problem [2, 9, 10]: workload balancing can be likened to balance the number of nodes or flow for each DMA and to minimize the pipe-cut or the number of boundary valves [2].

The second one, based on community detection [11, 12], stems from graph clustering and, specifically, from the Social Network Theory (SNT). Indeed, as in a social network, the importance of each element of water network depends on the interrelation degree with other elements. In a water system, the interrelation depends on the topologic and hydraulic features but, essentially, on the topologic and energetic redundancy that have a different influence on each network element. The SNT uses the concept of *density* and *centrality* [13,14] as measures of this influence using different metrics (or indices), mainly the *edge betweenness* [15] of a given node or pipe that measures the amount of paths that connect two given nodes and that pass through that node or pipe and *modularity* [16] between two or more possible community (a sub-set of network elements) that measures if there are many edges within each community and only few between them. Some clustering algorithms have been proposed in the literature, used essentially in SNT and other research fields [11, 12, 15], based on the maximization of the modularity of the graph community structure. Recently, some procedures have been applied in the clustering of water distribution network [4, 17, 18, 19].

In this paper, two community structure algorithms, recently proposed in the scientific literature were compared with two graph partitioning techniques, to define a water network clustering of an Italian real water system. After this first phase, the optimal positioning of gate valves and flow meters were found using a suitable Genetic Algorithm [2] involved by authors in other works.

Finally a large battery of performance indices was computed [20]. Essentially, the average path length [21] and the modularity [15], as surrogates for topologic redundancy, and node pressures and resilience [20], as a surrogate for energy redundancy, were computed.

2. Network clustering

Many of the physical and social processes can be described by complex networks or graphs. Also a water distribution network can be naturally considered as a *simple graph* $G=(G,E)$, where V is the set of n vertices (or nodes) and E is the set of m edges (or links), or as a *weighted graph*, if some vertices or edges have associated weights indicated respectively with w_i (e.g. demand, elevation, etc.), with $i=1..n$, or with ε_l (length, diameter, flow, dissipated power, etc.), with $l=1..m$.

Network (or graph) clustering consists in defining C_k clusters (or network subsets) where each node $i \in V$ belongs uniquely to one of the clusters $C_1, C_2, \ldots, C_k$ such that $C_i \cap C_j = 0$, for $i \neq j$, and $\bigcup_i C_i = V$.

In next sections the main differences between graph partitioning and community structures approaches are synthetically descripted presenting also the four algorithms compared in this paper to obtain a water network clustering.

2.1. Graph partitioning algorithms

A k-way graph partitioning problem consists in defining C_k clusters linked by e_{ij} edge-cuts that connect vertices belonging to different cluster:

$$N_{ec} = \sum_{i \in C_k \Rightarrow j \notin C_k} e_{ij} \tag{1}$$

minimizing the total number of the edge-cuts:

$$\min\left(N_{ec}\right) \tag{2}$$

If the edge-cuts have associated weights ε_l:

$$W_{\varepsilon} = \sum_{i \in C_k \Rightarrow j \notin C_k} \varepsilon_l \tag{3}$$

the goal of graph partitioning is minimizing the sum of the edge-cut weights:

$$\min\left(W_{\varepsilon}\right) \tag{4}$$

The minimization of the relations (1) and (2) has to be obtained by balancing the number of vertices n_k or the associated weights ϖ_k belong to each cluster C_k. This constraint is achieved by minimizing the balance index I_B:

$$I_b = \frac{k \times \max\left(C_k\right)}{n} \tag{5}$$

where $max(C_k)$ can be the size of the largest subset n_k or the maximum node weight ϖ_k obtained by the k-way partitioning algorithm.

Recently, graph partitioning algorithms have been applied to water network clustering problem; in this paper two procedures developed by the authors, have been used and synthetically described in next paragraphs.

