SILESIAN UNIVERSITY OF TECHNOLOGY PUBLISHING HOUSE

SCIENTIFIC PAPERS OF SILESIAN UNIVERSITY OF TECHNOLOGY 2020
ORGANIZATION AND MANAGEMENT SERIESNO. 144

APPLICATION OF GAME TREE STRUCTURES IN THE PROCESS OF OBTAINING KNOWLEDGE

Adam DEPTUŁA

Opole University of Technology, Faculty of Production Engineering and Logistics, Department of Management and Production Engineering; a.deptula@po.edu.pl, ORCID: 0000-0003-3017-5998

Purpose: The purpose of the article is to present dependency graphs and parametric game tree structures as tools in generating and acquiring knowledge.
Design/methodology/approach: The thematic scope of work concerns the author's method of generating graphs and decision trees. The scope of work includes the analysis of computational assumptions of guidelines supporting knowledge generation and decision making.
Findings: The paper presents a method of generating game tree structures that allow to change the values of decision parameters in the issues of decision making and knowledge generation.
Research limitations/implications: Further development directions of the presented issues should be conducted in the field of computer implementation of the developed algorithms.
Practical implications: The most important in this regard will be the selection of the optimal programming environment with the possibility of installing the program in laboratory room systems for decision support and knowledge management for students. At a later stage, the use of tools in various problems in companies with a managerial and technical profile.
Social implications: The application of method can improve the quality of decision alignment and give access to problem solving of various technical problems.
Originality/value: A novelty is the use of parametric game tree structures as an alternative method to induction trees and multi-valued logical trees. Because game structures, unlike other methods, are built directly from the graph.

Keywords: dependence graph; game tree structures, artificial intelligence, optimization.

Category of the paper: Research paper, viewpoint.

1. Introduction

Decision support system (SWD) is a very defining term. It was popularized by (Keen, and Scott Morton, 1978) although the first works on this subject were created over ten years earlier. Usually, it is defined very generally. For example: a decision support system is an IT system that provides information in a given field using analytical decision models with access to databases in order to support the effective operation of decision-makers in a comprehensive and poorly structured environment (Klein, and Methlie, 1992). In turn, (Spraque, and Carlson, 1982) have defined decision support systems as interactive, computer-based systems that help decision-makers use data and models to solve poorly structured problems. Decision support systems enable creating models, building scenarios, and using numerical algorithms. By combining them with the types of decisions typical of artificial intelligence systems, the classic SWD model can be extended in the following directions:

- Expert advice in a specific area of the problem – explaining the results of inference,
- Intelligent decision support – help in formulating queries,
- Intelligent support in the process of building a model.

Many models have been developed over the last years regarding the rules of construction of computer-aided decision support systems in management. Decision making often comes down to a process of limited rationality in which choices are made on the basis of structural rules with simultaneous access to limited information. Evolution in management support systems has resulted in the use of an area involving solutions based on artificial intelligence. These actions enabled automation of the assessment of proper management based on specific rules, and as a consequence led to the emergence of Intelligent Decision Systems and Expert Systems. Artificial intelligence in the field of management should serve to improve the work of the managerial staff by providing information that facilitates the decision-making process. The use of knowledge in the form of knowledge databases has allowed expanding the capabilities of classic decision support systems. Thus, a new SWD class called hybrid systems was created.

There are many definitions of the term "hybrid system". These definitions should be divided into two categories: first, classic – describes complex systems intended for processing analog and digital signals; second, semantic – defines the structure of the system using many methods. In the second approach, therefore, a hybrid system is called a system that integrates any number of interacting, heterogeneous approaches to data processing (in particular knowledge) (Iglesias et al., 1996; Chung- Min et al., 2013; Kuo- Hao, 2014).

The aim of hybrid systems in the field of decision support is to achieve synergy between classic SWD and artificial intelligence methods. Such a system should combine modelling possibilities, typical for SWD, with symbolic processing appropriate for artificial intelligence. Classic decision support systems had direct access to databases in the company and often assumed their own data sets necessary for their functioning. We define the hybrid system as an IT tool supporting the process of making complex and poorly structured decisions within a given class, enabling: supporting the analysis of the decision-making process; designing better decision-making learning tools; development of easy dialogue; supporting the selection of system components to create solutions (Kumar Kar, 2015). In addition to the classical view of the decision-making process, there is also an understanding of this process as knowledge-based.

2. Decision-making process supported by knowledge

This approach assumes that a decision consists of fragments of knowledge describing the essence of action that is necessary to be taken. Along with the development of information technology, the nature and possibilities of optimization methods and decision support systems are changing. There is a wide range of research on the development of methodologies supporting decision-making processes and management control, design methodologies and systems of varying scale of complexity involving artificial intelligence.

A large role can be attributed to IT systems of management, in which computer technology is used to solve management problems. One of the directions of development strengthening the role of information systems is to combine different methods of processing, inference and seeking the knowledge developed separately in the framework of artificial intelligence into one coherent hybrid advisory system. There are two general approaches to creating such hybrid systems: *CI* – Computational Intelligence and *SC* – Soft Computing (Rudas, 2004).

In the case of *CI*, the criteria are filled in by systems incorporating the following methods: neural networks, genetic algorithms, fuzzy logic, evolutionary programming and life simulations. The second approach (SC) is another developmental step in the theory of building hybrid systems using artificial intelligence methods. It assumes that created advisory systems process additional structured information, and thus inform about a specific structure, hierarchy and semantics. The *SC* approach focuses on the creation of knowledge processing systems and uses elements of machine learning theory, chaos theory and probabilistic reasoning. Analysing the state of research related to decision support and control systems, one can notice the continuous development of computational intelligence methods, i.e.: neural networks, fuzzy logic, rough sets, expert systems and a combination of these methods, e.g. in hybrid systems. There are also multi-criteria decision support methods described by (Trzaskalik, 2014). The work describes in detail the methods, including: AHP, ANP, ELECTRE, PROMETHEE, verbal methods, TOPSIS, BIPOLAR, and interactive methods in terms of risk.

