

Modified spectral clustering method for graphs decomposition

Dinar Yakupov

*Institute of Technical Cybernetics & Informatics
Kazan National Research Technical University named after
A.N.Tupolev – KAI
Kazan, Russia
yaqup@mail.ru*

Vladimir Mokshin

*Institute of Technical Cybernetics & Informatics
Kazan National Research Technical University named after
A.N.Tupolev – KAI
Kazan, Russia
vladimir.mokshin@mail.ru*

Abstract—Among a large number of tasks on graphs, studies related to the placement of objects with the aim of increasing the information content of complex multi-parameter systems find wide practical application (for example, in transport and computer networks, piping systems, in image processing). Despite years of research, accurate and efficient algorithms cannot be found for placement problems. It is proposed to consider the solution of the allocation problem in the context of decomposition of the initial network into k regions, in each of which a vertex with some centrality property is searched. This article provides an analysis of sources for solving the problem of placement in graphs, as well as methods of decomposition of graph structures. Following the main provisions of the theory of spectral clustering, the disadvantages of the splitting applied criteria $Rcut$ and $Ncut$ are indicated. It is shown that the application of the distance minimization criterion $Dcut$ proposed in this paper allows to obtain high results in the decomposition of the graph. The obtained results are based on the examples of searching for sensor placement vertices in the known ZJ and D-Town networks of the EPANET hydraulic modeling system.

Keywords—graph, spectral, clustering, distance minimization, decomposition

I. INTRODUCTION

Graph models provide an opportunity to study an object based on its topology, without delving into the physical nature of the processes occurring in the system under consideration, which, in turn, greatly simplifies the calculations [1-3]. Among the many problems on graphs, studies related to splitting the original graph into a predetermined number of connected disjoint components have found wide practical application [4-8]. Methods of decomposition of graph structures make a significant contribution to the speed of search algorithms, which is especially important in conditions of restrictions on computational and time resources. However, widespread algorithms of spectral clustering based on minimization of $Rcut$ and $Ncut$ sections do not always allow to solve the problem of object placement in the best way. The reason for this is that the decomposition by these criteria takes into account the number of cut edges and the size of the resulting subgraphs, but does not take into account the distances between the vertices and the nature of their location within these subdomains. This paper provides an example showing that decomposing the original graph of 12 vertices and 12 edges into two parts, there are two splitting options that meet the $Rcut$ and $Ncut$ criteria, but when considering these options in terms of the distances between the vertices within these subdomains, the second option is preferred. The use of this criterion, designated in the paper as $Dcut$, in the decomposition of graphs allowed us to solve the problem of placing objects in the network with a high quality result, comparable and even better than spectral methods based on

$Rcut$ or $Ncut$ criteria, which confirms the applicability of the $Dcut$ criterion in spectral methods of clustering graphs.

II. THE ANALYSIS OF THE SOURCES

The solution of the problem of finding the optimal placement of objects in k nodes for a network with the number of vertices $|C|$ by the full search method requires $\frac{|C|!}{k!(|C|-k)!}$ iterations. For example, for a small network with $|C|=100$ and $k=5$ it is required $10^{7.88}$ iterations, which makes practical application of this method impossible. Trial-and-error, greedy, and stochastic algorithms are the most widely used approaches for solving the placement problem.

The trial-and-error algorithm is based on iterations. This approach provides a fairly close to optimal solution, but requires a lot of time.

The principle of the greedy algorithm is to make locally optimal decisions at each stage, assuming that the final solution will also be optimal. The solution is fast, but not accurate.

Solving by evolutionary computation, simulated annealing algorithms is based on choosing combinations of nodes placing objects on the basis of the probabilistic approach does not provide warranty of solution time and solution as a whole.

In [8, 9] the solution of the placement problem is proposed to be considered in the context of the decomposition of the initial network into k regions, in each of which a vertex with some property of centrality is searched. When decomposing a graph, it is necessary to minimize the number of edges connecting the vertices of different subdomains. A prerequisite for this is the connectivity of the selected subgraphs.

Many methods are used to solve the graph decomposition problem [10-15].

Since the 1990s, spectral graph theory has been used in many fields [16]. The main advantage of spectral graph theory is simplicity, so any system represented as a graph can be analyzed only by the spectrum of the associated matrix.

