CLUSTERS IN SELF-ORGANIZING SYSTEMS OF SIMILAR ELEMENTS APPLIED TO THREE-DIMENSIONAL NEURAL NETWORKS

VICTOR TRUBITSIN[1], MILAN SAGA[2], ALEKSANDR KORSHUNOV[1], ZUZANA SAGOVA[2]

[1]Udmurt Federal Research Center of the Ural Branch of the RAS, Izhevsk, Udmurtia

[2]University of Zilina, Faculty of Mechanical Engineering, Zilina, Slovakia

DOI: 10.17973/MMSJ.2026_06_2026117

zuzana.sagova@fstroj.uniza.sk

This paper addresses the problem of community detection (clustering) in self-organizing systems consisting of a large number of similar interacting elements. A three-dimensional neural network is chosen as a model example, where neurons act as elements and synaptic connections serve as edges of a weighted graph. An adaptation of the Louvain method, one of the most efficient algorithms for community detection in large graphs, to this class of systems is proposed. The mathematical foundations of the method are presented: definition of modularity considering the effective distance between elements (length of axons and dendrites), description of the two-phase iterative procedure, and formulas for modularity gain. Quality metrics specific to three-dimensional neural structures are discussed. The results can be used to analyze the functional organization of neuronal ensembles in neurophysiological studies.

KEYWORDS

Louvain method, Clustering, Modularity, Self-organizing systems, Neural networks, Graphs, Communities, 3D reconstruction, Effective distance

1 INTRODUCTION

Self-organizing systems consisting of many similar elements occur in various fields, from social networks and biological systems to technical complexes. A prominent example is the neural network of the brain, where billions of neurons form a complex connectivity structure that enables information processing. Understanding the principles of functional organization of such systems is impossible without identifying stable groups (communities) of elements within which connections are significantly denser than between groups. Such communities often correspond to functional modules performing specific tasks: in the brain these may be cortical columns processing sensory information, in social networks – groups of people with shared interests, in technical systems – subsystems responsible for particular functions [Demcak 2024].

Clustering (community detection) not only simplifies the analysis of a complex system but also reveals hidden patterns of its organization, predicts the behavior of individual elements, and helps develop management or intervention strategies. However, as system size grows (to millions or billions of elements), traditional clustering methods become inapplicable due to high computational complexity. For instance, hierarchical clustering in a naive implementation requires O(N3) operations [Murtagh 2012], spectral methods require O(N3) [Ding 2024, Saga 2011] for full diagonalization, although approximate versions exist. Algorithms based on random walks are more efficient but can still be resource-intensive for very large graphs.

In 2008, Blondel et al. proposed the Louvain method [Blondel 2008], based on heuristic optimization of modularity – a quality measure of partitioning introduced by Newman and Girvan [Newman 2004]. The method combines high speed (nearly linear complexity for sparse graphs) with good partitioning quality, does not require a priori knowledge of the number of communities, and can reveal hierarchical structure. Due to these properties, the Louvain method has become one of the most popular tools for analyzing large networks in various domains.

However, when applied to three-dimensional biological systems such as neural networks, the classical Louvain method does not account for the spatial arrangement of elements, which plays a key role in the formation of connections. In neural tissue, the probability of synaptic contact is determined not only by topology but also by the actual length of axons and dendrites connecting neurons [Stepanyants 2002]. Therefore, it is necessary to adapt the method by replacing geometric distance with effective distance along the neurites, which will more adequately model the structure of neuronal ensembles.

The aim of this work is to adapt the Louvain method for clustering three-dimensional self-organizing systems using neural networks as an example, considering the effective distance between elements (length of axons and dendrites). The paper provides a mathematical description of the method, discusses the modifications required to account for such distances, and proposes metrics for evaluating the quality of detected clusters in the context of neurobiological data.

2 MODULARITY AND ITS ROLE IN CLUSTERING

Consider a self-organizing system consisting of N similar elements. Each element i $(i = 1, . . . , N)$ is characterized by a set of parameters, in particular coordinates in space $r_i = (x_i, y_i, z_i) \in R^3$. Connections between elements may exist, with intensity described by a non-negative weight w_{ij}. Thus, the system is modeled as a weighted undirected graph $G = (V, E, W)$, where $V = \{v_1, . . . , vN\}$ is the set of vertices, $E \subseteq V \times V$ is the set of edges, and $W : E \rightarrow R^+$ is a weight function.

