ISSN 0146-4116, Automatic Control and Computer Sciences, 2013, Vol. 47, No. 6, pp. 331–341. © Allerton Press, Inc., 2013.
Original Russian Text © G.F. Krivoulya, A.S. Shkil', D.Ye. Kucherenko, 2013, published in Avtomatika i Vychislitel'naya Tekhnika, 2013, No. 6, pp. 59–72.

Analysis of Production Rules in Expert Systems of Diagnosis

G. F. Krivoulya, A. S. Shkil', and D. Ye. Kucherenko

Kharkov National University of Electronics,
Lenin ave., 14, Kharkov, 61126, Ukraine
e-mail: krivoulya@i.ua, shkil@opentest.com.ua, d_zin@ukr.net

Received March 8, 2013; in final form, April 30, 2013

Abstract—This paper examines the problem of the quality test of production rules that is basic for judgment about the technical state of a computer system. The object of the diagnosis is software. Its quality is assessed on the basis of an expert appraisal of the chosen attributes (diagnostic features) with the use of rules and procedures of fuzzy logic. The developed formal procedures to check the production rules for correctness by analysis of the cubic form of their presentation on the basis of the proposed alphabet and procedures are given.

Keywords: decision making support system, technical diagnosis, fuzzy logic, knowledge base, production rules

DOI: 10.3103/S0146411613060060

1. INTRODUCTION

A feature of the present-day computer systems of information processing and control is that, as man—machine systems, they contain hardware, software, and personnel. It is supposed that violation of the working power of any of these three components leads to the malfunction of the system as a whole. A method to improve the reliability of computer systems is technical diagnosis. In the diagnosis of a computer system, due to the complexity of the object and the lack of binary templates of its correct functioning, methods of expert diagnosis are often used with expert appraisals in a natural language as input information for the decision making support system (DMSS) on the technical state of the object. In dealing with this problem, it makes sense to use the mathematical apparatus of fuzzy logic within the system of fuzzy logical inference on the basis of linguistic variables (LV) [1]. This mechanism provides transparency of the algorithm of decision making, easy correction, and provides an opportunity to take into consideration the quantitative and qualitative characteristics of the object of diagnosis.

If the object of diagnosis is a software, then DMSS can be used in the analysis of its quality. The basis for judgment on the quality of software means is the international standard ISO 9126:1991 "Information Technology. Software Product evaluation. Quality characteristics and guidelines for their use." According to this standard, the software quality is defined as a set of characteristics (functional suitability, reliability, usability, efficiency, maintainability, portability) allowing all the concerned parties to satisfy their needs.

The core of any fuzzy inference system is the knowledge base which is represented in the form of production rules (PR). There is quite a large number of methods for PR creation, ranging from ones informally composed by an expert and based on his idea of the object of diagnosis to the heuristic and formal algorithms of synthesis of production rules [2, 3].

Despite the ways of PR creation being various, all of them should meet the universal formal requirements of correctness without account the semantic aspect of the PR as such. A correct PR system should meet some formal requirements, namely, to be complete, minimal, coherent, and consistent [4, 5].

As the matter stands today, there are a vast number of publications devoted to the problem of creating knowledge bases in the form of production rules and of their further use in expert systems of various applications. However, only some of them touch the problems of verification of production rules. Thus, for example, in [6] an approach of PR verification of redundancy is suggested, but it is shown that the presence of redundancy is not yet an indication of an error. In [7], the author also treats such features of production rules as synthetic and semantic redundancy, inconsistency; he analyzes situations where one rule is covered by another, cyclic features of sets of rules, the insignificance of conditions, and deadlock and unused rules. In these and other such works, for PR analysis, use is made of their classical form in a natural language. However, this approach works only in the manual analysis of a small number of PR, while the automation of this process requires one to develop an additional lexical analyzer allowing work with the rules presented in the natural language.

This work is devoted to the formalization of the presentation of production rules in the knowledge base and to the development of procedures of their analysis for correctness. In order to achieve this target, we have to do the following.

(1) develop a compact form of the PR record;

(2) develop procedures of formal verification of the PR for correctness;

(3) show the possibility to use the cubic form of the PR presentation as a knowledge base for fuzzy inference system with its next implementation in the Matlab system.

2. CUBIC FORM OF THE PRESENTATION OF THE PRODUCTION RULES

We consider the object of diagnosis, whose technical condition in the process of expert diagnosis is determined by four diagnostic features. The estimation of the diagnostic features (DF) is performed proceeding from a three-score scale, i.e., low (L), average (A), and high (H). The diagnosis result (DR) has five levels of gradation: very low (VL), low (L), average (A), sufficient (S), and high (H). For example, if the number of diagnostic features is four, then the number of inputs LV is $n = 4$, the number of terms of each variable is $m = 3$, and the number of terms of the output LV is 5.