At the same time, there is a continuous propagation of the state of knowledge in the so-called machine learning. It covers the problems of constructing systems whose operation increases along with the experience represented by a set of teaching examples. In this area, particular attention should be paid to trees and graphs (Pijls, and De Bruin, 2001; Horzyk, 2012; Qiao et al., 2016; Iordanov, 2010).

Among many decision trees used, inductive trees and multi-valued logic trees are very applicable (Petrović, and Pale, 2017; Fernando et al., 2012; Quinlan, and Rivest, 1989). There are numerous author/co-author works presenting the use of inductive decision trees, multi-valued logical trees (also with coefficients) and multi-valued logic equations as a decision support tool, discrete optimization and determining the importance of decision variables and knowledge generation, for example (Deptuła et al., 2016; 2018a; 2018b; 2018c).

Inductive decision trees and multi-valued decision trees – the Quine-McCluskey algorithm for the minimization of logic functions

In the generation of inductive decision trees, a series of tests are performed in a specific order, where the data samples as a result of each subsequent test are separated into subgroups of greater "purity", i.e. containing more and more samples from a given class only. The expansion of the tree is associated with an increase in entropy. The entrance to the next floor of the induction tree corresponds to the n- state in analytical analysis. All floors of the induction tree correspond to all possible states of n in the computational analysis.

In problems of induction of decision trees, entropy determines the most significant attribute. The information in the set of teaching examples is equal to:

$$I(E) = -\sum_{i=1}^{|E|} \frac{|E_i|}{|E|} \cdot \log_2\left(\frac{|E_i|}{|E|}\right) \tag{1}$$

where:

E – a collection of teaching examples,

$|E_i|$ - the number of examples of the *i-th* object,

$|E|$ - the number of examples in the training set E.

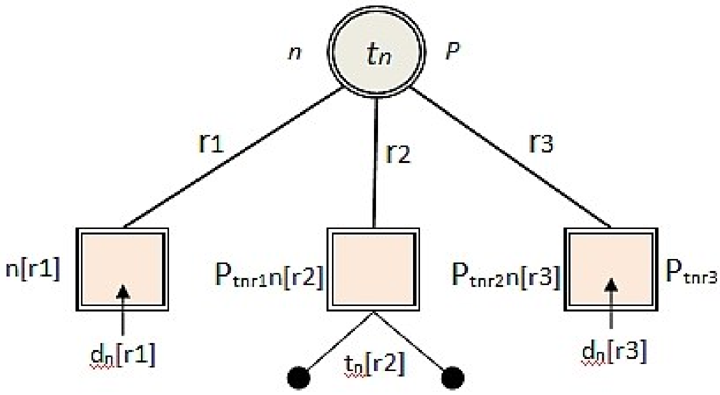

Figure 1. A sample “query by example” to the decision making system.

The Quine–McCluskey algorithm makes it possible to find all prime implicants of a given logic function that is there in a shortened alternative normal form SAPN. The terms of incomplete gluing and elementary absorption have the main role in the search for prime implicants and are used for the APN of a given logic function. The following transformation is called the consensus operation:

$$Aj_o(x_r) + \ldots + Aj_{m_r-1}(x_r) = A \tag{2}$$

where: $r = 1, \ldots, n$ and A is a partial elementary product, the literals of which possess variables belonging to the set $\{x_1, \ldots, x_{r-i}, x_{r+i}, \ldots, x_n\}$:

The following transformation is called the operation of reduction:

$$Aj_u(x_r) + A = A \tag{3}$$

where: $0 \le u \le m_r - 1$, $1 \le r \le n$, and A is a partial elementary product, the literals of which possess variables belonging to the set: $\{x_1, ..., x_{r-1}, x_{r+1}, ..., x_n\}$.

For example, multiple-valued logical function $f(x_1, x_2, x_3)$, where $x_1, x_2, x_3 = 0, 1, 2$, written by means of numbers (Canonical Alternative Normal Form): 100, 010, 002, 020, 101, 110, 021, 102, 210, 111, 201, 120, 022, 112, 211, 121, 212, 221, 122, there is one MZAPN (Minimal Complex Alternative Normal Form) after the application of the Quine-McCluskey algorithm based on the minimization of individual partial multi-valued logical functions having 13 literals (Fig. 2) (4):

$$f(x_1, x_2, x_3) = j_o(x_1)\big(j_o(x_2)j_2(x_3) + j_1(x_2)j_o(x_3) + j_2(x_2)\big) + \\ + j_1(x_1) + j_2(x_1)\big(j_o(x_2)j_1(x_3) + j_1(x_2) + j_2(x_2)j_1(x_3)\big). \tag{4}$$

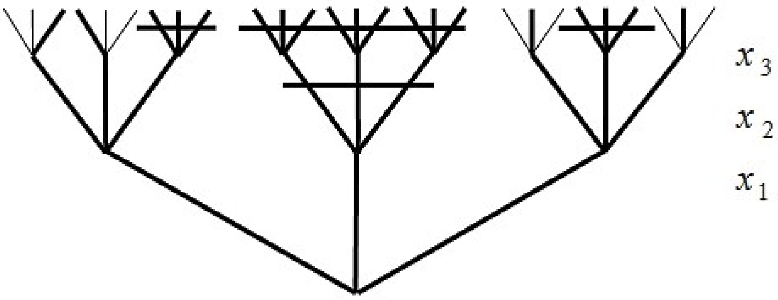

Figure 2. A multi-valued decision tree for the parameters x_1, x_2, x_3 with an appropriate layout of levels.

