UDC 519.715

The inverse problem of Q-analysis of complex systems structure in cyber security

V. I. Polutsyganova[1], S. A. Smirnov[1]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Educational and Research Institute of Physics and Technology*

Abstract

It is considered the inverse problem of Q-analysis. In the course of the research, an algorithm for the recovery of simplicial complexes from elementary simplex using local maps and a structural tree was developed. This algorithm will reduce the amount of data stored and improve the management process if the simplicial complex describes a real big complex system wich it can imagine cyber security system.

Keywords: Q-analysis, complex systems, system structure, simplicial complex, inverse problem, cyber security

Introduction

Q-analysis was first described by Atkin [1]. This approach has been used to formalize various problems in social sciences. It is now developed in the context of the application of hypergraphs to the analysis of various complex systems. The formalism of the simplicial complexes intervenes in a very wide range of applications. The approach allows to analyze the structure of a complex system, taking into account the variety of connections between its components. In order to use Q-analysis, it is necessary to formalize the description of the system in terms of the simplicial complexes. Such complexes are also visible in cyber security systems.

The simplicial complex is called the finite set of simplexes, which satisfies the following conditions [5]:

1. Together with any simplex, its facets of all dimensions belong to this set;
2. Two simplexes can intersect (have common points) only along the entire facet of any dimension and thus only on one facet.

The simplicial complex is a multidimensional topological structure, so it better describes the relationship between parts of the system and its elements than graphs. Also the concept of q-connectivity is used in this context, that is, the level of connectivity between simplexes in a complex. The complexes described in this work are well distributed to the search system for various vulnerabilities. At each level of such connectivity, a simplex complex can be described as a set of chains, that is, a graph whose nodes are simplexes and edges are connections which dimension is not less than a given level of connectivity (in current level). Such graphs are also called local maps [3] because they reflect the internal structure of the chains.

Definition 1. The local map of a simplicial complex is called the graph of the binary relation q-adjacent whose nodes are the simplexes of dimension k> q, and the edges correspond to the q-relation between them.

Definition 2. Q-adjacency is called the binary relation between simplexes in a simplicial complex, which occurs at the cross section of two simplexes and whose dimension is greater than or equal to "q" [7].

Definition 3. A Q-link is a binary relation that occurs when a q-junction is transitively closed in a simplicial complex [7].

Basing on the imitation of circuits of adjacent levels of connectivity, it is possible to build a corresponding tree [2]. The nodes of a Q-tree are chains of simplexes that are connected at a certain level of connectivity, and the depth levels of the trees are the same q-levels of connectivity. That is, a structural analysis of a complex system (a direct problem) results in a structural tree and local maps of the corresponding complex. The Atkin structural vector and the eccentricities of the simplexes are only a small part of the complete information about the structure of the system set up in this way. This method allows to quantify the set of connections between simplexes that form chains in a complex, which is important for the study of the structure of the system as a whole [4].

The question naturally arises about the inverse task. Suppose, that a system has a large number of elements (not simplexes), they make up certain subsystems that can be considered as simplexes or circuits of simplexes. Then the simplicial complex represents the structure of connections in the system, that knowledge gives the opportunity to visit the whole system. Working with large systems, we would like to reduce the amount of information on their structure for efficient storage and processing. Merging into a simplex of a vertex can be considered as a single vertex on some scale of representation of a simplicial complex and it creates opportunities for more efficient (scaled) processing of system related information.

In addition, management systems may encounter tasks related to individual subsystems, when it is advisable to work only with certain simplexes, without affecting the complex as a whole, that is, maintaining their external connections. Therefore, you need not only to decompose the system (which allows you to do Q-analysis), but also to synthesize it to its original state as needed.