2.1.1. Multi Level Recursive Bisection (MLRB)

Multi Level Recursive Bisection (MLRB) for water network clustering was proposed by [2] adapting the traditional phases of a MLRB [8]: a) coarsening; b) partitioning; c) uncoarsening (also with refinement). The coarsening phase simplifies the original graph $G_0 = (V_0,E_0)$ through a node aggregation that generates a sequence of smaller graphs $G_i = (V_i,E_i)$, each with fewer vertices such that $|V_i| < |V_{i-1}|$. Each graph G_{i+1}, obtained by the aggregation of the adjacent vertices of G_i that creates a new vertex v (also defined as *multinode*), is called a *coarser graph*. In this way, the edge that connects adjacent vertices collapses defining a, so called, *vertex matching*. Thus, the next level of a coarser graph G_{i+1} is constructed from G_i by finding a match of G_i and collapsing the matched vertices into multinodes. The next phase of an MLRB technique is to find a *k*-way partitioning by recursive bisection. First, G_i is subdivided into *2*-way partitions, and then each part is further subdivided into *2*-way partitions or bisections. Thus, a *k*-way partition can be solved by performing a sequence of *2*-way partitions. In this phase, an optimization procedure must be used to obtain a partitioning that minimizes the objective function (2) or (4) in compliance with constraint (5). Finally, the *uncoarsening* phase is achieved which, typically, consists of two steps: a) a projection from the coarser graph G_m back to the original graph G_0 by going through the graphs G_{m-1}, G_{m-2}, . . ., G_1 (*uncoarsening*) assigning the matched vertex of the previous level G_m to each multi-node *v* of G_{i+1} and b) a local optimisation of the partition (*refinement*) by moving a vertex from one partition to another and maintaining compliance with the constraint of relationship (5).

2.1.2. Multi Agent (MA)

This technique, proposed recently [22], integrates in a multilevel procedure for water network clustering: a Depth First Search (DFS) and a Multi-Agent (MA) algorithm [23]. The original graph G_0 is subdivided into *2*-way balanced partitions the $G_{01}=(V_1,E_1)$ and $G_{02}=(V_2,E_2)$, then each part is further subdivided in *2*-way balanced partitions (or bisections) satisfying the *Node Constrain* (5). This phase is carried out with a DFS algorithm [22] that starts from a node and explores as far as possible along each path (in “depth”) until there are no more adjacent unvisited nodes; only then it starts a new path. The application of the DFS algorithm makes it possible to identify a new graph structure of the network, composed of *trees* and *branches,* called a *DFS forest graph*, starting from a generic node of the graph. Then *DFS forest graph* is divided in two subgraph (bisected): $G_{01}=(V_1,E_1)$ and $G_{02}=(V_2,E_2)$ that can be considered a possible bisection of G_0 if each one is a connected graph or, in other words, if for each couple of vertices *s* and *t* there is a path that links *s* and *t*, otherwise this condition is iteratively sought moving each not connected vertex from a subset to the other one until G_1 and G_2 are connected. Then, each bisection is followed by an optimization phase, performed by multiagent algorithm that generates new optimized bisection $G_1=(V_1,E_1)$ and $G_2=(V_2,E_2)$. The goal of this phase is *minimizing* the number of the total edge-cuts N_{ec} (or the associated weights W_ε) by the ant algorithm proposed in [22, 23].

2.2. Community structure

In the last years, clustering algorithms have been proposed to understanding and analyzing complex systems and networks [12, 13, 14, 15, 16]. These algorithms are based on different criteria to divide the data set (or nodes) into clusters identifying some mathematical characteristics between elements to assign to a cluster rather than to another. Specifically, traditional global clustering algorithms used the measure of similarity [4], while, more recently, community structure algorithms use the measure of network density [11] to define clusters.

Community structure is a bottom-up hierarchical algorithm exploiting network density property as the quality measure of the clustering [24]. A network division is good in terms of density if there are many edges within communities (intra-clusters) and only a few between them (inter-clusters) based on hierarchical methods [11], edge betweenness [11] and modularity [15, 16]. The discovery and analysis of community structure in networks is a topic of considerable recent interest, but most methods proposed so far are unsuitable for very large networks because of their computational cost [15]. Various approximate optimization methods are available and, generally, for community structure considering a scheme based on a standard “greedy” optimization algorithm [15].

Some applications to the problem of the water network clustering and partitioning showed a good performance of community structure algorithms [4, 17, 18, 19]. In this paper two algorithms, based on the measure of the edge betweenness [11] and modularity [16] have been used and synthetically described in next paragraphs.