Entire transformation is described by the so-called Quine-McCluskey algorithm based on the minimization of individual partial multiple-valued logic functions. There are author's applications of inductive decision trees and logical multi-valued logical trees in the optimization of machine systems and many other areas of science as a tool for decision support systems. In particular, work (Deptuła, and Partyka, 2018b; Deptuła, 2014a) describes the use of multi-valued decision trees in the design and management methodology. The inductive tree determines the importance of the attribute's validity from the most important one placed in the root, by classifying an example. In work (Deptuła, A., and Deptuła, A.M, 2016), trees were used inductively to generate knowledge in the innovation process. However, in works (Deptuła, and Partyka, 2016; 2017; Deptuła, 2014b) induction trees have been used in technical problems.

Dependency graphs and parametric structures with cycles are a better generalization in machine learning than decision trees.

3. Dependency graph and game-tree structures

In the graphs of parametrically analysed dependencies there are the so-called 'bound' decisions. The results obtained in subsequent decisions depend on the initial decisions, which allows the creation of dynamic models. The decision process and the space of the possible states of the analysed system are described in parametrical from on the top of each graph.

A graph is an ordered pair *G*=(*X, R*), in which *X* is a finite set of elements called vertices of a graph, and R is a set of pairs $(x_i, x_j)(x_i, x_j \in X)$ called the edges of the graph. In the case of parametric graphs, the notation introduced by (Deptuła, and Partyka, 2011) defines the signs: *G* = (*Q, Z*), where Z is a set of pairs $(z_i, z_j)(z_i, z_j \in Z)$. Figure 3 shows an example of a parametric graph *G*.

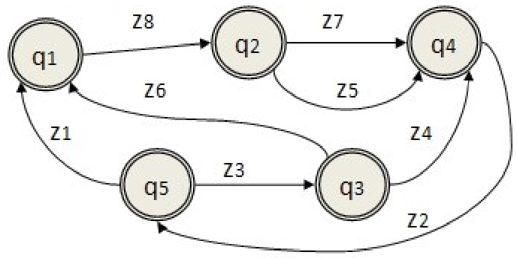

Figure 3. An oriented dependence graph.

The oriented game graph shown in Fig. 3 is composed of a set of vertices *Q*:

$$Q = \{q_1, q_2, q_3, q_4, q_5\} \tag{5}$$

and of a set of edges *Z*, that is an ordered pair of vertices

$$Z = \{z_1, z_2, z_3, z_4, z_5, z_6, z_7, z_8\} \tag{6}$$

The path in the *G*=(*Q, Z*) is the edge sequence $(z_{i_1}, z_{i_2}), (z_{i_2}, z_{i_3}), ..., (z_{i_{k-1}}, z_{i_k})$ in which for each $j \in \{2,3,...,k\} (z_{i_{j-1}}, z_{i_j}) \in R$ and vertices $q_{i_1}, q_{i_2}, ..., q_{i_k}$ are different pairs. Vertex q_{i_1} is called the beginning of the path, and the top q_{i_k} - the end of the road. As a result of a graph distribution from the chosen vertex, a tree structure with cycles is obtained in the first step and then, a general game tree structure is obtained. Each of them has an appropriate analytical formulation G_i^+ and G_i^{++}.

A game tree structure is a part of the systematic searching method. A start vertex q_i is chosen in the first step. The algorithm of graph distribution of dependence on parametric structures has been presented, among others, in the works (Deptuła, and Partyka, 2018a).

The graph in Figure 3 is distributed into 5 structures forming the D set:

$$D = \left\{G_{\{q1\}}^{++}, G_{\{q2\}}^{++}, G_{\{q3\}}^{++}, G_{\{q4\}}^{++}, G_{\{q5\}}^{++}\right\} \tag{7}$$

Figures 4-5 show the structures from each of the vertices: q_1, q_2, q_3, q_4, q_5 of the graph from Figure 3.

