Computer Science Journal of Moldova, vol.4, no.2(11), 1996

Knowledge base technology: a developer view

G.Ginkul S.Solowiev

Abstract

In present paper we have endeavoured to tell about some reasonings, conclusions and pricticals results, to which we have come being busy with one of most interesting problems of modern science. This paper is a brief report of the group of scientists from the Laboratory of Artificial Intelligence Systems about their experience of work in the field of knowledge engineering.

The researches in this area was started in our Laboratory more than 10 years ago, i.e. about in the moment, when there was just another rise in Artificial Intelligence, caused by mass emerging of expert systems. The tasks of knowledge engineering were being varied, and focal point of our researches was being varied too. Certainly, we have not solved all the problems, originating in this area. Our knowledge still has an approximate nature, but nevertheless, the outcomes obtained by us seem rather important and interesting. So, we want to tell about our experience in building of knowledge-based systems, and expert systems, in particular.

1 Introduction

Let us look through the problems of expert systems building as a common direction for the given paper. Just a mass development of the expert systems has allowed the knowledge engineering finally to be considered as independent discipline[1].

The expert systems are intended to solve the problems in those areas of human practice where professional experience and domain knowledge play a principal role. In expert systems the decision making is being carried out by means of imitation of reasonings of highly skilled professionals — human experts. The formalized human knowledge is a kernel of expert system, which is called its knowledge base (Fig.1). Other components of the expert system are intended to transformate the knowledge and are stipulated not so much by knowledge, but by properties of used formal structures.

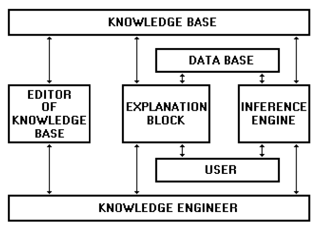


Fig.1. Structure of Expert system

The technology of building of knowledge-based systems, and in particular of the expert systems building, is named by knowledge engineering. Knowledge engineering has passed a number of stages of development. At first, the efforts of developers were concentrated on solution of the problem of knowledge representation. In outcome, there are four basic methods (models) of knowledge representation: production systems (rule-based systems), semantic networks, frames and predicate logic of the first order. Each of the models permits certain tolerance at implementation of its components. The last circumstance makes possible the design of combined models of knowledge representation. In the Laboratory of Artificial Intelligence Systems the method of simple alternatives was developed. It allows simultaneously to represent productions and formulas of predicate logic. The method of simple alternatives is considered in Section 2. The method can be used in tasks of recognition and classification and is oriented to attribute models of problem domains.

The method of simple alternatives has passed the practical test during development of the expert system shell FIACR, which is described in Section 3. FIACR is a typical representative of expert system shells. The knowledge base of FIACR is oriented to the use of systems of simple alternatives for knowledge representation. During implementation of FIACR some details of the method of simple alternatives were investigated in addition.

Other method of knowledge representation, developed in the Laboratory of Artificial Intelligence Systems, bases on use of finite predicates, including fuzzy predicates. The method is designed for diagnostic tasks and uses the formalism of partitioned boolean matrixes. It is considered in Section 4.

Evolution from narrows specialized programs up to general purpose tools is in general typical for software. Emerging of a great of expert system shells was the next stage of development of expert systems. It seemed, that the expert system shell can be utilized in any problem domain — the human expert only has to "fill" the knowledge base. However, this method turned out unproductive. There is the whole series of reasons, which hamper the extracting of expert's knowledge. As a result, the problem of knowledge acquisition has attacted attention of scientists and today it remains the most poorly by developed problem of knowledge engineering. Some obstacles braking knowledge acquisition are considered in Section 5.

It is impossible to create an universal technique of knowledge acquisition. However it is possible to facilitate and to speed up this process using special programming tools, automatizated systems of knowledge acquizition. Sections 6 and 7 are devoted to automatizated methods of knowledge acquisition. Two widespread approaches of automatizated dialogue with human expert are considered: contrasting of decisions and consulting by correspondence.

The methods of contrasting of decisions are intended for construction of taxonomic hierarchy of problem domain. Contrasting of decisions is used as methodical procedure stimulating the human expert to tell his (her) knowledge. The well-known representative is the repertory grid method. The methods of contrasting of decisions are discussed in Section 6.

At last, the special class of automatizated methods of knowledge eliciting, namely expert games, is considered in Section 7. The main idea of expert games is to force the human expert to make suppositions being based on incomplete information. The general game principles are being used to stimulate the human expert. The role of knowledge engineer is reduced to explanation of expert's game operations from the point of view of selected method of knowledge representation.

2 The systems of simple alternatives

There are four basic methods of knowledge representation: production systems (rule-based systems), semantic networks, frames and predicate logic of the first order. Besides, a practice has provided examples of combined methods of knowledge representation, in particular:

1. The data base of production system can have a structure of a semantic network.
2. The inference procedures in semantic networks and/or frames can be formed by means of productions or by means of predicate logic.
3. The system of frames can be organized in the form of semantic network or in the form of production system.
4. Frames can be used as templates for analysis of the data base of production system.

The new potentialities for combining of methods of knowledge representation arise in the case of use of intermediate knowledge representation languages. As examples, we refer to CM-frames[1] or descriptive

[1] V.Lozovsky. Network models. Artificial Intelligence. Vol.2, Models and methods: Reference book. Ed.D.Pospelov, Moscow, Radio i Svyazi, 1990, pp.28–49

tables[2], which are intended for implementation of semantic networks. In the Laboratory of Artificial Intelligence Systems the method of simple alternatives was developed. It supposes simultaneous representation of productions and formulas of predicate logic[2]. The method is based on elementary basic structure, which allows effectively to carry out the inference.

Note 1. Every case in the problem domain for which the decision making could be required we shall name as problem case. In different problem domains for the expression "a characteristic feature describing a problem case" the different synonyms are used: "symptom", "property", etc. We shall consider, that problem domain is described by means of attributes having alternative values. The pair "attribute: value" we shall name "fact".

Note 2. In practice, an attribute can be numerical, for example, such attributes as Growth or Weight. In the systems, where principal role is assigned to symbolic processing, values of numerical attributes are replaced by aggregated ones as Large, Small, etc. Further, we shall consider only attributes with finite number of values.

Definition 1 *We shall name a model of problem domain as attribute model, if*

(1) any domain case is described as a sequence of facts;

(2) facts are formed of attributes with finite range of values.