The clustering problem consists of partitioning the vertex set V into K disjoint subsets $C_1, C_2, . . . , C_K$ (communities) such that vertices within a community are more strongly connected than with vertices from other communities. The partition should reflect the internal organization of the system and be useful for interpreting its functioning. In the context of neural networks, vertices are neurons, edges are synaptic contacts (axons, dendrites), and weights may reflect synaptic strength, spike frequency, or number of contacts.

Modularity is a scalar function that evaluates the quality of partitioning graph vertices into communities. Introduced by Newman and Girvan [Newman 2004], it compares the actual density of edges within communities to the expected density in a random graph that preserves the vertex degrees of the original network. The idea is that a good partition should have significantly more internal connections than would be expected by random chance.

Mathematically, modularity is defined as the sum over all pairs of vertices of contributions: if an edge exists between two vertices and they belong to the same community, and the expected number of edges is less than the actual number. For a weighted undirected graph, the classical modularity formula is:

$$Q = \frac{1}{2m}\sum_{i,j}\left[A_{ij} - \frac{k_i k_j}{2m}\right]\delta(c_i, c_j), \quad (1)$$

where: A_{ij} is the weight of the edge between vertices *i* and *j* (0 if no edge).

$k_i = \sum_j A_{ij}$ is the weighted degree of vertex *i*; m = ½ $\sum_j k_i$ is the total weight of all edges; ci is the index of the community to which vertex *i* is assigned; $\delta(c_i, c_j)$ is the Kronecker delta, equal to 1 if $c_i = c_j$ and 0 otherwise.

The quantity $(k_i k_j)/2m$ represents the expected weight of the edge between *i* and *j* in a null model – the so-called configuration model, where edges are placed randomly while preserving vertex degrees. Subtracting this expectation from the actual weight gives a measure of how much stronger (or weaker) the real connection is compared to random.

In the Louvain method, vertices are iteratively moved between communities to increase the value of *Q*. Each move that yields a positive gain *ΔQ* improves the partition quality. Thus, modularity serves as an objective function guiding the algorithm toward a local optimum. Importantly, it does not require a priori specification of the number of communities – it is determined automatically during optimization.

Modularity has limitations. There is a well-known resolution limit: for very large networks, the method may fail to detect small but well-defined communities because their contribution to the total sum falls below the sensitivity threshold [Fortunato 2010]. Moreover, the maximum modularity may correspond to a partition that does not match the true structure, especially if the graph has a heterogeneous degree distribution.

To overcome these limitations, a resolution parameter γ was introduced [Reichardt 2006], which scales the contribution of the null model:

$$Q = \frac{1}{2m}\sum_{i,j}\left[A_{ij} - \gamma\frac{k_i k_j}{2m}\right]\delta(c_i, c_j), \quad (2)$$

For $\gamma > 1$, the null model gives a larger expected weight, so only very dense connections contribute positively – this promotes the detection of small communities. Conversely, γ < 1 allows vertices to merge into larger clusters. By varying γ, one can explore the hierarchical organization of the system at different scales.

The choice of the null model $(k_i k_j)/2m$ is not arbitrary. This model is the simplest that preserves vertex degrees (configuration model). It has an important property: the sum of expected weights of all edges equals m, ensuring normalization. Moreover, it corresponds to the maximally random graph with a given degree distribution, making it a natural benchmark for comparison. Thus, modularity is a powerful and flexible tool for graph clustering research, and despite known limitations, it remains one of the most widely used measures due to its interpretability and efficient optimization.

