ICIT 2015 The 7th International Conference on Information Technology
doi:10.15849/icit.2015.0061

Technical object as a system of "growing" structure

Marek Młyńczak
Wroclaw University of Technology
Faculty of Mechanical Engineering
Chair of Logistic System Operation,
Transportation System Operation and Hydraulic Units
ul. Wyspianskiego 27
50-370 Wrocław, Poland

***Abstract*: Modern technical objects are complex systems of multilevel hierarchical structure. In the literature, problem of system structure, its catalogues and spare parts catalogues, as well as formal definition of the structure is rarely undertaken. Failure analysis of technical objects shows that at reliability field test, the knowledge about its structure is not always a priori required. It leads to the concept of "growing structure" which is created continuously while operational events appear. Formal definition of hierarchical structure compared to growing one is in the paper presented.**

Keywords—technical object, system, reliability field test, construction structure

I. INTRODUCTION

The complexity of modern technical objects in operation enforces the needs of creative use of operation performance theory, especially in the field of study of these processes for operation rationalization of these objects. An important issue of process management operation is operational data collection creating the base of knowledge about the system and its operation. Information base (knowledge base) is developed in the process of operation using the diagnostic methods, data verification and processing up to final assessment of machine effectiveness. The database is created in order to use it in decision making process and should contain information about operational states, time, failures identification, etc. Machine system operators should collect all these data in the simplest and reliable way but also protecting data from losing it.

During the operation of the facility, many physical transformation processes of energy and mass take place. It is accompanied by the aging and degrading processes leading to a total or partial loss of its performance. This justifies the need of performing diagnostic tests in the operation and collection of data documenting the variability of these properties over time. Systems for collecting and processing information about events are essential for optimal management of a process operation of the facility.

In machine operation an information base used to control the operation process of the object forms a set of data (database, knowledge base) with the algorithms of processing, transmission and storage of the data.

Also in the earlier, design phase, technical objects should satisfy several technical, economical and quality requirements. Some of the most important are those, which have an effect on: availability, safety and operational costs. Most of computer aided design systems help in creating technical object, what is based on data covering attributes ensuring: consistency, dimensional synchronization and functionality. The purpose of this paper is to analyze the possibility of applying operational, reliability and safety data into database in more effective way. Random events disturb proper process and lower effectiveness of operation through undesired failures, hazardous events and other economic consequences. Object to be designed is modelled in systemic approach that considers elements described by attributes due to functionality, reliability and safety.

An important element of the information system operation management is an algorithm of dynamic information processing of its functional structure based only on observed events in real operation. This algorithm is an ordered set of instructions of input data transforming into information to make it useful in recognizing an item or object module (source of an event) in its functional structure. Data processing algorithm in the collection of information for decision-making is described as mathematical decision-making model.

The presented information encoding algorithm about events in the operation of a technical object creates a knowledge base about its structure and is an important element supporting the process of its operation.

Created knowledge base allows for rational use of the potential capacity of the facility and its individual components, by reducing the risk of errors resulting from the lack of the necessary information on how to use the principles handling, material supply and economically reasonable management [13].

II. MODELING OF COMPLEX TECHNICAL OBJECTS

The problem of modelling complex technical objects appeared in reliability theory in the early publications of Birnbaum, Saunders, Barlow and Proshan, where the object is defined as a system composed of elements [2]. The system structure is most often treated as a hierarchical structure, assigning sets of components to a higher level of decomposition by certain relations [3, 4, 12, 15].

A review of the many definitions of the system shows that "*elements ordering is the weakest point, structure definition requires establishing the existence of system elements to be in certain order, relationships and linkages between structure and function of the system*" [15]. Next, the author proposes a definition of the structure as a "generalized characteristics of the specific properties of the system, fixing in the abstract form elements, relationships, feedbacks, their arrangement and organization."

Object structure is the basis for modelling an operational database in the research of real mechanical object [9]. Quoted here are some examples of hierarchical, multi-level structure of the object and relevant documents to collect data on failures and other operational events.

Most studies of complex technical systems are based on the two-level decomposition where the system is made up of elements attributed with a hierarchical system of codes [9, 13, 16, 17].

In the systems with redundancy it is taken into account the complex structure of the object by assigning redundant elements two-level code. Sample block A of higher level is decomposed into elements A1, A2, A3 having code of higher level A and *i*=1,2,3 of lower one [11].

