

INTERACTIVE SYSTEMS FOR DESIGNING MACHINE ELEMENTS AND ASSEMBLIES

Wojciech Kacalak¹, Maciej Majewski¹, Keith Douglas Stuart², Zbigniew Budniak¹

¹ Koszalin University of Technology, Faculty of Mechanical Engineering, Koszalin, Poland

² Polytechnic University of Valencia, Department of Applied Linguistics, Valencia, Spain

Corresponding author:

Maciej Majewski

Koszalin University of Technology

Faculty of Mechanical Engineering

Raclawicka 15-17, 75-620 Koszalin, Poland

phone: (+48) 94 3478-352

e-mail: maciej.majewski@tu.koszalin.pl

Received: 6 June 2015

Accepted: 27 July 2015

ABSTRACT

The article describes the development of fundamentals of machine elements and assemblies design processes automation using artificial intelligence, and descriptions of structural elements' features in a natural language. In the proposed interactive automated design systems, computational artificial intelligence methods allow communication by speech and natural language, resulting in analyses of design engineer's messages, analyses of constructions, encoding and assessments of constructions, CAD system controlling and visualizations. The system is equipped with several adaptive intelligent layers for human biometric identification, recognition of speech and handwriting, recognition of words, analyses and recognition of messages, enabling interpretation of messages, and assessments of human reactions. The article proposes a concept of intelligent processing for analysis of descriptions of machine elements' structural features in a natural language. It also presents the developed methodology for similarity analysis between structural features of designed machine elements and corresponding antipatterns allowing normalization of parameters of the analysed structural solutions.

KEYWORDS

design systems, interactive systems, intelligent interfaces, antipatterns, computer-aided design.

Non-repeatability of design processes and uncertainty level of results

Many tasks related to designing machine and device components are isolated cases. Any attempt at forecasting working properties of machine components and assemblies entails the necessity to face the fact, that the conditions of their manufacturing, installation and operation are described with incomplete and uncertain, as well as inaccurate information.

The presented research involves the development of intelligent interactive automated systems for designing machine elements and assemblies using descriptions of structural elements' features in a natural language. Realization of the automated design processes is in conditions of uncertainty and with

non-repeatable processes. We propose a new concept [1] which consists of a novel approach to these systems, with particular emphasis on their ability to be truly flexible, adaptive, human error-tolerant, and supportive both of design engineers and data processing systems.

The foundation of interactivity is bi-directional communication between the data processing system and the user. Results of evaluation, performed with artificial intelligence, of the designed solutions are used at as early as the design stage.

The aim of the research is to develop the basis for new design processes of higher level of automation, and object approach to problems and application of voice communication between the design engineer and data processing systems. Some of the most significant disadvantages of the systems responsible

for creation of construction representation, present in the current design systems, are the following:

- Creating designs by performing graphical operations – using slow communication interfaces such as: keyboard, tablet, and mouse – on basic components like graphical symbols and lines.
- Drawing still takes too much of a designer’s thought processing.
- Data completion and processing takes place on layers – containing graphical elements of a certain type – which makes it more difficult to take advantage of benefits of object-oriented approach towards geometrical components of an item, e.g. a shaft’s step, or a particular cut-in.
- Data completion and processing taking place on layers containing graphical elements of a certain type (lines, circles, ...) makes it necessary for technological processes design software to perform recognition operations of basic geometric objects, based on analysis of their graphical representation (it has to reconstruct drawings in a technological aspect).
- Instead of storing information as objects representing an item’s basic components, graphical information is stored.
- Storing designs in formats characteristic of obsolete vector graphics systems which use basic image components (lines) instead of objects, which can be automatically rendered by a code interpreter.
- The disadvantage mentioned above can be explained by analogy with comparison of different methods of data representation: technical drawings, description of structure of documents (XML) and webpages (HTML). The conclusion to be drawn here is that instead of storing a vector graphics image, we simply can store item’s characteristics, and an interpreter will reconstruct the image under any operating system, using universal software. After making amendments to the item its “image” will be stored again as a set of characteristics.

The comparison of the proposed new automated design system with the present system of carrying out design tasks is shown in Fig. 1.

The condition for effectiveness of automated systems of machine components and composite machines design is mainly the implementation of the following mechanisms:

- Creating elementary objects, as components of an element, based on description created or passed in real time in a form of voice or symbols.
- Eliminating defects and operator’s errors in the process of automatic interaction with the technical documentation generation system.

- Automatic technological classification of machine components using artificial neural networks.
- Fuzzy inferencing for normalization of structural features, as inputs for the automatic technological classification system.

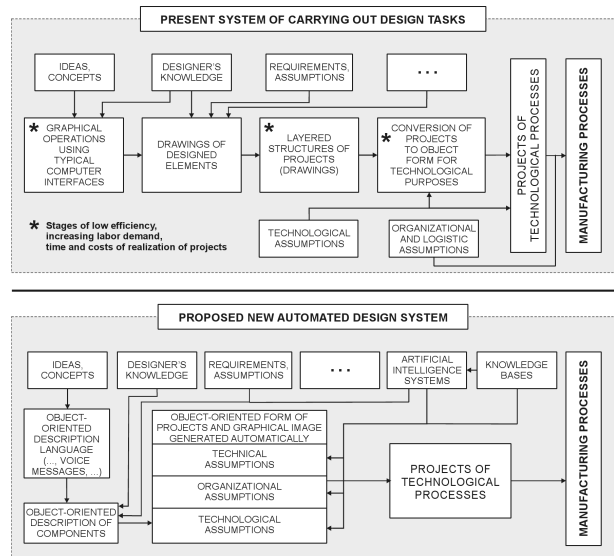


Fig. 1. Comparison of the proposed new automated design system with the present system of carrying out design tasks.

In the design many advanced methods for problem solving and task fulfilment are used, in which the designer’s explicit and implicit knowledge and skills are required for determination of product features to be created. The product features will be the outcome of many complex processes and actions, and will be assessed by the user in various operational conditions and various technical states.

In the presented, new design system the proposed methods for storing items’ characteristics allow any advanced integration of characteristics representation of structures and technological processes, as well as organization guidelines. However, there still will be cases where a symbolic or verbal description might be ambiguous, and for such cases graphical data will remain important. Also, data coming from processes of shape and dimension reconstruction, in graphical and numerical form, will serve as supporting data.

In complex design tasks liberating the designer from the necessity of manually using slow interfaces will allow to eliminate the intermediate phase (composing an image using graphical symbols), which relegates object-oriented approach to elements to the layer-oriented method, disadvantageous to further implementations of a project.

Many advanced methods for solving problems and executing design tasks are used when designing

constructions. Explicit and tacit knowledge, as well as designer's skills are used to determine properties of a product, which will be later manufactured. Properties of such a product will be the result of many complex processes, and work of many manufacturers, and users will test and judge those properties in different operational conditions and at different grades of technical condition. The time spent designing a component is the key element to the whole creation process of new constructions, which should meet certain strength and structural requirements, and also be faultless.