3 MODIFICATION OF THE LOUVAIN METHOD FOR 3D NEURAL SYSTEMS

Three-dimensional self-organizing networks such as neural networks have additional characteristics that must be considered when adapting modularity for the Louvain method [Blondel 2008, Newman 2004]. First, the probability of a connection between two elements depends not so much on the Euclidean distance between their centers as on the actual path length that axons and dendrites must traverse to form synaptic contact. Therefore, it is advisable to introduce the concept of effective distance l_{ij}, measured in units of neurite length (axons and dendrites). Given a complete morphological reconstruction of neurons or full control over the network modeling process, l_{ij} can be computed as the path length from the soma of neuron *i* to the presumed contact site with neuron j along the axon of *i* and the dendrite of *j*. In the absence of such data, one can use the approximation $l_{ij} = \kappa \cdot d_{ij}$, where $d_{ij} = \| r_i - r_j \|^2$ is the Euclidean distance, and κ > 1 is the average tortuosity coefficient of neurites, determined experimentally.

With this in mind, we modify the null model by replacing Euclidean distance with effective distance:

$$Q_{3D} = \frac{1}{2m}\sum_{i,j}\left[A_{ij} - \gamma . f(l_{ij}) . \frac{k_i k_j}{2m}\right]\delta(c_i, c_j), \quad (3)$$

where the spatial decay function *f* (*l*) now depends on the effective distance and can be chosen, for example, as: exponential $f(l) = e^{-\beta l}$, power-law $f(l) = l^{-\eta}$, or Gaussian $f(l) = \exp(-l^2/(2\sigma^2))$. The specific form of the function is determined by the problem and the type of interaction. The parameters *β, η, σ, κ* are selected based on empirical or assumed data about the connectivity structure of the network elements. As a rule, neurons in the brain that are close are more likely to form connections (the principle of wiring economy), but due to axonal tortuosity the effective distance can be substantially larger than the Euclidean distance. Experimental data show that the probability of synaptic contact decays roughly exponentially with distance along neurites [Stepanyants 2002, Meunier 2010]. Thus, choosing $f(s) = e^{-\beta s}$ with parameter *β* determined from empirical data is justified.

Another important aspect to consider is that the expressions for modularity in the form (1, 2) are written in the classical formulation for undirected graphs. In neural networks, information transmission is generally unidirectional: from the neuron body via the axon, synapse, and dendrite to another neuron. [Stepanov 2014]. The signal is not transmitted back along the same path. In this case, the network is a directed graph. For a directed graph, the modularity function takes the form [Leicht 2008]:

$$Q_{directed} = \frac{1}{m}\sum_{i,j}\left[A_{ij} - \gamma . f(l_{ij}) . \frac{k_i^{out} k_j^{in}}{m}\right]\delta(c_i, c_j) \quad (4)$$

Where:

$k_j^{out} = \sum_j A_{ij}$ is the out-degree,

$k_j^{in} = \sum_i A_{ij}$ is the in-degree.

Here $\frac{k_i^{out} k_j^{in}}{m}$ is the expected number of edges from *i* to *j* in a random configuration model that preserves vertex degrees. Thus, modularity measures the difference between the actual edge density within communities and the expected random density.

The presence of the resolution parameter γ in the modularity expression (4) allows adjusting the scale of detected communities. This can be useful because biological neural networks have hierarchical structures: microcolumns, macrocolumns, areas. By varying γ, one can investigate the organization at different levels.

4 LOUVAIN ALGORTIHM

The Louvain method for partitioning a network into clusters consists of two phases repeated iteratively. Phase 1 (local optimization). Initially, each vertex is in its own community. Then, for each vertex v, all neighbouring communities (including its own) are considered, and the modularity gain *ΔQ* for moving v into community *D* is computed. The move is performed if *ΔQ* is positive and maximal. The process is repeated until no further improvements are possible.

For classical modularity, the change when moving vertex v from community C to D is given by:

$$\Delta Q = \frac{k_{v.in(D)} - k_{v.in(c)}}{2m} - \gamma . \frac{s_v . (S_D - S_C + s_v)}{4m^2} \tag{5}$$

This is the general formula for weighted modularity. For an unweighted graph ($s_v = k_v$, m is the number of edges, $\gamma = 1$):

$$\Delta Q = \frac{k_{v.in(D)} - k_{v.in(c)}}{2m} - \frac{k_v . (S_D - S_C + s_v)}{4m^2} \tag{6}$$

where:

is the weight of edges from v to other vertices in community C;

$S_X = \sum_{i \in X} s_i$ is the total strength of vertices in X;

$s_i = \sum_j w_j$ is the total weight of vertex i;

$m = \frac{1}{2} \sum_i s_i$ is the total weight of all edges in the graph.