Creating structure of technical objects is done usually using a hierarchical approach; however, it is mostly intuitive process. Catalogues of spare parts are built by assigning elements to components, further to the subassemblies, assemblies, units and systems, which is a typical example of a hierarchical structure [10]. An example of v-belt stretcher subassembly of cooling subsystem of diesel engine is shown in Fig. 1. Elements are numbered 1 to 25 and correspond to higher decomposition level (cooling subsystem).

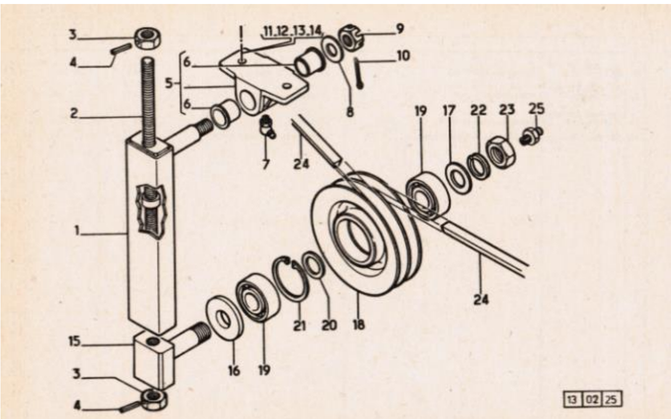

Fig. 1. An example of v-belt stretcher subassembly – catalogue page

III. THE ALGORITHM OF SYSTEM DECOMPOSING

Technical system is characterized by a hierarchical structure consisting of set of elements and its properties and is described in the form of triplet (1) [5, 7, 8]:

$$OM = \langle E, W, R \rangle \quad (1)$$

where: $E = \{e_i\},\ i = 1,2,\dots,n$ - a set of elements (subsystems, components),

$W = \{w_{is}\},\ s = 1,2,\dots,m_i$ - a set of distinguished properties (attributes) of elements,

$R = \{R_l(w_{is})\},\ l = 1,2,\dots,r_i$- a set of relations assigned to a set of properties of elements.

As time passes, elements of the object appears in "information process" giving in time section sign (event), that operator should react on it. Information regards one of many attributes attached to the element (Fig. 2).

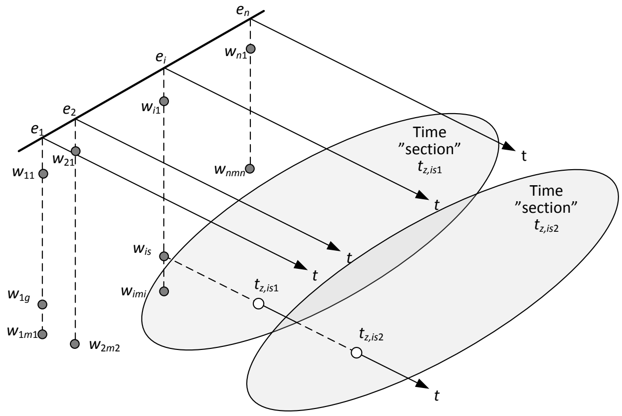

Fig. 2. Time sections with regard to events created by *i*-th element

The system is subjected to a multi-level decomposition into subsystems, in practice, called systems, units, assemblies, subassemblies, components, and so on. Later, the name of element is reserved exclusively for single-element module, referred to as non-disintegrable. The first level of decomposition is the main division, and it forms n_1 main modules (2):

$$E_{j_1};\ j_1 = 1, 2, \dots, n_1 \quad (2)$$

Since each of the components E_j has to belong to only one main module and so their set sum gives the set of all elements (3), i.e.:

$$\bigcup_{j_1=1}^{n_1} E_{j_1} = E \quad (3)$$

If for some $j_1 = 1, 2, \dots, n_1$ module E_{j_1} is composed of only one element e_i, i.e. $E_{j_1} = \{e_i\}$, then the module is no longer decomposed and element e_i is identified as the only element of the module E_{j_1}. Especially, when the number of units is equal to the number of elements, namely $n_1 = n$, then the system is completely decomposed at the first level, and it ends up decomposition process. If $n_1 < n$, then decomposition process continues to decompose the second level. Let n_1^* denotes the number of single-element modules decomposed at the first

doi:10.15849/icit.2015.0061

level of the system, i.e. non-disintegrable. These items are excluded from the further process of decomposition. Then, the number $d_1 = n_1 - n_1^*$ of multi-element modules remains to be decomposed further at lover level. Decomposition of each multi-element module proceeds in the same way as at first level and is continued until all multi-element modules are decomposed.