To simplify our further analysis, we assume that the weights of all the input LV are equal. The ranges of the changes in the variables and the type of membership functions do not influence the ways of the PR synthesis, so the principles of their selection are ignored in this work. The process of the synthesis of production rules as a heuristic algorithm is found in [3]. Here is a PR fragment presented in the classical expanded form:

$$\begin{array}{ll} & \{\mathrm{DF}_1 = \mathrm{L}\}\ \mathrm{AND}\ \{\mathrm{DF}_2 = \mathrm{L}\}\ \mathrm{AND}\ \{\mathrm{DF}_4 = \mathrm{A}\}\ \mathrm{OR} \\ \mathrm{IF} & \{\mathrm{DF}_1 = \mathrm{A}\}\ \mathrm{AND}\ \{\mathrm{DF}_3 = \mathrm{L}\}\ \mathrm{AND}\ \{\mathrm{DF}_4 = \mathrm{L}\}\ \mathrm{OR} \\ & \{\mathrm{DF}_1 = \mathrm{L}\}\ \mathrm{AND}\ \{\mathrm{DF}_2 = \mathrm{L}\}\ \mathrm{AND}\ \{\mathrm{DF}_3 = \mathrm{A}\}, \\ \mathrm{THEN} & \mathrm{DR} = \mathrm{VL}. \end{array} \tag{1}$$

Due to the use of structures of the natural language, the expended form of the PR presentation (1) is easily readable, but the further formal analysis of the PR is difficult, and with an increase in their number is practically impossible.

To formalize the process of the synthesis and analysis of the PR base, we use the presentation of the LV term conjunctions in the vector form, similarly to cubic presentations of logic functions in a multiform alphabet. As a rule, the cubic calculation is used in the construction of tests and the analysis of digital schemes presented in tabular form [8]. Thus, the PR from (1) is presented in the cubic form as

$$\mathrm{DR}^{\mathrm{VL}} = \{\mathrm{LLXA}, \mathrm{AXLL}, \mathrm{LLAX}\}. \tag{2}$$

Each conjunction from (2) has a corresponding cube, the rank of which is determined by the number of insignificant LV (coordinates equal to X). In the present context, X is understood as follows: an LV can take any value because the change of its value does not lead to a change in the output LV; i.e., it is insignificant.

For the formalization and, next, the automation of the work with the given cubic form of the PR presentation, we introduce a many-valued alphabet of cubic calculation A3 consisting of three primitives L, A, H ($m = 3$). These symbols take clear-cut values, while remaining as the equivalents to the terms of fuzzy variables.

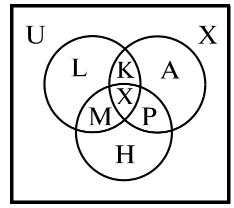
The number of symbols of this alphabet A3 will be $2^m = 2^3 = 8$:

$$\mathrm{A3} = \{\mathrm{L},\mathrm{A},\mathrm{H},\mathrm{X} = \{\mathrm{L},\mathrm{A},\mathrm{H}\},\ \ \mathrm{K} = \{\mathrm{L},\mathrm{A}\},\ \ \mathrm{P} = \{\mathrm{A},\mathrm{H}\},\ \ \mathrm{M} = \{\mathrm{L},\mathrm{H}\},\varnothing(\mathrm{U})\}, \tag{3}$$

where $\mathrm{X} = \mathrm{L} \cup \mathrm{A} \cup \mathrm{H}$ is a universe, and U is, on the one hand, a symbol of the alphabet serving to close the alphabet relative to the theoretic-multiple operations and, on the other hand, it is the result of the intersection operation and it notifies the empty set $\varnothing$.

Figure 1 presents a Euler diagram showing the interconnection between all the symbols of the suggested alphabet.

Thus, each PR can be presented as a set of n-digit vectors (cubes of appropriate rank), where n is the number of input LV. The value of each digit of the vector is determined by a many-valued alphabet assigned by m primitives. In this, each conjunction from the expanded form of the PR presentation is presentable by one n-digit vector. The notion "vector" in this aspect is a synonym of the notion "cube" in the cubic calculation. The rank of the cube is determined by the number of symbols "X" in conjunction. For example, the cube LLLL has the 0th rank, and the cube LLLX the 1st rank.



Euler diagram for alphabet A3.

We introduce theoretical-multiple operations of intersection, union, complement, and algebraic sum in the A3 alphabet.

The operation of intersection ($\cap$) of two n-dimensional vectors $A = a_1, a_2, \ldots, a_n$ and $B = b_1, b_2, \ldots, b_n$, where n is the number of vector digits, is denoted as $C = A \cap B$, where $C = c_1, c_2, \ldots, c_n$, and determined in the following way:

$$C = A \cap B = \begin{cases} \varnothing, & \text{if } a_i \cap b_i = U, \text{ at least for one of } n \text{ digits}, \; i = \overline{1,n} \\ (a_1 \cap b_1),\, (a_2 \cap b_2),\, \ldots, (a_n \cap b_n) & \text{otherwise.} \end{cases} \tag{4}$$

The intersection operation is digit-wise (the result is found for each digit independently) and the rules of its fulfillment in each digit are shown in Table 1.

A special case of the intersection operation is the absorption operation ($\in$). The vector A is absorbed by the vector B(A belongs to B, where $A \in B$), if $A \cap B = A$. If in the absorption similar vectors operate, then the result will be one of these vectors.

The operation of the integration ($\cup$) of two n-dimensional vectors $A = a_1, a_2, \ldots, a_n$ and $B = b_1, b_2, \ldots, b_n$, where n is the number of digits, is denoted as $C = A \cup B$, where $C = c_1, c_2, \ldots, c_n$, and is found as

$$C = A \cup B = \{(a_1 \cup b_1), (a_2 \cup b_2), \ldots, (a_n \cup b_n)\}. \tag{5}$$

The operation of integration is digital, and the rules of its fulfillment in each digit are shown in Table 2.

The operation of addition for one n-dimensional vector $A = a_1, a_2, \ldots, a_n$, where n is the number of digits, is denoted as $C = \tilde{A}$, where $C = c_1, c_2, \ldots, c_n$. By analogy with the analytical description of logical functions, the operation of addition is often called the logical inversion (negation), and the rules for its fulfillment in each digit are shown in Table 3.

The operation of the algebraic sum (an analogue of the logical sum by module 2) of two n-dimensional vectors $A = a_1, a_2, \ldots, a_n$ and $B = b_1, b_2, \ldots, b_n$, where n is the number of vector digits, is denoted by $C = A + B$, where $C = c_1, c_2, \ldots, c_n$, and defined as

$$C = A + B = \{(a_1 + b_1), (a_2 + b_2), \ldots, (a_n + b_n)\}, \tag{6}$$

$$\text{where } a_i + b_i = (a_i \cap \overline{b_i}) \cup (\overline{a_i} \cap b_i). \tag{7}$$

The operation of the algebraic sum is digit-wise, and its implementation rules in each digit are shown in Table 4.

Table 1. Intersection operation

$\cap$	L	A	H	K	P	M	X	U
L	L	U	U	L	U	L	L	U
A	U	A	U	A	A	U	A	U
H	U	U	H	U	H	H	H	U
K	L	A	U	K	A	L	K	U
P	U	A	H	A	P	H	P	U
M	L	U	H	L	H	M	M	U
X	L	A	H	K	P	M	X	U
U	U	U	U	U	U	U	U	U

Table 2. Union operation

∪	L	A	H	K	P	M	X	U
L	L	K	M	K	X	M	X	L
A	K	A	P	K	P	X	X	A
H	M	P	H	X	P	M	X	H
K	K	K	X	K	X	X	X	K
P	X	P	P	X	P	X	X	P
M	M	X	M	X	X	M	X	M
X	X	X	X	X	X	X	X	X
U	L	A	H	K	P	M	X	U

Table 3. Addition operation

A	L	A	H	K	P	M	X	U
$\tilde{A}$	P	M	K	H	L	A	U	X

Table 4. Operation of algebraic sum

+	L	A	H	K	P	M	X	U
L	U	K	M	A	X	H	P	L
A	K	U	P	L	H	X	M	A
H	M	P	U	X	A	L	K	H
K	A	L	X	U	M	P	H	K
P	X	H	A	M	U	K	L	P
M	H	X	L	P	K	U	A	M
X	P	M	K	H	L	A	U	X
U	L	A	H	K	P	M	X	U

The code distance d between two vectors (cubes) will be the number of different digits (coordinates), i.e., the number of characters not equal to U in the vector obtained as a result of their algebraic sum. For example, if A = LLLH and B = LLLA, then C = A + B = UUUP. Then, the code distance between them is $d = 1$ (the result of the sum is only one value not equal to U).

3. ANALYSIS OF THE PRODUCTION RULES FOR CORRECTNESS

As the production rules of expert diagnosis are formed, generally, by an expert based on his subjective ideas on the object of diagnosis, they need to be verified for correctness, namely, verified for completeness, consistency, coherence, and minimalistic.

Now we consider a PR system where to each of five terms (VL, L, A, S, H) of the output PR y a different set of cubic forms of the vector record $k^{\langle \mathrm{VL,L,A,S,H} \rangle} = \{k_j^{\mathrm{VL}}, k_j^{\mathrm{L}}, k_j^{\mathrm{A}}, k_j^{\mathrm{S}}, k_j^{\mathrm{H}}\}$ corresponds, where k is the set of vectors belonging to one of the terms of the output LV (VL, L, A, S, H), where $j = 1, 2, 3...$ is the ordinal number of the vector inside each set k. For example, if the term VL has three vectors, then we obtain the following record: k_1^{VL}, k_2^{VL}, k_3^{VL}.

A knowledge base is complete if for each input vector $\forall T = (T_1, T_2, .., T_{m^n})$ the mechanism of fuzzy inference can obtain a certain value of the output variable $y \in E_y$; i.e., there is at least one rule that assigns to the input vector T_i the linguistic value of the output variable y (linguistic completeness).

Thus, the collection of sets $k^{\langle VL, L, A, S, H\rangle}$ should cover all the m^n of input vectors T_i, where $i = \overline{1, m^n}$, m is the number of terms of the input variable, n is the number of input variables, and m^n is the number of all possible combinations of the input LV (vectors). In other words, there is no such vector that would not belong to some set:

$$\begin{gathered} T_i \in k^{\langle VL,L,A,S,H\rangle} \Leftrightarrow T_i \cap k^{\langle VL,L,A,S,H\rangle} = T_i, \quad \text{where} \\ T_i \in k_j^{VL} \Leftrightarrow T_i \cap k_j^{VL} = T_i, \quad T_i \in k_j^{L} \Leftrightarrow T_i \cap k_j^{L} = T_i, \\ T_i \in k_j^{A} \Leftrightarrow T_i \cap k_j^{A} = T_i, \quad T_i \in k_j^{S} \Leftrightarrow T_i \cap k_j^{S} = T_i, \\ T_i \in k_j^{H} \Leftrightarrow T_i \cap k_j^{H} = T_i. \end{gathered} \tag{8}$$

The proposed method of the PR verification for completeness is square relative to their number.

The knowledge base is consistent (coherent) if it does not contain inconsistent rules: rules with the same linguistic conditions but different inferences, for example, rules R_1 and R_2:

$$R_1\text{: IF } \{x_1 = L\} \text{ AND } \{x_2 = L\}, \text{ THEN } y = VL,$$
$$R_2\text{: IF } \{x_1 = L\} \text{ AND } \{x_2 = L\}, \text{ THEN } y = L.$$

These rules are inconsistency, since one and the same condition (LL) activates different output terms (VL and L).

Thus, the pairwise intersection of all the sets within the set of $k^{\langle VL, L, A, S, H\rangle}$ should yield the empty set

$$\begin{gathered} k_j^{VL} \cap \{k_j^{L}, k_j^{A}, k_j^{S}, k_j^{H}\} = \varnothing, \quad k_j^{L} \cap \{k_j^{A}, k_j^{S}, k_j^{H}\} = \varnothing, \\ k_j^{A} \cap \{k_j^{S}, k_j^{H}\} = \varnothing, \quad k_j^{S} \cap \{k_j^{H}\} = \varnothing. \end{gathered} \tag{9}$$

The proposed method of checking the consistency of the PR is linear relative to the number of cubic forms of the PR representation.

The knowledge base is coherent if for any pair of adjacent rules the values of the output LV also are adjacent, i.e, the conditions of these rules differ only by one of the subconditions, and, in these subconditions, the same LV is used with different values.

For example, the rules

$$R_1\text{: IF } \{x_1 = L\} \text{ AND } \{x_2 = L\}, \text{ THEN } y = VL,$$
$$R_2\text{: IF } \{x_1 = L\} \text{ AND } \{x_2 = A\}, \text{ THEN } y = L$$

are adjacent, as the terms L and A of the variable x_2 are adjacent.

The coherence of the knowledge base provides a smooth change in the output variable without sharp ups and downs, thus approaching the real conditions of operation of the object of the diagnosis. The proposed method of PR testing for coherence is linear relative to the number of adjacent rules.

The minimalistic knowledge base is the base from which you cannot remove any PR without violating its completeness. The minimalist in this context will be considered in the framework of the proposed PR structure (local minimum). Thus, the verification for minimality will be reduced to the verification of the following three sets of conditions:

(1) From the viewpoint of minimization and creation of dead-end forms (combining vectors with the code distance $d = 1$). For example, LLL $\cup$ LLA $\cup$ LLH = LLX.

(2) From the viewpoint of liquidation of the redundancy: $A \in B$, if $A \cap B = A$, where A and B are fuzzy sets belonging to one output term. For example, A = LLL and B = LLX. Then, LLL $\cap$ LLX = LLL; i.e., the vector LLL is absorbed by the vector LLX.

(3) From the viewpoint of covering (all the simple conjunctions are cubes of the 0-rank and should be covered by the minimal number of dead-end cubes of the k-rank). Consider, for example, the coverage consisting of three cubes $k^A = \{LXL, XHH, LHM\}$. Then, cube LXL covers the vectors {LLL, LAL, LHL}, cube XHH covers the vectors {LHH, AHH, HHH}, and cube LHM covers the vectors {LHL, LHH}. Having analyzed this coverage, it is possible to conclude that the vectors LHL and LHH belong to cubes LXL and XHH, respectively; therefore, the cube LHM is redundant.

Thus, the verification for minimalist provides us not only with the possibility to reduce the PR number but also to check the existing rules for redundancy and, therefore, to obtain a minimized knowledge base. The proposed method of the PR verification for redundancy is linear relative to the number of cubic forms of the PR presentation.

4. AN EXAMPLE OF THE VERIFICATION OF PRODUCTION RULE BASE FOR CORRECTNESS

As the object of diagnosis, we consider software that represents a web service of storage and exchange of documents (Copia.org.ua). In the course of this experiment, we limit ourselves to the analysis of the quality of the software from the user's point of view, i.e., to the quality in use. The standard (ISO 9126-4) offers a set of four criteria (attributes):

(1) The effectiveness, i.e., the software ability to allow the users to achieve their purposes with accuracy and completeness in the given context of use.

(2) The productivity, i.e., the software ability to provide the users with the ability to consume the amount of resources required to achieve the needed result in the given context of application.

(3) The safety, i.e., the software ability to achieve acceptable levels of risk concerning harm to people, business, and other software in the relevant context of its use.

(4) The satisfaction of the expectations, i.e., the software ability to satisfy the needs of the user in the given context of application.

These attributes are qualitative characteristics estimated by expert appraisals, which are as follows: as a result of the consideration of the presented document and the program itself, the expert composes his own opinion on how much the software meets the required criteria of quality. In view of the fact that the considered criteria are of a subjective character, their values are better presented linguistically instead of numerically. Then, every feature of quality can be presented as a linguistic variable, which values are names of fuzzy terms.

In this way, we obtain four input LV (diagnostic features (DF)): effectiveness (DF_1), productivity (DF_2), safety (DF_3), and satisfaction of expectations (DF_4) and one output variable software quality (diagnosis result (DR)). The parameters and the form of the membership functions of these variables can be optional according to the expert's choice, because it does not affect the algorithm of the PR synthesis and the analysis for correctness. According to the approach proposed in [1], it is possible to write 81 PR (because of the high number, they are omitted here). In view of the fact that the weights of the all input LV (diagnostic features) are equal, the PR can be written in a compact form where, for example, the vectors ALLL, LALL, LLAL, and LLLA take the following form: {LLLA (4)}, where in brackets the number of vectors (PR) is given. After these transformations, for all the PR, we obtain a set of compact forms of the PR presentation (Table 5).

Considering the universe $X = L \cup A \cup H$ and many-valued alphabet A3, the production rules can be presented in a minimized form. In doing this, it is necessary to carry out the integration of the vectors belonging to one output set that differ from each other only in one digit. For example, the vectors LLLA and LLLH belong to the output term L and differ from each other only in one digit. Therefore, they can be united: LLLA $\cup$ LLLH = LLLP. Consider a situation where three vectors AAAL, AAAA, and AAAH belong to the same output vector "A". Then, the result of the union will be AAAL $\cup$ AAAA $\cup$ AAAH = AAAX.

Having performed such transformations for each output set, we obtain a compact minimized record of the production rules (Table 6).

Comparing Tables 5 and 6, we can conclude that the total number of conjunctions has insignificantly decreased (76 conjunctions instead of 81), but the compactness and visualization of the PR presentation have considerably improved (9 instead of 15 different kinds of conjunctions).

We carry out the analysis for the correctness of the suggested set of production rules:

$$k^{\langle \mathrm{VL,L,A,S,H} \rangle} = \{k_1^{\mathrm{VL}}, k_1^{\mathrm{L}}, k_2^{\mathrm{L}}, k_1^{\mathrm{A}}, k_2^{\mathrm{A}}, k_3^{\mathrm{A}}, k_1^{\mathrm{S}}, k_2^{\mathrm{S}}, k_1^{\mathrm{H}}\},$$

where $k_1^{\mathrm{VL}} = \mathrm{LLLL}$, $k_1^{\mathrm{L}} = \mathrm{LLLP}$, $k_2^{\mathrm{L}} = \mathrm{LLAA}$, $k_1^{\mathrm{A}} = \mathrm{AAAX}$, $k_2^{\mathrm{A}} = \mathrm{LLHP}$, $k_3^{\mathrm{A}} = \mathrm{LAHP}$, $k_1^{\mathrm{S}} = \mathrm{HHHK}$, $k_2^{\mathrm{S}} = \mathrm{HHAA}$, and $k_1^{\mathrm{H}} = \mathrm{HHHH}$ are compact minimized forms of vectors.

Table 5. PR compact form

Output term	Input vector	Number of conjunctions
VL	LLLL	1
L	LLLA	4
	LLLH	4
	LLAA	6
A	LLAH	12
	AAAL	4
	LLHH	6
	AALH	12
	AAAA	1
	HHLA	12
	AAAH	4
S	HHHL	4
	HHAA	6
	HHHA	4
H	HHHH	1
	Total number of rules:	**81**

Table 6. PR compact minimized presentation

Input vector	Output term	Number of conjunctions
LLLL	VL	1
LLLP	L	8
LLAA		6
AAAX	A	4
LLHP		18
LAHP		24
HHHK	S	8
HHAA		6
HHHH	H	1
	Number of rules:	**76**

The knowledge base is complete if the number of sets $k^{\langle \mathrm{VL,L,A,S,H} \rangle}$ covers all $m^n = 3^4 = 81$ input vectors T_i, where $i = \overline{1,81}$, $m = 3$ is the number of terms of the input variable, and $n = 4$ is the number of input variables. We consider the vector T_1 = LLLA, which should be covered at the minimum by one cube; i.e., the result of the intersection with which will yield a nonempty set. The compact minimized form of the vectors presentation has made it possible to decrease the number of required checks; thus, instead of checking the vector T_1 = LLLA with four vectors {ALLL, LALL, LLAL, LLLA}, it is necessary to check the result of

Table 7. Check of production rules for consistency

$\mathrm{LLLL} \cap \{\mathrm{LLLP}, \mathrm{LLAA}, \mathrm{AAAX}, \mathrm{LLHP}, \mathrm{LAHP}, \mathrm{HHHK}, \mathrm{HHAA}, \mathrm{HHHH}\} = \varnothing$ $\Rightarrow k_1^{\mathrm{VL}} \cap \{k_1^{\mathrm{L}}, k_2^{\mathrm{L}}, k_1^{\mathrm{A}}, k_2^{\mathrm{A}}, k_3^{\mathrm{A}}, k_1^{\mathrm{S}}, k_2^{\mathrm{S}}, k_1^{\mathrm{H}}\} = \varnothing;$
$\mathrm{LLLP} \cap \{\mathrm{AAAX}, \mathrm{LLHP}, \mathrm{LAHP}, \mathrm{HHHK}, \mathrm{HHAA}, \mathrm{HHHH}\} = \varnothing \Rightarrow k_1^{\mathrm{L}} \cap \{k_1^{\mathrm{A}}, k_2^{\mathrm{A}}, k_3^{\mathrm{A}}, k_1^{\mathrm{S}}, k_2^{\mathrm{S}}, k_1^{\mathrm{H}}\} = \varnothing;$
$\mathrm{LLAA} \cap \{\mathrm{AAAX}, \mathrm{LLHP}, \mathrm{LAHP}, \mathrm{HHHK}, \mathrm{HHAA}, \mathrm{HHHH}\} = \varnothing \Rightarrow k_2^{\mathrm{L}} \cap \{k_1^{\mathrm{A}}, k_2^{\mathrm{A}}, k_3^{\mathrm{A}}, k_1^{\mathrm{S}}, k_2^{\mathrm{S}}, k_1^{\mathrm{H}}\} = \varnothing;$
$\mathrm{AAAX} \cap \{\mathrm{HHHK}, \mathrm{HHAA}, \mathrm{HHHH}\} = \varnothing \Rightarrow k_1^{\mathrm{A}} \cap \{k_1^{\mathrm{S}}, k_2^{\mathrm{S}}, k_1^{\mathrm{H}}\} = \varnothing;$
$\mathrm{LLHP} \cap \{\mathrm{HHHK}, \mathrm{HHAA}, \mathrm{HHHH}\} = \varnothing \Rightarrow k_2^{\mathrm{A}} \cap \{k_1^{\mathrm{S}}, k_2^{\mathrm{S}}, k_1^{\mathrm{H}}\} = \varnothing;$
$\mathrm{LAHP} \cap \{\mathrm{HHHK}, \mathrm{HHAA}, \mathrm{HHHH}\} = \varnothing \Rightarrow k_3^{\mathrm{A}} \cap \{k_1^{\mathrm{S}}, k_2^{\mathrm{S}}, k_1^{\mathrm{H}}\} = \varnothing;$
$\mathrm{HHHK} \cap \{\mathrm{HHHH}\} = \varnothing \Rightarrow k_1^{\mathrm{S}} \cap \{k_1^{\mathrm{H}}\} = \varnothing;$
$\mathrm{HHAA} \cap \{\mathrm{HHHH}\} = \varnothing \Rightarrow k_2^{\mathrm{S}} \cap \{k_1^{\mathrm{H}}\} = \varnothing.$

the intersection only with one compact vector LLLP belonging to the term "L" . The results of the checks for the vector $T_1 = \mathrm{LLLA}$ are as follows:

$$T_1 \in k_1^{\mathrm{VL}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{LLLL} = \varnothing; \qquad T_1 \in k_1^{\mathrm{L}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{LLLP} = \mathrm{LLLA};$$
$$T_1 \in k_2^{\mathrm{L}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{LLAA} = \varnothing; \qquad T_1 \in k_1^{\mathrm{A}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{AAAX} = \varnothing;$$
$$T_1 \in k_2^{\mathrm{A}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{LLHP} = \varnothing; \qquad T_1 \in k_3^{\mathrm{A}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{LAHP} = \varnothing;$$
$$T_1 \in k_1^{\mathrm{S}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{HHHK} = \varnothing; \qquad T_1 \in k_2^{\mathrm{S}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{HHAA} = \varnothing;$$
$$T_1 \in k_1^{\mathrm{H}} \Leftrightarrow \mathrm{LLLA} \cap \mathrm{HHHH} = \varnothing.$$

The result of the analysis has shown that the vector $T_1 = \mathrm{LLLA} \in k_1^{\mathrm{L}} = \mathrm{LLLP}$. In the same way, we check the remaining 80 vectors out of the knowledge base. From this analysis, we conclude that this knowledge base is complete.

The result of the check for consistency is given in Table 7.

Therefore, this knowledge base is consistent.

The compact form of the vectors presentation has allowed us to decrease the number of required checks because checked are not all the input vectors but only their compact forms.

A knowledge base is connected if the PR set $k^{\langle \mathrm{VL, L, A, S, H} \rangle}$ has adjacent rules. The compact minimized form of the vectors presentation made it possible to decrease the number of required checks because checked are not all the input vectors but their compact forms. Thus, for the considered PR set $k^{\langle \mathrm{VL, L, A, S, H} \rangle}$, the adjacent rules are the following:

$$\mathrm{LLLL} \in k_1^{\mathrm{VL}} \text{ and } \mathrm{LLLA} \in k_1^{\mathrm{L}}, \quad \mathrm{LAAL} \in k_2^{\mathrm{L}} \text{ and } \mathrm{LAAA} \in k_1^{\mathrm{A}}, \quad \mathrm{LHHA} \in k_3^{\mathrm{A}} \text{ and } \mathrm{LHHH} \in k_1^{\mathrm{S}},$$
$$\mathrm{HHAL} \in k_3^{\mathrm{A}} \text{ and } \mathrm{HHAA} \in k_2^{\mathrm{S}}, \quad \mathrm{HHHA} \in k_1^{\mathrm{S}} \text{ and } \mathrm{HHHH} \in k_1^{\mathrm{H}}.$$

Applying to the pairs of vectors the operation of the algebraic sum, it is possible to find the significance of LV. We consider two adjacent rules $\mathrm{LLLL} \in k_1^{\mathrm{VL}}$ and $\mathrm{LLLA} \in k_1^{\mathrm{L}}$, then $\mathrm{LLLL} \oplus \mathrm{LLLA} = \mathrm{UUUK}$; i.e., one digit has a value different from U; therefore, this LV is significant. Thus, having determined all the adjacent rules of the considered base of knowledge, it is possible to reach a conclusion about its coherence.

The base of knowledge is minimal if from the PR set $k^{\langle \mathrm{VL, L, A, S, H} \rangle}$ it is impossible to remove any conjunction (vector) and from the conjunction any letter (value of a term) without violation of the knowledge

Table 8. Check of production rule for redundancy

Output term	Set	Check
L	k_1^{L} and k_2^{L}	$k_1^{\mathrm{L}} \cap k_2^{\mathrm{L}} = \mathrm{LLLP} \cap \mathrm{LLAA} = \varnothing$
A	k_1^{A}, k_2^{A} and k_3^{A}	$k_1^{\mathrm{A}} \cap k_2^{\mathrm{A}} = \mathrm{AAAX} \cap \mathrm{LLHP} = \varnothing$ $k_1^{\mathrm{A}} \cap k_3^{\mathrm{A}} = \mathrm{AAAX} \cap \mathrm{LAHP} = \varnothing$ $k_2^{\mathrm{A}} \cap k_3^{\mathrm{A}} = \mathrm{LLHP} \cap \mathrm{LAHP} = \varnothing$
S	k_1^{S} and k_2^{S}	$k_1^{\mathrm{S}} \cap k_2^{\mathrm{S}} = \mathrm{HHHK} \cap \mathrm{HHAA} = \varnothing$

base's completeness. Also, each vector should have only a minimal number of significant inputs LV. In checking for minimality, the following three groups of conditions are analyzed:

(1) In terms of creating dead-end forms, cube $k_1^{\mathrm{A}} = \mathrm{AAAX}$ was obtained through combining three vectors: AAAL, AAAA, and AAAH.

(2) In terms of liquidating the redundancy (the result of checking for redundancy is shown in Table 8).

Therefore, the considered knowledge base has no redundant PR at all.

(3) In terms of coverage, it is enough to analyze Tables 5 and 6 in order to make conclusions that all the simple conjunctions are covered by the minimal number of cubes.

Thus, this analysis has shown that this base of production rules is correct and can serve as the basis for the fuzzy inference system. For further PR use in the Matlab set, it is necessary to pass from the cubic form of the PR record to the expanded form. An example of the transition for the cube $k_1^{\mathrm{L}} = \mathrm{LLLP}$ is as follows:

1. If (DF_1 is L) and (DF_2 is L) and (DF_3 is L) and (DF_4 is A) then (DR is L).
2. If (DF_1 is L) and (DF_2 is L) and (DF_3 is A) and (DF_4 is L) then (DR is L).
3. If (DF_1 is L) and (DF_2 is A) and (DF_3 is L) and (DF_4 is L) then (DR is L).
4. If (DF_1 is A) and (DF_2 is L) and (DF_3 is L) and (DF_4 is L) then (DR is L).
5. If (DF_1 is L) and (DF_2 is L) and (DF_3 is L) and (DF_4 is H) then (DR is L).
6. If (DF_1 is L) and (DF_2 is L) and (DF_3 is H) and (DF_4 is L) then (DR is L).
7. If (DF_1 is L) and (DF_2 is H) and (DF_3 is L) and (DF_4 is L) then (DR is L).
8. If (DF_1 is H) and (DF_2 is L) and (DF_3 is L) and (DF_4 is L) then (DR is L).

5. EXAMPLE OF FUZZY INFERENCE IN THE DECISION MAKING SUPPORT SYSTEM

We consider the results of the operation of the decision making support system about the PR quality based on the fuzzy inference. The system is based on the 76 PR earlier checked for correctness. The parameters of the linguistic variables are given in Table 9.

The process of the PR's quality appraisal in the process of its operation can be viewed as a diagnostic experiment based on the values of the diagnostic features. Taking into account the fact that we are interested in the quality of the program at the stage of its operation, we will speak about dynamic testing when testing is performed on functional patterns in the scale of real time. In the course of the diagnostic experiment, each diagnostic feature was appraised by an expert using a 100-score system in the following way: the efficiency = 85, the productivity = 87, the safety = 21, and the satisfaction of the expectations = 45. As the algorithm of fuzzy inference, the Mamdani algorithm was taken, where, at the stage of accumulation, the method of the boundary sum and the method of the maximum were taken, and, as the method of defuzzication, the method of the "center of gravity" was used. For analysis of the diagnosis result the mathematical system Matlab 7.5 was used, namely, a special package of fuzzy inference: the Fuzzy Logic Toolbox (license no. 532868 for the Mathwork Academic product). Table 10 provides the results of the operation of the fuzzy inference system.

In using the operation of the maximum as the method of accumulation, the system of fuzzy inference produced the following results: the "software quality" = 57.8, which to a high degree ($\mu_{\mathrm{A}}(57.8) \approx 0.60$) belongs to the average level (A) of the software quality. In the use of the operation of the boundary sum,