The simplest way to account for effective distance when computing spatially weighted modularity is to assign edge weights as a decreasing function of distance, for example

$$w_{ij} = \exp\left(-\frac{d_{ij}^2}{\sigma^2}\right) or\ w_{ij} = \frac{1}{d_{ij}^{\alpha}}$$

Then all the above formulas remain valid – it suffices to substitute these weights into the definitions of s_i, m, $k_{v,in(\cdot)}$. Modularity is now interpreted as “spatially weighted”, but mathematically it is ordinary weighted modularity.

Phase 2 (aggregation). After Phase 1 completes, a new graph is built whose vertices are the discovered communities. The weight of the edge between two new vertices (communities) is the sum of the weights of all edges between vertices of the original communities:

$$A'_{ab} = \sum_{i \epsilon C_a} \sum_{j \epsilon C_b} A_{ij}$$

Then Phase 1 is applied again to the resulting graph. The process repeats until modularity no longer increases.

The algorithm has the property of monotonically increasing modularity and has complexity $O(N \log^2 N)$ for sparse graphs, allowing it to handle systems with millions of elements.

5 CLUSTERING QUALITY METRICS

In addition to modularity itself, it is useful to employ additional metrics that account for geometry and effective distances.

Clustering quality metrics are divided into two main types: internal and external. Internal metrics evaluate the quality of a partition using only the information contained in the graph itself (connectivity structure, weights, distances). External metrics require a ground-truth partition (e.g., provided by experts) and measure the degree of agreement with it. Usually, the ground truth partition is unknown, so internal metrics play the main role. Clustering quality metrics are crucial in the analysis of self-organizing systems because they allow quantitative assessment of how well the detected communities correspond to the internal structure of the system [Coranic 2023]. In the context of applying the Louvain method to three-dimensional neural networks, these metrics are needed for several tasks:

1) Assessing the adequacy of the partition – determining whether the found clusters truly reflect functional or structural modules rather than being artifacts of the algorithm.
2) Comparing different clustering options – choosing optimal parameters of the method (e.g., the resolution parameter γ or the decay coefficient *β*).
3) Comparing with results of other algorithms.
4) Validation on known data – if expert labels are available for part of the system, metrics allow estimating the accuracy of agreement.
5) Investigating hierarchy and scales – metrics can help study how community structure changes with parameter variation and identify stable clusters.

Let us consider in more detail the purpose, examples of use, and objectivity of quality metrics.

Average silhouette for vertex *i:*

$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}} \tag{7}$$

where $a(i)$ is the average effective distance from i to other vertices in its own community, and $b(i)$ is the minimum average effective distance to vertices of another community. Averaging over all vertices gives an overall assessment of cluster compactness and separability. Values close to 1 mean that the vertex is well grouped; negative values indicate possible misclustering.

Spatial coherence of community C:

$$Coh(C) = \frac{1}{|C|} \sum_{i \epsilon C} exp\left(-\frac{s_{iC}}{\sigma_0}\right) \tag{8}$$

where s_{iC} is the effective distance from vertex i to the centroid of the community (the centroid is computed in Euclidean space, but distances are taken as effective), and σ_0 is a scale (e.g., the average intra-cluster effective distance). This metric is close to 1 if all points are concentrated around the center in terms of effective distances and decreases with spread.

Sphericity of a cluster:

$$S(C) = \frac{(36\pi V_C^2)^{1/2}}{A_c} \tag{9}$$

where V_C and A_C are the volume and surface area of the convex hull of the cluster points in Euclidean space. For a compact spherical cluster $S \approx 1$, for an elongated one it is smaller. This metric complements the previous ones by characterizing the geometric shape.

Internal edge density – the ratio of the number (or weight) of edges inside a community to the total possible number of edges (accounting for weights). It allows comparison of communities of different sizes.

Clustering quality metrics can be used to tune algorithm parameters. These metrics allow not only evaluating the partition but also comparing different choices of parameters γ, *β*, and *κ*.