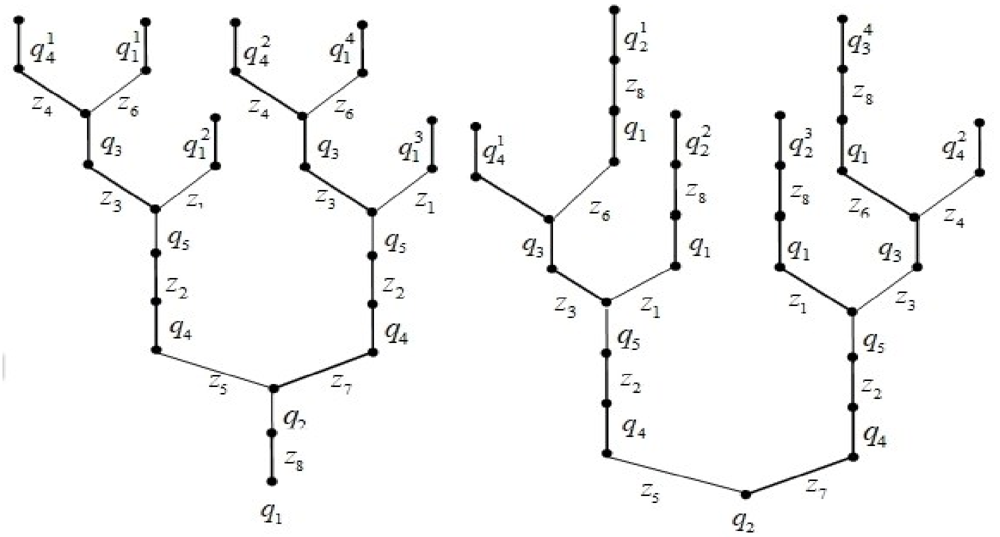

Figure 4. Game-tree structures with initial vertices: q_1, q_2

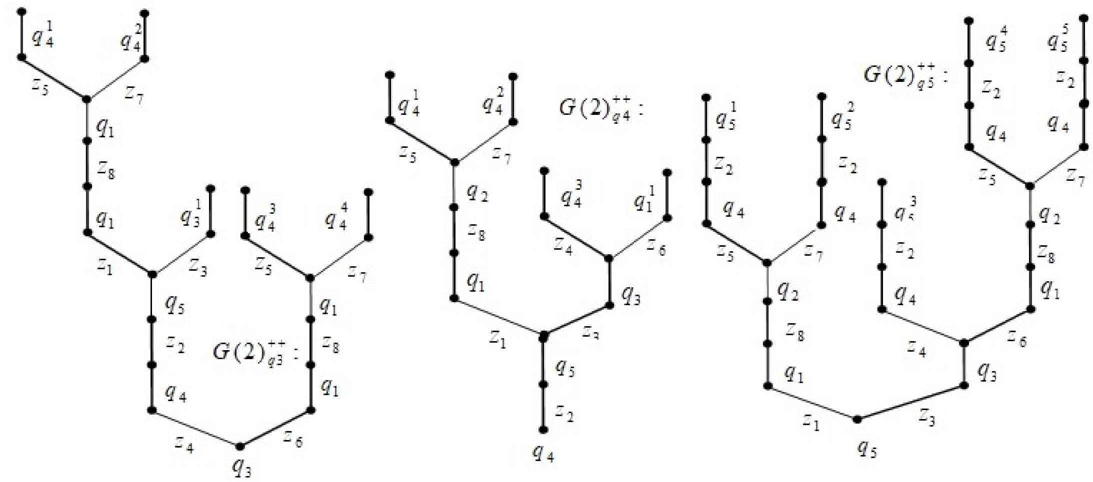

Figure 5. Game-tree structures with initial vertices: q_3, q_4, q_5

Each structure has its own analytical formula. For example, the structures in Figure 4 are described by the following formulas (8):

$$\begin{cases} G_{q3}^{++} = ({}^{0}q_3({}^{1}z_4q_4({}^{2}z_2q_5({}^{3}z_1q_1({}^{4}z_8q_2({}^{5}z_5q_4^1, z_7q_4^2)^5)^4, z_3q_3^1)^3)^2, \\ z_6q_1({}^{2}z_8q_2({}^{3}z_5q_4^3, z_7q_4^4)^3)^2)^1)^0 \\ G_{q4}^{++} = ({}^{0}q_4({}^{1}z_2q_5({}^{2}z_1q_1({}^{3}z_8q_2({}^{4}z_5q_4^1, z_7q_4^2)^4)^3, z_3q_3({}^{3}z_4q_4^3, z_6q_1^1)^3)^2)^1)^0 \\ G_{q5}^{++} = ({}^{0}q_5({}^{1}z_1q_1({}^{2}z_8q_2({}^{3}z_5q_4({}^{4}z_2q_5^1)^4, z_7q_4({}^{4}z_2q_5^2)^4)^3)^2, z_3q_3({}^{2}z_4q_4({}^{3}z_2q_5^3)^3, \\ z_6q_1({}^{3}z_8q_2({}^{4}z_5q_4({}^{5}z_2q_5^4)^5, z_7q_4({}^{5}z_2q_5^5)^5)^4)^3)^2)^1)^0 \end{cases} \quad (8)$$

Application of dependence graphs and game tree structures allows presenting a sequence of changes of arithmetic values of parameters in order to obtain the required behaviour of the system (for example machine system). Unlike traditional dependence graphs and tree classifiers, the dependence graph with game tree structures includes connection of importance rank of vertices (states) and height of the tree structure.

Game tree structures as tools for transforming knowledge in design methodology

There are many author/co-authored works describing the application of the parametrically-parametric structures in the transformation and retransformation of knowledge in design methodology. For example, works (Deptuła, and Partyka, 2018a; Deptuła et al., 2017) describe the use of parametric structures in the design and analysis methodology of automatic gearboxes.

Example 1

Figure 6 shows an example of a model drawing and functional diagram of an exemplary automatic transmission gearboxes (Deptuła, and Partyka, 2018a; Deptuła et al., 2017).

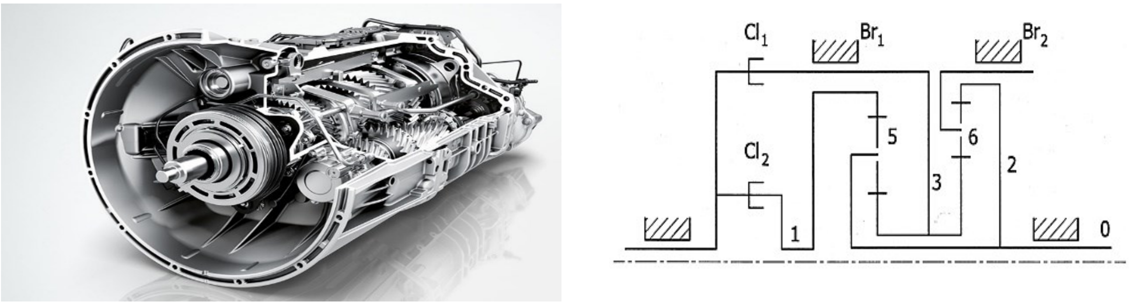

Figure 6. The Functional diagram of an exemplary automatic gearbox, where: *Cl* – clutch, *Br* – brake.

In the case of Hsu rules, the graph is built according to the following rules: geometrical dimensions are omitted and kinematic pairs are considered: rotary, "planet-yoke" and meshing. A set of equations describing transmission kinematics is generated. Figure 7 shows a dependence graph of the analysed transmission and game structures G_{q1}^{+} with cycles from the initial vertex q_1.