1. Problem discussion

In the studied sources, the analysis is considered for the application and use of the results of its work [8-10]. Studies of system and network topologies are based, as a rule, on their graph representation, when bijective correspondences are established between the sets of system modules and graph vertices, as well as between the sets of communication lines and graph edges. The set of studied/optimized topological properties of the cyber system also has a bijective image in the description of the graph and in the set of its characteristics, which allows for a comparative analysis using the characteristics of the descriptions of the corresponding graphs. However, due to the combinatorial nature of the problems of analysis, synthesis, and reconfiguration of the topology of computer cyber systems and communication networks, their solution is traditionally based on exhaustive, heuristic, or stochastic approaches. At the same time, a non-linear increase in the number of states in large systems exacerbates the problem of achieving an acceptable compromise between the quality of control processes and their relevance, leads to the emergence of emergency situations and to unstable dynamics of the systems. The main drawback of the traditionally used matrix-list descriptions of system graphs lies in the low-level nature of the binary relations between vertices specified by these descriptions, while the routes and cycles used to assess the quality of structures, by definition, are multi-place relations on the set of graph vertices[12].

This provides some (incomplete) information about all nodes, simplexes, chains in the complex. As stated earlier, this is a very large amount of data. Sometimes it is enough to know only the set of simplexes and the levels of connectivity between them. This study proposes a new approach to the analysis, synthesis, and control of structurally complex systems in cyber security based on the use of their structural tree and local maps. We will work with subsystems or simplexes and circuits, store information about connections, that is, trees and local maps, and, if necessary, restore the overall structure of the system. So far, the formulation of tasks in this form is unknown to the authors. It is possible only with the exception of the general meta-rule of “act locally, think globally”. The main purpose of this study is to prove the possibility of correct restoration of a structurally complex system in cyber security (simplicial complex) by information about its structural tree and local maps.

In addition, this methodology can be used to build complex systems with a defined structure of connections, but unknown nodes. Such problems are typical in linear systems, when the analysis of the system itself is not difficult, and the construction of a system with given characteristics is not a trivial task.

In this article, we will analyze and synthesize not a large system, but with non-trivial binary relationships. This will allow you to use this methodology not only for compact data storage, which will reduce data load, but also allow you to synthesize new systems only on known attributes.

2. Methodology

To solve this problem, an appropriate synthesis method has been developed, that is, a certain algorithm for the recovery of the simplicial complex has been constructed. The basic idea behind a recovery is that local maps give us the topology of connections at each level of connectivity, and the structural tree defines the rules of inheritance between chains when the level of connectivity changes. More details about the algorithms for constructing local maps and structural tree can be found in the article [8]. Therefore, it is well known how and what simplexes are related. But there is a problem: how or which nodes / edges / facets are connected by simplexes. To do this, you need to enter a certain equivalence ratio [5]. Therefore, it was hypothesized that there is a certain isomorphism that, if necessary, will return the simplex in the way it was built into the complex.

A graph can exist in different forms having the same number of vertices, edges, and also the same edge connectivity. Such graphs are called isomorphic graphs. Note that we label the graphs in this chapter mainly for the purpose of referring to them and recognizing them from one another.

Two graphs G1 and G2 are said to be isomorphic if:

- their number of components (vertices and edges) are same;
- their edge connectivity is retained.

If G1 ≡ G2 then:

- |V(G1)| = |V(G2)|
- |E(G1)| = |E(G2)|

Degree sequences of G1 and G2 are same.

If the vertices {V1, V2, .. Vk} form a cycle of length K in G1, then the vertices {f(V1), f(V2),… f(Vk)} should form a cycle of length K in G2.

All the above conditions are necessary for the graphs G1 and G2 to be isomorphic, but not sufficient to prove that the graphs are isomorphic.

- (G1 ≡ G2) if and only if (G1− ≡ G2−) where G1 and G2 are simple graphs.
- (G1 ≡ G2) if the adjacency matrices of G1 and G2 are same.
- (G1 ≡ G2) if and only if the corresponding subgraphs of G1 and G2 (obtained by deleting some vertices in G1 and their images in graph G2) are isomorphic.