The combined use of several independent metrics, as well as analysis of the stability of results with respect to parameter variation and noise, can increase the reliability of conclusions. It must be remembered that metrics are not absolute truth but tools to help make informed decisions. Quality metrics are necessary for objectifying the clustering process, but their application requires critical thinking and consideration of the specifics of the system under study.

Overall, the approach proposed above provides a powerful tool for identifying structural modules in self-organizing systems, especially when combined with three-dimensional morphological data [Kuric 2022].

6 DISCUSSION

To evaluate the performance of the Louvain method for cluster detection in a neural network, a network Net-0 containing 500 neurons was constructed. In all computer experiments, each neuron had 1 to 3 dendrites and one axon. The coordinates of the neuron centre, the number of dendrites, and the coordinates of dendrite and axon endpoints were randomly set in a 3D cube with side lengths equal to 1 in x, y, z. After creating the neurons, a self-organization procedure of the neural network was launched. At the first stage, the positions of axon terminals were optimized based on a procedure maximizing the total energy of the force field acting on each axon from all neurons. The next step, after the axon network formation procedure was completed, was dendrite growth and establishment of connections with the nearest axons. After the formation of connections and reaching equilibrium, the network clustering procedure using the Louvain method was carried out.

Figure 1 shows the formed neural network Net-0. Three types of neurons were considered during network formation: input neurons, which receive information only from outside the cell via dendrites; output neurons, through whose axon's information can be transmitted outside the simulation domain; and ordinary neurons, which receive and transmit information strictly within the simulation domain. Table 1 presents statistics on the main parameters and clustering metrics for this network.

As can be seen from the table, for 500 neurons, as a result of the neural network self-organization procedure, only 790 directed connections were formed. This number of connections is determined by the condition that a neuron can transmit information only in one direction: from the axon to the dendrite of another neuron. Information cannot be transmitted back along the same line. Thus, the neural network is a type of directed graph.

As already mentioned, modularity is the fundamental metric that the Louvain algorithm tries to maximize. Modularity is a measure of the "quality" of partitioning a network into communities. It compares the density of edges within communities with the edge density that would be expected if edges were randomly distributed in the graph. If a community has many more edges than a random graph with the same degree distribution, the modularity is high. Modularity values typically range from -0.5 to +1.

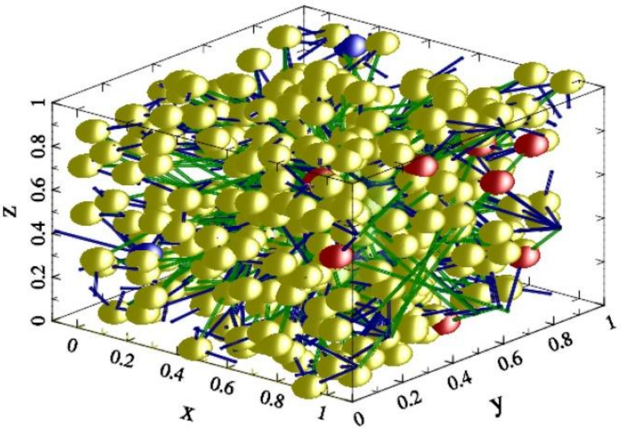

Figure 1. View of the formed neural network of 500 neurons (Net-0). Spheres denote neuron bodies: input neurons in blue, output neurons in red, ordinary neurons in yellow. Blue lines indicate dendrites; green lines indicate axons

Positive values (e.g., 0.3) mean that there are significantly more edges within clusters than between them. A value of zero means the partition is no better than random. Negative values indicate that the graph has virtually no communities (or the partition is poor). Thus, modularity allows assessing how well the graph is divided into isolated groups of nodes. A value of +1 would mean perfect clustering, where all edges are strictly within communities and there are no inter-cluster edges. In real complex networks (social graphs, biological networks) this is practically unattainable. The table shows that the modularity value in our calculations is less than 0.12, which indicates weak clustering in the sparse neural network.