2.2.1. Edge betweenness community (EBC)

A commonly used procedure for finding network communities is based on a divisive algorithm that uses *edge betweenness* as a metric to identify the boundaries of communities [11]. The betweenness $c_B(l)$ of an edge *l* is defined to be the number of geodesic (e.g., shortest) paths between vertex pairs that run along the edge *l*, summed over all vertex pairs, as follows [25]:

$$c_B(l) = \sum_{s,t \in V} \frac{\sigma(s,t \mid l)}{\sigma(s,t)} \tag{6}$$

where V is the set of nodes, $\sigma(s, t)$ is the number of shortest *(s,t)-paths*, and $\sigma(s, t|l)$ is the number of those paths passing through edge l.

This expression allows to extend to the edges the definition of *betweenness centrality* proposed by Freeman [26] for the nodes; it is possible to define the *edge betweenness* of an edge as the number of *shortest paths* between pairs of nodes that run along it, in order to find the edges that, in a network, are most *between* other pairs of nodes. The algorithm proposed by [11] is based on the idea that instead of trying to construct a measure that provides which edges are most central to communities, it is more simple identifying the edges which are least central or, in other terms, those edges which are most *between* communities. Thus, rather than defining communities by adding the strongest edges to an initially empty vertex set, an optimal community cluster can be defined by progressively removing edges with high value of edge betweenness from the original graph [11]. So, this algorithm identifies edges in a network that lie between communities and then removes them, leaving behind just the communities themselves. The algorithm removes the edge with highest betweenness and repeating this process until no edges remain and, if two or more edges tie for highest betweenness, then one can either choose one at random to remove, or simultaneously remove all of them [11]. The entire progress of the algorithm from start to finish can, as with the hierarchical clustering method, be represented as a dendrogram. Girvan and Newman [11] have proposed an algorithm that appears to achieve natural communities structure of networks through three definitive features thus: (1) it is a divisive method, in which edges are progressively removed from a network, by contrast with the agglomerative hierarchical clustering method; (2) the edges to be removed are chosen by computing betweenness scores; (3) the betweenness scores are recomputed following the removal of each edge [11].

2.2.2. Fast greedy community (FGC)

Fast greedy community algorithm belongs to the family of community structure methods and it is based on the maximization of the metric (or index) of modularity, defined as follows:

$$I_Q = \sum_i \left(e_{ij} - a_i^2\right) \tag{7}$$

where e_{ij} is the fraction of edges in the network that connect vertices in the group (or cluster) i to those in the group j, and a_i is the fraction of ends of edges that are attached to vertices in community i [16]. The algorithm FGC, used in this paper, is a recent important improvement of the previous algorithm of Newman [15], based on the same greedy optimization that runs far more quickly by exploiting some shortcuts in the optimization problem and using more sophisticated data structures. In this way, the algorithm allows to extend community structure analysis also to very large networks [16]. The aim of the algorithm is to have a cluster layout that maximizes the modularity optimizing I_Q of all possible divisions. The authors refer that values of I_Q greater than about 0.3 is a good indicator of significant community structure in a network. The algorithm involves finding the changes in I_Q that would result from the amalgamation of each pair of communities, choosing the largest of them, and performing the corresponding amalgamation [16].

3. Network partitioning

The second phase of the proposed heuristic optimization approach consists in to define the best position of the flow meters and boundary (or gate valves) to insert in the boundary pipes (or edge-cuts) between districts previously obtained by clustering algorithms. In this way, a physical partitioning of water network is achieved.

Once obtained the set N_{ec} of the edge-cuts, it is necessary to choose how many and which of these boundary pipes have to be interrupted with N_{bv} gate valves or, equally, have to be used for installing N_{fm}=(N_{ec}-N_{bv}) flow meters. Thus, it occurs to define which pipes have to be interrupted among all the possible combinations N_c expressed by a binomial coefficient of (N_{ec} N_{fm}) [2]. Therefore, also in this case, the problem is practically unsolvable with an exhaustive search of best solution, and a heuristic optimization method based on a Genetic Algorithm (GA) is used maximizing the total node power of the network [2]:

$$\max\left(\gamma\sum_{i=1}^{n} Q_i H_i\right) \tag{8}$$

where γ is the specific weight of water, and Q_i and H_i are the water demand and head at each network node.