Fiedler in [17] showed that the eigenvector that corresponds to the second smallest eigenvalue of the Laplacian matrix can be used to solve the problem of bipartite graph decomposition. Hagen and Kahng [18] introduce the criterion of rational sections ($Rcut$) to assess the quality of decomposition. Shi and Malik in [19] use conclusions by Fiedler for iterative bipartite partitioning and introduce the normalized sections criterion ($Ncut$). The development and application of the theory of spectral clustering are also considered in [20-25].

III. THEORY OF SPECTRAL CLUSTERING OF GRAPHS

A. Fundamentals

The class of spectral decomposition methods [26-29] combines elements of graph theory and linear algebra. They are based on the application of the properties of eigenvalues and vectors of the Laplacian matrix of the graph.

The eigenvectors contain information about the topology of the graph. Based on the problems, the spectral graph theory uses: the main eigenvector [30], Fiedler eigenvector [17], a group of the first k eigenvectors [19]. A review of spectral clustering methods is presented in [31, 32].

Spectral clustering algorithms consist of three main steps:

1) For the original graph G , the adjacency matrix W , the matrix of degrees of vertices D , the Laplacian matrix L are forming. In addition to the non-normalized Laplacian matrix, its normalized equivalents are also used, for example, the Laplacian matrix normalized by the random walk method [19] or the symmetric normalized Laplacian matrix [20].

2) Determination of eigenvalues and eigenvectors of the non-normalized or normalized Laplacian matrix, which are used in the formation of the matrix of eigenvectors U .

3) Division of the set of vertices into k clusters by classical clustering methods, for example, the k-means method in relation to the matrix U .

B. Graph cut point of view

Methods of spectral clustering are aimed at obtaining such subgraphs that the difference between the constituent elements of the subdomain is minimal with the maximum difference between the subgraphs. In this case, the subgraphs must be connected and balanced in size. To implement these conditions, the criteria proposed in [18, 19]:

$$Rcut(C_k) = \sum_{i=1}^k \frac{cut(C_i, G)}{|C_i|} \rightarrow \min \quad (1)$$

$$Ncut(C_k) = \sum_{i=1}^k \frac{cut(C_i, G)}{vol(C_i)} \rightarrow \min \quad (2)$$

where G is the initial graph, C_i is i -th subgraph, k is the number of sub-areas to divide the original graph, $cut(C_i, G)$ is the sum of the weights of the cut edges, $|C_i|$ is the quantity of vertices in the subgraph i , $vol(C_i)$ is the sum of the weights of edges in subgraph i .

It should be noted that the values of both criteria tend to a minimum with decreasing edge sections and with balancing subgraphs ($|C_1| = |C_2| = \dots = |C_k|$ or $vol(C_1) = vol(C_2) = \dots = vol(C_k)$). According to [32], the $Rcut$ criterion is preferred for decomposition by non-normalized matrices, and $Ncut$ is preferred for decomposition by normalized matrices.

However, the criteria under consideration do not always clearly indicate a solution. Figure 1 shows a graph with 12 vertices and 12 edges. The weight of each edge, according to the figure, is 10.

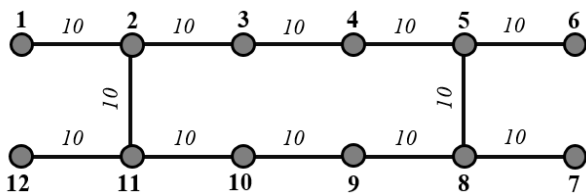


Fig. 1. The original graph with 12 vertices and 12 edges.

When searching for the optimal decomposition on $k=2$ subgraphs, we get two variants (figure 2).

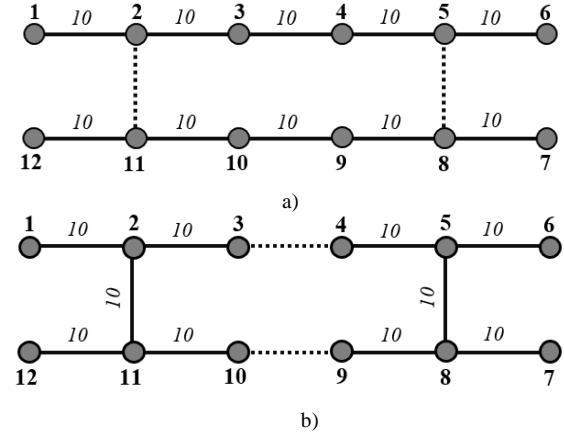


Fig. 2. Decomposition of the original graph into 2 parts: a. with a partition on edges 2-11 and 5-8; b. with a partition on edges 3-4 and 9-10.