Table 1. Main characteristics and clustering metrics of Net-0

Parameter	Value
Number of neurons	500
Decay coefficient βn	10
Connections (directed)	790
Communities (min-max)	141-147
Modularity	0.1196
Spatial coherence	0.1662
Overall quality	0.4131
Max. number of neurons in a community	14

Another metric given in Table 1 is spatial coherence. "Coherence" of a cluster (community) means how densely and strongly nodes within a community are connected compared to connections with nodes from other communities. In the Louvain method, cluster "coherence" refers to their structural density and isolation, formally measured by modularity (*Q*). In fact, it is an iterative process of greedy modularity maximization, which itself serves as a quantitative measure of the quality (or "coherence") of the found graph partition into communities. This is not physical or semantic coherence but purely topological (network) coherence.

It should be noted that this metric is not built into the Louvain method, which only cares about the presence or absence of graph edges. It is added when graph nodes have geographic coordinates (e.g., 3D coordinates or coordinates in feature space). The meaning of spatial coherence is that it evaluates how compact and continuous clusters are in space (or in the physical world). A good cluster in spatial data should consist of points that are close to each other, not scattered across the map. One use of this metric is noise detection. In the process, the Louvain method might merge two distant regions into one cluster only because they have many shared connections (e.g., migration flows), but physically they are far apart. Low spatial coherence would indicate that this cluster, although well connected in the network, is actually disconnected. An example cluster for network Net-0 is shown in Fig. 2. As can be seen, the neurons belonging to this cluster are located at considerable distances from each other. Compactness here is understood not in terms of distance in 3D space, but in terms of connectivity and proximity in units of axon+dendrite distances. For interpreting the results, it must be remembered that coherence values close to 1 mean that the cluster points are maximally close to each other and maximally distant from other clusters (in the given distance space). This is ideal compactness. Values close to zero mean that clusters overlap or points lie on the boundary.

Another metric presented in the table is Overall quality. This is an integral metric that combines modularity (connection structure) and spatial coherence (geography). This may be needed to find a balance between pure network structure and spatial reality. Typically, one looks at the trade-off: the Louvain algorithm gives maximum modularity, but at the expense of spatial coherence. If necessary, parameters can be tuned to find a sweet spot where both metrics are acceptable. For example, as $Q_{total} = \alpha_1 * Modularity + \alpha_2 * Coherence$. The metric takes values in the range from 0 to 1.

The three metrics discussed actually complement each other. Modularity answers the question: "Are the graph nodes well grouped according to the principle 'I am friends with those who are usually friends with my friends'?" Spatial coherence answers: "Are these groups compact on the map/in space?" Overall quality tries to answer: "How good is this result overall (taking into account both connections and geography)?"

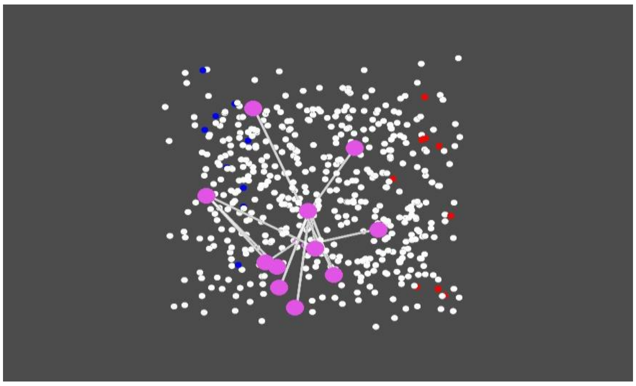

Figure 2. Example of a cluster for network Net-0. Spheres denote neuron bodies: input neurons in blue, output neurons in red, ordinary neurons in gray. Connections between neurons within the cluster are shown by lines

Figure 3 shows the number of communities as a function of the number of neurons in the cluster, calculated for Net-0. As can be seen from the figure, 22% of communities consist of a single neuron. These neurons remain free and do not form communities with other neurons. Of course, this does not mean that they are isolated and cannot receive or transmit information.