4. Performance indices

Two categories of Performance Indices (PI) have been involved to test different sectorization layouts using a Demand Driven Approach [27]:

a) *energy performance indices (EPI)*, measured traditionally by mean node pressure h_{MEAN}, maximum node pressure h_{MAX}, minimum node pressure h_{MIN} and standard deviation node pressure h_{SD} and, most recently, by the resilience index I_R [28] – based on the comparison between the power dissipated in the network to satisfy the total demand and the maximum power that would be dissipated internally in order to satisfy the constraints in terms of demand and head at the nodes, and by the resilience deviation index I_{RD} [29] – based on the comparison among the resilience indices of the original and partitioned network;

b) *topological performance indices (TPI),* measured by the number n_k of nodes belong to each cluster k; by the balance index, I_B; by the total number of the edge-cut, N_{EC}, and by the worst number of the edge-cut, N_{WEC}, computed as the maximum number of the boundary pipes (with flow meters or gate valves) that connect the nodes in a cluster to the rest of the network, and by the modularity index I_Q, computed by relation (7). Another topological index, named Average Path Length (APL) [20], as measure of betweenness centrality, is computed by finding the shortest path between all pairs of nodes, adding them up, and then dividing by the total number of pairs, as follows:

$$APL = \frac{\sum_{\forall s \neq t} \sigma(\mathrm{s,\ t})}{\frac{1}{2} n \cdot (n+1)} \tag{9}$$

where n is the total number of the network vertex, $\sigma(s,\ t)$ is the shortest path between two nodes s and t, computed as the number of edges or the sum of the weights of the edges respectively for unweighted/weighted network (when there is no path between a pairs of nodes, the distance is assumed infinite – thus expressing the condition of non-reachability of the isolated nodes) [20]. Practically the APL index determines the average degree of separation between any pair of nodes.

In this study, a new index defined as Average Path Length Deviation, is proposed:

$$I_{APLD} = \left(\frac{I_{APL} - I^*_{APL}}{I_{APL}}\right) \cdot 100 = \left(1 - \frac{I^*_{APL}}{I_{APL}}\right) \cdot 100 = \left(1 - \frac{\sum_{\forall s \neq t} \sigma(\mathrm{s,\ t})^*}{\sum_{\forall s \neq t} \sigma(\mathrm{s,\ t})}\right) \cdot 100 \tag{10}$$

where I_{APL} and I^*_{APL} are the average path length of the original and of the WNS layout, respectively. This index immediately indicates the average path length percentage deviation between the WNS and original water network, with higher values of I_{APLD} indicating a worse WNS.

5. Case study

The case study is Parete network [2], located in a densely populated area in the South of Caserta (Italy), with 10,800 inhabitants, with two sources and many loops and a design pressure h_i^* =25 m equal for each node i. All WNPs have been obtained with k=4 DMA inserting the same number of flow meters, N_{fm} = 6.

In the first column of the Table 1 and 2, the energy and topological indices of the original network are reported. The network has a low value of the resilience index I_R (Table 1) showing a low energy redundancy and, consequently, the insertion of gate valves to define a WNP is more difficult because the hydraulic performance can significantly worsen. With reference to topological indices (Table 2), only the average path length index can be computed for the original network I_{APL}=8.80 and it shows naturally the best value obtained that is very useful to the comparison with WNP layouts.

Table 1. Energy Performance Indices (EPI).

EPI	Network	EBC	FGC	MA	MLRB	EBC	FGC	MA	MLRB
		no weight				with weight			
h_{MIN}	21.36	21.31	20.73	21.42	21.49	21.63	19.65	21.18	21.18
h_{MEAN}	31.05	30.49	30.46	30.78	30.45	31.06	29.50	30.75	30.77
h_{MAX}	50.47	50.48	50.63	50.44	50.43	50.39	50.24	50.51	50.50
h_{SD}	5.66	5.93	6.20	5.72	5.55	5.47	5.56	5.95	5.93
I_R	0.358	0.332	0.336	0.342	0.321	0.348	0.278	0.343	0.340
I_{RD}	-	5.50	4.23	2.48	8.38	0.94	20.84	2.24	3.20

Table 2. Topological Performance Indices (TPI).