Machine components design is a very time-consuming and responsible task. It is the first phase of the manufacturing process, and its aim is to anticipate the impact of external factors on the designed component. Therefore designers must be able to correctly combine knowledge of many fields. To their aid come new design-aiding systems for both individual machine components, as well as complex mechanical systems in a concept phase.

Currently, the demand from computer-aided design is that it meet requirements [2] such as error handling, geometrical uncertainty, automatic error detection, as well as intelligent design aiding [3]. As part of the last of the listed postulates, the authors, using their own artificial-intelligence-based solutions related to bi-directional voice communication [4] between technological devices and their operators, have begun works the aim of which is to devise complementary fundamentals for creating intelligent and interactive systems for designing machine components and assemblies, based on their attributes described in a natural language.

Many research centers all over the world have begun to work on features which would supplement CAD systems with simple mechanisms of speech input [5, 6] (in the form of a simple interface for recognizing selected basic shapes). Those attempts share a simple postulate, which is to make it easier to work with traditional systems (through a simple interface for limited control of a CAD system).

The objective character of design features' descriptions

A part of the proposed system is our symbolical language used for defining structural properties of selected machine components during the design stage. Its purpose is to enable creation of descriptions of fundamental objects, such as elements of a structure.

In order to ensure unambiguous description and correct semantics of the language, a method of distinguishing individual elements constituting machine

parts has been defined. It has been specified that each structural object has its own attributes (e.g. length, diameter, etc.) and is comprised of constituent objects. As an example, for a structural object being a machine shaft, constituent objects are distinguished based on their position, dimensions and shape, as well as their surface layer properties. Constituent objects are simple polyhedra and they have their own attributes: dimensions, shape and position. In the given example a step of the shaft is a constituent object.

Any changes of dimensions, shape and surface layer properties are made with modifiers. A modifier also has its own attributes, which are defined in relation to the modified constituent object (parent object), to another constituent object, or to another modifier. A modifier cannot exist independently. Therefore, a modifier is an integral part of its parent object. Examples of a modifier are chamfer, a hole, or a thread.

The main assumption of the symbolical notation of structural features of machine parts' elements is that their description is object-oriented. Objects are characterized by attributes and permitted operations. Attributes are properties or variables associated with the object, and operations are functions or operations performed on object's attributes for the purpose of auto-modification or in order to influence other objects. Access to an object and its attributes is possible only through a defined interface. The interface interacts with an object's instance – which can be perceived as a living specimen (instance) of a species (object). Functionality of an object is directly connected with the attributes associated to it.

A disadvantage of the – currently used – structural description, usually in the form of a tree, less often graphs, is lack of description of relations between objects. When it comes to creation of technological and organizational processes, object-based description is important due to the manner of data notation.

Object-oriented notation for structural features requires creation of classes representing sets of objects, as well as methods (functions) allowing to perform an array of operations on particular objects, such as creation, deletion, modification, moving, rotation, etc. Objects of one class share many similarities, but they can insignificantly differ from each other. A class defines which operations and attributes are associated with an object, but values assigned to attributes may vary between instances.

Complex structures can be described in many ways. It can be done using an XML-like language, a specialized language similar to HTML, or a sym-

bolical language. We have defined assumptions for creation of a language, whereby a structural object can be easily divided into constituent objects, and modifiers can be distinguished as elements for altering dimensions, shape and properties of constituent objects. Their symbolical notation is also simple and easy to read.

Symbolical language is a form of machine notation used by a system, which generates, and interprets such a notation for automatic real-time creation of designed components. For the convenience of designing engineers, a higher-level language can be used, such as a constructional-XML.

Design tasks automation system's structure

The presented research proposes a concept and architecture of intelligent and interactive automated machine components and assemblies design systems [1] using descriptions of structural elements'

features in a natural language for design processes in conditions of uncertainty and with non-repeatable processes. Execution of design processes using this system (Fig. 2) allows:

- to have a higher level of design by unleashing designers' full creative potential and liberating them from carrying out the process of creating graphical representations of designed components,
- to speed up the design process, especially in the case of complex components,
- easier modification and rating of multiple solution variants,
- to automatize the most labour-intensive tasks in machine components design,
- to create an object-oriented language for describing structures, and also to create methods of symbolical notation,
- to improve the processing and data archiving operations,
- to create an artificial intelligence system aiding the design processes.

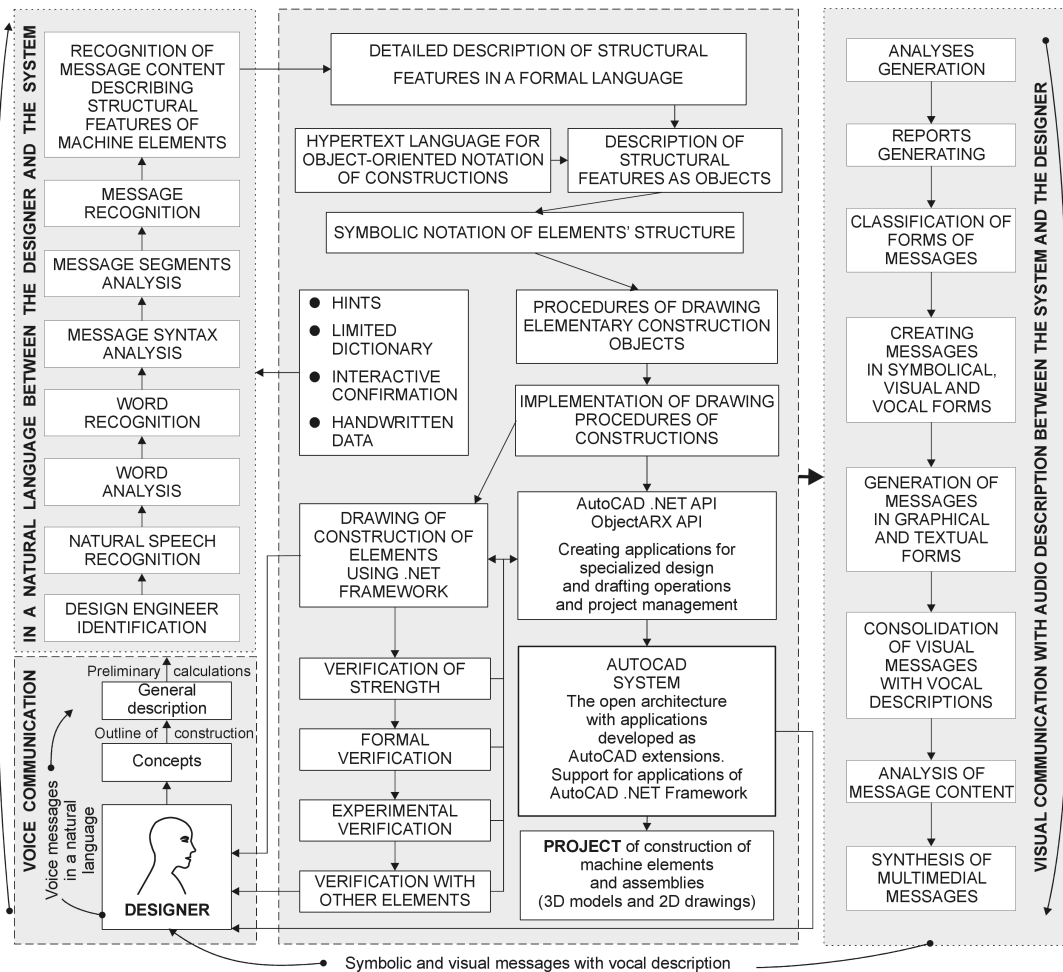


Fig. 2. Concept of design processes using interactive automated systems.

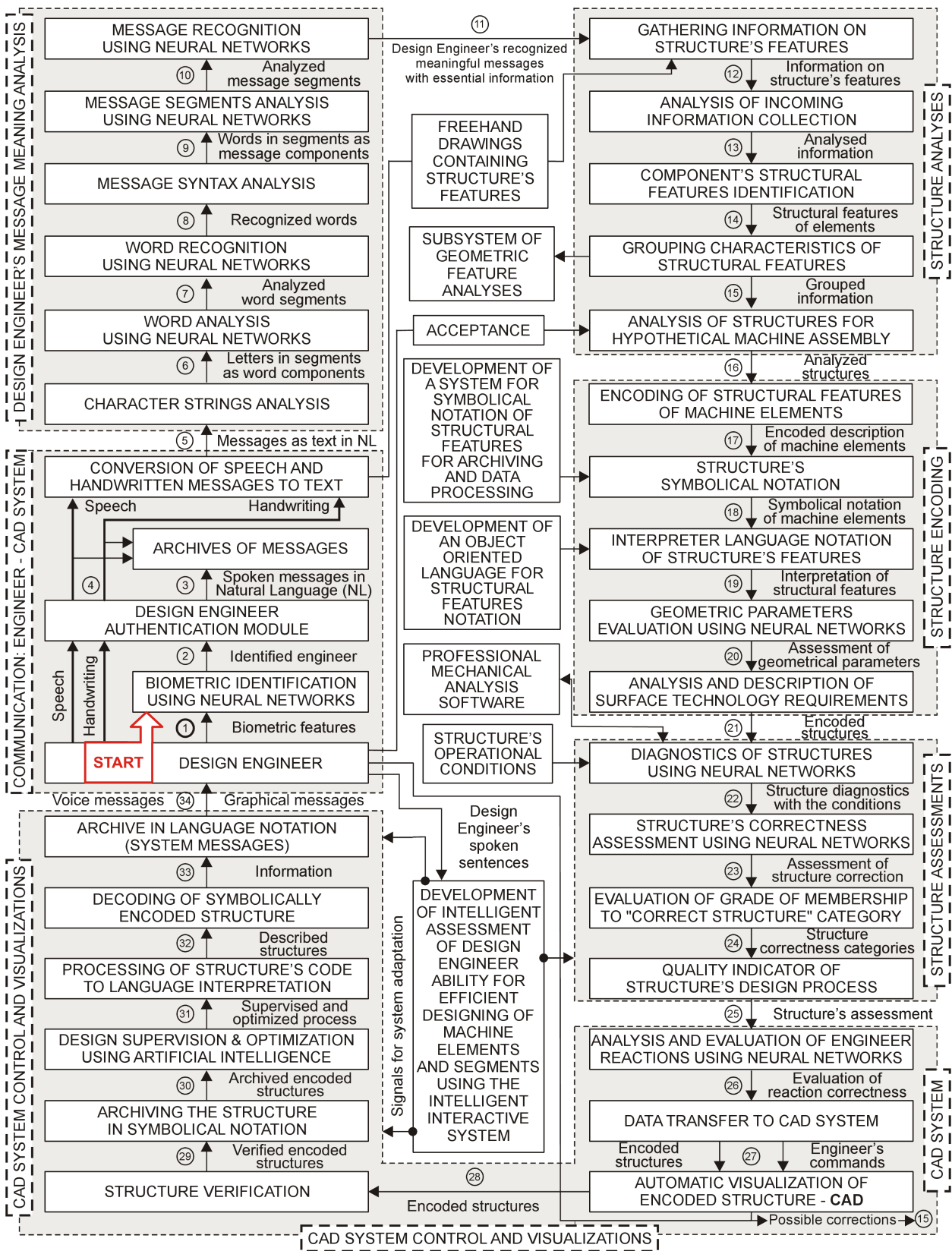


Fig. 3. The architecture of interactive automated machine components and assemblies design systems.

Implementation of intelligent interaction systems aims at increasing efficiency and comfort of designers and speeding up creation of new designs. Development of fundamentals of this new and promising design methodology comprises execution of important and labour-intensive tasks:

- development of fundamentals of effective conversion of voice messages into a symbolic object-oriented notation,
- improvement of designed methods of handwriting and graphical symbols recognition,
- research on effectiveness and correctness of design's features during its development, occurring automatically with continuous integration of incoming information regarding its attributes.

Research on creation of an intelligent and interactive automated design system comprise:

- creation of fundamental principles of structure elements features' notation development,
- creation of fundamentals of a new language for notation, archiving and processing of data related to construction,
- description (a hypertext-based object-oriented construction description language),
- creation of improved methods for voice messages recognition and processing,
- development of created handwriting recognition algorithms,
- creation and testing of fundamental procedures for using artificial intelligence to create structures' symbolic notation based on their natural-language description,
- verification of created methods for structure notation creation based on their description in a natural language for the following classes of machine components: shafts, axles, spindles, cog-wheels, disks and more,
- examination of quality of generated projects with uncertainty rating for achieving results matching their originals, and results of dependent tasks using incomplete and uncertain data,
- creation of fundamentals of a new symbolic notation language, and an object-oriented construction description language,
- determination of direction for further research.

The proposed interactive automated design system (Fig. 3) [1] contains many specialized modules and it is divided into the following subsystems [7–14]: subsystem for communication between designers and the intelligent CAD system, subsystem for design engineers' voice messages content analysis, construction analysis subsystem, construction notation subsystem, construction rating subsystem, subsystem for visualization and CAD system control,

design process optimization subsystem, construction decoding subsystem.

In this system, artificial intelligence methods allow communication by speech and natural language, resulting in analyses of design engineer's messages, analyses of constructions, encoding and assessments of constructions, CAD system controlling and visualizations. The system is equipped with several adaptive intelligent layers for human biometric identification, recognition of speech and handwriting, recognition of words, analyses and recognition of messages [15], enabling interpretation of messages, and assessments of human reactions.

Designing with natural language as the communication interface between the designer and the system

The developed fundamentals of machine elements and assemblies design processes automation using artificial intelligence contain a concept of a system for analysis of incoming information on structure's features.

A variation of the used natural language (English, Polish) corresponds to the glossary of nomenclature used during the design process, and to the libraries developed for the purpose of processing speech (as part of the voice user interface). It is possible to develop interfaces based on text, graphics, and symbols, depending on the user's preferences.

The system performs intelligent processing for analysis of descriptions of structural features of machine elements and assemblies in a natural language. This proposed intelligent semantic-based system for text corpus analysis is shown in abbreviated form in Fig. 4A. It consists of a text corpus processing subsystem and an intelligent processing subsystem for description analysis.

In the text corpus processing subsystem, words are isolated from text extracted from the text corpus, which are developed into various combinations of word clusters based on the statistical models of word sequences. The developed word clusters representing appropriate N-gram models are processed further for training hybrid probabilistic neural networks [22, 23] with learning patterns of words and clusters.

In the intelligent processing subsystem, word clusters are retrieved from descriptions of structural elements' features in a natural language using a parser. In the next step, word clusters are extracted by the parser using lexical and grammar patterns. The separated words are processed for letter strings iso-

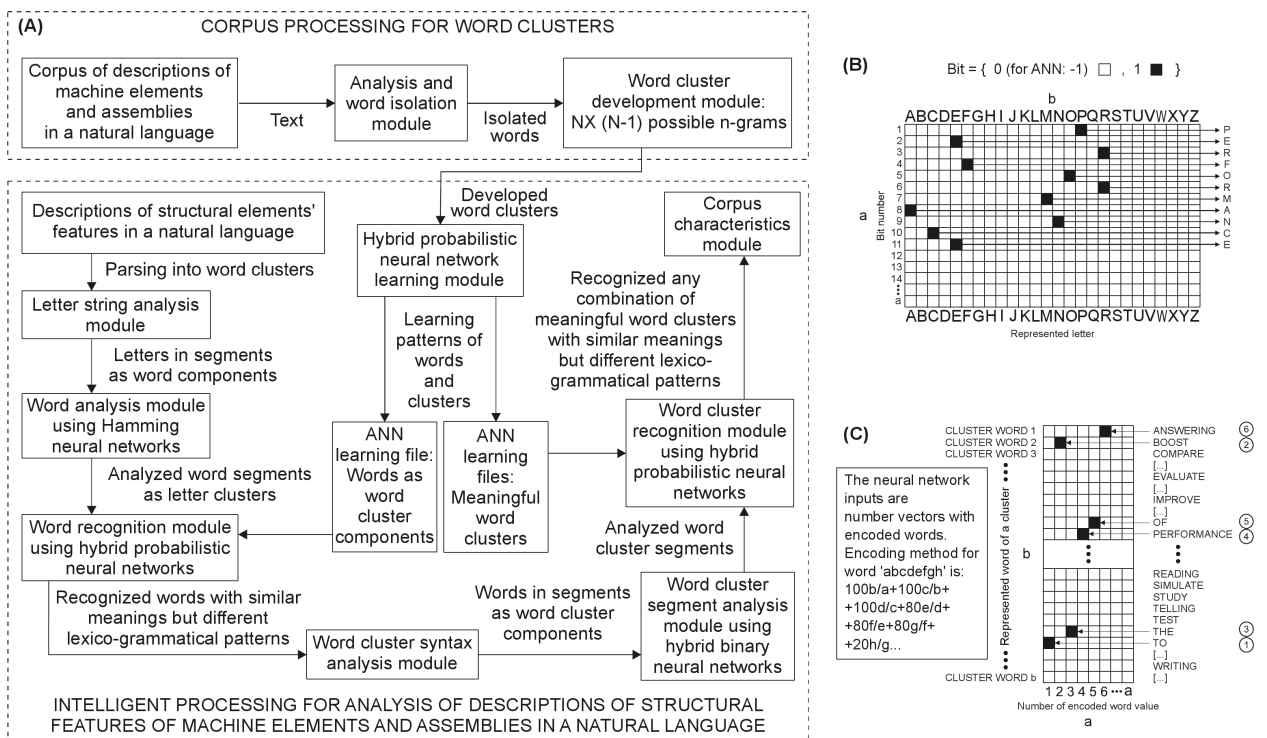


Fig. 4. Diagram of the proposed system for analysis of incoming information on structure's features, (B) inputs of the word recognition module, (C) inputs of the word cluster recognition module.

lated in segments as possible cluster word components. This analysis has been carried out using Hamming neural networks which are described in detail in [15–19]. The output data of the analysis consists of processed word segments [20, 21]. Individual word segments treated here as isolated possible components of the cluster words are inputs (Fig. 4B) of hybrid probabilistic neural networks [22, 23] for recognizing words. The networks use learning files containing words and are trained to recognize words as word cluster components, with words represented by output neurons. The intelligent cluster word recognition method allows for recognition of words with similar meanings but different lexico-grammatical patterns. In the next stage, the words are transferred to the word cluster syntax analysis module. The module creates words in segments as word cluster components properly, which are coded as vectors. Then they are processed by the module for word cluster segment analysis using hybrid binary neural networks [23].

The analyzed word cluster segments become inputs of the word cluster recognition module using hybrid probabilistic neural networks (Fig. 4C). The module uses multilayer probabilistic neural networks, either to recognize the cluster and find its meaning or else it fails to recognize it. The neural networks

of this module use learning files containing patterns of possible meaningful word clusters. The intelligent analysis and processing allow for recognition of any combination of meaningful word clusters with similar meanings but different lexico-grammatical patterns. The overall detailed results of the intelligent analysis are subject to processing for text corpus characteristics and its linguistic description including: statistical analysis, checking occurrences, and validating linguistic rules.

The proposed intelligent system for analysis of incoming information on structure's features contains hybrid probabilistic neural networks which are described in detail in [22–24] (Fig. 5). The network consists of five layers: cluster processing, cluster input, cluster pattern, summation and output layers. This pattern classifier can become effective tools for solving classification problems of lexico-grammatical structures, where the objective is to assign cases of clusters of letters or words to one of a number of discrete cluster classes. The classifier places each observed vector of cluster data x in one of the predefined cluster classes $k_i, i = 1, 2, \dots, K$ where K is the number of possible classes in which x can belong. The effectiveness of the cluster classifier is limited by the number of data elements that vector x can have and the number of possible cluster classes K .

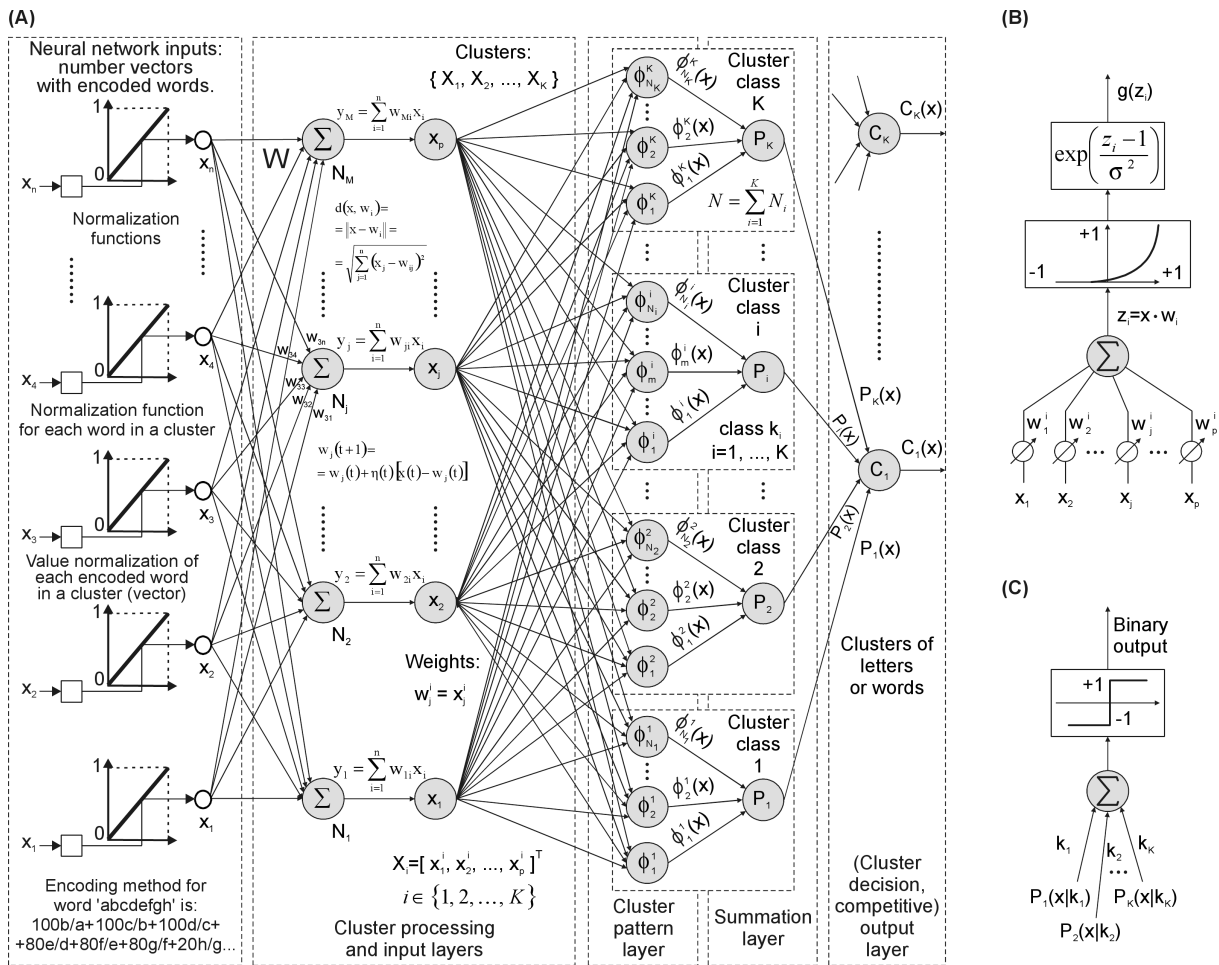


Fig. 5. The hybrid probabilistic neural networks for recognition of clusters of letters or words [23], (B) Neuron of the pattern layer, (C) Neuron of the output layer.

The probabilistic neural network offers a way to interpret the network's structure in the form of a probabilistic density function. It approximates the probability that vector x belongs to a particular class k_i as a sum of weighted Gaussian distributions centred at each cluster training sample. The output of the model is an estimate of the cluster class membership probabilities.

Antipatterns in assessment of design solutions' quality

The role of antipatterns in the machine design process

The antipattern concept, which is successfully applied in software engineering, can serve as inspiration to using it in other engineering-related tasks. The most favourable scenario would be a task based on heuristic concept creation, in which the designer wouldn't be limited to using patterns, but instead he

would be expected to come up with innovative solutions characterized by a high level of non-obviousness in relation to the established knowledge.

Issues connected with optimal design have been elaborated in many publications on subjects like design method selection [25, 26]. Methods such as programming tasks, equality constraints with excess variables, dynamic programming, and optimization of variable-topology structures are useful when it comes to accepting decisions regarding the value of a structural feature. When making such decisions, stress and distortions, as well as correspondence of geometries and topologies from a finite set of combinations with other structural elements are taken into account. However, usefulness of the mentioned methods in conception development is limited.

The set of methods used for solving problems connected with designing machine components comprises many topics related to data analysis and exploration [27–31]. This set of methods can be expanded with many more details, such as: multidimensional

mensional scaling, tree-based classifiers, neural classifiers, search algorithms (including heuristic ones), data-clustering algorithms, and content-based search methods.

The drawback of using design patterns is that not only do they impede innovation, but also they increase the odds of a scenario, where poor application of a pattern produces additional design errors, or contributes to delayed detection of such errors and may result in repeated application of solutions worse than the possible correct ones, coming at the same cost. Antipatterns are generic definitions of possible instances of incorrect solutions to design problems. Work on theoretical and experimental basics of antipattern definition methodology, their features and exceptional matching factor measure evaluation clas-

sification ability are important lines of research, important to the automated systems of aided design process, which aspire to being prevalent in the near future.

Antipatterns can comprise multiple attributes: (AFD) the function served by the designed component or structural system, (ASG) the component's structural group, (AIS) the definition of an incorrect solution, (AEC) the error cause, (AER) the error result, (AEI) the error's importance, (AES) other error similarity, and design solution features (AF1), (AF2)..(AFn). The distinct features of antipatterns are their geometrical and physical properties. Antipatterns can also be classified based on the root causes of flawed designs, the most common of which are presented in Table 1.

Table 1
Root causes of designs with errors.

	The name of an ERROR's cause for each symbol
EPI	a lack of correct evaluation of PRODUCT'S IMPORTANCE for the future manufacturing program
EOE	a lack of project's economical OUTCOME EVALUATION
ETP	an underestimation of TECHNOLOGICAL PROBLEMS' impact on the product's quality
EMR	a lack of analysis of MAINTENANCE REQUIREMENTS
ECP	a lack of analysis of CUSTOMER PREFERENCES in relation to quality, modernity, durability, ease of operation, and visual design
EDG	DOMINANCE of GRAPHICAL tasks over conceptual analysis in the process of structure design
EDT	DOMINANCE of TRUST in results of massive data processing over trust in knowledge
ESE	a deferral of SOLUTION EVALUATION, with overemphasized trust in ease of maintenance
EAF	an introduction of an ABUNDANT FUNCTIONALITY
ECF	incorrectly COMBINED FUNCTIONS in the designed elements
EAE	existence of ABUNDANT ELEMENTS, which do not serve any functions
ELH	a LACK of HARMONY in functionality, incorrect division of functions into elements of the structure
ELV	a LACK of a VERIFICATION (competent and multicriterial) at each stage, and a lack of quality evaluation, supposed to be carried out with the required procedures
EDR	DEADLINE RUSH due to short time
EAT	incorrectly ASSIGNED TASKS within the team
EDV	either a neglect or awareness of DEFERRED VERIFICATION of structural design effects and unclear evaluation of responsibility
EAS	stubborn ADHERENCE to SOLUTIONS (known and otherwise correct)
EIM	INCOMPATIBLE METHODOLOGIES, preferences, and design criteria among team members
EOC	application of OLD COMPONENTS and norms in the designed structure
ELE	a LACK of EVALUATION of a need for scalable, modular and unifiable solutions
ENE	an excessive (implying a lack of integration) or an insufficient (implying differentiation) NUMBER of ELEMENTS
EDS	too easily DEFORMABLE STRUCTURE
ECM	incorrect CHOICE of MATERIALS
EDI	DIFFICULT INSTALLATION, limited or costly parts replacement ability
EQI	an insufficient emphasis on the QUALITY of INTERACTING surfaces and the properties of outer layers
EAA	an excessively limited ABILITY to ADAPT structural features to operating conditions
EED	a lack of EVALUATION of DYNAMICS of operating conditions
EEO	a lack of EVALUATION of OCCURRENCES, which are not characteristic of the typical operation (conditions pacing)

Evaluation measures of correspondence between the designed structure and antipatterns

Determination of measures for evaluation of correspondence between the designed structure and antipatterns requires normalization of compared features. Antipatterns' linguistic attributes can be used for selection by (AFD), (ASG), (AIS), (AEC), (AER), (AEI), (AES), (AF1), (AF2) ... (AFn). Input attributes require normalization to fit in the (0,1) range. Below are listed examples of attributes used for antipatterns selection:

- (AFD) – the function served by the designed element or aggregate of components – rotating element, housing, hood, frame, cantilever, adaptor, guide, cylinder, valve, handle, floor, pipe, spring etc.,
- (ASG) – component's structural groups, for AFD='rotating' possible options are: straight or step shaft, spindle, disk, ring, roller, ball, and additionally: crankshaft, axle, toothed wheel etc.,
- (AIS) – incorrect solution's definition, for AFD='rotating' and ASG='shaft' such a definition can be: easy deformation, excessive difference between steps' diameters, incorrect diameter tolerance, high level of steps, lack of exit zone, incorrect dimensions of splineway etc.

Evaluation measures of correspondence will depend on the set of parameters typical of the aforementioned attributes, and they should be subjected to normalization, so that they fall within the range of (0,1) using fuzzy sets. Error's importance (AEI) should be – after normalization – attached to the set of attributes describing structural features. Normalization by the shape of the classification function should take into account the meaning of the value to the result of normalization (flow direction). The developed methodology for design solutions evaluation with antipatterns has been presented in Fig. 6. An application example of the design solutions evaluation methodology is a seven-step shaft ($n = 7$), presented in Fig. 7.

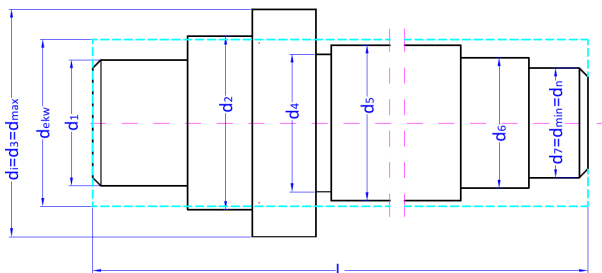


Fig. 6. A step shaft.

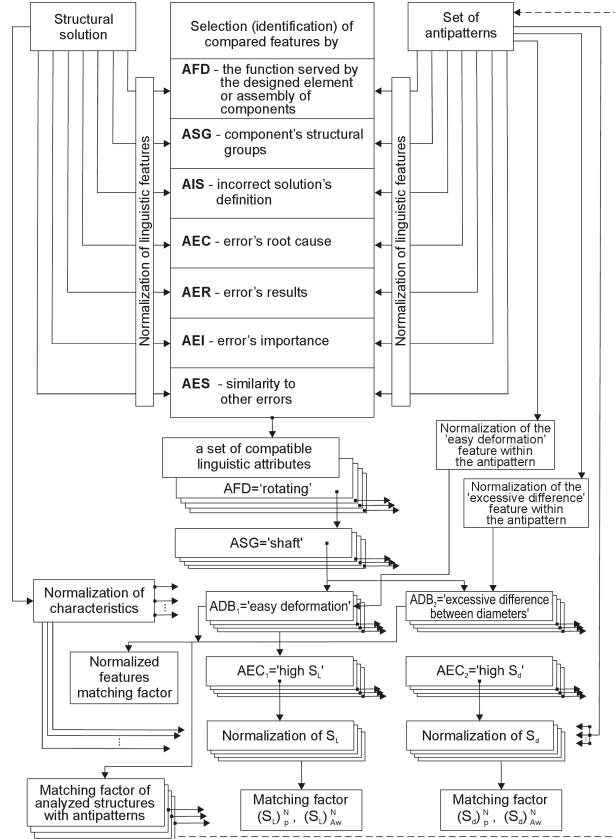


Fig. 7. Design solutions and antipattern correspondence evaluation methodology.

The selected shaft antipattern has been, for the designed element's function AFD='rotating', has been chosen from the structural group ASG='step shaft'. The incorrect solution AIS1='easy deformation' is characterized by shaft buckling due to an axial force. The cause of the error AEC1='high SL=L/dekw' is such, that a flexible shaft, due to axial compression, may be subject to buckling (bending). In the case of the given example importance AEI1 is high and it comprises similarity to other errors AES1: stability loss, which leads to inevitable physical destruction, which usually involves other interacting elements, small stiffness, large deviations, noise, vibrations, unstable operation, etc.

Another incorrect solution is AIS2='excessive diametral steps', the cause of which AEC2='high S_d ' may lead to excessive stress cumulation caused by indentation. The S_d value is given by the formula:

$$S_d = \sqrt{S_s \cdot S_m}, \tag{1}$$

where

$$S_s = \max \in \left\{ \frac{d_j}{d_{j-1}}, \frac{d_k}{d_{k+1}} \right\}, \tag{2}$$

$$j \in \{2, 3, \dots, n\}, \quad k \in \{1, 2, \dots, n-1\}, \tag{3}$$

$$d_i \in \{d_1, d_2, \dots, d_n\}, \tag{4}$$

$$S_m = \frac{d_{\max}}{d_{\min}}, \quad (5)$$

$$d_{\max} = \max \{d_1, d_2, \dots, d_n\}, \quad (6)$$

$$d_{\min} = \min \{d_1, d_2, \dots, d_n\}. \quad (7)$$

To lessen the indentation's impact it is advised to assume $S_m = d_{\max}/d_{\min} \leq 1,2$ and also to introduce a possibly large intermediate radius or a conical transition. The importance of the AEI₂ error is considerable and it comprises similarities to other errors AES₂: stress cumulation, fatigue endurance, difficult installation, etc.

More examples of applied methodology for design solution and antipattern correspondence evaluation are presented in Figs. 8, 9 and 10. Selected

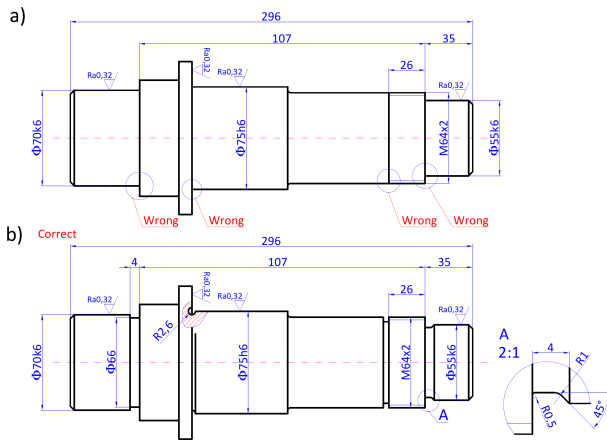


Fig. 8. A design of a shaft manufactured with machine cutting with tool entrance and exit zones: a) antipattern, b) correct design.

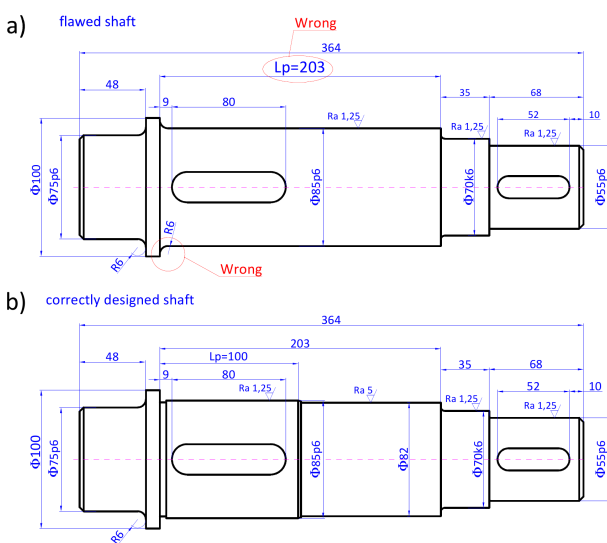


Fig. 9. Design of a shaft with friction fit's mounting surfaces of different lengths.

shaft antipatterns for the designed element's function AFD='rotating' have been chosen from the structural group ASG='step shaft'.

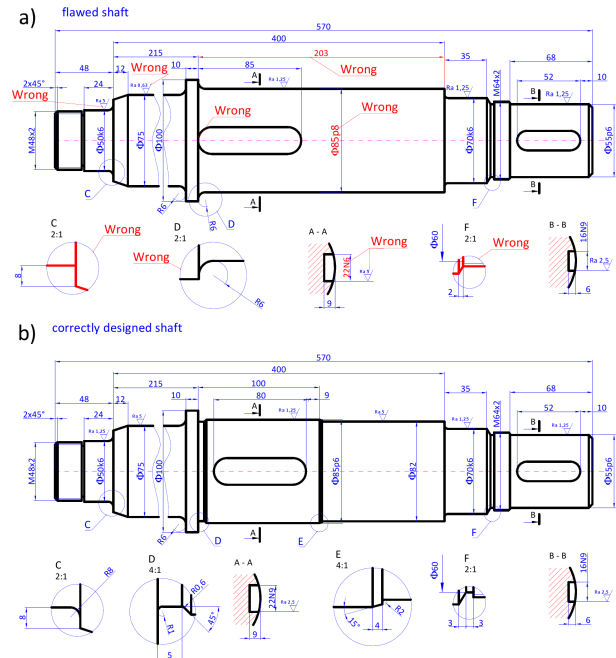


Fig. 10. A machine shaft with highlighted errors: a) an antipattern, b) correct design.

In the example presented in Fig. 8 AIS='tool exit zone' is an incorrect solution, the cause of which AEC='tool exit zone's shape' is the fact, that the shaft doesn't provide a zone for the tool to freely move around and exit certain zones during the manufacturing process. In the case of the given example importance of AEI is high and it comprises similarity to other errors AES: manufacturing errors, producibility, poor installation, etc.

Figure 7 presents a shaft designed with tool's reach and exit ability in mind, in accordance with the producibility criterion and not displaying the aforementioned antipattern characteristics.

In the example presented in Fig. 9 the incorrect solution is AIS₁'=installation of fitted elements', the cause of which AEC₁'=the L_p fitted surface's length' is such, that the shaft has a mounting surface with the diameter $d = F85p6$ and length $L_p = 203$ mm, which is much longer than the width of the element mounted on it, e.g. a toothed wheel. The cause of the error (AEC₁) results in difficult installation – that is forcing the element through a significant length. In the case of the given example the importance of the error AEI₁ is average and it comprises similarity to other errors AES₁, such as: causing damage of the mounting surface, costs of forcing the shaft

through, etc. At the same time, the shaft has an error AIS_2 ='imprecise axial fitting of the mounted components'. The cause of the error AEC_2 ='filleting radius R ' indicates the possible imprecision of components axial mounting (lack of support for the fitted component from the end collar), in which the size of the mounted component's phase F is smaller than the radius of the shaft step's filleting R . The error's importance AEI_2 is average, and its similarities shared with other errors AES_2 are: noise, unstable operation, vibrations, increased wearing out of the fitted elements' surfaces, etc.

An example of an attributes set for the aforementioned cases can be as follows: (AEIN) – error importance normalized value, (LVN) – normalized length value L to the diameter equivalent d_{ekw} , (DVN) – maximal S_m diameter value normalized to the smallest one, (STN) – normalized ratio of surface's roughness parameter S_t to this surface's size tolerance T , etc.

The measure of correspondence with an antipattern for the given examples can be the sum of absolute distance values between the parameters of an antipattern and the designed element, the sum of squared distances or their geometrical distance.

Before performing the evaluation of correspondence between the structural features of a designed machine elements and antipatterns, it is possible to verify the ability to classify of the used parameters. The requirement would be to have a set of design solutions, for which we want to determine the differentiation ability with each of the parameters.

Presented antipatterns comprise a generic definition of possible incorrect design solutions. The developed methodology for evaluation of correspondence between a design solution and an antipattern, illustrated with step shafts, allows to normalize parameter characteristics of selected features of analysed design solutions, and, then, to determine the antipattern's and the analysed structure's matching factor. Work on theoretical and experimental basics of antipattern definition methodology, their features and exceptional matching factor measure evaluation classification ability are important lines of research, important to the automated systems of aided design process, which aspire to being prevalent in the near future.

Conclusions

Works aiming to develop basics of automation of processes in designing machine elements

and assemblies with the use of artificial intelligence in uncertainty and unrepeatability of processes have been started. In automated design systems, which use a natural-language description of structural features and an intelligent interface of natural speech and hand-drawn sketches, application of design antipatterns – especially in combination with artificial-intelligence methods – carries a lot of meaning to effectiveness and development of such systems.

The devised concepts for creating a design and object-oriented notation of information about its characteristics, in comparison to the known and used solutions, have benefits resulting from lack of constraints related to graphical operations performed by the designer, as well as from the advantages of object-oriented data notation and lack of constraints (amount and type of data) related to the structure of classes representing object types.