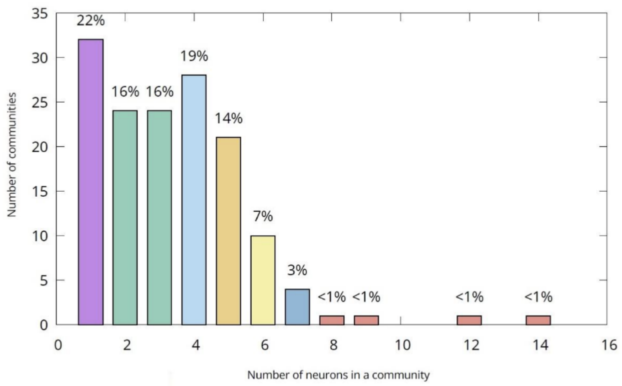

Figure 3. Number of communities as a function of the number of neurons per cluster for network Net-0

It should be noted that the actual partition of the network into clusters depends on the order in which neurons and communities are traversed during iterations. By running the Louvain method multiple times with different initial conditions (or with small perturbations of the data), one can assess which communities are reproduced more often. High stability indicates the reliability of the detected clusters. Figure 3 shows the cluster distribution for a single run. More reliable data can be obtained by averaging the distribution over several independent runs performed on the same formed networks. Figure 4 shows the distribution curve obtained by averaging over 3, 4, and 5 independent runs. It can be seen that as the number of runs increases, a bimodal structure emerges in the region of fewer than 6 neurons per cluster, with maxima at 2 and 4 neurons per cluster and a minimum at 3 neurons. The number of clusters with more than 5 neurons drops sharply. At the same time, there are several large individual clusters of 9, 12, and 14 neurons each.

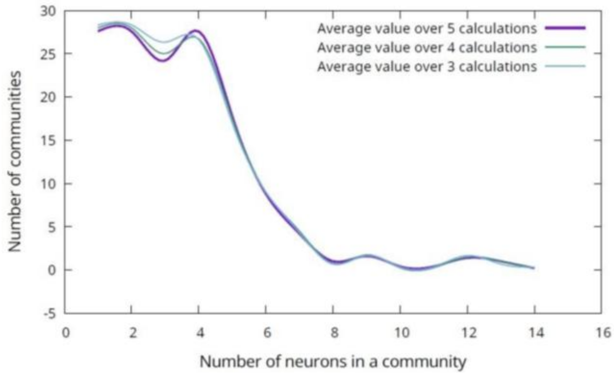

Figure 4. Number of communities as a function of the number of neurons per cluster for a network of 500 neurons. Different lines show graphs for averaging over different numbers of independent runs

The maximum possible community size (number of neurons in it) depends primarily on the size of the neural network. The larger the number of nodes in the network, the larger the clusters that can form. As discussed above, the size of communities can also be tuned by the parameter γ in equation (1). For $\gamma < 1$ communities become larger, and for $\gamma > 1$ they become smaller. In this work, all clustering calculations were performed with $\gamma = 1$.

From Fig. 4 it can be seen that for a network of 500 elements, almost 22% of neurons do not belong to any community. Thus, it can be concluded that the network of 500 neurons contains regions with free neurons and regions with neurons organized into clusters. In other words, during the self-organization of the neural network, a network with internal large-scale structure emerges.

The proposed parameter of the number of communities with a given number of neurons allows evaluating and describing the internal structure of a self-organized network.

7 CONCLUSIONS

This paper provides a theoretical justification for applying the Louvain method to cluster three-dimensional self-organizing systems using neural networks as an example. A modification of the modularity function is proposed that considers the effective distance between elements, measured in units of axon and dendrite length, which more adequately reflects the actual connectivity structure. The algorithmic phases are described, and formulas for modularity gain in the spatially weighted case using effective distances are given. Quality metrics specific to three-dimensional data are discussed.

Future research may focus on developing methods for automatic parameter selection, accounting for directed connections, and integration with neuronal functional activity data. The proposed mathematical framework can be used in the development of software tools for analyzing large volumes of neuroanatomical data.

ACKNOWLEDGMENTS

This work was supported by the VEGA project No. 1/0423/23, "Experimental research and simulation of dynamic properties of composite structural elements manufactured by 3D printing" and the KEGA project No. 012ZU-4/2025, "Implementation of methods and proposal of educational modules for the design of automated and robotic production systems with an aspect related to safety and risk elimination".

REFERENCES

[Blondel 2008] Blondel, V.D., Guillaume, J.L., Lambiotte, R. and Lefebvre, E. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 2008, Vol. 2008, No. 10, P10008.