Due to the finite number of elements of the system, decomposition process finally completes and the number of levels of the system depends on the number of its components and the complexity of the functional structure. Let's introduce further designations:

l – number of decomposition levels ($l \in$N), N is the set of natural numbers,

n_i – number of modules on the i-th decomposition level of the system ($1 \leq i \leq l$, $n_i \in \boldsymbol{N}$),

n_i^* – number of single-element modules on the i-th decomposition level,

d_i – number of multiple modules on the i-th decomposition level.

Of course, the equality (4) exists:

$$\sum_{i=1}^{l} n_i^* = n \quad (4)$$

The total number of decomposition $d = \sum_{i=1}^{l} d_i$ is (5):

$$d = \sum_{i=1}^{l} n_i - n \quad (5)$$

As a result of decomposition process of the full system, all elements belong to single-element modules. As an example of the concept of identifying the elements in complex system let's look at the decomposition of the system consisting of 18 elements (Table I).

In this example it is l=5 levels of decomposition. The number of modules at different levels is summarized in Table II.

TABLE I. NUMERICAL DETAILS OF SYSTEM DECOMPOSITION

Decomposition level i	***Number of elements n_i***	**Number of multi-element modules n_i^***	**Number of single-element modules d_i**
1	4	1	3
2	7	2	5
3	10	7	3
4	6	5	1
5	2	2	0

Code assignment procedure requires, in the first step, to distribute all multiple modules, so that the structure at the lowest decomposition level consists only with non-disintegrable elements e_i, i.e. $E_{j_1 j_2 \dots j_k} = \{e_i\}$. Then e_i element is attributed to the index of the module, or code $j_1 j_2 \dots j_k$. It is written as: $e_i^{j_1 j_2 \dots j_k}$. Number k indicates that the element passes k levels of decomposition. After full decomposition of the system, each element is given a unique code indicating its location in the functional structure. In the given example the e_7 has a code 3-2-1, so that we denote it: $e_7^{3,2,1}$, and e_8 has a code 3-2-2-1 so we denote it: $e_8^{3,2,2,1}$.

Let $K = \{k_1, k_2, \dots, k_n\}$ denotes the set of codes of decomposed system. The function code (6) is called element coding. This function is a bijection, and the inverse function assign element with e given code.

$$Code: E \rightarrow K \quad (6)$$

Presented coding system gives a complete knowledge of the structure of the system. Introduction set-theoretic formalism [1], taking into account the effect of degradation of the system on its declining utility can set a new direction of research on modelling the process of operation of the facility.

IV. TECHNICAL OBJECT AS A SYSTEM OF INDETERMINATE A PRIORI STRUCTURE (DYNAMIC STRUCTURE – "GROWING SYSTEM")

In the real operation is often necessary to identify the components of the complex technical object while and-end are not supplied with catalogues describing object structure what saves an interest of technical services or protect intellectual property rights relating to technical solutions.

Description for the elements in the structure of the object becomes important if the complexity of the object is significant, i.e. entire system has hundreds or thousands of components or, on the other hand, if the knowledge of the position of element is not required for all items. The needs to identify the object element in the system appear each time when this element creates information that the end user may use. It means he has to undertake certain decision or actions, based on the event like: the element failed, item does not meet user requirements or spare part should be ordered.

In reliability field test of bucket wheel excavator SRs-2000 it were observed damages of 86 elements among 260 identified main modules on 3 decomposition levels [5]. Similar observation took place in the study of bucket loader L-220, where there are 259 elements failed among 3334 modules stored in the catalogue [6]. These examples show redundancy of the catalogues in relation to operational needs. The idea of dynamic catalogue would save time and memory of database and it can grow according to new events that appear in operation.

It is assumed that at the beginning of the service is not known object structure and its construction is "learnt" as a result of the occurrence of events related to the individual modules. Elements of the system create modules (subsystems) with common characteristics. Technical analysis allows for introduction of sub-elements for each structure as the appearance of information generated by them. The key to proper placement of elements in the structure is their characteristics and properties of components which have been already identified.

doi:10.15849/icit.2015.0061

Element is the smallest part of the object i.e. not undergoing to further decomposition. Particularly, if the module is a single – piece module, it is called an element. Otherwise, we call it module. At the moment, if appears an event associated with a new module or component modification of the object structure is performed. Modification of the structure is done using three types of relations, in which the new module can be recognised comparing to the known modules. These relations are: equivalence, subordination and the primacy relationship. Type of relationship for the new module to the already known modules is determined based on its properties.