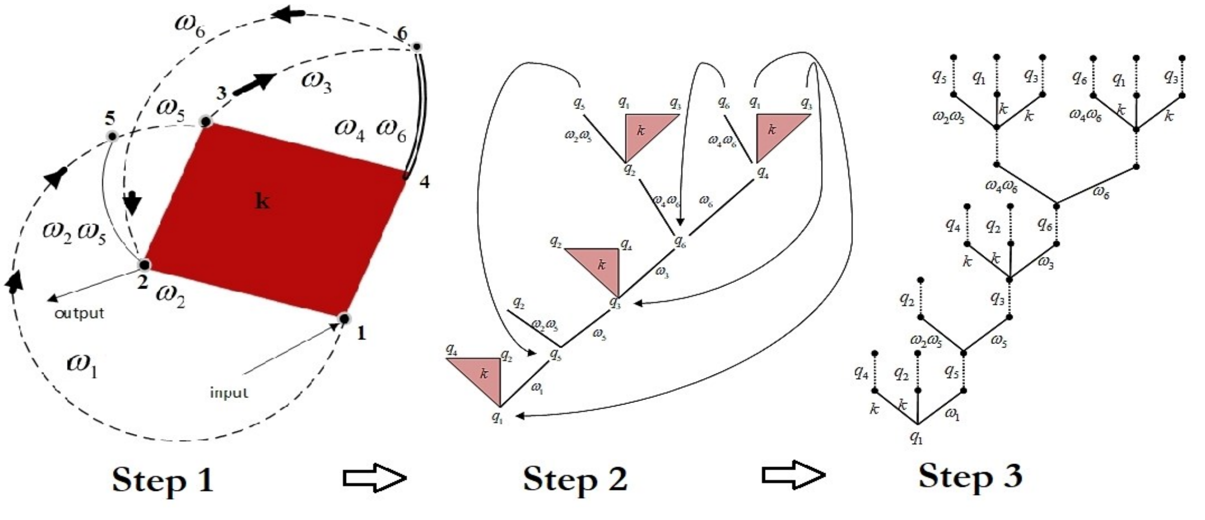

Figure 7. Functional Hsu graph with the path from the input to the output for the transmission in terms of signal flow graph (step 1), game structures G_{q1}^{+} with cycles (step 2) and without cycles from the initial vertex (step 3).

The usage of graph-based methods for analysis of the exemplary gear with a closed internal loop has been described and performed step by step giving a detailed explanation to all activities. The methods are relatively simple, algorithmic and general. This confirms the usefulness of these methods for checking of correctness of gear analysis.

4. Game-tree structures as tools for obtaining knowledge

The deduction structure for the majority of expert systems has been proposed as an inference rule. In this type of systems, the inference structure is created by a network of transition operators, leading from nodes describing initial situations to final conclusions (Kotsiantis, 2013).

The elemental link in parametric parameters is the relation of implication, that is the rule of the deduction of "IF, THEN" or IF. The inference mechanism for a given parametric structure and the initial state can be described using the scheme in Figure 8.

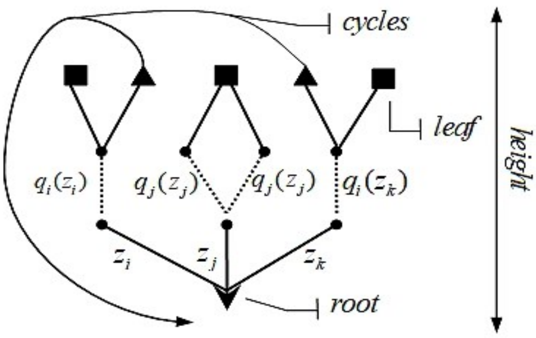

Figure 8. Functional generating a graph of the inference function (parametric structure).

After checking the states and rules, the path (path) is selected on the structure that was considered optimal (or the working rule). The popular tree search algorithm can be used to search (calculate) the parametric structure. In particular, a special coefficient of complexity index has been developed.

4.1. Complex coefficient of complexity for game-tree structures

The Complex coefficient of complexity $L\,G_i^{++}(\ )$ is applied to the description of the shape and properties of the game-tree structures previously obtained by the decomposition of the dependency graph on the *i*- th vertex. The level of structure's complexity is determined by the complex coefficient of complexity $L\,G_i^{++}$:

$$L^K(G_i^{++}) = \sum_{w\in W(L)} \frac{d(w_i)}{h(w_i)+1} + \frac{L}{\sum_{l\in L} \frac{1}{h_{l_i}}} \tag{9}$$

where:

$L^K G_i^{++}$ ( ) – complex coefficient of the complexity of structure G_i^{++},

w_i - *i* - *i-th* node,

$d(w_i) = \deg(w_i)$ - degree of *i-th* node branching (amount of node branching),

$h(w_i)$ - distance from the *i-th* node root

$W(L)$ - set of all nodes,

L – number of leaves for the i-th node branching $\deg(w_i) \geq 2$ ( ),

h_{l_i} - amount (complexity) of the *i-th* leaf.

Figure 9 shows an example game-tree structure with different L and L^K coefficients.

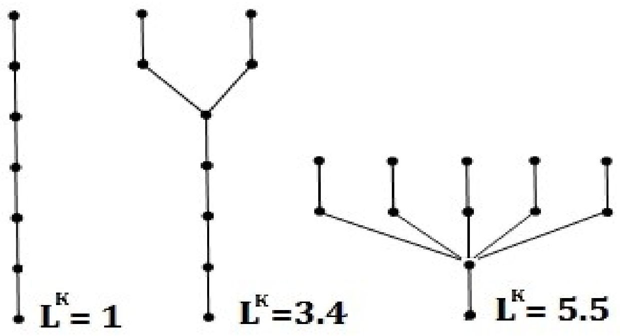

Figure 9. Tree-game structures with different complexity L^K coefficients.