[Coranic 2023] Coranic, T. and Mascenik, J. Experimental Measurement of Dynamic Characteristics of Structural Units. Processes, 2023, Vol. 11, No. 12, 3333. DOI: 10.3390/pr11123333

[Demcak 2024] Demcak, J., Zidek, K., Krenicky, T. Digital Twin for Monitoring the Experimental Assembly Process Using RFID Technology. Processes, 2024, Vol. 12, No. 7. DOI: 10.3390/pr12071512

[Ding 2024] Ding, L., Li, C., Jin, D., Ding, S. Survey of spectral clustering based on graph theory. Pattern Recognition, 2024, Vol. 151, 110366.

[Fortunato 2010] Fortunato, S. Community detection in graphs. Physics Reports, 2010, Vol. 486, No. 3-5, pp. 75-174.

[Kuric 2022] Kuric, I., et al. Approach to Automated Visual Inspection of Objects Based on Artificial Intelligence. Applied Sciences, Vol. 12, Issue 2, DOI: 10.3390/app12020864

[Leicht 2008] Leicht, E.A. and Newman, M.E.J. Community structure in directed networks. Physical Review Letters, 2008, Vol. 100, No. 11, 118703.

[Meunier 2010] Meunier, D., Lambiotte, R., Bullmore, E.T. Modular and hierarchically modular organization of brain networks. Frontiers in Neuroscience, 2010, Vol. 4, p. 200.

[Murtagh 2012] Murtagh, F. and Contreras, P. Algorithms for hierarchical clustering: an overview. WIREs Data Mining and Knowledge Discovery, 2012, Vol. 2, No. 1, pp. 86-97.

[Newman 2004] Newman, M.E.J. and Girvan, M. Finding and evaluating community structure in networks. Physical Review E, 2004, Vol. 69, No. 2, 026113.

[Reichardt 2006] Reichardt, J. and Bornholdt, S. Statistical mechanics of community detection. Physical Review E, 2006, Vol. 74, No. 1, 016110.

[Saga 2011] Saga, M. Bednar, R. Vasko, M. Contribution to Modal and Spectral Interval Finite Element Analysis. Vibration Problems ICOVP, 2011, Vol. 139, pp. 269-274. DOI: 10.1007/978-94-007-2069-5_37

[Stepanov 2014] Stepanov, P. Nikitin, Y. Diagnostics of Mechatronic Systems on the Basis of Neural Networks with High-Performance Data Collection. In: Brezina, T., Jablonski, R. (eds). Mechatronics, 2013, pp. 433-440. https://doi.org/10.1007/978-3-319-02294-9_55

[Stepanyants 2002] Stepanyants, A., Hof, P.R., Chklovskii, D.B. Geometry and structural plasticity of synaptic connectivity. Neuron, 2002, Vol. 34, No. 2, pp. 275-288.

CONTACTS:

VICTOR Yu. TRUBITSIN, Doctor of Physical and Mathematical Sciences
Chief Research Scientist Machine Learning and Big Data Processing Laboratory
Udmurt Federal Research Center of the Ural Branch of the RAS, Institute of Mechanics
T. Baramzina Str 34, 426067 Izhevsk, Udmurtia
E-mail: tvy@udman.ru

MILAN SAGA, Dr. h. c. prof. Dr. Ing.
University of Zilina, Faculty of Mechanical Engineering, Department of Applied Mechanics
Univerzitna 8215/1, 010 26 Zilina, Slovakia
E-mail: Milan.Saga@fstroj.uniza.sk

ALEKSANDR I. KORSHUNOV, Doctor of Technical Sciences, Professor
Udmurt Federal Research Center of the Ural Branch of the RAS, Institute of Mechanics
T. Baramzina Str 34, 426067 Izhevsk, Udmurtia
E-mail: kai@udman.ru

ZUZANA SAGOVA, Assoc. prof. Ing. PhD.
University of Zilina, Faculty of Mechanical Engineering, Department of Automation and Production Systems
Univerzitna 8215/1, 010 26 Zilina, Slovakia
E-mail: zuzana.sagova@fstroj.uniza.sk

LICENSE CREATIVE COMMONS: