Fundamenta Informaticae (2013) 1–16
IOS Press

A Domain Knowledge as A Tool For Improving Classifiers

Jan G. Bazan*
Institute of Computer Science, University of Rzeszów,
Pigonia 1, 35-959 Rzeszów, Poland and
Institute of Mathematics, The University of Warsaw
Banacha 2, 02-097 Warszawa, Poland
bazan@univ.rzeszow.pl

Sylwia Buregwa-Czuma
Institute of Computer Science, University of Rzeszów,
Pigonia 1, 35-959 Rzeszów, Poland
sczuma@univ.rzeszow.pl

Andrzej W. Jankowski
Institute of Computer Science, Warsaw University of Technology,
Nowowiejska 15/19, 00-665 Warsaw, Poland and
Institute of Mathematics, The University of Warsaw
Banacha 2, 02-097 Warszawa, Poland
a.jankowski@ii.pw.edu.pl

Abstract. This paper investigates the approaches to an improvement of classifiers quality through the application of a domain knowledge. The expertise may be utilizable on several levels of decision algorithms such as: feature extraction, feature selection, a definition of temporal patterns used in an approximation of the concepts, especially of the complex spatio-temporal ones, an assignment of an object to the concept and a measurement of the objects similarity. The domain knowledge incorporation results then in the reduction of the size of searched spaces. The work constitutes an overview of classifier building methods efficiently utilizing the expertise, worked out latterly by Professor Andrzej Skowron research group. The methods using domain knowledge intended to enhance the quality of classic classifiers, to identify the behavioral patterns and for automatic planning are discussed. Finally it answers a question whether the methods satisfy the hopes vested in them and indicates the directions for future development.

Keywords: rough set, concept approximation, ontology of concepts, discretization, behavioral pattern identification, automated planning, wisdom technology

1. Introduction

Definability of concepts is a term well-known in classical logic (see, *e.g.*, [22]). Yet in numerous applications, the concepts of interest may only be defined approximately on the basis of available, incomplete

*Address for correspondence: Institute of Computer Science, University of Rzeszów, Pigonia 1, 35-959 Rzeszów, Poland

information about them (represented, *e.g.*, by positive and negative examples) and selected primary concepts and methods for creating new concepts out of them. It brings about the necessity to work out approximate reasoning methods based on inductive reasoning (see, *e.g.*, [13, 23, 24, 25]).

Classifiers also known in literature as *decision algorithms*, *classifying algorithms* or *learning algorithms* may be treated as constructive, approximate descriptions of concepts (decision classes). These algorithms constitute the kernel of *decision systems* that are widely applied in solving many problems occurring in such domains as *pattern recognition*, *machine learning*, *expert systems*, *data mining* and *knowledge discovery* (see, *e.g.*, [23, 24, 25, 12]).

In the literature there can be found descriptions of numerous approaches to constructing classifiers, which are based on such paradigms of machine learning theory as *classical and modern statistical methods*, *neural networks*, *decision trees*, *decision rules*, *inductive logic programming*, and *rough sets* (see, *e.g.*, [23, 24, 27, 28, 26] for more details).

The aforementioned methods of classier building demand some kind of representation of the information about an object instance or object features. Moreover, in case of the training objects (applied for a concept learning or a classification), the information about affiliation of objects to decision classes is added to information about the features used in learning. The typical model of data employed to classifier construction is a rectangular data table, which in the rough set theory is called a decision table [28]. The decision table consists of finite number of the rows called objects and the columns called the attributes. The rows represent the information about a single real object instance and for that reason are called the objects. The columns describe the features of the objects expressed as numerical or textual (symbolical) values, and are called the attributes. One of the columns (usually the last one) represents the decision class membership of an object and is named a decision attribute. The classifier construction approach based on a decision table often comes up against difficulties, because in many applications it is hard to build the table for a construction of the efficient classifier. The reasons for this may be e.g., difficulties with the definition of the proper features approximating given concept (the feature extraction problem) or difficulties with the selection of appropriate features approximating given concept from available data set (the feature selection problem). Moreover, in many cases the problems with the definition of an objects assignment to the concept (especially if the concept is defined in a complex way in a natural language), the approximation of the defined concept based on an attribute with a large number of values, the measurement of objects similarity in the context of decision attribute values (e.g. in a situation when the value of a decision attribute is a complex object, such as a behavioral graph, plan or task performance algorithm etc. (see, e.g., [3]). The difficulties often result in the non-optimal effects of classic methods of classifier construction for a given decision problem which are based on established heuristics for selection, reduction, discretization, etc. The above-mentioned difficulties arise especially in the case where the approximation of complex spatio-temporal concepts is needed. Such concepts are expressed in a natural language on far superior level of the abstract then sensory data, usually used for concept approximation. The examples of the concepts are: *safety drive by car*, *the patients behaviour associated with life threatening*, *a plan of an effective collective task performance by a group of robots* etc.

A crucial limitation of the existing methods is, among other things, the fact that an effective approximation of complex concepts requires discovery of extremely complex patterns. Intuitively, such concepts are too far in the semantical sense from the available concepts, eg., sensory ones. As a consequence, the size of spaces which should be searched in order to find patterns crucial for approximation are so large that an effective search of these spaces very often becomes unfeasible using the existing methods and technology. Thus, as it turned out, the ambition to approximate complex concepts with high quality from available concepts (most often defined by sensor data) in a fully automatic way, realized by the existing systems and by most systems under construction, is a serious obstacle since the classifiers obtained are often of unsatisfactory quality (see, eg., [33, 30, 3]).

Recently, it has been noticed in the literature (see, eg., [11, 30, 3]) that one of the challenges for data mining is discovery of methods linking detection of patterns and concepts with domain knowledge. The latter term denotes knowledge about concepts occurring in a given domain and various relations among them. This knowledge greatly exceeds the knowledge gathered in data sets; it is often represented in a natural language and usually acquired during a dialogue with an expert in a given domain. One of the ways to represent domain knowledge is to record it in the form of the so-called concept ontology where ontology is usually understood as a finite hierarchy of concepts and relations among them, linking concepts from different levels (see, eg., [16]).

The concepts occurring in such ontologies might be treated as certain types of clues helpful with approximation of complex concepts with the help of the so called sensory concepts, which are concepts simple enough to be approximated effectively only with the help of available data sets.