For each of the game-tree structures complex coefficient can be calculated.

For example, for the parametric structure G_2^{++} in Figure 7, the value of the coefficient is (12):

$$L^K(G(2)_{q1}^{++}) = \left(\sum_{w \in W(L)} \frac{d(w_i)}{h(w_i)+1} + \frac{L}{\sum_{l \in L} \frac{1}{h_{l_i}}} \right) = \left(\frac{1}{0+1} + \frac{2}{\frac{1}{3}+\frac{1}{3}} \right) + \left(\frac{2}{2+1} + \frac{2}{\frac{1}{2}+\frac{1}{2}} \right) + \left(\frac{2}{6+1} + \frac{2}{\frac{1}{2}+\frac{1}{2}} \right) + \tag{10}$$

$$\left(\frac{2}{8+1} + \frac{2}{\frac{1}{2}+\frac{1}{2}} \right) + \left(\frac{2}{6+1} + \frac{2}{\frac{1}{2}+\frac{1}{2}} \right) + \left(\frac{2}{8+1} + \frac{2}{\frac{1}{2}+\frac{1}{2}} \right) = 15{,}69,$$

For each of the game-tree structures complex coefficient can be calculated.

4.2. Parametrically structured structures in the analysis of the production process

Game-tree structures can be used in the analysis of decision-making processes in manufacturing enterprises. The purpose of the distribution of the graph will then be to determine which of the selected stages of the production process are the most important from the point of view of the adopted criteria.

Example 2

The example used for the analysis concerns a simplified scheme for selecting a variant of production order. In the description, only the most important stages of the decision-making process were selected. The analysed company produces metal products. The decision problem concerns the alternative method of production of metal screws. The values given in the example refer to bulk quantities, which are not described in detail.

The vertices and edges of the dependency graph mean:

Q1 – cost: <z_1 - low; z_2 - high>;
Q2 – execution time: <z_{13}> - selected for a given order (specified);
Q3 – completion of production order: <z_7> complete order;
Q4 – risk assessment <z_4> acceptable risk;
Q5 – transfer of the final product: <z_5 - the product is not suitable for sale; z_6 - product for sale>;
Q6 – quality control: <z_{11}> - product defects;
Q7 – order evaluation: <z_9 - for acceptance; z_8 - for correction; z_3 - for further evaluation>;
Q8 – production corrections: <z_{10} - increase in the number of working hours>
where: <z_1 - from 1 to 5 PLN per pack; z_2 - from 5 PLN to 7 PLN per pack; z_{13} - less than one day; z_5 - acceptable number of defects => 2% from the size of the production lot; z_6 - acceptable number of defects = <2% from the size of the production lot; z_{10} - increase of working hours to 2 hours on commission; z_4 - risk assessment for low or medium risk on a three-level scale: low, medium, high.

Figure 10 presents a dependency graph describing the process of selecting a variant of production order.

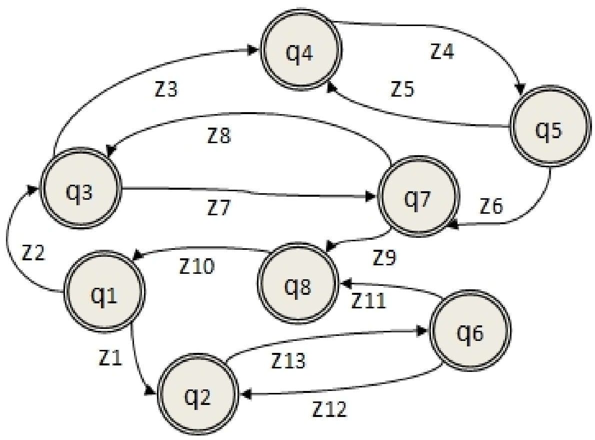

Figure 10. Dependency graph describing the process of selecting a variant of production order.

Acting in accordance with the algorithm and assuming that the start vertex is q_1, it is possible to transform the oriented dependence graph presented in Figure 10 into an analytical formulation $G^+_{q_1}$, and then we obtain the following formulation as a result of the operation (3).

$$G^+_{q1} = (^0 q_1(^1 z_2 q_3(^2 z_7 q_7(^3 z_8 q_3, z_9 q_8(^4 z_{10} q_1)^4)^3), z_3 q_4(^3 z_4 q_5(^4 z_5 q_4, z_6 q_7)^4)^3)^2, \\ z_1 q_2(^2 z_{12} q_6(^3 z_{11} q_8, z_{13} q_2)^3)^2)^1)^0 \quad (11)$$

The next step is to obtain a tree structure that plays a parametrically defined expression.

$$G_{q1}^{++} = ({}^{0} q_1 ({}^{1} z_2 q_3 ({}^{2} z_7 q_7 ({}^{3} z_8 q_3{}^{1}, z_9 q_8 ({}^{4} z_{10} q_1{}^{1})^{4})^{3}), z_3 q_4 ({}^{3} z_4 q_5 ({}^{4} z_5 q_4{}^{1}, z_6 q_7 ({}^{5} z_8 q_3{}^{2}, \\ z_9 q_8 ({}^{6} z_{10} q_1{}^{2})^{6})^{5})^{4})^{3})^{2}, z_9 q_8 ({}^{6} z_{10} q_1{}^{2})^{6})^{5})^{4})^{3})^{2}, z_1 q_2 ({}^{2} z_{12} q_6 ({}^{3} z_{11} q_8 ({}^{4} z_{10} q_1{}^{3})^{4}, z_{13} q_2{}^{1})^{2})^{1})^{0} \quad (12)$$

It is possible to return to an earlier vertex and even to a start vertex from an appropriate end vertex, so we obtain a game tree structure presented in Fig. 11.