Definition 2 *The system of simple alternatives is a triple of finite sets* (S, C, A)*, where* S *is a set of the basic facts,* C *is a set of auxiliary facts (and* $S \cap C = \emptyset$*),* A *is a set of some subsets from* $S \cup C$*. Elements of* A *we will name as alternatives.*

[2] G.Andrienko, N.Andrienko. Building knowledge bases in ESKIZ system. In "Application AI Systems", Kishinev: Stiintsa, 1991, pp.73–79

According to the definition, an alternative is a finite n-dimensional vector consisting of facts. Facts may be basic or auxiliary. The basic facts correspond the real features of problem domain. The auxiliary facts have not informative interpretation. They establish connections between alternatives. For example, alternative

$$< animal : bird\ animal : fish >$$

consists of two basic facts.

Set of alternatives, joined together, makes up a system of simple alternatives. For example, the system of two simple alternatives is the following:

$$\{ < animal : bird\ animal : fish\ animal : mammal > \\ < extremities : wings\ animal : fish > \}$$

Note 3. In the last example, it would be correct to write the system of simple alternatives as follows:

$$\begin{aligned} S &= \{ < animal : bird\ animal : mammal\ animal : fish > \\ &\quad < extremities : wings\ extremities : fins > \} \\ C &= \{\} \\ A &= \{ < animal : bird\ animal : fish\ animal : mammal > \\ &\quad < extremities : wings\ animal : fish > \} \end{aligned}$$

However, if we assume that every fact enters at least into one alternative, then it is possible to skip set S and/or set C. So, any system of simple alternatives can be described by enumeration of the only set A.

During consultation the every fact can be confirmed or be refuted. According to the definition, each alternative can contain the only one confirmed fact. In the last example, if the fact "$finitenesses : wings$" is confirmed, then the fact "$animal : fish$" and the fact "$animal : mammal$" should be refuted. If this condition is not satisfied, then the set of confirmed facts makes up a conflict set for the given system of simple alternatives. Thus, any system of simple alternatives permits the finite number of combinations of facts[3].

Let a system of simple alternatives $R = (S, C, A)$ is given and consider a set of characteristic functions $f : S \cup C \to \{0, 1\}$.

Definition 3 *Characteristic function F satisfies the system of simple alternatives R, if for any alternative $< h1, \ldots, hm >$ from A the equality $f(h1) + \ldots + f(hm) = 1$ is true.*

The set of characteristic functions satisfying the system of simple alternatives R, we shall name a class of recognition for R and designate it $K(R)$. The class of recognition consists of correct combinations of basic facts.

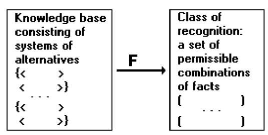


Fig.2. System of alternatives and its class of recognition

Therefore we can consider two tasks: "direct" and "inverse". The "direct task" is to detect a class of recognition of the given system of simple alternatives. In [2] is shown how the "direct task" can be resolved. However, from the point of view of knowledge engineering the "inverse task" is more interesting: to form a system of simple alternatives if its class of recognition is given. An algorithm of solution of the "inverse task" is discussed in [4]. Thus, the representative potentialities of systems of simple alternatives are enough rich. They make possible, at least theoretically, to generate any class of recognition.

In [5] the inference procedure in the systems of simple alternatives is analyzed. Let us assume, that the class of recognition of system of simple alternatives was formed as a result of enumeration of all permissible domain cases. Besides, a few facts (but not all) of some domain case are already confirmed. It is necessary to specify (as far as it possible) all unknown facts for the given domain case. It is a typical task of recognition.

It was proved, that inference in the systems of simple alternatives is reduced to solution of a system of boolean equations. The complete solution is being achieved only by complete exhaustive search of different non-conflict domain cases. In practice, we can use the simplified inference procedure. It is based on "deleting" of refuted facts from all simple alternatives in knowledge base. Various modifications of this procedure allow to perform the effective inference in systems of simple alternatives. The inference procedure should be considered as a constituent of the method of alternatives.

Now consider the task of knowledge base formation. From the point of view of systems of simple alternatives it consists in the following. Given a set of facts, which are sufficient for specifying of the problem domain, human expert sequentially selects some subset of facts (generally speaking, very limited) and joins the facts in an integral functional unit. Actually he (she) creates a mini-system of simple alternatives. We shall name such mini-systems of simple alternatives - as knowledge modules. As a rule, knowledge base is being formed of separate independent modules. From the point of view of systems of simple alternatives the independence of modules means that every module has its own auxiliary facts. During inference the modules can make up a sequence of "condition-conclusion" steps for reaching of given purpose. In [3] the knowledge modules are described, which have a practical interest.

First of all one should emphasize, that the simple alternative $< s1, s2, \ldots, sm >$ is a knowledge module, which links the alternative facts from the set $S = \{s1, \ldots, sm\}$. The alternative works in two ways:

- if some fact from the simple alternative is confirmed, then remaining facts from this module will be rejected;

- if all facts except one of alternative are refuted, then the sole non-refuted fact will be confirmed.

From the point of view of human expert, the module of alternative fixes a complete list of facts such, that: (a) the facts are pairwise incompatible and (b) each of decisions can permit only one fact from the alternative.

The module of incompatibility for facts $s1, s2, \ldots, sm$ can be written as the alternative $< s1, \ldots, sm, c >$, where c is an auxiliary fact. The module allows to form incomplete lists of incompatible facts. In contrast to alternative, the only first way of work is possible. In other words, module of incompatibility can only refute facts. We shall designate the module of incompatibility $< s1, \ldots, sm, c >$ as $[s1, \ldots, sm]$.

The module of prohibition[3] for facts $s1, s2, \ldots, sm$ is the following system of simple alternatives

$$
\begin{array}{l}
\{\ [s1, c1], \\
\ldots \\
[sm, cm], \\
< c1, \ldots, cm >\ \}.
\end{array}
$$

The class of recognition of the module of prohibition consist of all permissible subsets of its basic facts except the subset $(s1, s2, \ldots, sm)$ as itself. Module of prohibition refutes one fact from the initial list of facts, if all other facts have been confirmed.

The module of implication for facts $s1, s2, \ldots, sm$ is the following system of simple alternatives

$$
\begin{array}{l}
\{\ [s1, c1], \\
\ldots \\
[sm, cm], \\
< s0, c1, \ldots, cm >\ \}
\end{array}
$$

The module works in two ways. The first way corresponds the rule "IF $s1$ and ...and sm, Then $s0$". The second way corresponds the

[3] A.Zakrevsky. The eliciting of implicative rules in the boolean attribute space and pattern recognition. // Cybernetics, 1982, No.1, pp.1–6

reasoning "by contradiction", i.e. if both the fact-conclusion is refuted and all facts from condition except the only fact are confirmed, then the sole non-confirmed fact will be refuted. Usually, the module of implication is used to represent the "condition-conclusion" rules having a nature of the law. For example, "an increased content of chlorine in ground reduces in loss of a plant".