Given graphs G and H, an isomorphism from G to H is a bijection $\phi : V(G) \to V(H)$ such that $\phi(g)$ is adjacent to $\phi(g0)$ if and only if g is adjacent to g0 . When such an isomorphism exists, we say that G and H are isomorphic and write G =∼ H. The notion of isomorphism is central to a broad area of mathematical research encompassing algebraic and structural graph theory, but also combinatorial optimization, parameterized complexity, and logic. The graph isomorphism (GI) problem consists of deciding whether two graphs are isomorphic. It is a question with fundamental practical interest due to the number of problems that can be reduced to it. Additionally, the GI problem has a central role in theoretical computer science as it is one of the few naturally defined problems in NP which is not known to be polynomial-time solvable or NP-complete. While there is a deterministic quasipolynomial algorithm for the GI problem, regardless of its worst case behavior, the problem can be solved with reasonable efficiency in practice. In relation to the context of this paper, it is valuable to notice that the discussion around graph isomorphism has branched into the analysis of many equivalence relations that form hierarchical structures. Prominent instances are, for example, cospectrality, fractional isomorphism, etc[11].

The search for isomorphism, i.e. establishing the fact of isomorphism or non-isomorphism of graphs G1 and G2 belongs to the class of NP-hard problems and therefore, in practice, indirect isomorphism criteria are used that characterize the graph structure - numerical invariants. The invariant I of the graph G should be fairly easy to calculate, and if the graphs G1 and G2 are isomorphic, then it must necessarily be I (G1) = I (G2), but the converse statement is generally not true: the equality of the invariants of two graphs does not guarantee their isomorphism. logically substantiated classification of graph invariants will allow grouping different invariants when comparing their ability to distinguish non-isomorphic graphs, as well as identifying “empty” classification groups in which there are no invariants known today; such groups provide a "hint" for constructing new invariants.

The first classification feature divides the graph invariants into two classes: scalar and vector. A scalar invariant is an integer or real number that characterizes the graph as a whole, for example: the number of vertices and edges, diameter, cyclomatic and chromatic numbers, Wiener and Randic indices, etc. It is obvious that the ability of such invariants to distinguish between non-isomorphic graphs cannot be high, and it is likely that sufficiently large sets of non-isomorphic graphs can have the same values of the scalar invariant. Vector invariants contain more information about the structure of the graph and, therefore, have a higher ability to distinguish non-isomorphism in the analyzed sets of graphs. Note that instead of the term “vector invariant”, it is more correct to use the term “invariant in the form of a multiset”, since, firstly, the order of the mutual arrangement of the elements of the vector invariant does not matter and, secondly, it may contain repeating elements.

For Vertex Scalar invariants of the group, each element of the multiset is a scalar numerical characteristic of some graph vertex, showing its specific role in the overall graph structure and depending on the local "device" of the graph in the vicinity of this vertex. This feature could be:

- degree (valence) of the vertex v, which is numerically equal to the cardinality of the set of vertices adjacent to it N(v): δ(v)=|N(v)|; the degree of the second order of the vertex v, which is numerically equal to the number of vertices, the shortest path to which from the given vertex v is exactly equal to 2: δ2 (v)=|{u∈V, dist(u,v)=2}|; [13]
- eccentricity of vertex v, equal to the shortest distance to the vertex farthest from vertex v, expressed in the number of edges;
- resistive peak eccentricity v, which is based on the idea of measuring distances between pairs of peaks using analogies from the theory of electrical circuits;
- coefficient of local clustering of the vertex v, which shows how close the subgraph H = (N(v), D) of the graph G, generated by the set of vertices N(v) adjacent to v, is close to the complete graph.

In the Vertex MultiSet invariants of the group, each element of the multiset is a vector characteristic of the vertex v of the graph G, showing its “role” in the overall structure of the graph relative to other vertices. There are two possible approaches to constructing such vector characteristics (i.e., multisets):

- "near" - the elements of the multiset are scalar characteristics of vertices from the "near environment", i.e. vertices adjacent to v;
- “far”, in which the elements of the multiset are the distances (geodesic or resistive) between the current vertex v and all other vertices.

The Pairs of Vertex invariants of a group can be called "matrix" since their multisets are composed of elements of an n × n matrix, each element of which is a numerical characteristic of some relation between a pair of vertices. This is usually the distance (geodesic or resistive). For undirected graphs, it suffices to consider the upper triangular part of such a matrix of size n(n – 1)/2. Thus, taking into account line graphs, the PV group includes 4 invariants (see Table 1) [13].