The aim of the paper is to review existing methods of classifier building using successfully a domain knowledge, that were developing lately by Professor Andrzej Skowron group. The main attention is focused on the methods that were elaborated featuring the authors of the present paper. Beyond the theoretic depiction, we describe an implementation of these methods and their application to the real life data. The descriptions are rather brief (further details can be found in indicated literature). The main goal of the authors was to summarize whether the expectations for the methods were exceeded and to indicate directions in which they should be developing.

The organization of the paper is following. In Section 2 we discuss the approaches aimed to improve the quality of classic methods of classifier construction using a domain knowledge, putting emphasis on the methods of attribute discretization. In Section 3 we describe the approach to the identification of behavioral patterns or high risk, paying particular attention to plans of novel method construction within the scope. Finally in the Section 4 we debate on automatic planning methods exploiting domain knowledge. Moreover, we describe our suggestions regarding further development of the methods.

2. Improving the classical methods of classifier construction

As already mentioned, the main motivation of using the domain knowledge to improve the quality of classifiers is the fact that the expertise may be useful to selection of the classification model proper for given data set using specified paradigm of classifier construction (*e.g.*decision rules, decision trees, statistical methods etc.). The space of available classifiers using determined classifier construction paradigm may be very extensive. Meanwhile, for practical requirements, only one or few classifiers are needed, which will work as successful as possible not only for training data but also for testing one. Each classic method is based on some kind of heuristic delivering specified classifier. If the heuristic does not consider the domain knowledge in the area enough, the efficiency of constructed classifiers may be unsatisfactory for testing data, although effective for training set. The classic example is a situation in which the method of classifier building prefers certain numerical attribute, perfectly discriminating decision classes of some medical problem in a training sample, but the experts consider it meaningless in diagnosis, so the classifier should not select the feature. The case illustrates that the domain knowledge is often of great value for choosing better classifier.

The situation arises from the fact that there are often relatively little data sets at our disposal, which are not statistically representative to the explored decision problems. In that case, additional domain knowledge application seems to be the only way of retrieving successive classifiers in practise.

One way of using the domain knowledge for classifier improvement is the direct application of it for the enhancement of the effectiveness of existing methods using decision table. The approach was utilized for a long time. For example, expert's hints regarding weights of decision classes may be incorporated for the generation of rules. The indications may be used for the classifier construction or for the classification of new objects. Building therefore a rule classifier for certain disease identification (e.g., two decision classes: "ill" - a patient suffers from a given disease, "healthy" - a patient is not ill), an augmentation of the decision class weight often allows for a diminution of the number of false positive cases - patients classified as "healthy". It matters for the enhancement of so called specificity of decision classes. Instead the application of attribute weights may for example, be useful at choosing an appropriate reduct from the assigned reduct set (a reduct is a minimal set of attributes maintaining the discernibility of the objects like the whole set of attributes), which will be employed for further classifier construction.

Another example of the improvement of the classifier quality is a discretization of attributes supported by the domain knowledge. The discretization of attributes is often employed for a classifier building. Here, we consider the supervised discretization, i.e. we mean the methods of discretization that use a decision attribute's value for the training cases. There are a lot of supervised discretization methods based on different heuristics. In the paper we describe an approach premised on the generation of a local discretization decision tree (see, eg., [8, 26, 5]). The binary tree is constructed using repeated divisions of a given data set into two groups of objects (e.g. patients) by means of chosen attribute's values. For instance, for a numerical attribute a (with plentiful organized values), the partition of the patients may be performed using the value v of a given attribute, in a manner that patients with values of the attribute a greater or equal to v belong to one group, and the another group consists of patients whose values of the attribute are less then v. Let's notice that the partition of the object set may also take place using a symbolic attribute (non numerical, with a modest amount of values). For example, for the attribute b with symbolic values the split may be performed using some value v in that way, that patients whose value of the attribute b is equal to v belong to one group, and the patients with the value of the attribute b different from v to the another one. The way of the selection of an attribute and its value (for numerical attributes often called a cut) applying for the partition, is a key element of the discussed method for local discretization tree building and should be related to the analysis of the decision attribute values for training objects. As a measure of a cut quality, a number of pairs of objects discerned by a cut with different decision attribute's values can be used. For instance, when a given cut c (a cut is a value of chosen attribute) divides the objects into two groups of M and N size and the number of objects with $C0$ and $C1$ class equals $M0$ and $M1$ in one group, and the other group contains $N0$ and $N1$ objects of $C0$ and $C1$ class respectively, then the number of object pairs discriminated by the cut c amounts to: $M0 * N1 + M1 * N0$. If we compute the value of this measure for all potential pairs (attribute, value), then we can greedily choose one of pairs and divide the whole data set into two parts based on it.

Accordingly, a root of a tree contains the entire set of objects. Next we recursively use the same splitting procedure to appearing parts, which are assigned to tree nodes on more lower levels. The stopping criterion for a division is constructed in a manner that a given part is not split (becomes a tree leaf) when it contains the objects with single decision class (alternatively the objects of a given class constitute specified percent which is treated as a parameter of the method) or further partitions do not yield any results (all potential cuts do not discern the pairs of the objects with distinct classes any more).

An exemplary decision tree is presented on Fig. 1.