TABLE II. AN EXAMPLE OF A COMPLETE SYSTEM DECOMPOSITION

Decomposition level	**System elements that correspond to events describing operational process**																	
	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}	e_{15}	e_{16}	e_{17}	e_{18}
1	$\boldsymbol{E_1}$[a]	E_2			E_3					E_4								
2		$\boldsymbol{E_{21}}$	E_{22}		E_{31}		E_{32}			$\boldsymbol{E_{41}}$	E_{42}		E_{431}		E_{432}			
3			$\boldsymbol{E_{221}}$	$\boldsymbol{E_{222}}$	$\boldsymbol{E_{311}}$	$\boldsymbol{E_{312}}$	$\boldsymbol{E_{321}}$	E_{322}			$\boldsymbol{E_{421}}$	$\boldsymbol{E_{422}}$						
4								$\boldsymbol{E_{3221}}$	$\boldsymbol{E_{3222}}$				$\boldsymbol{E_{4311}}$	$\boldsymbol{E_{4312}}$	$\boldsymbol{E_{4313}}$	$\boldsymbol{E_{4321}}$		
5																	$\boldsymbol{E_{43221}}$	$\boldsymbol{E_{43222}}$

a. in bold, a single element module is represented

At the beginning of system structure creating, an object (system) is identified at the highest level of decomposition E. As soon as new event associated with the module e_1 appears, knowledge about the object is described as: $E = \{e_1\}$.

The first step of iterative structure modification requires analysis of the next event in the operation which may cause either a change in its structure as well as in the codes. The event may be related to:

1. the same element e_1 (just record the attribute for the element e_1)
2. new element e_2 which belongs to the same module (subsystem) as element e_1 on equivalent level of decomposition (siblings),
3. new element e_2 which belongs to the same module (subsystem) as element e_1 on subordinate level of decomposition (child),
4. new element e_2 which belongs to the same module (subsystem) as element e_1 on superior level of decomposition (parent),

According to the assumptions above, there is a description of the modification of the structure according to the scheme:

- ad 1. (as above): If you assume that the next event during the observation operating system is associated with existing in the structure module e_i then the structure of the system is not modified and information is saved as change of attribute of element e_1: $E = \{e_i\}$.
- ad 2. (as above): Elements e_1 and e_2 form subsystems at the same level of decomposition, so that: $E = \{E_1, E_2\}; E_1 = \{e_1\}, E_2 = \{e_2\}$.
- ad 3. (as above): Element e_2 is an element of lower level of decomposition over e_1, so that: $E = \{E_1, E_{11}\};\ E_1 = \{e_1\}, E_{11} = \{e_2\}$,
- ad.4. (as above): Element e_2 is a super system element relatively to E_1, so that: $E = \{E_1, E_{11}\};\ E_1 = \{e_2\}, E_{11} = \{e_1\}$.

Table III shows an example of the development of the structure for the first seven elements. The table columns are related to the successive moments of event appearing in the system.

The example above shows code modification of originally single-element system requires algorithmic approach, taking into account the relationship between the elements (modules) already present in the structure and new elements: $e_i - R - E_j$.

The relation R shows the algorithm of code change for any system module:

- $R^{=}$ – equivalence position in the structure of the elements e_i and E_j (siblings) (7),

$$\left(e_i - R^{=} - E_j\right) \wedge (i = j_1 j_2 \dots j_k) \Rightarrow j = j_1 j_2 \dots (j_k + 1) \quad (7)$$

- R^{-} – subordinate position in the structure of elements e_i and E_j (child) (8)

$$\left(e_i - R^{-} - E_j\right) \wedge (i = j_1 j_2 \dots j_k) \Rightarrow j = j_1 j_2 \dots j_k j_{k+1} \quad (8)$$

- R^{+} – primacy position in the structure of the elements e_i and E_j (parent) (9).

doi:10.15849/icit.2015.0061 © ICIT 2015 (http://icit.zuj.edu.jo/ICIT15)

$$\left(e_i - R^+ - E_j\right) \wedge (i = j_1 j_2 \dots j_k) \Rightarrow j = i \wedge i = j_1 j_2 \dots j_{k-1} \tag{9}$$

In this case, the set of codes changes and after registration of the *i*-th element code set will take the form of *i*-element set: $K_i^* = \{k_1^*, k_2^*, \dots, k_i^*, \dots\}$, wherein $K_n^* = K$. Function $Code: E \rightarrow K_i^*$ maps the set of elements in the set of codes.