The module of production for facts $s1, s2, \ldots, sm$ is the following system of simple alternatives

$$\begin{aligned} &\{\ [s1, c1, c1'], \\ &\ldots \\ &[sm, cm, cm'], \\ &< s0, c1, c1', \ldots, cm, cm' > \} \end{aligned}$$

In contrast to implication, the only first way of work is possible for the module of production, so the reasoning "by contradiction" is impossible. Production is intended mainly for one-direction subjective instructions. For example, "IF there is a necrosis on leafs of plant, THEN conclude that the surplus of chlorine is in the ground".

Theoretically, the considered types of knowledge modules is excessive. However it is meaningful to eliminate this redundancy only at the level of computer implementation. On the contrary, it is necessary to expand the expressive abilities of knowledge representation during the dialogue with expert.

3 The expert system shell FIACR

The method of systems of simple alternatives was tested during development of the expert system shell FIACR[6]. From the point of view of functionalities, FIACR is the typical shell. It is intended for building of expert systems supporting decision making in the tasks of recognition. FIACR can be applied in the problem domains, where observable features, intermediate judjements and final decisions can be represented by means of enumeration of attributes accepting alternative values.

The work with FIACR is being carried out in the form of dialogue between the system and the user. The user is in conditions of specific domain case and needs in justified decision making. The attribute the value of which is interesting for the user, is being considered as target attribute. The decision making consists in conclusion about the value of target attribute. After obtaining the next observable fact (facts) from the user the expert system tries to make a decision. If it turns out, that the system "knows" not enough, then it continues to ask the user. The dialogue finishes as soon as one of values of the target attribute is confirmed.

In practice, the problem domain can include hundreds of attributes. In FIACR the set of attributes is divided into non-intersected classes, which are named subsystems. So, three different attributes with the same title "COLOUR" could be simultaneously represented in knowledge base, but they belong to different subsystems, for example, "LEAF", "STEAM" and "FRUIT". This simplifies the selection of titles for attributes.

For each fact the concept of "condition" is defined. The fact can be confirmed, refuted, uncertain or conflict. At the beginning of the dialogue all facts are uncertain. After obtaining the new information from the user or as a result of inference the fact can be refuted or be confirmed. The conflict condition indicates the presence in knowledge base of two possible independent conclusions, one of which confirms some fact and the other refutes the same fact.

FIACR supports four different classifications of attributes[7]: observable or non-observable, target or non-target, compulsory or non-compulsory, declarative or procedural.

1) Observable and non-observable attributes. Some attributes are used as questions in the dialogue. These attributes correspond to the observable parameters of the problem domain. The values of non-observable attributes become known only as a result of inference.

2) Target and non-target attributes. Before beginning of the dialogue the system should "know" about purposes of the consultation. Therefore some attributes are being declared as target attributes. As a rule, the target attributes are being chosen from

non-observable attributes.

3) Compulsory and non-compulsory attributes. The system should "know", which attributes always exist in problem domain, though the system don't know their values in the meantime.

4) Declarative and procedural attributes. FIACR provides use of declarative knowledge and procedural knowledge. Procedural attributes are identified with computer programs. If during the dialogue the system confirms a value of procedural attribute, then it temporarily interrupts the dialogue and the computer program is being executed. The value of attribute is being transmitted to the program as argument.

As well as any other shell, FIACR is based on a rigid structure of knowledge modules [8]. Potentialities of the system permit to use five types of knowledge modules: alternative, incompatibility, prohibition, production and implication. Simple alternative is the basic module. Eventually, all knowledge modules are being represented in the form of systems of simple alternatives. Moreover, every attribute is being represented as simple alternative. Besides, some alternatives carry out additional functions of data structures. Thus, for all logic structures in knowledge base a principle of double use of alternatives is satisfied.

To expand representative facilities of knowledge representation[9]:

- the use of negative facts is supposed;
- The module of implication and the module of production can have a lot of consequences. For this, the concept of "composite fact" was introduced.

Dialogue with the user in FIACR is based on the mechanism of replacing of the questionnaires [10]. Every questionnaire is a list of questions, and every question corresponds to an observable attribute. Rules for replacing the questionnaires (simple alternatives) are represented in the knowledge base. The questionnaire with the title "START" is being performed the first. It contains the most common questions. The mechanism of replacing the questionnaires allows to avoid extremal forms of the dialogue with user, namely:

- Use of unique list of questions;
- Choice of the sole (isolated) question on every step of dialogue.

To fill the knowledge base an input language is developed[11]. Loading the knowledge modules into the knowledge base is being carried out with the help of special compiler. At the same time, FIACR has the dialogue editor, which allows to build a knowledge base without use of the input language.

The special place in the system FIACR is assigned to knowledge base debugging. Methods of debugging are subdivided into testing methods and research methods. Testing is intended to check the knowledge base on particular (test) examples. The purpose of testing is to localizate erroneous knowledge modules. In [12] the test is defined from the point of view of systems of simple alternatives and criterions of completeness of set of tests [13] are considered.

The research methods are intended for discovering of poorly developed fragments of knowledge base. They are subdivided into static methods and dynamic methods. Methods, which use inference engine, belong to the dynamic methods; the others are static ones. Static methods are constructed in the form of independent utilities. Every utility tests a strictly fixed type of knowledge modules. Usually static methods investigate domain-independent characteristics of knowledge bases.

The dynamic methods are based on the concept of "conflict set of the facts" (see Section 2). The main idea is to detect all theoretically admissible conflict sets of facts and to present them for analysis by human expert. In [14] the methods of examination of conflict sets of facts are considered. In [15] the concept of "minimum conflict set of facts" was introduced and the algorithm of generation of all minimum conflict sets of facts is offered.

The expert system shell FIACR was elaborated up to the level of commercial prototype[16]. FIACR has been used to build the expert systems in such domains, as:

- prognosis of type of relations between fitohormones (in the Institute of Ecological Genetics of the Academy of Sciences of Moldova, Kishinev);
- Searching of defects and restoring of machine tools with numerical programm control (in the Institute of Problems of Control of the Academy of Scienses of the USSR, Moscow);
- Pre-start monitoring of technical condition of the object (in the Institute of Problems of Control of the Academy of Scienses of the USSR, Moscow).

Besides, FIACR was used in tens of technological and industrial centers of the former USSR.

4 Method of partitioned boolean matrices

Other developed method of knowledge representation bases on the use of finite predicates, including fuzzy predicates. The method is designed for diagnostic tasks and uses the formalism of partitioned boolean matrices, proposed by professor A.Zakrevsky [17].

Let be given a set of attributes $\{a_1, a_2, \ldots, a_n\}$, and $A_1, A_2, \ldots, A_n$ are finite ranges of values of these attributes. Any problem case can be represented as an element from the universum $M = A_1 \times A_2 \times \ldots \times A_n$. The set of correct problem cases (all permissible combinations of facts) P can be represented by means of enumeration of some elements of universum M, i.e. $P \subset M$.