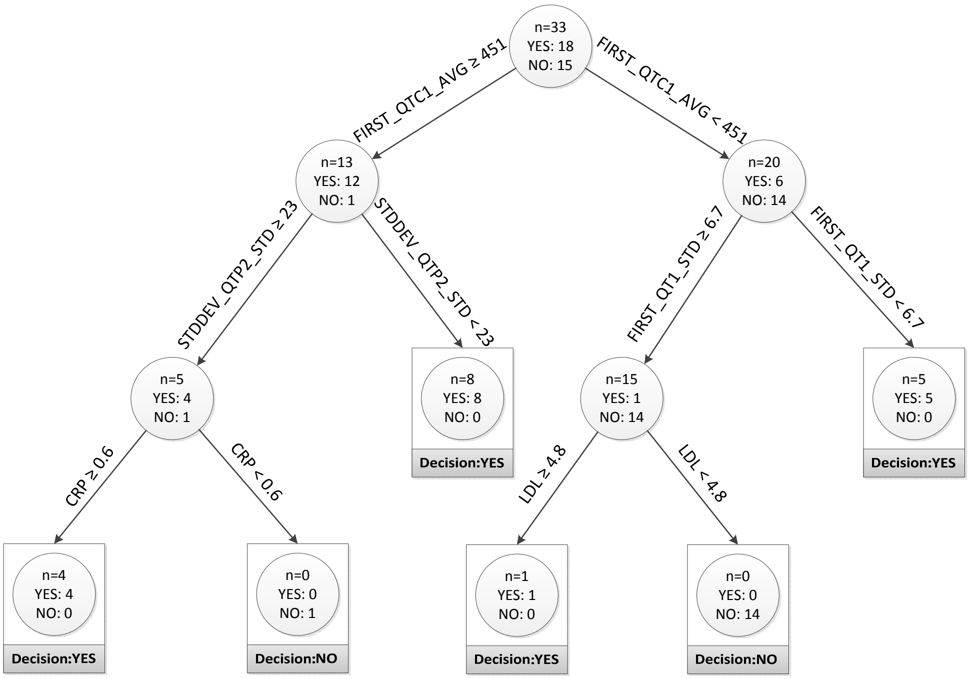

Figure 1. The exemplary decision tree from [5]

Note that the above tree can be handled explicitly as a classifier because the test objects may be classified by determination to what leaf do they belong. It is possible, because, thanks to determined cuts, one can retrace the route of object's affiliation from the root down to the leaves and thereon classify the object to the decision class dominating in the leaf. In the paper we pay the main attention to the fact that the construction of the tree provides the cut set which may serve to the construction of new binary attributes for a given decision table. Information about the cuts may also be grouped by attributes and the new values of the input decision table attributes may be determined. Such a decision table we call the discretized table and the values of all of its attributes are symbolic and arise from initial numerical values. For example, if for an attribute a, the cuts $c1$ and $c2$ are delimited, then the values of the attribute a after a discretization may be valued at $v1$ (the value associated with a range of the original attribute values from $c1$ to $c2$), $v2$ (the value related to a range of the original attribute values $a < c1$) or $v3$ (the value associated with a range of the original attribute values: $a >= c2$).

The way of a cut quality measure computation, described above, may be modified using the domain knowledge. For instance, for a problem of the identification of the patients requiring a revascularization based on Holter ECG recording (see [5]), the measure of a cut quality may be amended through the integration of the knowledge about the number of narrowed vessels in a particular patient. Namely, a given cut receives the specific number of points for each of distinguished pair of the patients, where the points are assigned according to the Table 1.

Notice that the cut receives 3 points for a discernment of the patient with untouched arteries from the patient having only one narrowed vessel. Instead of, the distinction of patients with normal arteries from those having 3 narrowed vessels assigns only 1 point. It is a way to make border between positive and

Table 1. The weights relevant to the number of vessels involved in coronary heart disease

	0	**1**	**2**	**3**
0	0	3	2	1
1	3	0	1	1
2	2	1	0	1
3	1	1	1	0

negative examples more exposed. We are interested in distinguishing patients with normal and narrowed (one, two or three) vessels, that is why we put the biggest weights between negative examples of the concept and the remainder. Class differences are most subtle between patients with normal arteries and with only one vessel changed, compared with divergence between normal and tree-vessel disease. The cut discriminating patients with normal and one vessel disease is assigned the biggest weight.

The points of particular cuts are summed and the cut with the biggest number of points is chosen, at a given level of a tree building. The data analysis employing the above method of a cut quality measurement performed for clinical and laboratory data together with 24-hour Holter ECG monitoring in limited group of patients, have led to the development of preliminary methods with 94% of sensitivity and 80% of specificity in predicting the necessity of revascularization (see [5]). The results are much better then those with 78% of sensitivity and 73% of specificity received using the same method of classifier building, but without implementation of the domain knowledge (see [6]).

The foregoing circumstances indicate that using additional domain knowledge may yield in the enhancement of the classifier quality. The described example of the domain knowledge implementation is rather simple and certainly does not exhaust a topic within the scope of a discretization advancement using the expertise.

Notice that in the above method, the quality of the cuts is modified by extra information about a patient (the number of narrowed arteries). It is possible, because we add another attribute to the original binary decision one, for the sake of the computation of a measure of the cuts. Moreover, the proposition of the method of making the use of the information is introduced, which enables the computation of a cut measure based on the weights table. It is easy to imagine that a quality of a cut discerning a pair of objects could depend on the decision attribute in a more complex way. For example, for a pair of objects more, in some sense, diverse one from another referring to the decision value, the value of a cut measure could have an increased value in a more subtle way. It requires still novel methods to measure a similarity between objects in the context of the decision attribute values. The situation is especially troublesome when the values of the decision attribute are complex in some sense (e.g. a vector of values, a behavioral pattern, a plan etc.). The deduction arises then, that the incorporation of supplemental domain knowledge is needed to measure the similarity of objects in the context of the decision attribute value. An exemplary method of that type could be the method based on specially constructed ontology of the concepts, similar to the one used to measure the similarity between plans (see Section 4). Notice that carrying out the research towards the issue will require high level of medical experts involvement. The experts will have to define the medical ontology describing the patient's similarity in the context of e.g. the revascularization necessity. Predictably the resemblance measure based on the ontology may significantly enhance the quality of the identification of the revascularization necessity in testing patients.

3. Methods of Behavioral Pattern Identification

Many real-life problems may be modeled with the help of the so-called *complex dynamical systems* (see, *e.g.*, [2, 10]) or, putting it in an other way, *autonomous multiagent systems* (see, *e.g.*, [17]) or *swarm systems* (see, *e.g.*, [29]). These are sets consisting of complex objects which are characterized by the persistent changes of parameters of their components over time, numerous relationships among the objects, the possibility of cooperation/competition among the objects and the ability of objects to perform more or less complicated actions (see, *e.g.*, [3] for more details). The study of collective behavior in complex dynamical systems is now one of the more challenging research problems (see, *e.g.*, [10, 20, 29]). For example, an efficient complex dynamical systems monitoring very often requires the identification of the so-called *behavioral patterns* or a specific type of such patterns called *high-risk patterns* or *emergent patterns* (see, *e.g.*, [1, 10, 20, 21]).

They are complex concepts concerning dynamic properties of complex objects expressed in a natural language on a high level of abstraction and describing specific behaviors of these objects. Examples of behavioral patterns may be: *overtaking one vehicle by another vehicle*, *driving a group of vehicles in a traffic jam*, *behavior of a patient under a high life threat*, etc. These types of concepts are difficult to identify automatically because they require watching complex object behavior over longer period of time and this watching usually is based on the identification of a sequence of less complex spatio-temporal concepts (see, *e.g.*, [3]). Moreover, a crucial role for identification of a given behavioral pattern is played by the sequence of less complex concepts which identify it. For example, in order to identify the behavioral pattern of *overtaking one vehicle by another*, it should first be determined whether the overtaking vehicle approaches the overtaken vehicle; next, whether the overtaking vehicle changes lanes appropriately and overtakes the vehicle; and finally, to determine that the overtaking vehicle returns to the previous lane driving in front of the overtaken vehicle. The methodology of a dynamical system modeling outlined in the paper enables approximation of behavioral patterns on the basis of data sets and domain knowledge expressed using a concept ontology.

The starting point for the method of identification of behavioral patterns is a remark that identification of behavioral patterns requires an observation of a complex object over a longer period of time called a time window. To describe complex object changes in the time window, the so-called temporal patterns are used, which are defined as functions determined on a given time window. These patterns, being in fact formulas from a certain language, also characterize certain spatial properties of the complex object examined, observed in a given time window. They are constructed using lower level ontology concepts (in case of hierarchical ontology) and that is why identification whether the object belongs to these patterns requires the application of classifiers constructed for concepts of the lower ontology level.

On a slightly higher abstraction level, the spatio-temporal concepts (also called temporal concepts) are directly used to describe complex object behaviors. Those concepts are defined by an expert in a natural language and they are usually formulated using questions about the current status of spatio-temporal objects. The methods are based on approximating temporal concepts using temporal patterns with the help of classifiers. In order to do this a special decision table is constructed called a temporal concept table. The rows of this table represent the parameter vectors of lower level ontology concepts observed in a time window. Columns of this table are determined using temporal patterns. However, the last column represents membership of an object, described by parameters (features, attributes) from a given row, to the approximated temporal concept (see Fig. 2).

Temporal concepts may be treated as nodes of a certain directed graph which is called a behavioral

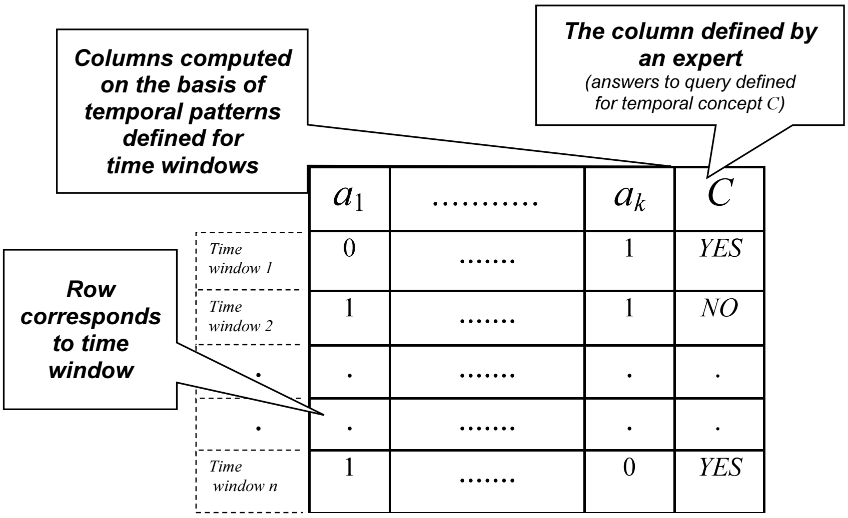

Figure 2. The scheme of a temporal concept table

graph (see Fig. 3). Links (directed edges) in this graph are the temporal relations between temporal concepts meaning a temporal sequence of satisfying two temporal concepts one after another. The method of behavioral pattern identification is based on interpreting the behavioral graph as a complex classifier enabling identification of a behavioral pattern described by this graph. This is possible based on the observation of the complex object behavior for a longer time and checking whether the behavior matches the chosen behavioral graph path. It is determined if the behavior matches the behavioral pattern represented by this graph, which enables a detection of specific behaviors of complex objects.

It is worth mentioning that in practical applications for identifying behavioral patterns, you may need to make successive levels of the hierarchy of complex objects with an even more complex structure. For such higher levels of the hierarchy, the identification of behavioral patterns requires observation of objects on the so-called time paths (sequences of temporal concepts defined on the basis of the sequences of time windows or on the basis of clusters of time windows sequences) [3]. In this case, the features of complex objects describe their behaviors observed on the time paths. The values of these features determine the indiscernibly classes or the similarity classes that are time windows sequences. In case of approximation of time concepts at this level, conditional attributes may be features whose values are represented for instance as Petri nets that are consistent with a given collection of time windows. At the next level, objects can be represented by models of concurrent processes that satisfy certain constraints responsible for the interaction of these processes. Such an approach may be necessary for efficient approximation of very complex concepts, which concern the interaction of many processes (such as modeling the behavior of a patient suffering from several diseases remain together in continuous interaction).

In this direction, a computer system RoughICE (ang. Rough Set Interactive Classification Engine) has been created [31]. The RoughICE is a software platform supporting, among other things, the approximation of spatio-temporal complex concepts with the given concept ontology in the dialogue with the user. The main aim of the RoughICE platform is to facilitate construction of complex, hierarchical classifiers based on data sets, domain knowledge and interaction with human experts. For instance, the software makes it possible to construct classifiers for identification of behavioral pattern or high risk