We define such an isomorphism as a reflection of the set (denote such a set C) of nodes of the simplex itself f: C → C. The essence of such a mapping is that it renames the nodes in the simplicial complex obtained after restoration according to the names that were in the original complex.

Hypothesis. For any two simplicial complexes in which the structure (number of nodes, simplexes, local maps, and structural tree) coincides, there is an isomorphism f: C → C that rearranges the nodes in the simplex so that the complexes become identical.

At this stage of the study there is no doubt about the existence of such an isomorphism, so we will use this assumption to describe the algorithm of complex recovery.

Therefore, to reconstruct a complex, it is necessary to have a structural vector in the concept of Q-analysis, that is, the number of symmetry chains at each q-connectivity level, local maps or incidence matrix between the simplexes at these levels, and the structural tree. In general, the recovery algorithm looks like this:

- Incoming data:
- Structural tree;
- Local maps;
- Sets of simplexes according to local maps;

The output of the algorithm:

- Simplicial complex of complex system.

Algorithm:

1. At the level of maximum dimension q = n, we use simplexes of this dimension, based on the "leaves" of the structural tree at that level. We add simplexes of this dimension.
2. For q = n-k> 0:

- If the link is inherited from the previous level, then we store it and do not process it at the current level.

- We form q-chains of (q + 1) -chains according to the inheritance structure given by the structural tree. We determine by the local map of the q-chain exactly which simplexes (q + 1) -chains will stick together.
If existing links of level q = n-k between the corresponding simplexes connect their face (simplex) by face (simplex) with dimension n-k, while maintaining the correspondence of the previous level connections. We "suppress" simplexes of dimension q = n-k. We check the presence of simlexes that are not separate chains.
- If all the links of the local map are processed, we proceed to the next (q + 1) chain of the structural tree.
- If all the chains of the level are worked out - we move to the level above: q = n-k + 1.

3. For the level q = 0:

- If there are level 0 connections in the local map, that is, simplex connections across the vertex, then we connect the corresponding simplexes across the vertex, so as not to break the other-level connections. We "simplify" simplexes of dimension q = 0, that is, they are not connected to other simplexes 0-simplex points.
- If there are no more 0-connections or 0-simplexes - the complex is built.

4. If necessary, we use isomorphism transformations (see Hypothesis) to obtain a symmetric complex corresponding to the system.
5. The algorithm is complete.

The definition of isomorphism will be specified in the course of the inverse algorithm. In the first step, a subset of nodes is formed by simplexes of maximum dimension. In the next step, the simplexes are glued in accordance with the scheme predetermined by local map, while the sub-complexes (facets) of the different simplexes are identified - each such sub-complexes corresponds to the real subsystem, which correspond as a common part of the higher-level subsystems. That is, the nodes that form it are identified with precision by permutation according to the subsystem connection information. Subsequent bonding may clarify the correspondence between the nodes of the complex (points) and the real elementary (unstructured) subsystems. Some nodes uniquely correlate with such subsystems, and some up to permutations that can characterize the internal stuctural symmetry of the system. In addition, one must not forget about "pasting" individual simplexes that are not separate chains, because they are parts of some simplexes of a larger dimension.

This algorithm makes it possible to recover a simplicial complex from a structural tree and local maps, but an important step is a well-defined isomorphism. Conflict may arise if several simplexes are connected through the same simplex but smaller dimension. But the local map at the appropriate level is always shown which simplexes are interconnected, and the isomorphism "returns" the simplexes so that the input complex of Q-analysis and the output of the recovery algorithm are identical.

3. Example

For a better understanding of the algorithm calculations, we give an example of a complex of small dimension and number of simplexes in the composition.