Characteristic function for the set P can be represented in the form of n-place predicate $\varphi\ (a_1, a_2, \ldots, a_n)$, for which the set M is range of definition.

Now consider the task of diagnostics. The exposition of some problem case is being considered as partial (incomplete) information about some element from the set P, i.e. only the m $(m < n)$ of values of attributes are known: $a_j \in F_j \subset A_j$ $(j = \overline{1, m})$. It is necessary to find minimum subsets $B_j \subset A_j$ $(j = \overline{1, m})$ such, that for any $a_j \in B_j$ the equality $\varphi\ (a_1, a_2, \ldots, a_n) = 1$ is true.

In [18] the given task is reformulated as follows. To represent elements from the universal set M, the concept of partitioned boolean vector is introduced. If attribute $a1$ has values $A_{11}, A_{12}, \ldots, A_{1m1}$, attribute a_2 has values $A_{21}, A_{22}, \ldots, A_{2m2}, \ldots$ attribute has values A_{n1}, $A_{n2}, \ldots, A_{nmn}$, then the appropriate partitioned boolean vector has the length $m1 + m2 + \ldots + mn$. The vector consists of sections; sections are been ordered in correspondence with the attributes from the set $\{a_1, a_2, \ldots, a_n\}$. Every element of every section corresponds to the value of the attribute. The value of each element of the partitioned boolean vector can be 0 or 1, accordingly to whether is confirmed the corresponding value of the attribute or not.

To represent finite predicates the sets of partitioned boolean vectors are used. The partitioned boolean vectors are interpreted as elementary conjuctions of one-place predicates, and sets of vectors (i.e. matrix of conjuncts) are interpreted as disjunctive normal forms of final predicates. Finite predicate $\varphi\,(a_1, a_2, \ldots, a_n)$ can be defined as a conjuction of elementary disjunctions $d_1, d_2, \ldots, d_k$. This conjuctive normal form can be represented as a partitioned boolean matrix D, consisting of lines — disjunctions $d_1, d_2, \ldots, d_k$ and can be considered as the set of equations $d_1 = 1,\ d_2 = 1,\ \ldots,\ d_k = 1$. The set of radicals of this system is the characteristic set of predicate φ.

Obtained theoretical results [19] provide a list of functions to work with partitioned boolean matrices. Inference in the knowledge base represented by means of the matrix D is reduced to determination of simple disjunctions for the predicate represented by the matrix D.

In [20] the given method was generalized for fuzzy case, i.e. when the set P is fuzzy in sense of L. Zadeh[4]. The main idea of the proposed method is the operation of fuzzy resolution. The operation is applied to two initial disjunctions of the k-th section and, as a result, it forms the third disjunction — a resolvent. The k-th section of the resolvent is the conjuction of corresponding sections of initial disjunctions, and the other sections are disjunctions of corresponding sections of initial disjunctions. It is important, that the contents of the resolvent depends

[4]L.Zadeh. The concept of a linguistic variable and itd application to approximate reasoning. N.Y.: Elsevier P.C., 1973

on the form of premises. It was proved, that for deriving the most strong resolvent it is necessary to transform the initial disjunctions into the k-th reduced form. The criterions of complexity of disjunctions are formulated in [21]. Basing on the premises (i.e. without the formation of resolvent) the criterions allow to determine, whether or not the resolvent is a consequence of one of disjunctions.

5 Problem of knowledge acquisition

Knowledge acquisition means a process of knowledge elicitation from human expert or knowledge extraction from other sources and transmission the obtained knowledge to expert system. Usually the following ways of knowledge acquisition are being considered: analysis of special texts, restoring of regularities from empirical data, elicitation of personal knowledge of human experts. Further we shall consider only the elicitation of personal knowledge of human experts.

Originally it seemed that human expert can fill knowledge base himself (herself). Such method is named as "direct extraction". After unsuccessful attempts to implement this method in complete volume, it became clear, that personal knowledge and professional skills of human expert counteract the direct extraction procedure.

In the first place, the expert's knowledge have not a modular structure. Human expert understands steps of inference in the uniform "compilated" form. This is the result of numerously reiterations of the same steps of reasoning in practice.

Secondly, the most important knowledge of human expert have non-verbalized nature, because they were gained immediately from practice, without using of natural language. In other words, the most important knowledge of human expert exist in the form of images or associations. Therefore, it is very difficult to verbalize them and it is much more difficult to structure them.

Thirdly, to state knowledge correctly the human expert should deeply comprehend a course of his (her) own reasonings. However it is impossible to rely on self-observations of human expert. The skill to apply knowledge and the skill to tell about them are the two different things. From the point of view of the second skill the human expert is not a highly skilled professional. Certainly, human expert could explain the decisions made by him (her), however this explanation differs from the genuine course of reasonings. Besides, logical steps and regularities which the expert has told during the explanation are valid only within the case of the given example.

These natural obstacles seriously hamper the direct knowledge extraction. As a result, the other method is more acceptable, when the specialist of special genre — knowledge engineer — tries to form knowledge base together with the human expert. Knowledge engineer should elicit and to structure expert's knowledge, so that further they could effectively be used within framework of the selected knowledge representation formalism. Knowledge engineer should not only use his (her) engineering skills, but also demonstrate nonstandart, creative facilities.

The use of special systems, allowing (at least partially) to automate gathering, structuring and systematization of knowledge can essentially increase the effectiveness of knowledge acquisition. However the task of automatization of dialogue with human expert is nontrivial. Despite abundance of theoretical researches the shortage of constructive methods in this area is being obviously felt.

At initial stages of knowledge acquisition the simple interview still remains a widely used method. Simple interview looks as a friendly talk. In dialogue with expert the knowledge engineer tries to find out the general structure of the problem domain and to make out the essence of decision making task. Special attention the knowledge engineer should give to elicition of target decisions. Only then it is possible to give the definite answer: either the knowledge engineer has run into already investigated case of "good" decisions or he(she) should go an unknown way. General feature of the "good" decisions is simplicity of their titles and possibility to enumerate the complete list of decisions. For example, the following decisions are "good": list of defects, directory of products, taxonomy of microorganisms. Thus, simple interview allows to fix main concepts of problem domain.

Structured interview is used for more detailed analysis of these concepts. Structured interview looks as a set of questions and answers.

Knowledge engineer uses outcomes of the simple interview and makes ready a list of questions being subject to discussion. Structured interview allows to reveal the structure of concepts. Experienced knowledge engineer should know how to use the methods of zonding[22].

Simple interview and structured interview poorly lend themselves to automatization. In practice, the methods of indirect extraction are much more useful, i.e. when the expert solves different problem cases given by the knowledge engineer, and then the knowledge engineer reconstructs steps of expert's reasonings. There are two basic approaches to organization of indirect knowledge eliciting: contrasting of decisions and consulting by correspondence.