TP	Network	EBC	FGC	MA	MLRB	EBC	FGC	MA	MLRB
		no weight				with weight			
n_1	-	44	44	46	44	43	12	46	47
n_2	-	53	57	46	47	55	32	46	47
n_3	-	31	44	46	46	24	108	46	45
n_4	-	56	39	46	47	62	32	46	45
I_B	-	1.22	1.24	1.00	1.02	1.35	2.35	1.00	1.05
N_{EC}	-	14	14	18	16	13	13	23	27
N_{WEC}	-	5	4	4	4	4	4	6	5
I_Q	-	0.71	0.72	0.73	0.73	0.70	0.57	0.73	0.72
I_{APL}	8.80	9.82	9.93	9.87	9.78	9.43	10.35	10.31	11.27
I_{APLD}	-	10.36	11.32	10.77	9.07	6.61	14.95	14.61	21.91

The results reported in the Table 1 and 2 were obtained with k=4 clusters (or DMAs) with or without weight “flow” on the pipes.

All WNP layouts obtained with four algorithms without weight show a good energy and topological performance with deviation indices I_{RD} < 8.38% (corresponding to MLRB algorithm) and I_{APLD} < 11.32% (corresponding to FGC algorithm) with the best results obtained with MA algorithm in terms of I_{RD} = 2.48% and with MLRB in terms of I_{APLD} = 9.07%, respectively. Also the index of modularity I_Q is higher than 0.3 and substantially equal for each algorithm (within the range 0.71-0.73) as well as node pressure indices (i.e. within the range of h_{min} = 20.73-21.49 m) practically equal to h_{min} = 21.36 m of the original network. Without weight also the balance index I_B is good, as showed in Figure 1, in which the four DMAs are illustrated with a different colour of nodes. As predictable, the index I_B is better for graph partitioning algorithms (I_B = 1 with the same number n_k=46 for k = 1..4 obtained with MA algorithm) than community structure algorithms because the constraint of the relation (4).

Finally, for simulation results without weight, the number of edge-cut is similar for four algorithms with values from N_{EC} =14 (with EBC and FGC) up to N_{EC} =18 (with MA); also the number of worst edge cut is similar with values from N_{WEC} =4 (with FGC, MA and MLRB) up to N_{WEC} =5 (with EBC).

The best WNP layout was obtained with EBC algorithm with weight “flow” on pipes, both in terms of energy and topological deviation indices, with excellent values of I_{RD} = 0.94% and I_{APLD} = 6.61%. Good results in terms of EPI were obtained also with graph partitioning algorithms (I_{RD} = 2.24% with MA and I_{RD} = 3.20% with MLRB) although the index of average path length is significantly worse (I_{APLD} = 14.61% with MA and I_{APLD} = 21.91% with MLRB) than results obtained without weights. Also the index of modularity (I_Q = 0.70 with EBC, I_Q = 0.73 with MA and I_Q = 0.72 with MLRB) and the balance index (I_B = 1.35 with EBC, I_B = 1.00 with MA and I_B = 1.05 with MLRB) are very similar to the previous case obtained without weight on the pipe. While overall very bad results were obtained with FGC algorithm with I_{RD} = 20.84, I_Q = 0.57, I_{APLD} = 14.95 and I_B = 2.35 and h_{min} = 19.65 m.

In the Figure 2 four WNP layouts, obtained with weights on pipes, show a different shape and dimension of each DMA with a clear worsening of layout b) FGC in terms of the balance index with only 12 nodes in DMA1.

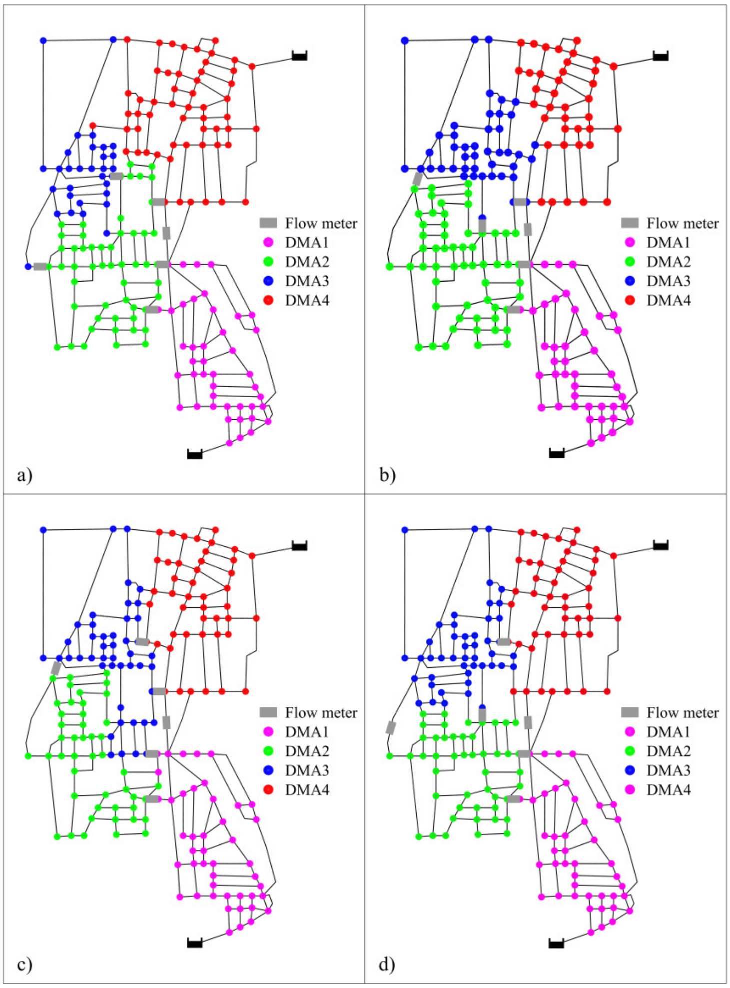

Fig. 1. WNP obtained without weights: a) EBC algorithm, b) FGC algorithm, c) MA algorithm and d) MLRB algorithm

Finally, a multi-criteria evaluation can be simplified with the use of *radar diagram*, reported in Figure 3, where the six vertices represent each performance index standardized to a unit scale. In each diagram, a comparison of four algorithms was illustrated in the case obtained with (a) and without (b) weight on pipes.

It is evident that, as illustrated in Figure 3a, the MA and MLRB graph partitioning algorithms outperform community structure algorithms without weight; while, as illustrated in Figure 3b, the best result with weight on pipes, was obtained in a multi-criteria analysis, with EBC algorithm followed by MA and MLRB algorithms. The radar diagram shows also clearly the worst multi-criteria results obtained with FGC with weight “flow” on pipes.