Let the structural vector be Q = {1, 3, 3, 4}. The structural tree has the form:

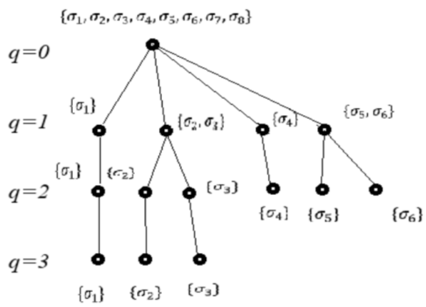

Figure 1: Structural tree for simplicial complex

The local maps for each level look like this:

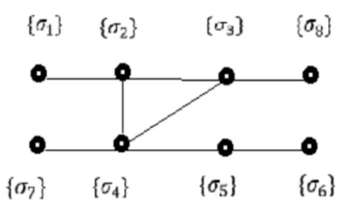

Figure 2: Local map for level q = 0

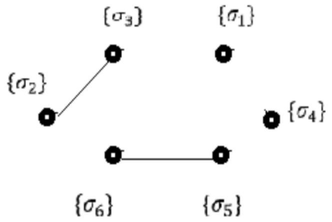

Figure 3: Local map for level q = 1

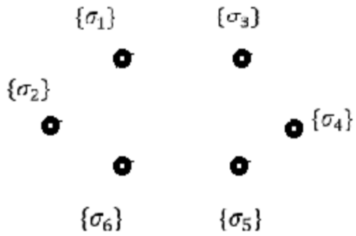

Figure 4: Local map for level q = 2

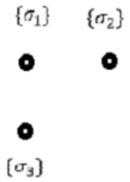

Figure 5: Local map for level q = 3

So, we begin the restoration of the simplicial complex:

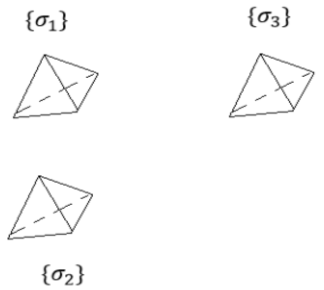

Figure 6: Simplicial complex for level q=0

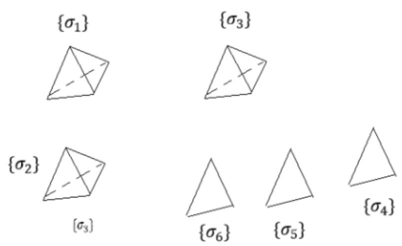

Figure 7: Simplicial complex for level q=1

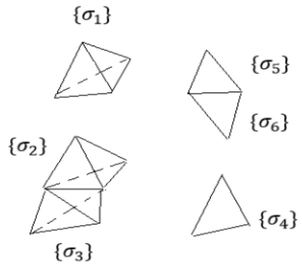

Figure 8: Simplicial complex for level q=2

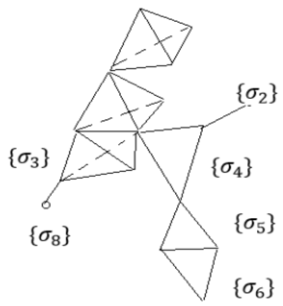

Figure 9: Simplicial complex for level q=3

When using the recovery algorithm, it is needs to remember that the simplicial complex presents some real system. In the process of joining a simplex to a complex, we considered that all nodes / edges / facets are equivalent, that is, by choosing a facet or node to attach another simplex, one can choose any existing one (except when one node / edges / facets with several simplexes). Therefore, after recovery, we use an isomorphism that will determine each vertex according to the connections that were in the real system. If it is not defined before the algorithm starts, then expert evaluation can be used to accurately determine the transformation. That is, specialists can provide information on how subsystems (in our terminology - simplexes) fit into the system (complex). Therefore, using isomorphism, the nodes will be determined according to the original simplicial complex.

Using the algorithm of recovery (or synthesis), we must not forget that this is some real system such as cyber security systems. Therefore, after "gluing" simplexes on vertices, faces and edges, it is necessary to apply the isomorphism defined by us, which places the subsystems in the correct orientation. In itself, isomorphism can be determined on the basis of expert judgment, ie specialists provide information on how subsystems (simplexes) are embedded in the system (complex).

Conclusions

Thus, the developed recovery algorithm makes it possible to reduce the amount of data on the structure of the system and improve management capabilities. The amount of data is reduced due to the fact that instead of storing all nodes and relations of the simplicial complex, it is sufficient to know only the simplexes and their dimensions, and, in addition, it is necessary to have a structural graph and local maps at each level of q-connectedness in order to be able to fully recover a structure of the system. The most important part of the synthesis of a complex is the determination of an isomorphism that will ensure that the complex being analyzed corresponds to the complex formed at the output of the recovery algorithm. Although this part is formed, it may not be sufficiently formalized, but there is no doubt about its feasibility (at least through busting).