Next, according to (1) and (2), we define the values of the $Rcut$ and $Ncut$ criteria for the first (figure 2.a) and second (figure 2.b) variants of the partition:

$$Rcut_1 = \frac{cut_{2,11} + cut_{5,8}}{|C_1|} + \frac{cut_{2,11} + cut_{5,8}}{|C_2|} = 6.67 \quad (3)$$

$$Ncut_1 = \frac{cut_{2,11} + cut_{5,8}}{vol(C_1)} + \frac{cut_{2,11} + cut_{5,8}}{vol(C_2)} = 0.8 \quad (4)$$

$$Rcut_2 = \frac{cut_{3,4} + cut_{9,10}}{|C_1'|} + \frac{cut_{3,4} + cut_{9,10}}{|C_2'|} = 6.67 \quad (5)$$

$$Ncut_2 = \frac{cut_{3,4} + cut_{9,10}}{vol(C_1')} + \frac{cut_{3,4} + cut_{9,10}}{vol(C_2')} = 0.8 \quad (6)$$

From calculations it is clear that from the point of view of $Rcut$ and $Ncut$ both variants of splitting give the same result. However, when solving placement problems, it is important to consider the distances between all vertices in subgraphs. Tables 1 and 2 show the distances between vertices in subgraphs when decomposing by the first (figure 2.a) and second variants of the partition (figure 2.b).

It can be seen from the tables that the second variant of splitting the graph into 2 parts provides a smaller distance length in subgraphs, which indicates a greater degree of grouping of vertices in subgraphs in the second variant of decomposition.

TABLE I. DISTANCES BETWEEN VERTICES IN SUBGRAPHS (DECOMPOSING BY THE FIRST VARIANT OF THE PARTITION)

Subgraph C_1						
Nodes	1	2	3	4	5	6
1	0	10	20	30	40	50
2	10	0	10	20	30	40
3	20	10	0	10	20	30
4	30	20	10	0	10	20
5	40	30	20	10	0	10
6	50	40	30	20	10	0
Sum: 700						
Subgraph C_2						
Nodes	7	8	9	10	11	12
7	0	10	20	30	40	50
8	10	0	10	20	30	40
9	20	10	0	10	20	30
10	30	20	10	0	10	20
11	40	30	20	10	0	10
12	50	40	30	20	10	0
Sum: 700						

TABLE II. DISTANCES BETWEEN VERTICES IN SUBGRAPHS (DECOMPOSING BY THE SECOND VARIANT OF THE PARTITION)

Subgraph C_1'						
Nodes	1	2	3	10	11	12
1	0	10	20	30	20	30
2	10	0	10	20	10	20
3	20	10	0	30	20	30
10	30	20	30	0	10	20
11	20	10	20	10	0	10
12	30	20	30	20	10	0
Sum: 580						
Subgraph C_2'						
Nodes	4	5	6	7	8	9
4	0	10	20	30	20	30
5	10	0	10	20	10	20
6	20	10	0	30	20	30
7	30	20	30	0	10	20
8	20	10	20	10	0	10
9	30	20	30	20	10	0
Sum: 580						

C. Distance minimization criteria

Let us consider the solution of the problem of placing $k=2$ objects on the basis of the preliminary decomposition of the graph according to options 1 and 2.

The problem under consideration can be formulated as follows. There is a graph G whose nodes are characterized by certain parameter estimates P_i , and whose edges are characterized by weights W_j . After setting the next object in the vertex, the estimates are recalculated according to the formulas:

$$P_S = 1 \quad (7)$$

$$P_i = \max(P_y \cdot 1/W_{i,y}) \quad (8)$$

where P_S is evaluation of the deterministic value of the parameter in the node setup of the object, P_y is evaluation of the deterministic value of the parameter of node neighbor, $W_{i,y}$ is the weight of edge connecting two adjacent vertices i and y .

It is necessary to find such an arrangement of objects in the nodes that provides a minimum of the average value of the uncertainty estimation of the target parameter:

$$F = 1 - \text{mean}(P) \rightarrow \min \quad (9)$$

The values of the estimates P_i of each vertex for the decomposition variants 1 and 2 are shown in figure 3:

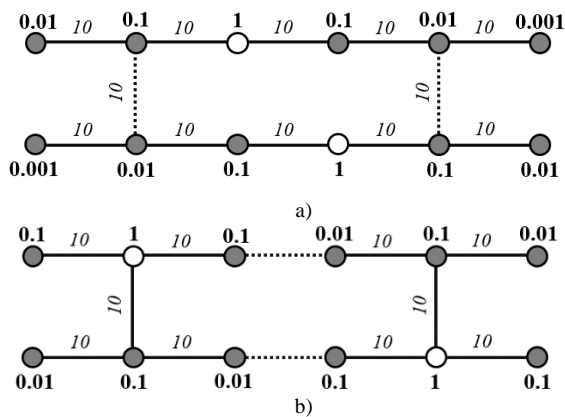


Fig. 3. Values of P_i scores: a. splitting by edges 2-11 and 5-8; b. splitting by edges 3-4 and 9-10.

White color indicates the vertices where the objects are placed, next to the vertices are given the values of the determinism estimates P_i . As a result, we get:

$$F_1 = 1 - \text{mean}(P) = 0.796 \quad (10)$$

$$F_2 = 1 - \text{mean}(P') = 0.780 \quad (11)$$

The value of F_1 is greater than F_2 , which means that the placement based on the decomposition of the second option gives better results. This result is mainly due to the fact that the vertices in the subgraphs of the second variant of the decomposition are grouped more tightly.

Thus, the use of a criterion that takes into account the length of distances in subgraphs is justified in solving the problems of placing objects on graphs. In this paper, we propose the following criterion for minimizing distances (12):

$$Dcut(C_k) = \sum_{i=1}^k \left(\frac{\text{cut}(C_i, G)}{\text{vol}(C_i)} * \frac{\text{dist}_i}{|C_i| * (|C_i| - 1)} * \frac{|C|}{|C_i|} \right) \rightarrow \min \quad (12)$$

where dist_i is the sum of the distances between all vertices in subgraph i , $|C|$ is the number of all nodes in the original graph.

D. The algorithm of nodes priority distribution

The structurally optimal number of subgraphs can be estimated by the largest difference between the eigenvalues of the normalized Laplace matrix. Figure 4 shows the first 10 eigenvalues of the normalized Laplace matrix for the D-Town network graph.

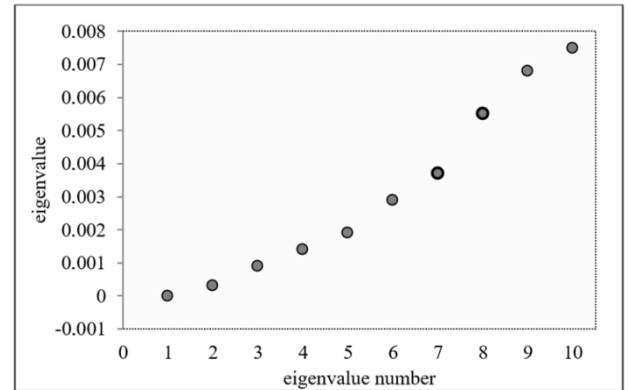


Fig. 4. Eigenvalues of the normalized Laplace matrix.

In the figure, you can see that the largest difference is between the eigenvalues equal to 7 and 8, which means that the best partition of this graph corresponds to 7 subgraphs.

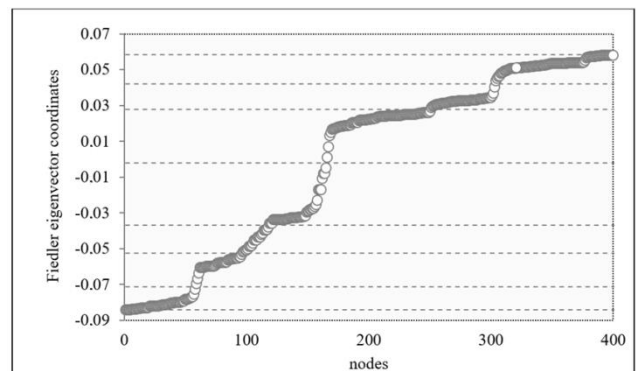


Fig. 5. Fiedler eigenvector coordinates (7 subgraphs).

Consider the plot (figure 5) of the values of elements of the Fiedler eigenvector for the graph under consideration.

Each node of the graph corresponds to the value of an element of the eigenvector. Dotted lines mark the boundaries between clusters. The nature of the graph shows that, indeed, you can distinguish 7 grouped sections.

But what happens if you need to divide the graph into more parts? Figure 6 shows the values of elements of the Fiedler eigenvector for the graph under consideration, divided into 10 subgraphs. It is obvious that the boundaries between subgraphs become less unambiguous, and the probability of some nodes falling into neighboring subdomains increases, which can lead to subgraphs of an unrelated structure. As the number of subdomains increases, the probability of disjoint subgraphs will increase.

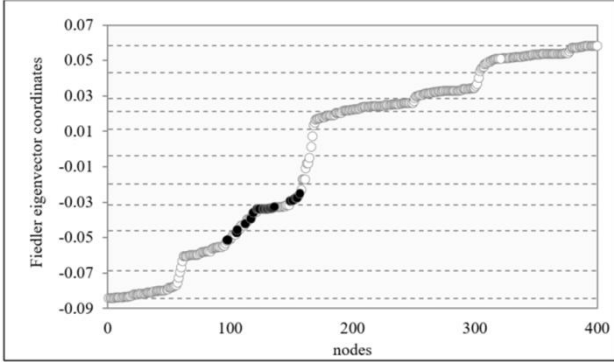


Fig. 6. Fiedler eigenvector coordinates (10 subgraphs).

Figure 7 shows the pipeline network represented by the original graph model divided into 10 subgraphs. Multi-colored sections correspond to different subgraphs, and white nodes are the Central points of the subgraphs. The area highlighted in the figure is part of subgraph "4" and is not related to the second half. The values of the Fiedler eigenvector elements of these "cut off" nodes are marked in black in figure 6. As you can see, these nodes are spread across three clusters with borders $(-0,069; -0,046]$, $(-0,046; -0,032]$, $(-0,032; -0,019]$, which corresponds to the original subgraph "4" and neighboring subgraphs "2" and "3".

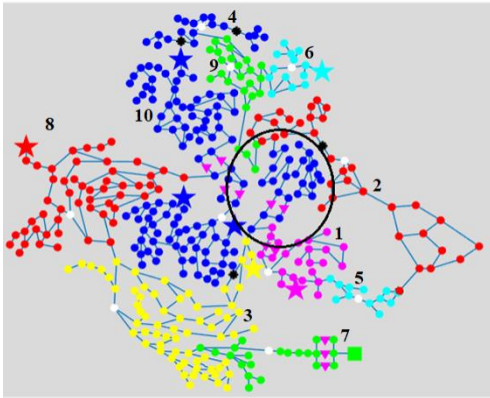


Fig. 7. The graph divided into 10 sub-areas. The "cut off" area is highlighted.

However, with the growth of the number of k clusters, the probability of formation of disconnected subgraphs increases.

Connectivity of subgraphs is a necessary condition, for example, when solving problems of division, path search, etc. The solution of this connectivity problem is using proposed priority node distribution algorithm:

Input: graph $G(V, E)$, number of subdomains k .

Output: the k connected subgraphs.

Steps:

- Step 1. A standard procedure for spectral clustering of the graph is performed.
- Step 2. Subgraphs are formed.
- Step 3. Connectivity is checked. If the subgraph is connected, go to step 10. If the subgraph is not connected, go to step 4.
- Step 4. The boundary node is located (the vertex with the largest distance from the values of the components of the eigenvector to the cluster centroid).
- Step 5. Determined by the neighboring boundary vertices that are not part of the current subgraph.
- Step 6. Subgraphs of neighboring nodes found in step 5 are defined.
- Step 7. The subgraph whose centroid is closest to the boundary vertex is determined.
- Step 8. This boundary node is passed to the subgraph defined in step 7.
- Step 9. Go to step 2.
- Step 10. If the subgraph is the last one, exit, otherwise go to the next subgraph (step 3).

Using this nodes priority distribution algorithm guarantees the connectivity of the resulting subgraphs at low computational cost.

IV. CASE STUDY

The proposed solution, based on the application of a criterion that takes into account the distance lengths in subgraphs, was tested on the example of solving the problem of placing pressure sensors in water supply networks ZJ and D-Town of EPANET hydraulic modeling system (figure 8).

ZJ is a network with 114 nodes and 164 pipes, D-Town has 407 nodes and 459 pipes. Nodes (consumers) of the considered networks are characterized by pressure determinism estimates, and edges (pipelines) are characterized by lengths L_j . After installing the next sensor in the network, the determinism estimates are recalculated by the formulas:

$$P_s = 1 \quad (13)$$

$$P_i = \max(P_y \cdot \alpha_1 \cdot \alpha_2^2 \cdot f(L_{i,y})) \quad (14)$$

where P_s is assessment of determination of pressure values in the node setup of the sensor, P_y is assessment of determination of pressure values of the node-neighbor, α_1 is the estimated error of determination of specific resistance of the pipeline, α_2 is estimating the error of determining the values of water consumption, $f(L_{i,y})$ is a function of the length of the pipeline section to the next node.

The task is to arrange these sensors in the nodes in such a way that provides a minimum of the average value of the estimation of the uncertainty of pressure in the network (9).

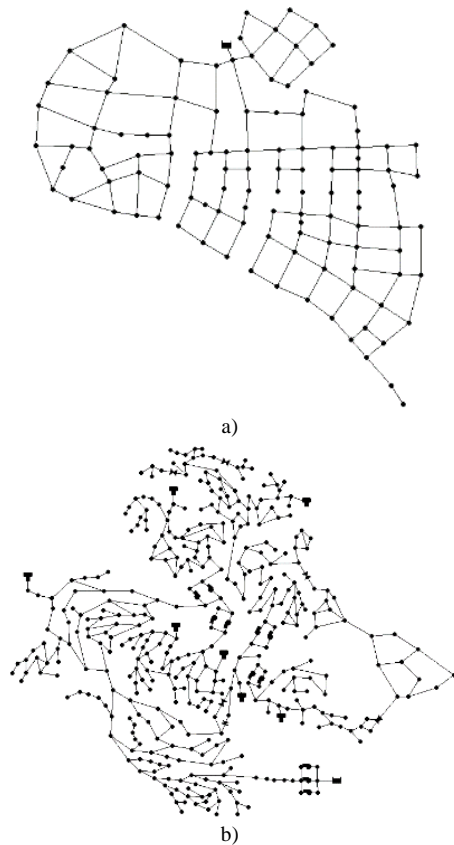


Fig. 8. Water supply networks: a. ZJ, b. D-Town.

For the ZJ network, options for installing sensors in the number from 1 to 10 are considered, for the D-Town network - from 1 to 20. Nodes with the highest centrality in the group are selected as sensor placement vertices.

Solutions are considered: trial and error (*TE*), greedy algorithm (*Gr*), algorithms based on spectral clustering (*SC*). Algorithms based on spectral clustering (*SC*) are considered in the context of using various criteria: *SCr* - spectral clustering by *Rcut* criterion, *SCn* - spectral clustering by *Ncut* criterion, *SCd* - spectral clustering by *Dcut* criterion. The criteria to assess the effectiveness of the algorithms: 1) average uncertainty estimates (*F*), 2) number of iterations ($\sum Iter$), 3) elapsed time (*T*), 4) accuracy rate ($1 - \delta$), where δ is the relative error between the results of the considered algorithm and the algorithm of trial-and-error, 5) the highest relative error ($max(\delta)$).

Tables 3 and 4 show the results of calculating the performance indicators of the algorithms. The best accuracy scores are shown in bold.

TABLE III. PERFORMANCE INDICATORS OF ALGORITHMS (ZJ NETWORK) WITH DIFFERENT CRITERIAS (RCUT, NCUT, DCUT)

Indicator	TE	Gr	SCr	SCn	SCd
<i>F</i>	0.571	0.590	0.572	0.570	0.570
$\sum Iter$	1140	55	10	10	10
<i>T, min</i>	18.9	0.6	1.7	1.8	1.8
$1 - \delta, \%$	100.0	96.8	100.1	100.2	100.2
$max(\delta), \%$	0.0	7.6	1.1	1.0	1.0

TABLE IV. PERFORMANCE INDICATORS OF ALGORITHMS (D-TOWN NETWORK) WITH DIFFERENT CRITERIAS (RCUT, NCUT, DCUT)

Indicator	TE	Gr	SCr	SCn	SCd
<i>F</i>	0.535	0.591	0.561	0.542	0.539
$\sum Iter$	8140	210	20	20	20
<i>T, min</i>	188.5	4.6	6.5	6.6	6.7
$1 - \delta, \%$	100.0	89.5	95.2	98.7	99.3
$max(\delta), \%$	0.0	17.4	9.8	7.3	3.2

The solution obtained by the trial-and-error algorithm was chosen as the reference solution. The main problem of this algorithm is the necessary time and computational resources (1140 and 8140 iterations take 18.9 and 188.5 minutes, respectively). The fastest (0.6 and 4.6 minutes) and at the same time less accurate is greedy algorithm (96.8% and 89.5%). Algorithms based on spectral clustering showed close to the reference result at low computational cost (about 1.7-1.8 and 6.5-6.7 minutes). The application of the *Rcut* criterion provides a solution with an accuracy of 100.1% and 95.2%, the *Ncut* criterion - 100.2% and 98.7%. The best results among the methods of spectral clustering for the *Dcut* criterion are 100.2% and 99.3% in relation to the results of the trial-and-error algorithm.

Thus, the application of the proposed *Dcut* distance minimization criterion for graph decomposition allowed us to solve the problem of placing objects in the network with a high quality result, comparable and even better than spectral methods based on *Rcut* or *Ncut* criteria, which confirms the applicability of the *Dcut* criterion in spectral methods of graph clustering.

V. CONCLUSION

This article offers a look at the problems of solving the problems of placing objects in the network in the context of finding a solution in pre-defined subdomains obtained using the tools of the theory of spectral clustering of graphs by the criterion of minimizing distances in the desired subgraphs. The analysis of sources for solving the problem of placement in graphs, as well as methods of decomposition of graph structures are given. It is shown that many combinatorial problems on graphs can be solved with acceptable accuracy and in a short time, performing a search not in the entire set of graph elements, but on local sets grouped by a certain criterion. The proven theory of spectral clustering of graphs is proposed as a decomposition tool. Following the main provisions of this theory, the disadvantages of the applied criteria for splitting *Rcut* and *Ncut* are indicated. It is shown that the application of the *Dcut* distance minimization criterion proposed in this paper allows us to obtain good results when decomposing a graph. The obtained results are based on the examples of searching for sensor placement vertices in the known ZJ and D-Town networks of the EPANET hydraulic modeling system.

Further work will be aimed at studying the possibilities of applying the *Dcut* criterion in solving various combinatorial problems on graphs using spectral clustering methods.

ACKNOWLEDGMENT

Authors thank all colleagues from The Department of Automated Information Processing Systems & Control of KNRTU named after A.N. Tupolev.

REFERENCES

- [1] V.V. Mokshin, I.M. Yakimov, R.M. Yulmetyev and A.V. Mokshin "Recursive-regression self-organization of complex systems analysis and control models," *Nelineinyy mir*, vol. 7, no. 1, pp. 66-76, 2009.
- [2] V.V. Mokshin, "Parallel genetic algorithm for selection of significant factors influencing the evolution of a complex system," *Vestnik Kazanskogo gosudarstvennogo tekhnicheskogo universiteta im. A.N. Tupoleva*, no. 3, pp. 89-93, 2009.
- [3] I.M. Yakimov, A.P. Kirpichnikov, V.V. Mokshin, "Modeling of complex systems in GPSS W simulation environment with advanced redactor," *Vestnik Kazanskogo tekhnologicheskogo universiteta*, vol. 17, no. 4, pp. 298-303, 2014.
- [4] I.R. Saifudinov, V.V. Mokshin, P.I. Tutubalin and A.P. Kirpichnikov, "Optimization model of graph representation for allocation of significant structures on the example of visual data preprocessing," *Vestnik tekhnologicheskogo universiteta*, vol. 21, no. 5, pp. 121-129, 2018.
- [5] G.I. Galimova and D.T. Yakupov, "Statistical analysis of the urban water consumption for administrative-business sector," *Amazonia Investiga*, vol. 7, no. 17, pp. 414- 425, 2018.
- [6] I.R. Saifudinov, V.V. Mokshin, A.P. Kirpichnikov, P.I. Tutubalin and L.M. Shamin, "Method of allocation of significant areas in graphic information for decision support systems," *Vestnik tekhnologicheskogo universiteta*, vol. 21, no. 6, pp. 146-152, 2018.
- [7] V.V. Mokshin and I.M. Yakimov, "Method of formation of complex system analysis model," *Informatsionnye tekhnologii*, no. 5, pp. 46-51, 2011.
- [8] G.I. Galimova, I.D. Galimyanov, D.T. Yakupov and V.V. Mokshin, "Application of Spectral Clustering Methods in Pipeline Systems Graph Models," *Helix*, vol. 9, no. 5, pp. 5607- 5614, 2019.
- [9] A. Di Nardo, C. Giudicianni and R. Greco, "Sensor placement in water distribution networks based on spectral algorithms," *EPiC Series in Engineering*, vol. 3, pp. 593- 600, 2018.
- [10] A. George and J.W. Liu, "Computer Solution of Large Sparse Positive Definite Systems," Prentice-Hall, Englewood Cliffs, NJ, 1981.
- [11] K. Schloegel, G. Karypis and V. Kumar, "Graph partitioning for high performance scientific simulations," *CRPC Parallel Computing Handbook*, Morgan Kaufmann, 2001.
- [12] B.W. Kernighan and S.Lin, "An efficient heuristic procedure for partitioning graphs," *The Bell System Technical Journal*, vol. 49, no. 2, pp. 291-307, 1970.
- [13] C.M. Fiduccia and R.M. Mattheyses, "A linear time heuristic for improving network partitions," *Proc. 19th IEEE Design Automation Conference*, pp. 175-181, 1982.
- [14] A.L. Zheleznyakova, "Efficient decomposition methods of unstructured adaptive grids for high-performance calculations in solving computational aerodynamics problems," *Fiziko-himicheskaya kinetika v gazovoy dinamike*, vol. 18, no. 1, 2017.
- [15] V. Dongen and S. Graph, "Clustering by Flow Simulation," University of Utrecht, The Netherlands, 2000.
- [16] D. Cvetkovic and S. Simic, "Graph spectra in computer science," *Linear Algebra Appl.*, no. 434, pp. 1545-1562, 2011.
- [17] M. Fiedler, "Algebraic connectivity of graphs," *Math. J., Czechoslov.*, no. 23, pp. 298-305, 1973.
- [18] L. Hagen and A. Kahng, "New spectral methods for ratio cut partitioning and clustering A," *IEEE Trans. Comput.-Aided Des. of Integrated Circuits and Systems*, vol. 11, no. 9, pp 1074-1085, 1992.
- [19] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, no. 22, pp. 888-905, 2000.
- [20] A.Y. Ng, M.I. Jordan and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in Neural Information Processing Systems (NIPS)*, vol. 14, pp. 849-856, 2002.
- [21] E. Ary-Catro, G. Chen and G. Lerman, "Spectral clustering based on local linear approximations," *Electronic Journal of Statistics*, vol. 5, pp. 1537-1587, 2011.
- [22] G. Chen, S. Atev and G. Lerman, "Kernel spectral curvature clustering (KSCC)," *Dynamical Vision Workshop, IEEE 12th International Conference on Computer Vision, Kyoto, Japan*, pp. 765-772, 2009.
- [23] G. Chen and G. Lerman, "Spectral curvature clustering (SCC)," *Int. J. Comput. Vision*, no. 81, pp. 317-330, 2009.
- [24] Q. Guo, H. Li, W. Chen, I.-F. Shen and J. Parkkinen, "Manifold clustering via energy minimization," *Proceedings of the Sixth International Conference on Machine Learning and Applications, Washington, DC, USA*, pp. 375-380, 2007.
- [25] U. Luxburg and M. Belkin, "Consistency of spectral clustering," *Ann. Statist.*, no. 36, pp. 555-586, 2008.
- [26] H.D. Simon, "Partitioning of unstructured problems for parallel processing," *Comput. Syst. Eng.*, vol. 2, pp. 135-148, 1991.
- [27] R.D. Williams, "Performance of dynamic load balancing algorithms for unstructured mesh calculations," *Technical Report, California Institute of Technology*, 1990.
- [28] T.F. Chan and D.C. Resasco, "A framework for the analysis and construction of domain decomposition preconditioners," *Technical Report CAM-87-09, University of California, Los Angeles*, 1987.
- [29] D.A. Spielman and S.-H. Teng, "Spectral partitioning works: Planar graphs and finite element meshes," *Linear Algebra and its Applications*, vol. 421, pp. 284-305, 2007.
- [30] D.E. Plotnikov, P.A. Kolbudaev and S.A. Bartalev, "Identification of dynamically homogeneous areas with time series segmentation of remote sensing data," *Computer Optics*, vol. 42, no. 3, pp. 447-456, 2018. DOI: 10.18287/2412-6179-2018-42-3-447-456.
- [31] B. Mohar, "The Laplacian spectrum of graphs," *Graph Theory, Combinatorics, and Applications, Wiley: Hoboken, NJ, USA*, pp. 871-898, 1991.
- [32] U. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, pp. 395-41, 2007.