6 Knowledge acquisition based on contrasting of decisions

The methods of contrasting of decisions are intended for building of a taxonomic hierarchy of problem domain. It is supposed, that the initial set of decisions is already known. The main idea consists that knowledge engineer requires that the expert specifies the features of similarity and/or distinction of different decisions. As a result, methods of comparison of decisions allow to elicit the attributes and connections between the decisions and to form an attribute model of the problem domain. Scenario of the dialogue depends on the nature of a problem domain and on the selected method of knowledge representation. Therefore, the scenario can vary, but every time in the framework of basic idea of comparison. The most known method of comparison of decisions is the repertory grid method[5].

The main idea of the repertory grid method is borrowed from experimental psychological analyses[6]. The essence of the method consists that triples of various decisions are being sequentially demonstrated to the human expert and he (she) should execute a special operation for each triple. Namely, the expert should name the attribute and specify

[5] J.Boose Expertise Transfer for Expert Systems, Elsevier, 1986

[6] G.Kelly. The psychology of personal constructs, Norton, 1955

two its values, which divide the given triple into pair of similar decisions and one contrast decision. Thus, two decisions should be characterized by the first value of the attribute, while the one contrast decision should be characterized by the second value of the attribute.

Consider an example. Let the triple of decisions: HUMMING-BIRD, OSTRICH and PARROT has been demonstrated to expert. The correct answer can sound as follows: "PARROT and HUMMING-BIRD can fly well and this fact distinguishes them from OSTRICH which can not fly". The answer of the human expert can be presented by a fragment of the semantic network with binary connections:

$$\text{PARROT} \quad \xrightarrow{\text{Fly ability}} \quad Good \quad \xleftarrow{\text{Fly ability}} \quad \text{HUMMING-BIRD}$$

$$\text{OSTRICH} \quad \xrightarrow{\text{Fly ability}} \quad None$$

The repertory grid method can be easily incarnated in computer. Its orientation on semantic networks with binary connections allows to develop the knowledge bases for tasks of recognition and classification.

In closer examination[24], the repertory grid method looks ideally balanced. On the one hand, expert has ONLY three variants for partition, and exhaustive search of the variants is not a complicated problem. On the other hand, expert has the FULL three variants and, therefore, an enough number of ways to select the most natural dividing attribute. The requirement of similarity of two decisions, firstly, limits an area of searching of suitable dividing attribute, and, secondly, makes the expert's reasonings more constructive.

Another advantage of the repertory grid method is that the elicited attributes reflect qualitative properties of decisions. Only the expert can rightly specify, what features of decisions are identical, and what are different. For example, from the point of view of recognition of vegetables the attribute "COLOUR" (green, red, violet, etc.) is an important feature, therefore all tints of the colour are reduced to one value. But if the development of plants is being investigated, then just the tints of colouring of leafs (dark-green, bright-green, etc.) are important.

The "classical" repertory grid method allows to specify only two values of attribute. However, the method can be supplemented by the procedure of elicitation of complete spectrum of values of the attribute [23], it is enough to ask the expert about values of the attribute for those decisions, which were not presented in the initial triple. In our example, knowledge engineer can demonstrate remaining decisions — for instance CHICKEN and EAGLE, that could stimulate the expert to specify other values of the attribute "Fly ability": "bad" and "very good".

$$\text{PARROT} \xrightarrow{\text{Fly ability}} Good \xleftarrow{\text{Fly ability}} \text{HUMMING-BIRD}$$

$$\text{OSTRICH} \xrightarrow{\text{Fly ability}} None$$

$$\text{CHIKEN} \xrightarrow{\text{Fly ability}} Bad$$

$$\text{EAGLE} \xrightarrow{\text{Fly ability}} Very\ good$$

At the same time any modifications of "classical" repertory grid method should be carried out with the extreme caution. Changing of initial conditions of the task can result the human expert incorrectly understands what it is required from him (her). To define the initial task more correctly a Schematic Picture or visual image can be used. With the help of Schematic Picture it is possible to explain a complicated idea. The idea of using of Pictures in knowledge engineering is discussed in [26] and an example of special image language is described. In short, knowledge engineer and expert accept the conventions about use of Pictures. As a result, they have in instruction a powerful image language, which allows them to formulate precisely nontrivial questions and answers. Application of the image language is especially effective in combination with computer graphics.

The shortages of repertory grids are of a specific nature. Sometimes the formally composed triple of decisions irritates the expert because of non-comparability of presented decisions. To solve this problem it is possible to filter all possible triples of decisions beforehand [28].

Other problem of repertory grids is caused by a non-verbalized nature of personal knowledge. All methods of comparison require from human expert to specify (to write or to say aloud) a title of attribute and its values. However this task is not simple. To facilitate the search of correct titles and statements a method of improvement of titles [30] was developed. It allows to standartize all titles of facts and to dismember them into titles of attributes and titles of their values. The method consist of three steps. At first, it requires the human expert to construct a general question for the set of given facts. Further, the expert should give all possible different answers to the question. At last, the special heuristic procedure uses methods of grammatics restoring [25] and extracts titles of attributes and titles of their values.

Another shortage of repertory grids is its monotone: the human expert should repeat the same operation for many different triples of decisions. The idea of organization of the repertory grid method in the form of game was proposed in [29]. It consists in division of the repertory grid procedure into a comparison stage and an explanation stage. At the stage of comparison the program sequentially shows various triples of decisions to the human expert. The task of the expert consists only in dividing decisions of every triple into two parts, containing one contrast and two similar decisions. This stage is being finished when the number of triples being analyzed is enough (by formal reasons) to recognize any decision from the complete list of decisions.

Note, that at the first stage the expert should only divide triples of decisions. The act of comparison is extremely simple, therefore the stage has a dynamic nature and it allows to analyze a much more number of different triples. At the second stage, human expert should explain the partitions he (she) made. For each of the made partitions the expert should define (enter from keyboard) a proper attribute and two its values. First of all, the expert specifies title of the attribute. Then he (she) can construct a complete spectrum of values of the given attribute. Note, every new attribute can make redunant (from the formal point of view) the further analysis of non-explained triples. The game task is to enter a minimum quantity of attributes at the second stage of the game.

The proposed modification of repertory grids allows to increase effectiveness of knowledge elicitation. Human expert has a time to analyze much more triples than in the case of synchronous dividing and explanation.