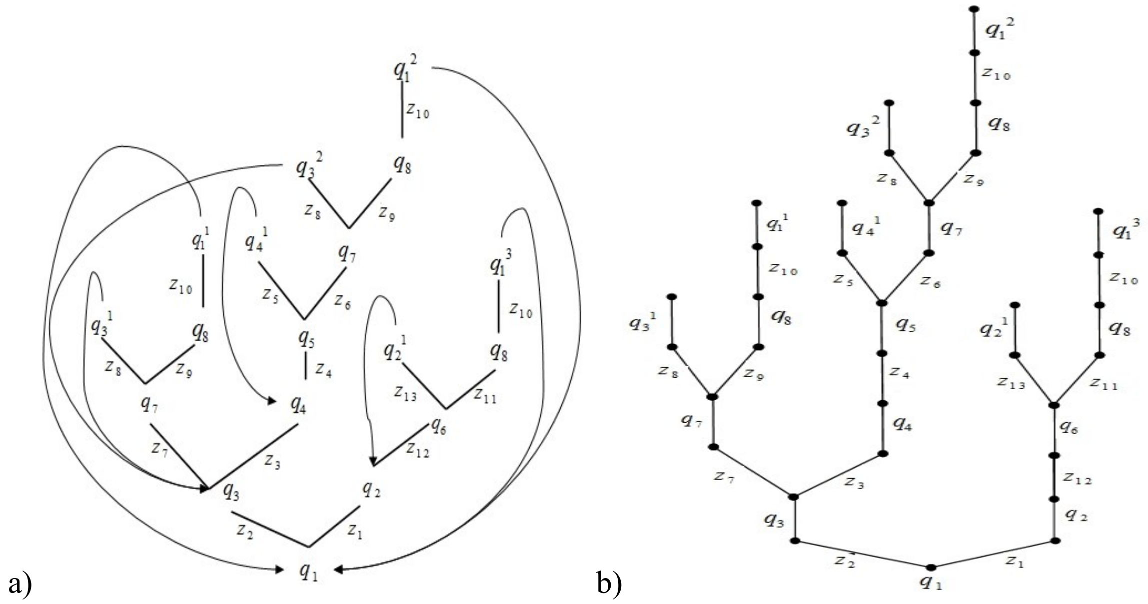

Figure 11. The tree structure with cycles and the initial vertex $q_1(a)$ and game tree structures from the initial vertex $q_1(b)$.

6. Conclusions

The game-tree structures method can be an excellent tool for gathering and transforming knowledge. After encoding, this knowledge can be repeatedly reproduced and globally distributed via the program. In expert systems, the question of structuring the inference process (obtaining knowledge) consists of two separate parts (Radosiński, 2001):

- mapping of deductive activities specific to a given subject field (defining the structure of inference).
- conducting the deduction process by the requesting mechanism (determination of the inference function).

This approach to the deduction process makes it possible to represent the inference structure in the form of a parametrically directed graph with a defined span (i.e. the maximum or the minimum number of arcs derived from one node).

Graph nodes in the inference structure represent individual information states, and edges are logical transition operations that result in state transformation. Initial nodes are variants of the initial problem description, end nodes are the ending of the inference process.

The inference structure can be specified on the parametric structure. The tree finds solutions – through the application mechanism – of specific problems assigned to the field of the subject expert system. The task of the dependency graph is to initiate the graph of the inference function by selecting one of the variants (the starting vertex) in the inference structure graph. Then the requesting mechanism searches the parametric structure - along with determining the function, up to the final node (solving the problem). The application of the method can improve the quality of decision alignment and give access to problem solving of various technical problems. When changes in the numerical values of input variables in the mathematical model are made, changes in the values of output variables are obtained. Other planned behaviour of the system (element) can be obtained in many ways by changing the numerical values of the input variables. In the optimization process, additional questions are asked, e.g.: which parameters should be changed? how to change them – increase?, without change?, decrease? in what order? etc. Such individual conception according to a given designer changes the numerical values of structural parameters in the mathematical model.

The practical application of graphs and parametric trees concerns the optimization of hydraulic and pneumatic systems as a generalization of signal flow graphs, for example Coates graphs. The second practical application of parametric game trees is finding the optimal number of teeth in the optimization of planetary gears previously modelled with a contour graph. In decision support systems, parametric game trees are used to find the objective function. In addition, logistics problems are a very wide practical application of this methods. In the case of knowledge generation, game-tree structures can be used to automatically generate a knowledge base.

References

1. Chung-Min, W., Ching-Lin, H., and Kuei-Lun, Ch. (2013). A Hybrid Multiple Criteria Decision Making Model for Supplier Selection. *Mathematical Problems in Engineering, 2013.*
2. Deptuła, A. (2014). Application of multi-valued weighting logical functions in the analysis of a degree of importance of construction parameters on the example of hydraulic valves. *Inter. J. of Appl. Mech. and Eng., 19(3),* pp. 539-548.
3. Deptuła, A. (2014). *Indukcyjne drzewa decyzyjne (entropia) jako odpowiednik zmodyfikowanych drzew logicznych w wyznaczaniu rangi ważności zmiennych decyzyjnych projektowanego układu.* XLIII Konf. Zast. Mat., Zakopane 2014, Inst. Mat. PAN, Warszawa.