The benefits of using antipatterns are lack of constraints when designing new solutions and lesser likelihood of making errors typical of antipatterns.

The devised solutions can be implemented in commercial products, which may considerably accelerate the processes of designing simple machine components, make more feasible the creation of more complex designs, as well as decrease the impact of limitations related to the amount of work that can be invested into potential corrections and analyses during the design creation process. However, in the case of elements characterized by complex shapes, difficult to define in linguistic terms, graphical interface-based methods will still find their application.

This project was financed from the funds of the National Science Centre (Poland) allocated on the basis of the decision number DEC-2012/05/B/ST8/02802.

References

- [1] Kacalak W., Majewski M., *New Intelligent Interactive Automated Systems for Design of Machine Elements and Assemblies*, ICONIP 2012, Doha, Qatar, in Lecture Notes in Computer Science, 7666, IV, 115–122, 2012.
- [2] Zeng Y., Horváth I., *Fundamentals of next generation CAD/E systems*, Computer-Aided Design, 44, 10, 875–878, 2012.
- [3] Goel A., Vattam S., Wiltgen B., Helms M., *Cognitive, collaborative, conceptual and creative – Four*

- characteristics of the next generation of knowledge-based CAD systems: A study in biologically inspired design*, Computer-Aided Design, 44, 10, 879–900, 2012.
- [4] Piegl L.A., *Ten challenges in computer-aided design*, Computer-Aided Design, 37, 4, 461–470, 2005.
- [5] Kou X.Y., Xue S.K., Tan S.T., *Knowledge-guided inference for voice-enabled CAD*, Computer-Aided Design, 42, 6, 545–557, 2010.
- [6] Kou X.Y., Tan S.T., *Design by Talking with Computers*, Computer-Aided Design and Applications, 5, 1–4, 266–277, 2008.
- [7] Kacalak W., Majewski M., *Intelligent system for automatic recognition and evaluation of speech commands*, ICONIP 2006, Hong Kong, China, in Lecture Notes in Computer Science, 4232, 298–305, 2006.
- [8] Majewski M., Kacalak W., *Intelligent system for natural language processing*, ICIC 2006, Kunming, China, Lecture Notes in Computer Science, 4114, 742–747, 2006.
- [9] Kacalak W., Majewski M., *Natural Language Human-Robot Interface Using Evolvable Fuzzy Neural Networks for Mobile Technology*, ICIC 2009, Ulsan, South Korea, Lecture Notes in Computer Science, 5754, 480–489, 2009.
- [10] Kacalak W., Majewski M., *E-learning systems with artificial intelligence in engineering*, ICIC 2009, Ulsan, South Korea, Lecture Notes in Computer Science, 5754, 918–927, 2009.
- [11] Kacalak W., Majewski M., Zurada J.M., *Intelligent e-learning systems for evaluation of user's knowledge and skills with efficient information processing*, ICAISC 2010, Zakopane, Poland, Lecture Notes in Computer Science, 6114, 508–515, 2010.
- [12] Lipinski D., Majewski M., *System for Monitoring and Optimization of Micro- and Nano-Machining Processes Using Intelligent Voice and Visual Communication*, IDEAL 2013, Hefei, China, Lecture Notes in Computer Science, 8206, 16–23, 2013.
- [13] Lipinski D., Majewski M., *Interactive Hybrid Systems for Monitoring and Optimization of Micro- and Nano-machining Processes*, ICONIP 2013, Daegu, Korea, Lecture Notes in Computer Science, 8227, 360–367, 2013.
- [14] Lipinski D., Majewski M., *Intelligent Monitoring and Optimization of Micro- and Nano-Machining Processes*, Mechatronics 2014, Lodz, Poland, Advances in Intelligent Systems and Computing, 317, 101–110, 2015.
- [15] Majewski M., Zurada J.M., *Sentence recognition using artificial neural networks*, Knowledge-Based Systems, 21, 7, 629–635, 2008.
- [16] Kacalak W., Stuart K., Majewski M., *Intelligent natural language processing*, ICNC 2006, Xi'an, China, Lecture Notes in Computer Science, 4221, 584–587, 2006.
- [17] Stuart K., Majewski M., *Selected problems of knowledge discovery using artificial neural networks*, ISNN 2007, Nanjing, China, Lecture Notes in Computer Science, 4493, 1049–1057, 2007.
- [18] Kacalak W., Stuart K., Majewski M., *Selected problems of intelligent handwriting recognition*, IFSA 2007, Cancun, Mexico, Advances in Soft Computing, 41, 298–305, 2007.
- [19] Stuart K., Majewski M., *A new method for intelligent knowledge discovery*, IFSA 2007, Cancun, Mexico, Advances in Soft Computing, 42, 721–729, 2007.
- [20] Stuart K., Majewski M., *Artificial creativity in linguistics using evolvable fuzzy neural networks*, ICES 2008, Prague, Czech Republic, Lecture Notes in Computer Science, 5216, 437–442, 2008.
- [21] Stuart K., Majewski M., *Evolvable neuro-fuzzy system for artificial creativity in linguistics*, ICIC 2008, Shanghai, China, Lecture Notes in Computer Science, 5227, 46–53, 2008.
- [22] Stuart K., Majewski M., Botella-Trelis A., *Selected problems of intelligent corpus analysis through probabilistic neural networks*, ISNN 2010, Shanghai, China, Lecture Notes in Computer Science, 6064, 268–275, 2010.
- [23] Stuart K., Majewski M., Botella-Trelis A., *Intelligent semantic-based system for corpus analysis through hybrid probabilistic neural networks*, ISNN 2011, Guilin, China, Lecture Notes in Computer Science, 2011, 6675, 83–92.
- [24] Specht D.F., *Probabilistic neural networks*, Neural Networks, 3, 1, 109–118, 1990.
- [25] Brown W.J., McCormick H.W., Thomas S.W., *AntiPatterns in Project Management*, New York: John Wiley & Sons, 2000.
- [26] Riel A.J., *Object-Oriented Design Heuristics*, Reading, MA: Addison-Wesley, 1996.
- [27] Dostatni E., Diakun J., Grajewski D., Wichniarek R., Karwasz A., *Functionality assessment of ecode-sign support system*, Management and Production Engineering Review, 6, 1, 10–15, 2015.
- [28] Zhong H., Levalle R.R., Moghaddam M., Nof S.Y., *Collaborative Intelligence - Definition and Measured*

- Impacts on Internetworked e-Work*, Management and Production Engineering Review, 6, 1, 67–78, 2015.
- [29] Wasilewska B., Knosala R., *Methods for Stimulating Creativity in Production Engineering*, Management and Production Engineering Review, 5, 4, 76–84, 2014.
- [30] Jakubowski J., Peterka J., *Design for manufacturability in virtual environment using knowledge engineering*, Management and Production Engineering Review, 5, 1, 3–10, 2014.
- [31] Ahram T.Z., Karwowski W., *Engineering sustainable complex systems*, Management and Production Engineering Review, 4, 4, 4–14, 2013.