An interesting idea of implementation of the contrasting of decisions is proposed in the role dialogue methods, which implicity mask the problem domain specifications by some well-known sphere of human activity. For example, to create the different cases of contrasting of decisions the scenario "Advertising agent" [31] uses primitive market nomenclature. Decisions are named by Goods, expert is Advertising agent. The role without words belongs to Competitor. The Competitor does not take participation in the dialogue, but it is necessary to take into account Competitor's influence. The task of Advertising agent is concrete: to leave the Competitor behind and to sell the Goods to Buyer. However the Buyer is rather fastidious: he seems inclined to purchase, but then begins to doubt and require new explanations. Taking into account modification of the nomenclature, the constructed predicates reflect the quality of goods. One step of the dialogue consists that Buyer describes to the Advertising agent the next doubtful situation and requires to enter new information, which refutes these doubts. The scenario "Advertising agent" was implemented in the system ES-KIZ. It is intended for knowledge elicitation in tasks of selection.

Usually, methods of comparison of decision are being utilized for elicitation of declarative knowledge. Tasks of control are a non-traditional area for these methods. In [32] the method of contrasting of decisions for tasks of control is proposed. Knowledge is being represented in the form of special semantic network, which is named as control scheme. The dialogue with human expert is intended for synthesis of the control scheme and consists of several stages. At first stages the human expert should explain the sequence of operations for reaching of given goal. Then the special procedure based on methods of formal grammatics restoration, generalizes the examined sequences of operations into so-called "transition diagram". Further dialogue with human expert allows to elicit the conditions of transitions in the diagram. Fragments of the diagram are being considered as outcomes of comparisons, which expert has to explain. The diagram with included conditions of transitions represents the required control scheme.

The attempt of generalization of repertory grids for fuzzy case is made in [33].

7 Expert games

The idea of consultation by correspondence as a method of knowledge acquisition consists in that knowledge engineer formally describes a domain case and requires the expert to make the decision. The different methods of consultation by correspondence support different forms of dialogue between knowledge engineer and human expert. Protocol analysis is the most elementary method: knowledge engineer shows to human expert a complete exposition of the domain case and then writes the protocol of expert's reasonings. The obtained protocol "of reasonings aloud" bears the text information about mechanisms of decision making. There are various variants of protocol analysis: scenario modelling, self-report, observation, etc. Knowledge elicitation tools make a hypothesis, that any protocol has a rigid structure. For example, authors of the system KRITON [7] assume that protocol consists of several segments appropriate to the stages of inference.

In some other methods of consulting by correspondence, for example in the Expert games[34], knowledge engineer shows to human expert only a part (fragment) of complete exposition of domain case and requires the expert explain steps of his (her) reasonings. The expert games is a special category of these methods[35]. They look as usual computer games with bonuses, sanctions and penalties. At the same time, expert games are "immersed" in the researched problem domain and use problem nomenclature.

The purpose of expert game is to stimulate the expert to make deci-

[7] J.Diederich, I.Ruhman, M.May. KRITON: a Knowledge-acquisition tool for expert systems Int.J. of Man-Machine Studies, Vol.26, 1987, No.1, pp.29–40

sions being based on incomplete information about domain case. During the game the information about some domain case is being implicity conveyed to the expert. The rules of expert game require the expert makes particular operations (conclusions). For this purpose the expert should use his (her) professional knowledge. The retrieval behaviour of the expert uncovers the "interior kitchen" of decision making. As a result, the task of knowledge engineer is reduced to explanation of expert's operations from the point of view of selected method of knowledge representation.

Switching of expert's attention to the pure game tasks allows to attract the expert and to mask the original purposes of the dialogue. Actually, knowledge engineer can keep from expert, that the expert takes participation in complicated process of knowledge elicitation. At the same time it is necessary to remember, that not every person is being tempted to passion, therefore sometimes expert games can be meaningless.

As a class software products the expert games are based on four main principles.

Principle 1. In expert games the dialogue with expert is organized in the form of computer game.

Principle 2. Test examples are used as input data for expert games. Every test example consists of formal exposition of a domain case and the title of appropriate decision. The base of examples is created by knowledge engineer.

Principle 3. In outcome of expert game the protocol of expert's game operations is being formed.

Principle 4. In expert games there is an interactive block of protocol analysis, which allows knowledge engineer to re-build formal stimuli of the expert's game operations. This block consists of the two subsystems: subsystem of preliminary protocol analysis and subsystem of knowledge elicitation. The functionalities of subsystems depend on the hypotheses about style of reasonings of expert during the expert game. General architecture of expert game is represented on Fig.3.

In contrast to other automatized methods of knowledge acquisition, the expert games are much more poorly oriented to the construction

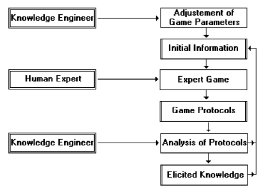


Fig.3. General architecture of expert game

of a final structure of knowledge base. Even in the ideal case, knowledge engineer should not expect to generate knowledge base after the expert's gaming at computer. The expert games deliver half-finished knowledge, which require finishings.

The game "Black Box" [37] was the first of expert games. "Black Box" assumes the attribute model of problem domain. Human expert and computer are competing parties in the game. Before beginning of the game, computer "knows" the set of potential decisions and, at least, one test example of one of decisions (i.e. set of facts describing domain case and title of appropriate decision — See Principle 2). The human expert should state the title of decision, that is known to computer. During the game, computer shows facts from the test example to the expert. It show not all facts at once, but one by one. Thus, the confirmed information about the domain case is being increased after every step of the game. Rules of the game require the expert make stakes on potential decisions. Through the stakes the human expert implicity formulates hypotheses about acceptability of making that or another decision.

Rules of the game permit to increase or to decrease the early made stakes after acquaintance with new facts. However, if expert has decreased a stake, then the removed part comes back to him (her) not wholly, but only partially, and the part being returned is being decreased in geometrical progression after each new round of the game. The points are used for payment and the some amount of money (parameter S) is being given to expert before start of the game. The winnings of expert depends on that, how well-grounded were the expert's stakes.

Formal parameters of the game are:

N – number of facts in the initial example,
S – initial amount of expert's money,
Q – geometrical progression factor,
L – minimum permissible size of the total sum stakes being made.

Using these parameters the knowledge engineer can control the rules of the game.

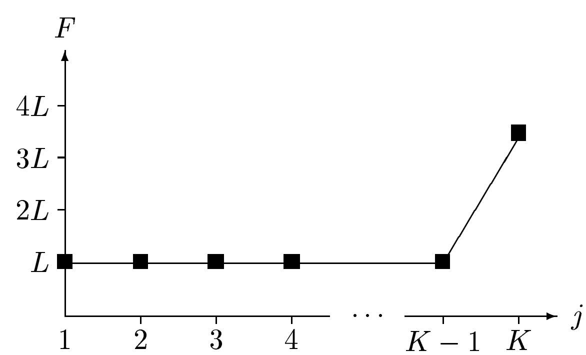


Fig.4 a) "Bad" behaviour of expert

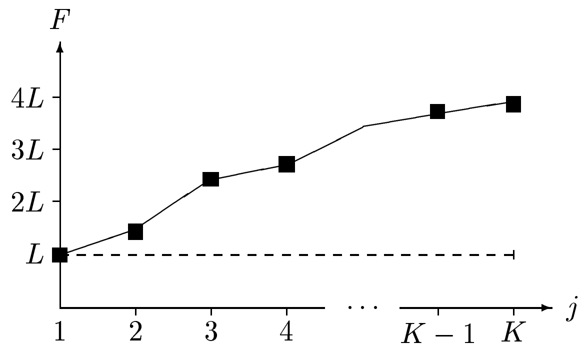


Fig.4 b) “Good” behaviour of expert

In [37] the various modifications of “Black Box” were discussed and influence of the game parameters to expert’s behaviour was investigated. It was proved, that there is the optimum strategy of expert’s behaviour, which allows to play “Black Box” with minimum losses. In common case, the optimum strategy is a function F, which depends on parameters of the game. The “most unfavorable” is such behaviour of human expert, when he (she) strictly follows the optimum strategy. Using of the optimum strategy “kills” initial intensions to elicit the knowledge. If expert adheres to the optimum strategy, he (she) only confirms (repeats) the initial test example (pic.4).

But in practice the expert seldom uses the optimum strategy. On the contrary, his (her) knowledge just counteracts to using of the optimum strategy.

The “Black Box” allows to elicit the prohibitions and productions (see Section 2). First of all, knowledge engineer should investigate the game protocol and to eliminate the evidently unpromising information. At the second stage, knowledge engineer builds the formal knowledge structures. Two various methods of protocol analysis are discussed in [39]. They result in the formation of productions.

In the first case, all facts, known by the expert on some step of the game, enter in the left part of module of production, and the decision, on which the expert has made stake, enters in the right part of the production. Besides, an unknown condition X is being added to the left part of production. Further, attempt is being made to determine the unknown condition X. The second method is that all stakes are being considered as feasible productions, then the coefficient of reliability is being put in the correspondence to every production and the most authentic productions are being selected.

Actually, in "Black box" the human expert analyzes observable facts and then performs direct inference procedure. On the contrary, in the expert game "Burime" [38] the human expert reasons in direction from decision up to observable facts, therefore he (she) imitates the inverse inference.

Initial information for "Burime" is the same as for "Black Box": computer "knows" all potential decisions and one test example. In the beginning of the game, computer tells to expert the title of decision from the test example and requires to guess all facts from the test example. Also, computer informs about the number of facts in the test example, for instance N facts. Trying to state the facts from the test example, human expert every time tells N facts to computer. Computer compares these facts and facts from the test example, and then informs about "exact hits". For example: "You have guessed right the 3 facts. Try once again" or "You have guessed right all N facts!". In the last case the game is be finished.

Protocol of the game represents a tree-like structure of expositions (Fig.5). The root of the tree is the exposition constructed at the first attempt. In a common case, the exposition $Zi - 1$ is a direct ancestor of exposition Zi $(1 < i), if\ Zi$ was present on the screen before construction Zi. The method of knowledge elicitation is based on the two hypothesises:

- Any exposition corresponds to the inference tree in a production system;
- As a rule, the new exposition Zj is generated from the previous exposition Zi by means of replacement of one of its subtrees.

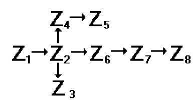


Fig.5. Tree-like structure of the game protocol

At the stage of preliminary analysis the tree of expositions is being divided into components of connectivity, which are being simplified by linearization procedure and then are being considered independently. In [39] the procedure of linearization is considered in detail. In common case, linearization is a selection of the main chain and then step by step inclusion of adjacent chains.

At the second stage the formal knowledge structures are being created. For this purpose, the recursive heuristic procedure of AND/OR Trees restoration was developed.

Except considered games, we have created a lot of others. They are based on various game principles and are intended for elicitation of special knowledge structures. For example, the expert games presented in [40,41] are oriented to elicitation of prohibitions and incompatibilities, and knowledge modules are being formed in correspondence with the inpur language of FIACR (see Sections 2–3).

8 Conclusion

The majority of our researches in knowledge engineering was supported by grants of the Academy of Science of the USSR. Obtained results are presented in about 100 scientific publications. They were reported at the scientific conferences in USA, France, Japan, Germany, Russia, Moldova, etc. Eight employees of the Laboratory have received degrees of doctors. The developed methods of knowledge engineering are implemented in the program systems FIACR, FIACR+, CAPRICE, EDIP, AFORIZM and ESKIZ. The systems are used in tens of scientific and industrial centers of Moldova and abroad.

References

[1] Yu.Pechersky, K.Naidenova. Expert systems: the state and perspectives. Preprint of the Institute of Mathematics of the Academy of Science of Moldova, Kishinev, Stiintsa, 1987, 38 p. (in Russian)

[2] Yu.Pechersky, S.Solowiev, G.Solowieva. The principles of expert recognition systems design. Intern. Journal of Pattern Recognition and AI, vol.3, No.1, 1989, pp.113–120.

[3] S.Solowiev, G.Solowieva. Issues of application of systems of alternatives for knowledge representation. Izvestiya of the Academy of Science of the USSR, Tehnicheskaya kibernetika, No.5, 1987, pp.80–82. (in Russian)

[4] S.Solowiev. Issues of the inference realization in systems of alternatives. In "Application AI Systems", Kishinev: Stiintsa, 1991, pp.11–18. (in Russian)

[5] S.Solowiev. The realization of inference engine in systems of alternatives. II-nd Soviet AI Conference, Minsk, 1990, Vol.1, pp.170–172. (in Russian)

[6] S.Solowiev, G.Solowieva. The comercial system FIACR. Handbook of Artificial Intelligence, Vol.1, Moscow, Radio i Svyazi, 1990, pp.362–369. (in Russian)

[7] S.Solowiev, G.Solowieva. The FIACR system. In "Problems of using of Artificial Intelligence". Varna, Bulgaria, 1987, pp.179–183.

[8] S.Solowiev, G.Solowieva. The architecture of the expert system FIACR. In "Theory and practice of programming". Kishinev: Stiintsa, 1989, pp.111–128. (in Russian)

[9] S.Solowiev, G.Solowieva. The issues of knowledge base organization in the system FIACR. In "Expert systems: the state and perspectives". Moscow, Nauka, 1989. pp.47–54. (in Russian)

[10] S.Solowiev. The explanation block in the system FIACR. I-st Soviet AI Conference, Pereslavl-Zalessky, 1988, pp.516–520. (in Russian)

[11] S.Solowiev, G.Solowieva. The input language of the system FIACR. In "The software for programming complexes", Matematicheskie issledovania, Vol.104, Kishinev, Stiintsa, 1988, pp.110–116. (in Russian)

[12] V.Sirbu. Knowledge base debugging in the system FIACR. In "Applied AI Systems", Kishinev: Stiintsa, 1991, pp.57–62. (in Russian)

[13] V.Sirbu. The debugging mehanism for knowledge bases using alternatives. In "The raising the software quality". Sevastopol, 1988. (in Russian)

[14] V.Sirbu. The authomatized debugging of knowledge base in the system FIACR. II-nd Soviet AI Conference, Minsk, 1990, Vol."Exposition", pp.130–132. (in Russian)

[15] V.Sirbu. The minimal conflict combinations of facts in the system FIACR. In "Applied knowledge-based systems", Kishinev: Stiintsa, 1992, pp.88–95. (in Russian)

[16] S.Solowiev. The expert system FIACR evolution. Proc. of INFO'89, Minsk, Vol.1, Part 2, pp.883–885. (in Russian)

[17] A.Zakrevsky, V.Levchenko, Yu.Pechersky. Instrumental diagnostic expert system EDIP based on finite predicates. In "Application AI Systems", Kishinev: Stiintsa, 1991, pp.24–40. (in Russian)

[18] V.Levchenko, A.Savinov. A dialog control and logical inference in finite predicates. In "Application AI Systems", Kishinev, Stiintsa, 1991, pp.40–46. (in Russian)

[19] V.Levchenko, A.Savinov. The matrix representation of predicates and its using for knowledge representation in expert systems. Izvestiya of the Academy of Science of the USSR, Tehnicheskaya kibernetika, No.5, 1993. (in Russian)

[20] A.Savinov. Fuzzy propozitional logic. Fuzzy Sets and Systems, vol.60, No.1, 1993.

[21] A.Savinov. Fuzzy propozitional logic for the knowledge representation. Proc. of EUFIT'93, Aachen, Germany, 1993, pp.1420–1426.

[22] G.Ginkul. Inerview methods for knowledge acquisition. In "Applied AI Systems". (Mathematical investigations, issue 123) Kishinev, Stiintsa, 1991. pp.66–72. (in Russian)

[23] Iu.Secrieru, The procedure for classifying concepts eliciting based on comparison of decisions. In "Advances in modelling & Analysis", Vol.34, N1, 1995, pp. 57–63.

[24] G.Andrienko, N.Andrienko, G.Ginkul, S.Solowiev. Automatizated systems for knowledge acquisition. Computer Science Journal of Moldova, Vol.1, No.3, 1993, pp.42–50.

[25] S.Solowiev. The task of grammatic inference. Matematicheskie issledovaniya, Kishinev, Stiintsa, 1985, pp.135–143. (in Russian)

[26] G.Andrienko, N.Andrienko. The game comparison procedures in knowledge engineering. Proc. of the III Soviet AI Conference, Vol.1., Tver, 1992. pp.93–96. (in Russian)

[27] N.Andrienko. An intelligent system to support solving of nonformal problems. In "Application AI Systems", Kishinev: Stiintsa, 1991, pp.46–57. (in Russian)

[28] G.Andrienko, N.Andrienko. Hypertext in the AFORIZM knowledge acquisition technology. In "Applied knowledge-based systems", Kishinev: Stiintsa, 1992, 3–14. (in Russian)

[29] Iu.Secrieru. Modificated repertory grid procedure for attribute problem domains. Computer Science Journal of Moldova. Vol.2, No.2, 1994.

[30] G.Ginkul, S.Solowiev. The methods of forming of attribute models of problem domains. III-st Soviet AI Conference, Tver, 1992, Vol.2, pp.99–101. (in Russian)

[31] G.Andrienko, N.Andrienko. The acquisition of procedural knowledge in the system AFORIZM: the task of planning. In "Methods and systems of decision making. Systems of knowledge processing in automated designing", RTU, Riga, 1992. (in Russian)

[32] G.Andrienko, N.Andrienko. The acquisition of procedural knowledge in the system AFORIZM: the task of selecting. In "Methods and systems of decision making. Systems of knowledge processing in automated designing", RTU, Riga, 1992. (in Russian)

[33] V.Levchenko, A.Savinov. Qualitative aggregation of Information in Fuzzy Attribute Model. Computer Science Journal of Moldova, Vol.2, No.2, 1994.

[34] G.Ginkul, S.Solowiev. An approach to knowledge base building. In "Introduce automation using micro-processor systems", 1986. p.35. (in Russian)

[35] G.Ginkul. The game approach to knowledge acquisition. Theory and practice of programming. Kishinev: Stiintsa, 1989, pp.41–47. (in Russian)

[36] A.Zinchenko. A system for logical dependences acquisition. In "Applied AI Systems", Kishinev: Stiintsa, 1991, pp.57–62. (in Russian)

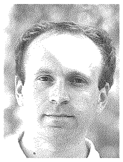
[37] G.Ginkul. Expert games: a mean to acquire of expert knowledge. Computer Science Journal of Moldova. Vol.1, No.2, pp.84–103.

[38] G.Ginkul. The game approach to knowledge acquisition: the expert game "Burime". In "Applied knowledge-based systems". Kishinev, Stiintsa, 1992, pp.42–59. (in Russian)

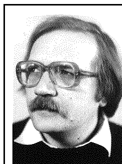
[39] G.Ginkul, S.Solowiev. CAPRICE: knowledge acquisition tool based on game approach. Computer Science Journal of Moldova, Vol.2, No.2, 1994, pp.140–150.

[40] Iu.Secrieru. The expert game for eliciting of prohibited sets of facts in attribute problem domains. In "Informatics and computer techniques". Kishinev, 1993, pp.98–101. (in Russian)

[41] G.Ginkul, IU.Secrieru. The procedure of knowledge base building in the form of expert game. Issues of automatization of managing. Proc. of All-Union Scientific Conference. Kishinev, Stiintsa, 1991, pp.90–93. (in Russian)



Gennady Ginkul is a senior scientific researcher at the Institute of Mathematics of the Academy of Sciences of Moldova. He is a Kishinev State University graduate. In 1991 he received his Ph.D. Degree in computer Science from Moscow State University. His research interests include knowledge acquisition and machine learning, cognitive science and man-machine interaction. He has published more than twenty papers.
phone: (373–2) 738130; fax: (373–2) 738027
e-mail: ginkul@math.moldova.su



Sergei Solowiev is Professor of Computer Science at the Institute of Mathematics of the Academy of Sience of Moldova. He received his M.S. and Ph.D. in computer science from Moscow State University in 1980 and 1986, respectively. S.Solowiev has authored some 40 papers. In 1996 he received a Professor degree from Tver State University. His interest span several areas of artificial intelligence, including knowledge acquisition and representation, knowledge-based machine translation, planning and problem solving.