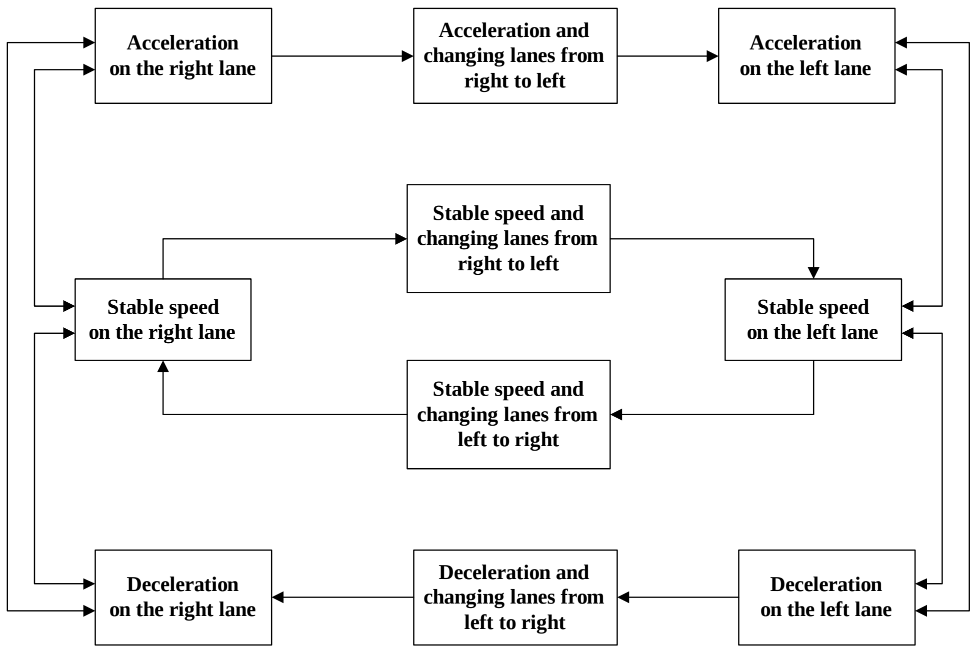

Figure 3. A behavioral graph for a single object-vehicle (data sets from the road simulator)

pattern.

The experiments have been performed on the data sets obtained from the road simulator and two real medical data sets. In case of the data sets obtained from the road simulator, our experiments were related to detection of the overtaking maneuver (see [3]). While, in case of medical data sets we investigated two medical behavioral patters: *higher-death risk in infants suffering from the respiratory failure* (see [7, 3]) and *sudden cardiac death risk by analyzing cardiovascular failure* (see [4]).

Experimental results showed that the suggested method of behavioral patterns identification gives good results, also in the opinion of medical experts (compatible enough with the medical experience) and may be applied in medical practice as a supporting tool for medical diagnosis and treatment evaluation.

It is also worthwhile mentioning that the method reported above concerns the identification of present behavior of the complex object, not complex behavior, which will take place in the future. For instance, if a coronary artery by-pass graft surgery is being performed, it would be interesting from the medical point of view to predict which behavioral pattern (in the context of circulatory system stability) the patient will match after the operation. Determining if the patient will match the dangerous behavioral pattern connected with a high death risk is extremely vital, for it requires undertaking certain remedial operations in advance.

Thus, developing the methods predicting the shifting of one behavioral pattern of a complex object into another is planned in our future research. It concerns a situation when a complex object matches first one pattern and then after a certain amount of time matches a different one and means an early prediction (when the object still matches the first pattern) if in a certain amount of time the object will match the second pattern (e.g. prediction of change of the patient's stable behavior into an unstable one during an intensive therapy; detecting the fact that a client or citizen who has been honest so far is planning a bank or tax fraud, early detection of change of a correctly working device pattern into a pattern of work with certain disruptions which may lead to a device malfunction and others). This method may be treated as a more advanced than the previous ones (we mean here the methods of the behavioral pattern identification itself) serving a more effective approximation of complex concepts. It is worth noticing that this issue is related to the domain of machine learning called *the concept drift* (see, *e.g.*, [19, 9]). Research works in this domain often concern construction of methods for reconstructing classifiers according to the changes (drift) of decision classes.

The method of predicting the shifting of one complex object from one behavioral pattern into another may be realized on the basis of the classifiers which will learn on specially prepared decision tables. Decision classes in these tables are connected with particular behavioral patterns. However, features are constructed as formulas in a certain language (e.g. one using elements of temporal logic) and are defined by experts. These formulas represent specific characteristics of the behavior of complex objects in the near and distant future. It is particularly vital that those types of characteristics include the behavior of a complex object in the context of a pattern (or patterns) of the behavior the object matches at the time of speaking. To the formulas describing historical behavior of objects may be added those which describe the planned changes in parameters or existing conditions of complex objects (e.g. performing a surgery). It may be expected that this approach will allow effective and relatively early prediction of behavioral pattern changes of complex objects.

4. Methods of Automated Planning

In monitoring the behavior of complex objects (*e.g.*, by means of behavioral patterns identification) there may appear a need to apply methods of automated planning of complex object behavior. For example, if during observation of a complex object, a behavioral pattern describing inconvenient or unsafe behavior of a complex object (i.e., a part of system state or trajectory) is identified, then the system control module may try, using appropriate actions, to change the behavior of this object in such a way as to lead the object out of the inconvenient or unsafe situation. However, this type of short-term interventions may not be sufficient to lead the object out of the undesired situation permanently. Therefore, a possibility of automated planning is often considered which means construction of sequences of actions alternately with states (of plans) to be performed by the complex object or on the complex object in order to bring it to a specific state. In literature, there may be found descriptions of many automated planning methods (see, *e.g.*, [14, 32]). However, applying the latter approaches, it has to be assumed that the current complex object state is known which results from a simple analysis of current values of available parameters of this object. Meanwhile, in complex dynamical systems, a complex object state is often described in a natural language using vague spatio-temporal conditions whose satisfiability cannot be tested on the basis of a simple analysis of available information about the object [3]. This type of conditions may be represented using complex spatio-temporal concepts. Identification of these conditions requires, however, an approximation of the concepts representing them with the help of classifiers. Therefore, in the paper, we describe automated planning methods of behavior of complex objects whose states are described using complex concepts requiring approximation.

Two kinds of methods for automated planning have been developed in the Professor Skowron group at the Institute of Mathematics, Warsaw University. The first kind of such methods is the Exhaustive Expert Forward Search (EEFS) method reported in [3]. This method works on the basis of data sets and a domain knowledge represented by a concept ontology. A crucial novelty in the methods mentioned here, in comparison with the already existing ones, is the fact that performing actions according to plan depends on satisfying complex vague spatio-temporal conditions expressed in a natural language, which leads to the necessity of approximation of these conditions as complex concepts. Moreover, these conditions describe complex concept changes which should be reflected in the concept ontology.

Behavior of complex objects is modeled using the so-called *planning rules* being formulas of the type: *the state before performing an action* $\to$ *action* $\to$ *state* 1 *after performing an action* $|$ ... $|$ *state* k *after performing an action*, which are defined on the basis of data sets and a domain knowledge. Each rule includes the description of the complex object state before applying the rule (that is, before performing an action), expressed in a language of features proposed by an expert, the name of the action (one of the actions specified by the expert which may be performed at a particular state), and the description of sequences of states which a complex object may turn into after applying the action mentioned above. It means that the application of such a rule gives indeterministic effects, i.e., after performing the same action the system may turn into different states.

Let us consider the planning rule from Fig. 4. This is the planning rule for treating RDS (respiratory distress syndrome) obtained from domain knowledge (see [3]). The rule may be applied when RDS with very severe hypoxemia is present. The application of the rule consists in performing a medical action utilizing the respirator in the MAP3 mode (see [3] for more medical details). As an effect of the application of this action at the following time point of observation (*e.g.*, the following morning) the patient's condition may remain unchanged or improve so as to reach the condition of RDS with severe hypoxemia.

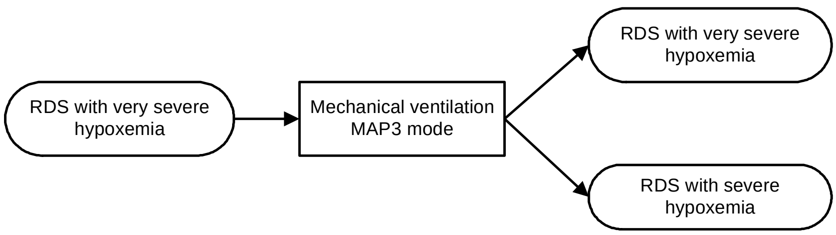

Figure 4. The medical planning rule

All planning rules may be represented in a form of the so-called *planning graphs* whose nodes are state descriptions (occurring in predecessors and successors of planning rules) and action names occurring in planning rules. In the graphical interpretation, solving the problem of automated planning is based on finding a path in the planning graph from the initial state to an expected final state. It is worth noticing that the conditions for performing an action (object states) are described by vague spatio-temporal complex concepts which are expressed in the natural language and require an approximation.

Let us consider planning graph from the Fig. 5, where the states are represented using ovals, and actions are represented using rectangles. Each link between the nodes of this graph represents a time dependencies. For example, the link between state s_1 and action a_1 tells us that in state s_1 of the complex object action a_1 may be performed, whereas the link between action a_1 and state s_3 means that after performing action a_1 the state of the complex object may change to s_3. An example of a path in this graph is sequence (a_2, s_2, a_3, s_4) whereas path $(s_1, a_2, s_2, a_3, s_3)$ is an exemplary plan in this graph.

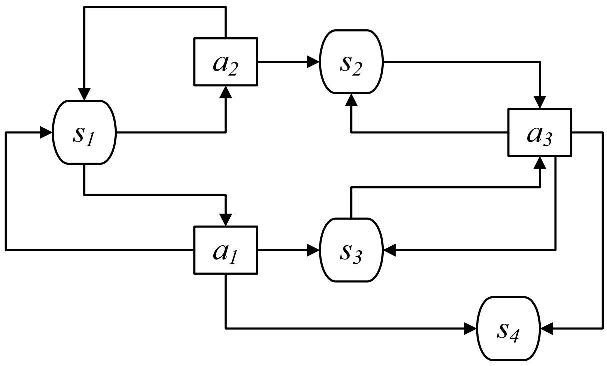

Figure 5. An exemplary planning graph

In the graphical interpretation, solving the problem of automated planning in this method is based on finding a path in the planning graph leading from the initial state to an expected final one.

For specific applications connected with the situation when it is expected that the proposed plan of a complex object behavior is to be strictly compatible with the determined experts' instructions (*e.g.*, the way of treatment in a specialist clinic is to be compatible with the treatment schemes used there), there has also been proposed an additional mechanism enabling to resolve the nondeterminism occurring in the application of planning rules. This mechanism is an additional classifier based on data sets and domain knowledge. Such classifiers suggest the action to be performed in a given state and show the state which is the result of the indicated action.

The automated planning method for unstructured objects has been generalized also in the case of planning of the behavior of structured objects (consisting of parts connected with one another by dependencies) (see [3] for more details).

The second kind of methods for automated planning that have been developed in the Professor Skowron group were methods of planning in the context of case-based planning, where plans (of treatment) are treated as complex decisions (see [15]). Then, the complex decision can be constructed on the basis of the complex decisions for training examples which are closest to the analyzed case. One of the most important steps of the case-based planning approach is the construction of relevant similarity measure for finding cases close to a given new testing case. In this approach, a combination of advice of medical expert and machine learning techniques has been applied to obtain useful domain knowledge for construction of the similarity measure. The approach is based on ontology constructed by an expert and on rough approximation of the ontology concepts. We test two kind of measures. The first one measures the closeness of two plans which are historical (past) treatment plans, i.e., treatment plans realized in the past for some patients. The second one measures the compatibility of two joined (treatment) plans. It tells us how relevant for the new case is the complex decision (i.e., plan) extracted from the known (training). It is worthwhile mentioning that the ontology selection is an important problem related to a more general problem of context selection in approximate reasoning. In particular, selection of ontology has great impact on the quality approximation of similarity. The method based on case-based planning is different from the Exhaustive Expert Forward Search (EEFS) method reported in [3]. The EEFS method is based on hierarchical networks of classifiers induced from data and ontology of concepts delivered by experts. Such classifiers are constructed using decision rules computed by rough set tools. The case-based methodology is used to predict some steps of treatment. The main advantage of the method in comparison with the EEFS method is that it is more understandable by medical doctors. Moreover, this method usually gives better results in terms of accuracy and coverage.

In construction and application of classifiers approximating complex spatio-temporal concepts, there may appear a need to construct, with a great support of the domain knowledge, a similarity relation of two elements of similar type, such as complex objects, complex object states, or plans generated for complex objects. Hence, a new method of similarity relation approximation has been proposed based on the use of data sets and a domain knowledge expressed mainly in the form of a concept ontology [3]. We apply this method to verify automated planning methods, that is, to compare the plan generated automatically with the plan suggested by experts from a given domain.

In order to check the effectiveness of the automated planning methods described, there were performed experiments concerning planning of treatment of infants suffering from the respiratory failure. Experimental results showed that the proposed methods give good results, also in the opinion of medical experts (compatible enough with the plans suggested by the experts), and may be applied in medical practice as a supporting tool for planning of the treatment of infants suffering from the respiratory failure.

During the attempt to execute the plan constructed there often appears a need to reconstruct the plan which means that during the plan execution there may appear such a state of a complex object that is not compatible with the state suggested by the plan. A *total reconstruction* of the plan (building the whole plan from the beginning) may computationally be too costly. Therefore, we proposed another plan reconstruction method called *a partial reconstruction*. It is based on constructing a short so-called *repair plan*, which rapidly brings the complex object to the so-called *return state* which appears in the current plan. Next, on the basis of the repair plan, a current plan reconstruction is performed through replacing its fragment beginning with the current state and ending with the return one.

In future, we are going to utilize the approach to plan reconstruction presented above, in utilitarian project related to real life data sets.

Moreover, we plan to elaborate a novel method for a plan generation based on inconsiderable changes of available plans. The method assumes the observation, that in order to generate an efficient plan for complex object (e.g. for a treatment plan of a patient) it is sometimes enough to discover a training plan derived from the therapy of the other patient, which may be easily modified for the needs of the current patient. The approach of such a type may be realized using the concept of so called a plan modification, which we define with the help of the domain knowledge and classifiers for approximation of such concepts. Furthermore, the method will require a special tool to verify whether the plans established after the modifications are compatible with the domain knowledge in a given area.

It should be emphasized that, in general, full and integrated approach for responsible automatic plan generation is a very difficult and complicated task because it requires extensive medical knowledge combined with sensor information about the state of a patient. In particular it requires advanced AI technology for representation, processing and integration of practical wisdom from many domains (by wisdom we understand here very advanced domain knowledge - see below). It is important to use appropriate wisdom granulation, and at once we have to avoid solutions when the holistic perspectives are lost or ignored. Moreover the action has to be on time. In many cases lack of any action has worse consequences then non prefect - but on time - action. In order to build any software systems for automatic plan generation at the beginning we have to select appropriate model of computation. We have started some experiments with models of interactive granular computations based on models for rough granular approach to Wisdom Technology [18].

The main idea of the Wisdom computational model is based on the following metaphorical wisdom equation: **WISDOM = INTERACTIONS + ADAPTIVE JUDGEMENT + KNOWLEDGE**. Intuitively speaking it means we have to integrate models of the following components:

1. INTERACTIONS - information about interactions between patient, medical stuff, sensors, communication channels, computers, especially ability to follow by physicians important phenomena related to patient status and state of environment, it should be combined with the ability to act in the form of appropriate interactions with the patient and the environment (for example implementations of some treatment plans and/or emergency plans for treatment plans modification);

2. ADAPTIVE JUDGEMENT - adaptive medical habits and necessary recommendations for emergency actions, adaptive reasoning rules, intuitions and adaptive instinctive reactions called all of them as the (adaptive) judgment, for quick adaptive assessment of changes on the basis of the situation perception and for quick translating of the judgment results for recommendations of performing of relevant actions;

3. KNOWLEDGE - having the access to right knowledge representation for the interpretation of the observed phenomena and drawing important conclusions. One of the key issues is appropriate concept selection and granulation for fast understanding and optimal knowledge processing about the observed (by interaction processes) phenomena.

We plan to develop a new framework for computations based on the Wisdom Technology. We expect that an approach to classifier construction based on the Wisdom Technology will allow to improve the quality of classifiers, especially constructed for complex concepts.

5. Conclusion

Many different aspects of the application of the domain knowledge in classifier building methods were presented. The benefits of such a strategy proceed from the potential practical use. The outlined results of the experiments performed on real life data sets showed that methods developed by Professor Andrzej Skowron group are realizable for implementation and launching and suggest their future importance and the usefulness in practise. The theoretical considerations indicate that there is still quite a lot of work to do in terms of linking detection of patterns and concepts with the domain knowledge. In our further work we would like to develop or refine the methods for the measurements of the objects similarity based on a concept ontology, the prediction of behavioral pattern changes of the complex objects, the approximation of a similarity relation and the generation of a plan reconstruction or plan modification. We also want to develop the Wisdom Technology. The innovative solutions will serve as a more effective approximation of the complex objects and subsequently will enable a more efficient classification.

Acknowledgement

We would like to express our deepest gratitude to our mentor, Professor Andrzej Skowron for the inspiration and comprehensive support during the whole of our scientific life. It goes without saying that without his great involvement in scientific research and creative influence he exerts on the scientific development of the Group of Mathematical Logic at the Institute of Mathematics, Warsaw University which remains under his supervision, the reported research in this paper would never have been possible.

This work was supported by the Polish National Science Centre grants 2011/01/B/ST6/03867 and 2012/05/B/ST6/03215 as well as by the Polish National Centre for Research and Development (NCBiR) under SYNAT - Grant No. SP/I/1/77065/10 in frame of the strategic scientific research and experimental development program: "Interdisciplinary System for Interactive Scientific and Scientific-Technical Information".

References

[1] Adam, N. R., Janeja, V. P., Atluri, V.: Neighborhood based detection of anomalies in high dimensional spatio-temporal sensor datasets, in: Proc. of the 2004 ACM symposium on Applied computing (SAC'04), Nicosia, Cyprus, ACM Press, New York, USA, 2004, 576–583.

[2] Bar-Yam, Y.: Dynamics of Complex Systems, Addison Wesley, New York, USA, 1997.

[3] Bazan, J. G.: Hierarchical classifiers for complex spatio-temporal concepts, Transactions on Rough Sets, IX, LNCS 5390, 2008, 474–750.

[4] Bazan, J. G., Bazan-Socha, S., Buregwa-Czuma, S., Pardel, P. W., Skowron, A., Sokolowska, B.: Classifiers Based on Data Sets and Domain Knowledge: A Rough Set Approach, in: Rough Sets and Intelligent Systems - Professor Zdzislaw Pawlak in Memoriam. Intelligent Systems Reference Library (A. Skowron, , Z. Suraj, Eds.), vol. 43, Springer-Verlag, Berlin Heidelberg, 2013, 93–136.

[5] Bazan, J. G., Bazan-Socha, S., Buregwa-Czuma, S., Pardel, P. W., Sokolowska, B.: Predicting the presence of serious coronary artery disease based on 24 hour Holter ECG monitoring, in: Proceedings of the Federated Conference on Computer Science and Information Systems (FedCSIS'2012) (M. Ganzha, L. Maciaszek, M. Paprzycki, Eds.), Wroclaw, Poland, 2012, 279–286.

[6] Bazan, J. G., Bazan-Socha, S., Buregwa-Czuma, S., Pardel, P. W., Sokolowska, B.: Prediction of coronary arteriosclerosis in stable coronary heart disease, in: Advances in Computational Intelligence , vol. 298 of Communications in Computer and Information Science (S. Greco, B. Bouchon-Meunier, G. Coletti, M. Fedrizzi, B. Matarazzo, and R. R. Yager, Eds.), Springer, 2012, 550–559.

[7] Bazan, J. G., Kruczek, P., Bazan-Socha, S., Skowron, A., Pietrzyk, J.: Rough set approach to behavioral pattern identification, Fundamenta Informaticae, **75**(1–4), 2007, 27–47.

[8] Bazan, J. G., Nguyen, H. S., Nguyen, S. H., Synak, P., Wróblewski, J.: Rough set algorithms in classification problems, in: Rough Set Methods and Applications: New Developments in Knowledge Discovery in Information Systems (L. Polkowski, T. Y. Lin, S. Tsumoto, Eds.), Springer-Verlag/Physica-Verlag, Heidelberg, Germany, Studies in Fuzziness and Soft Computing, vol. 56, 2000, 49–88.

[9] Brzezinski, D., Stefanowski, J.: Accuracy Updated Ensemble for Data Streams with Concept Drift, in: Proceedings of the 6th International Conference on Hybrid Artificial Intelligence Systems (HAIS 2011), Part II. LNCS (LNAI), vol. 6679 (Corchado, E., Kurzynski, M., Wozniak, M., Eds.), Springer, 2011, 155–163.

[10] Desai, A.: Adaptive complex enterprises, Communications ACM, **5**(48), 2005, 32–35.

[11] Domingos, P.: Toward knowledge-rich data mining, Data Mining and Knowledge Discovery, **1**(15), 2007, 21–28.

[12] Douzal-Chouakria, A., Amblard, C.: Classification trees for time series, Pattern Recognition, vol. 45, issue 3, 2011, 1076–1091.

[13] Gabbay, D. M., Hogger, C. J., J. A. R. (eds.): Handbook of Logic in Artificial Intelligence and Logic Programming, vol. I-V, Oxford University Press, New York, USA, 1994.

[14] Ghallab, M., Nau, D., Traverso, P.: Automated Planning: Theory and Practice, Elsevier, Morgan Kaufmann, CA, 2004.

[15] Góra, G., Bazan, J. G., Kruczek, P., Bazan-Socha, S., Skowron, A., Pietrzyk,J.: Case-based planning of treatment of infants with respiratory failure, Fundamenta Informaticae, **85**(1–4), 2008, 155–172.

[16] Guarino, N.: Formal ontology and information systems, in: Proc. of the First International Conference on Formal Ontology in Information Systems (FOIS'98), June 6-8, Trento, Italy, IOS Press, 1998, 3–15.

[17] Hoen, P. J., Tuyls, K., Panait, L., Luke, S., Poutré, J. A. L.: Overview of cooperative and competitive multiagent learning, in: Learning and Adaption in Multi-Agent Systems: First International Workshop (LAMAS 2005) (K. Tuyls, P. J. Hoen, K. Verbeeck, S. Luke, Eds.), Utrecht, The Netherlands, 2005, 1–46.

[18] Jankowski, A., Skowron, A.: Wisdom technology: A rough-granular approach, Lectures Notes in Computer Science, **5070**, 2009, 3–41.

[19] Kuncheva, L., I.: Classifier ensembles for detecting concept change in streaming data: Overview and perspectives, in: Proceedings 2nd Workshop SUEMA 2008 (ECAI 2008), Patras, Greece, 2008, pp. 5-10.

[20] Liu, J., Jin, X., Tsui, K.: Autonomy Oriented Computing: From Problem Solving to Complex Systems Modeling, Kluwer/Springer, Heidelberg, 2005.

[21] Mahoney, M., Chan, P. K.: Learning rules for anomaly detection of hostile network traffic, in: Proceedings of the Third IEEE International Conference on Data Mining (ICDM'03), December 19-22, Melbourne, Florida, USA, IEEE Computer Society, 2003, 601–604.

[22] Mendelson, E.: Introduction to Mathematical Logic, International Thomson Publishing, 1987.

[23] Michalski, R. et al. (Eds.): Machine Learning, vol. I-IV, Morgan Kaufmann, Los Altos, 1983, 1986, 1990, 1994.

[24] Michie, D., Spiegelhalter, D. J., Taylor, C. C.: Machine learning, neural and statistical classification, Ellis Horwood Limited, England, 1994.

[25] Mitchel, T. M.: Machine Learning, McGraw-Hill, Boston, MA, 1997.

[26] Nguyen, H. S.: Approximate Boolean Reasoning: Foundations and Applications in Data Mining, Transactions on Rough Sets, **V**, LNCS 4100, 2006, 334–506.

[27] Pawlak, Z., Skowron, A.: Rough sets and boolean reasoning, Information Sciences, 177, 2007, 41–73.

[28] Pawlak, Z., Skowron, A.: Rudiments of rough sets, Information Sciences, 177, 2007, 3–27.

[29] Peters, J. F.: Rough ethology: Towards a biologically-inspired study of collective behavior in intelligent systems with approximation spaces, Transactions on Rough Sets, III, LNCS 3400, 2005, 153–174.

[30] Poggio, T., Smale, S.: The mathematics of learning: Dealing with data, Notices of the American Mathematical Society (AMS), **50**(5), 2003, 537–544.

[31] Rough ICE. Project web site: `http://www.mimuw.edu.pl/~bazan/roughice/`.

[32] Wezel, W. V., Jorna, R., Meystel, A.: Planning in Intelligent Systems: Aspects, Motivations, and Methods, John Wiley & Sons, Hoboken, New Jersey, 2006.

[33] Zadeh, L., A.: A new direction in AI: Toward a computational theory of perceptions, AI Magazine, **22**(1), 2004, 73–84.