Iterative modification of the structure will be carried out until all modules are considered as single-element modules (elements).

TABLE III. MODIFICATION OF SYSTEM STRUCTURE WHILE CONSECUTIVE ELEMENTS ARE ADDED IN TIME

element	e_1	**e_2 (siblings for E_1)**	**e_3 (child for E_1)**	**e_4 (child for E_{11})**	**e_5 (parent for E_2)**	**e_6 (siblings for E_1 and E_2)**	**e_7 (child for E_3)**	e_i
	time of new element appearance →							
subsystem 1	$e_1 =>^b E_1$		$e_3 => E_{11}$	$e_4 => E_{111}$				…
subsystem 2		$e_2 => E_2$			$e_5 => E_2$ $e_2 => E_{22}$			…
subsystem 3						$e_6 => E_3$	$e_7 => E_{31}$	…
subsystem …		$^{b)}$ $e_i => E_j$ – means that new element e_i is labelling E_j						…

SUMMARY

The paper presents two approaches to systematic identification of elements in a complex system. These approaches are particularly important at the stage of creating data bank in operational research. The first approach is the traditional one leading to the determination of structural or functional relationships for all elements that create an object. This approach ensures full knowledge of the construction of the system at every stage of research and analysis, however, requires considerable work and oversizing database of object structure. Dynamic approach presented in the paper modifies the structure of the object after every event appearance related to the "new" element, yet not existing in the structure. Database starts with one element, which is entire object and grows as new information about components come from the operation. This may limit the size of database because, as it is seen from observation, only 10-30% of the object components provide operational information.

REFERECES

[1] K. Andrzejczak, "Set-theoretical approach to finite-state system reliability structure". Szczyrk: Proceedings of XXXIV Winter School of Reliability, 2006: 13-30.

[2] R.E. Barlow, F. "Proschan Mathematical theory of Reliability". New York: John Wiley and Sons Inc., 1965.

[3] D. Bobrowski, "Models and methods of mathematical theory of reliability". Warsaw: Scientific and Technical Publishing, 1985.

[4] E. Fidelis, S. Firkowicz, K. Grzesiak, K. Kołodziejski, K. Wisniewski, "Mathematical Foundations of reliability assessment". Warsaw: PWN, 1966.

[5] A. Gołąbek, M. Młyńczak, T. Nowakowski, "Computer system operational reliability assessment of open pit mine machines". Górnictwo odkrywkowe - Journal of Open Pit Mining 1992; 33 (3/4): 66-74.

[6] A. Gołąbek, M. Młyńczak, T. Nowakowski, "Assessment of operational reliability of L-220 loader". Overview of Mechanical Engineering - Mechanical Reviev 1990, 49 (18): 53-60.

[7] A. Gołąbek, "Procedures for testing and reliability assessment of machinery". Scientific Papers of the Institute of Machine Design and Operation, Wroclaw University of Technology; Series of Monographs; 21 Wroclaw: Ed. Wroclaw University of Technology, 1992.

[8] "Handbook of Performability Engineering". Ed.: K.B. Misra. London: Springer-Verlag, 2008.

[9] A. Kossiakoff, W.N. Sweet, S.J. Seymour, S.M. Biemer, "Systems Engineering Principles and Practice". Hoboken, NJ: John Wiley & Sons, Inc., 2011.

[10] "L220 Charger. Spare parts catalog". Factory Construction Equipment BUMAR-FADROMA. Wroclaw: AGPOL, 1986.

[11] A. Myers, "Complex System Reliability. Multichannel Systems with Imperfect Fault Coverage". London: Springer-Verlag, 2010.

doi:10.15849/icit.2015.0061

[12] "Reliability and maintenance of systems". Ed. Zamojski W. Wroclaw: Wroclaw University of Technology, 1981.

[13] T. Nowakowski, "Problems with analyzing operational data uncertainty". Archives of Civil and Mechanical Engineering 2010; 10 (3): 95-109.

[14] "Product Reliability. Specification and Performance". Ed. D.N.P.Murthy, M. Rausand, T. Østerås. London: Springer-Verlag, 2008.

[15] W. Sadowski, "Fundamentals of general systems theory". Warsaw: PWN, 1978.

[16] DJ Smith, "Reliability, Maintainability and Risk. Practical methods for engineers". Oxford: Butterworth-Heinemann, 2000.

[17] W.R. Wessels, "Practical Reliability Engineering and Analysis for System Design and Life-Cycle Sustainment". Boca Raton: CRC Press Taylor & Francis Group, 2010.