Table 9. Parameters of terms and membership functions of all LV

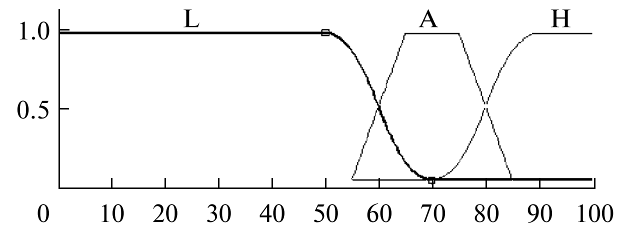
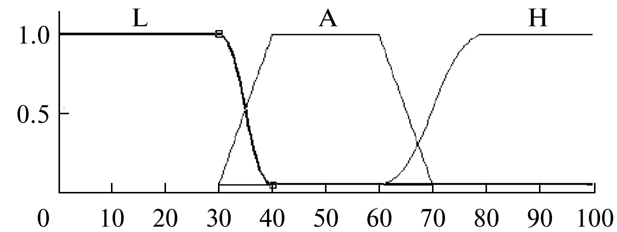
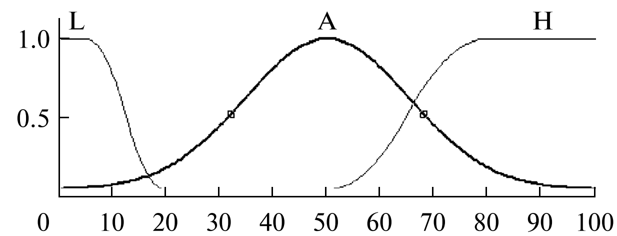
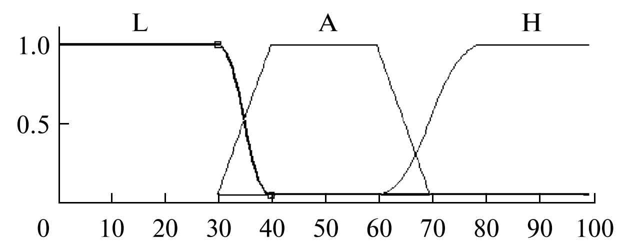
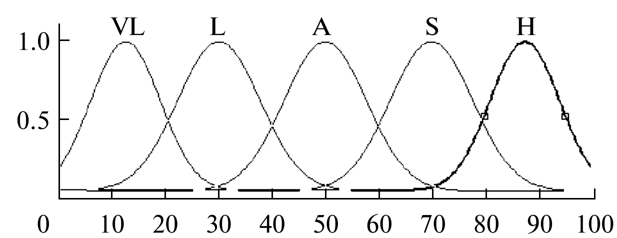
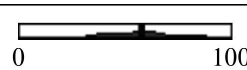
LV name	Terms	Ranges		Graph of membership function
Effectiveness	L A H	0 55 70	70 85 100	
Productivity	L A H	0 30 60	40 70 100	
Safety	L A H	0 20 50	30 80 100	
Satisfaction of the expectations	L A H	0 30 60	40 70 100	
Software quality	VL L A S H	0 15 35 55 75	25 45 65 85 100	

Table 10. Results of the fuzzy inference system

Method of accumulation	Kind of output fuzzy set	Result of defuzzification
Max		57.8
Sum		58.2

the "software quality" = 58.2, which to a high degree ($\mu_A(58.2) \approx 0.57$) also belongs to the average level (A) of the software quality.

6. CONCLUSIONS

In determination of the technical conditions of a computer system, in particular, of the integral index of the quality of the software, expert knowledge is vital. For automation of the decision making process concerning the quality appraisal of the software, it is recommended to use decision making support system that includes a block of fuzzy inference. The base of knowledge here is the production rules. Knowledge in the form of rules is easily formulated and perceived by experts, but the process of their creation and analysis is extremely difficult. We propose here an approach to the analysis of the knowledge base of the production rule for correctness allowing verification of rules obtained in any way be it an informal method of their making or a formal algorithm of synthesis. To simplify the process of the analysis of the production rules, a procedure of transformation of the expended form of their presentation into a cubic form was proposed, for which a many value alphabet of cubic calculation was introduced. The proposed theoretic multiple operations in this alphabet have enabled us to formalize the procedures of analysis of the production rules for correctness, by which it is possible to avoid errors in creating the knowledge base and, thus, to obtain an adequate result in the operation of the fuzzy inference system.

REFERENCES

1. Sivanandam, S.N., *Introduction to Fuzzy Logic Using Matlab,* Sivanandam, S.N., Sumathi, S., and Deepa, S.N., Eds., Berlin: Springer-Verlag, 2006.
2. Krivulya, G.F., Shkil', A.S., Kucherenko, D.E., and Garkusha, E.V., Fuzzy logic in expert estimation of IKT-Competences, *Vestn. Kherson. Nat. Tekh. Univ.,* 2011, no. 2 (41), pp. 13–22.
3. Krivulya, G.F., Shkil', A.S., Kucherenko, D.E., Competence as a support factor of the computer system operation, *Proc. IEEE Symp. EWD'2011,* Sevastopol, 2011, pp. 303–310.
4. Kargin, A.A., *Vvedenie v intellektual'nye mashiny. Kniga 1. Intellektual'nye regulyatory* (Introduction into Intellectual Machines. Book 1. Intellectual Regulators), Donetsk: Nord-Press, 2010.
5. Baralis, E. and Widom, J., An algebraic approach to rule analysis in expert database systems, *Proc. 20th Int. Conf. on Very Large Data Bases (VLDB 1994),* Santiago de Chile, 1994, pp. 475–486.
6. Dvorak, A., Stepnicka, M., and Vavrickova, I., Redundancies in systems of fuzzy/linguistic IF-THEN rules, *Proc. Eur. Soc. for Fuzzy Logic and Technology EUSFLAT-2011,* Aix-les-Bains, France, 2011, pp. 1022–1029.
7. Fogelqvist, P., *Verification of Completeness and Consistency in Knowledge-Based Systems: A Design Theory,* Uppsala, Uppsala Univ., 2011.
8. *Design of Digital Systems and Devices: Lecture Notes in Electrical Engineering*, Adamski, M., Barkalov, A., and Wegrzyn, M., Eds., Berlin: Springer-Verlag, 2011.

Translated by D. Shtirmer