Therefore, the inverse problem of Q-analysis can have both theoretical and practical value for structuring and managing complex systems. This method is described for structurally complex systems, which in turn are cyber security systems. Therefore, such a mathematically based method can be used to model system vulnerabilities and be used for cyber security.

Thus, the developed algorithm for the restoration of the simplex complex (or its synthesis) makes it possible to reduce the data stored and the development of new systems, if the task is to build a system with specified parameters. All simlexes must be preserved in the analysis task, and this methodology allows to store local simplex maps at each level of q-connectivity and a structural tree, which is sufficient to restore the structure of the system. The main part of this algorithm is an isomorphism, which makes it possible to unambiguously establish a correspondence between subsystems (simlexes) and their location in the system (complex).

Thus, the inverse problem of Q-analysis has practical and theoretical significance for the structuring, control and synthesis of complex systems.

References

[1] Atkin R. H. “Mathematical structure in human affairs”, Heinemann Educational Books, (1973); 143. doi: 10.1137/1018064.

[2] Pierre Mazzega, Claire Lajaunie and Etienne Fieux “Governance Modeling: Dimensionality and Conjugacy”, Graph Theory - Advanced Algorithms and Applications, 2018. doi: 10.5772/intechopen.71774.

[3] Beaumont J.R., Gatrell A.C. “An introduction to Q-analysis” Catmog 34, 1982. URL: https://alexsingleton.files.wordpress.com/2014/09/34-an-introduction-to-q-analysis.pdf.

[4] Duckstein, L. and Nobe, S. A. “Q-analysis for modeling and decision making”, European Journal of Operational Research, 103/3, (1997) 411–425. doi: 10.1016/S0377-2217(97)00308-1.

[5] Dobrovin B. A., Novikov S. P., Fomenko A. T. “Modern geometry. Method of the theory of homology ”, The main edition of the physics and mathematics literature, (1984) 344. URL: https://link.springer.com/book/9780387972718.

[6] Doktorova M. “Constructing simplicial complexes over topological spaces”, A thesis submitted dartmouth college, 2012. URL:https://digitalcommons.dartmouth.edu/cgi/viewcontent.cgi?article=1019&context=masters_theses.

[7] Jeffrey H. J. “Some structures and notation of Q-analysis”, Environment and Planning B Planning and Design, (1981). doi: 10.1068/b080073.

[8] Macgill S. M. “Q-analysis and transport: a false start?” Pergamon Journals Lid, 4, (1986) 271–281. doi: 10.1016/0191-2615(86)90043-3.

[9] Kachinsky A. B., Agarkova N. V. “The nature of the interconnectedness of the elements of the safety system for hydraulic structures”, System technology and information technology, 3, (2015) 72–83. URL: https://ela.kpi.ua/handle/123456789/14230.

[10] Johnson J. H. “A q-analysis of television programmes”, International Journal of Man-Machine Studies, 10, (1978) 461–479. URL: https://www. jstor .org /stable/24945189.

[11] Mančinska L., Roberson D. E., Šámal R., Severini S., and Varvitsiotis A. “Relaxations of graph isomorphism”, 44th International Colloquium on Automata, Languages, and Programming (2017) 1-14. doi: 10.4230/LIPIcs.ICALP.2017.76.

[12] Alexandrovich, Melent'ev “Метрика, цикломатика и синтез топологии систем и сетей связи.”, VI Международная конференция "Параллельные вычисления и задачи управления", (2012) 10-25. URL: https://www.researchgate.net/publication/344781086 _Metrika_ciklomatika_i_sintez_topologii_sistem_i_setej_svazi.
[13] Ivanchenko A.N., Ivanchenko K.N. “Analysis of invariants used for graph isomorphism testing and the principles of their systematization”, University News North Caucasian Region. Technical Sciences=Bulletin of Higher Educational Institutions, 4, (2021) 17-23. http://dx.doi.org/10.17213/1560-3644-2021-4-17-23/ (In Russ.)