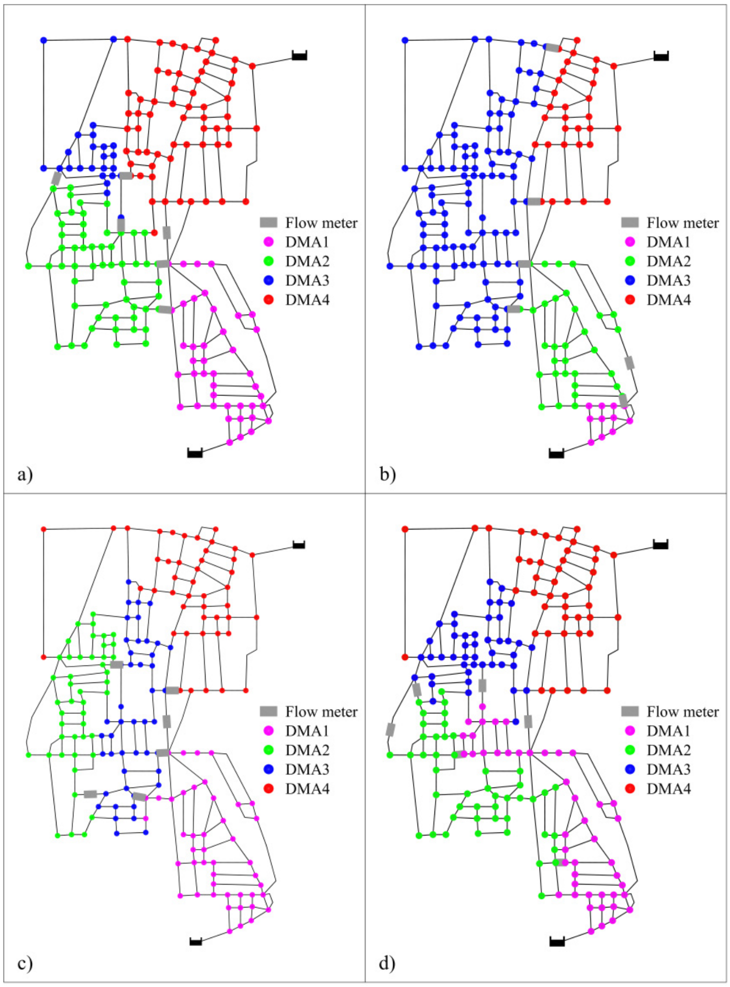

Fig. 2. WNP obtained with weights “flow” on pipes: a) EBC algorithm, b) FGC algorithm, c) MA algorithm and d) MLRB algorithm

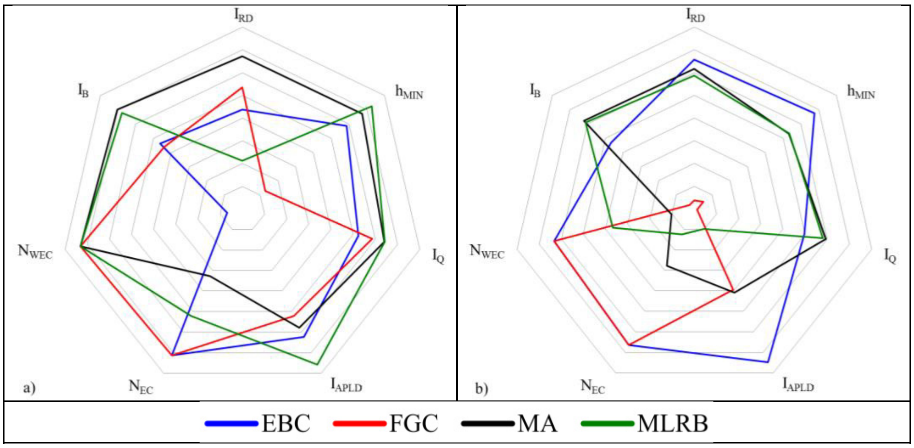

Fig. 3. (a) without weights; (b) with weights

6. Conclusion

The multi-criteria analysis proposed in this paper showed that community structure algorithms, used in social network theory, can represent an effective approach also to define water distribution network clustering and partitioning. In the case study, the best results in terms of energy and topological indices, very useful to evaluate energy and topological redundancy, were obtained with the edge betweenness community (EBC) (a community structure algorithm) and with the multi agent (MA) (a graph partitioning algorithm) coupled with a genetic algorithm to define heuristically the optimal positioning of flow meters and gate valves. Good results were obtained with or without weight on pipes. Next studies are required to evaluate the effectiveness of EBC algorithm on large water distribution networks and if other weights can improve clustering and partitioning phases.

References

[1] Water Authorities Association and Water Research Centre. *Leakage Control Policy and Practice*. Technical Working Group on Waste of Water. London: WRC Group, 1985.
[2] A. Di Nardo, M. Di Natale, G.F. Santonastaso, S. Venticinque, An automated tool for smart water network partitioning, Water Resources Management 27(13) (2013) 4493–4508.
[3] A. Di Nardo, M. Di Natale, G.F. Santonastaso, V.G. Tzatchkov, V.H. Alcocer Yamanaka, Divide and conquer partitioning techniques for smart water networks, Procedia Engineering 89 (2014) 1176–1183.
[4] L.S. Perelman, M. Allen, A. Preis, M. Iqbal, A.J. Whittle, Automated sub-zoning of water distribution systems, Environ. Model. Softw. 65 (2015) 1-14.
[5] A. Di Nardo, M. Di Natale, D. Musmarra, G.F. Santonastaso, V.G. Tzatchkov, V.H. Alcocer-Yamanaka, Dual-use value of network partitioning for water system management and protection from malicious contamination, Journal of Hydroinformatics 17(3) (2015) 361–376.
[6] Water Industry Research Ltd. 1999. *A Manual of DMA Practice*. London: UK Water Industry Research.
[7] A. Di Nardo, M. Di Natale, G.F. Santonastaso, F.P. Tuccinardi, G. Zaccone, SWANP: Software for automatic Smart WAter Network Partitioning, in D.P. Ames, N.W.T. Quinn, A.E. Rizzoli (Eds.), Proc. 7th Intl. Congress on Env. Modelling and Software, June 15-19, San Diego, CA, USA, 2014.
[8] G. Karypis, V. Kumar, Multilevel k-way partitioning scheme for irregular graphs, J. Parallel Distrib. Comp. 48 (1998) 96–129.
[9] G. Ferrari, G. Becciu, Hybrid graph partitioning approach for dividing a water distribution network into district metered areas, in: WDSA 2012: 14th WDSA Conference, 24-27 September 2012, Adelaide, South Australia. Barton, A.C.T. Engineers Australia, 2012, pp. 569–578.
[10] S. Alvisi, M. Franchini, A heuristic procedure for the automatic creation of district metered areas in water distribution systems, Urban Water J. 11(2) (2014) 137–159.
[11] M. Girvan , M. E. J. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. 99(12) (2002) 7821–7826.
[12] D. Combe, C. Largeron, E. Egyed-Zsigmond and M. Géry, A comparative study of social network analysis tools, Proc. International Workshop on Web Intelligence and Virtual Enterprises, Saint-Étienne, France, 2010, pp. 1–14.
[13] M. E. J. Newman, Scientific collaboration networks: II. Shortest paths, weighted networks, and centrality, Phys. Rev. E. 64(1) (2001).
[14] D. R. White and F. Harary, The cohesiveness of blocks in social networks: Connectivity and conditional density, Sociological Methodology, 31(1) (2001) 305–359.
[15] M. E. J. Newman, Fast algorithm for detecting community structure in networks, Phys. Rev. E. 69(6) (2004).
[16] A. Clauset, M. E. J. Newman, C. Moore, Finding community structure in very large networks, Phys. Rev. E. 70 (2004).
[17] K. Diao, Y. Zhou, W. Rauch, Automated creation of district metered area boundaries in water distribution systems, J. Water Resour. Plann. Manage. 139(2) (2013) 184–190.
[18] E. Campbell, D. Ayala-Cabrera, J. Izquierdo, R. Pérez-García, M. Tavera, Water Supply Network Sectorization Based on Social Networks Community Detection Algorithms 89 (2014) 1208–1215.
[19] O. Giustolisi, Ridolfi, L., New Modularity-Based Approach to Segmentation of Water Distribution Networks, J. Hydraul. Eng. 140 (10).
[20] A. Di Nardo, M. Di Natale, G.F. Santonastaso, V.G. Tzatchkov, V.H. Alcocer-Yamanaka, Performance indices for water network partitioning and sectorization, Water Science & Technology: Water Supply 70 (2014). doi:10.2166/ws.2014.132
[21] X.F. Wang, Complex Networks: Topology, Dynamics and Synchronization, Int. J. Bifurcat Chaos, 12(5) (2002) 885–916
[22] A. Di Nardo, M. Di Natale, R. Greco, G.F. Santonastaso, Ant Algorithm for Smart Water Network Partitioning, Procedia Engineering, 70 (2014) 525–534.
[23] F. Comellas, E. Sapena, A Multiagent Algorithm for Graph Partitioning, in F. Rothlauf et al. (Eds.), Applications of Evolutionary Computing. Lecture Notes in Computer Science. Springer, Berlin, 2006, 3907, pp.279-285.
[24] S. Fortunato, Community detection in graphs, Phys. Rep. 486 (2010) 75–174.
[25] U. Brandes, On Variants of Shortest-Path Betweenness Centrality and their Generic Computation, Soc. Networks, 30(2) (2008) 136–145.
[26] L. Freeman, A set of measures of centrality based upon betweeness, Sociometry 40(1) (1977) 35–41.
[27] L.A. Rossman, EPANET2 Users Manual, US E.P.A. Cincinnati (OH), 2000.
[28] E. Todini, Looped water distribution networks design using a resilience index based heuristic approach, Urban Water, 2(2) (2000) 115–122.