4. Deptuła, A., and Partyka, M.A. (2011). Application of dependence graphs and game trees for decision decomposition for machine systems. *Journal of Automation, Mobile Robotics & Intelligent Systems, 5, 3,* pp. 17-26.
5. Deptuła, A., and Partyka, M.A. (2016). *Znaczenie kolejności atrybutów dla zbiorów przykładów klasyfikowanych z wykorzystaniem indukcyjnych drzew decyzyjnych*. XLV Konf. Zast. Mat., Zakopane 2016, Inst. Mat. PAN, Warszawa.
6. Deptuła, A., and Partyka, M.A. (2017). Inductive Decision Tree Analysis of the Validity Rank of Construction Parameters of Innovative Gear Pump after Tooth Root Undercutting. *International Journal of Applied Mechanics and Engineering, 22, 1*, pp. 25-34.
7. Deptuła, A., and Partyka, M.A. (2018). Application of complex game-tree structures for the Hsu graph in the analysis of automatic transmission gearboxes. *Journal of Machine Engineering, 18, 4,* pp. 96-113.
8. Deptuła, A., and Partyka, M.A. (2018). Zastosowanie graficznych struktur decyzyjnych w metodologii projektowania i zarządzania. Tom 1 – Grafy rozgrywające parametrycznie. *Studia i Monografie, 482,* Oficyna Wydawnicza Politechniki Opolskiej, ISBN: 973-83-66033-10-8, Opole.
9. Deptuła, A., Drewniak, J., and Partyka, M.A. (2017). Application of dependence graphs and game trees in analysis of a planetary gear modeled with a contour graph. *Machine Dynamics Research, 3, 41.*
10. Deptuła, A., Łuszczyna, R., and Partyka, M.A. (2018). Zastosowanie graficznych struktur decyzyjnych w metodologii projektowania i zarządzania. Tom 2 – Decyzyjne wielowartościowe drzewa logiczne. *Studia i Monografie, 483*, Oficyna Wydawnicza Politechniki Opolskiej. ISBN 978-83-66033-11-5, Opole.
11. Deptuła, A., Macek, W., and Partyka, M.A. (2018). Analysis of loading history influence on fatigue and fracture surface parameters using the method of induction trees. *MATEC Web of Conferences, 252, 08003,* Lublin, pp. 1-6.
12. Deptuła, A., Osiński, P., and Partyka, M.A. (2018). Analysis of dimensional tolerances a series of 2PW-SE gear pumps using multi valued logical trees. *MATEC Web of Conferences, 252, 06002,* Lublin, pp. 1-6.
13. Deptuła, A., Osiński, P., and Radziwanowska, U. (2016). Decision Support System for Identifying Technical Condition of Combustion Engine. *Archives of Acoustics, 41, 3,* pp. 449-460.
14. Deptuła, A.M., and Deptuła, A. (2016). *Zastosowanie indukcyjnego systemu ekspertowego do procesu wyznaczania ważności kryteriów oceny ryzyka innowacyjnego*. XLV Konf. Zast. Mat., Zakopane 2016, Inst. Mat. PAN, Warszawa.
15. Fernando Otero, E.B., Freitas, A., and Colin, G. (2012). Inducing decision trees with an ant colony optimization algorithm. *Applied Soft Computing, 12, 11,* pp. 3615-3626.

16. Horzyk, A. (2012). *Information Freedom and Associative Artificial Intelligence* Information Freedom and Associative Artificial Intelligence. Berlin-Heidelberg: Springer Verlag. LNAI 7267, pp. 81-89.
17. Iglesias, C.A., Gonzalez, J.C., and Velasco, J.R. (1996). The role of hybrid systems in intelligent data management: the case of fuzzy/neural hybrids. *Proceedings of the Seminar on Intelligent Data Managemen*t, London, UK.
18. Iordanov, B. (2010). Hyper Graph DB: A generalized graph database. *Proceedings of the Web-Age Information Management Workshop* (WAIM2010).
19. Keen, P.G., and Scott Morton, M.S. (1978). *Decision Support Systems: An Organizational Perspective*. Addison-Wesley, Reading.
20. Klein, M., and Methlie, L.B. (1992). *Expert Systems. A Decision Support Approach with Applications in Management and Finance*. England: Addison-Wesley Publishing Company.
21. Kotsiantis, S.B. (2013). Decision trees: a recent overview. *Artificial Intelligence Review*, *39, 4*, pp. 261-283.
22. Kumar Kar, A. (2015). A hybrid group decision support system for supplier selection using the analytic hierarchy process, fuzzy set theory and neural network. *Journal of Computational Science Journal of Computational Science, 6.*
23. Kuo-Hao, Ch. (2014). A decision support system for planning and coordination of hybrid renewable energy systems. *Decision Support Systems, 64.*
24. Petrović, J., and Pale, P. (2017). *Decision trees in formative procedural knowledge assessment*. Electronics and Microelectronics (MIPRO). 40th International Convention on Information and Communication Technology, pp. 17-20, Opatija, Croatia.
25. Pijls, W., and De Bruin, A. (2001). Game tree algorithms and solution trees. *Theoretical Computer Science, 252, 1-2,* pp. 197-215.
26. Qiao, L., Yang, L., Hong, D., Yao, L., and Qin, Z. (2016). Knowledge Graph Construction Techniques. *Journal of Computer Research & Development*.
27. Quinlan, J.R., and Rivest, R.L. (1989). Inferring decision trees using the minimum description length principle. *Information and Computation, 80, 3*, pp. 227-248.
28. Radosiński, E. (2001). *Systemy informatyczne w dynamicznej analizie decyzyjnej*. Warszawa: PWN.
29. Rudas, I.J. (2004). *Hybrid Systems.* Encyclopedia of Information Systems, Elsevier Inc.
30. Spraque, R.H., and Carlson, E.D. (1982). *Building Effective Decision Support Systems*. Englewood Cliffs, NJ: Prentice-Hall, Inc.
31. Trzaskalik, T. (2014). Wielokryterialne wspomaganie decyzji. Przegląd metod i zastosowań. *Zeszyty Naukowe. Organizacja i Zarządzanie / Politechnika Śląska, 74.* Gliwice, pp. 239-263.